MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE MESTRADO EM SISTEMAS E COMPUTAÇÃO

GUILHERME BAESSO MOREIRA

UMA ONTOLOGIA PARA TRATAMENTO DE INCIDENTES DE SEGURANÇA DA INFORMAÇÃO

Rio de Janeiro
2018

INSTITUTO MILITAR DE ENGENHARIA

GUILHERME BAESSO MOREIRA

UMA ONTOLOGIA PARA TRATAMENTO DE INCIDENTES DE SEGURANÇA DA INFORMAÇÃO

Dissertação de Mestrado apresentada ao Curso de Mestrado 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: Prof. Julio Cesar Duarte - D.Sc.
Co-Orientador: Prof. Anderson Fernandes Pereira dos Santos - D.Sc.

Rio de Janeiro
2018

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 - Praia Vermelha
Rio de Janeiro - RJ CEP 22290-270

004.69 Moreira, Guilherme Baesso
S586e Uma ontologia para tratamento de incidentes de Segurança da Informação / Guilherme Baesso Moreira, orientado por Julio Cesar Duarte e Anderson Fernandes Pereira dos Santos - Rio de Janeiro: Instituto Militar de Engenharia, 2018.

125p.: il.

Dissertação (mestrado) - Instituto Militar de Engenharia, Rio de Janeiro, 2018.

1. Curso de Sistemas e Computação - teses e dissertações. 1. Defesa Cibernética. 2. Incidentes de Segurança. 3. Web Semântica. 4. Ontologias. 5. SPARQL. I. Duarte, Julio Cesar. II. dos Santos, Anderson Fernandes Pereira . III. Título. IV. Instituto Militar de Engenharia.

INSTITUTO MILITAR DE ENGENHARIA

GUILHERME BAESSO MOREIRA

UMA ONTOLOGIA PARA TRATAMENTO DE INCIDENTES DE SEGURANÇA DA INFORMAÇÃO

Dissertação de Mestrado apresentada ao Curso de Mestrado 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: Prof. Julio Cesar Duarte - D.Sc.

Co-Orientador: Prof. Anderson Fernandes Pereira dos Santos - D.Sc.

Aprovada em 02 de Maio de 2018 pela seguinte Banca Examinadora:

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

Prof. Anderson Fernandes Pereira dos Santos - D.Sc. do IME

Prof^{a}. Maria Claudia Reis Cavalcanti - D.Sc. do IME

Prof. Sidney Cunha de Lucena - D.Sc. da UNIRIO

Rio de Janeiro
2018

AGRADECIMENTOS

À minha esposa Keila, pela paciência, compreensão e companheirismo de sempre, e essa garra que eu tando admiro. Agora também Mestre, percorremos juntos esse caminho.

Aos meus filhos Daniel e Thiago, por me inspirarem a ser uma pessoa melhor.

À minha mãe Elza e minha tia Maria, pela sua história de luta, valores ensinados e toda dedicação que tiveram, mesmo em tempos difíceis, para garantir a minha educação. Vocês pavimentaram o caminho que me permitiu chegar até aqui.

À minha sogra Eni, pelo carinho e cuidado incondicionais com a minha família. Sem a sua ajuda não teria sido possível.

Aos meus gestores e amigos Flavio Moura, Luis Rodrigues, Marcia Vieira, Pedro Quevedo e Rodrigo Rosa, pela parceria e confiança. Serei eternamente grato por todas as oportunidades proporcionadas, inclusive a de ter viabilizado esta conquista.

À amiga Vanusa Calegario, a profissional mais dedicada e solícita que conheço, pela parceria de todas as horas, suporte e revisões criteriosas.

Ao meu concunhado Jonas Borges pelas diversas orientações ao longo dessa jornada, por toda a ajuda no âmbito familiar e por ter se tornado um amigo tão presente.

Ao amigo Felipe Castro, também Mestre pelo IME, que me incentivou a fazer o curso, apontou os caminhos, ajudou com as disciplinas obrigatórias e cuja dissertação inspirou trabalhos e quase rendeu uma parceria profissional.

Ao meu co-orientador, TC Anderson, pelas inestimáveis contribuições e por ter desempenhado este papel mesmo em meio a diversos compromissos e orientandos.

Ao meu orientador, TC Duarte, por ter aceitado sem hesitação o desafio de me orientar, mesmo tendo acabado de assumir importantes novas responsabilidades no IME e sabendo que minha proposta não passava ainda de uma ideia. Foram inúmeras revisões e edições de textos, reuniões, e-mails, cobranças na medida e uma postura sempre realista mas nunca pessimista. Uma orientação absolutamente precisa e exemplar, da qual terei sempre imensa gratidão.

A Deus, parceiro de todas as lutas e conquistas, por ter proporcionado tudo isso e muito mais.

“O que sabemos é uma gota; o que ignoramos é um oceano.”

ISAAC NEWTON

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE CÓDIGOS

RESUMO

A rápida evolução da tecnologia da informação nas últimas décadas levou a sociedade a um processo de crescente dependência nos sistemas computacionais e serviços baseados na Internet. Este cenário dinâmico e complexo implica em iniciativas cada vez mais desafiadoras na Defesa Cibernética e, embora a indústria empregue inúmeros esforços para garantir a Segurança da Informação, observa-se uma escalada na gravidade, volume e frequência de incidentes. Com ataques motivados por ganhos financeiros e políticos, utilizando cada vez mais recursos e, muitas vezes, financiados por Estados, a ocorrência de um incidente é praticamente inevitável. É preciso preparar-se para responder de forma rápida e eficiente aos ciberincidentes, porém as abordagens de segurança tradicionais estão muito mais voltadas para a prevenção do que para a resposta a incidentes.

O objetivo deste trabalho é apresentar um novo modelo de tratamento de incidentes, descrito como uma ontologia, que seja facilmente extensível e integrável com outros modelos propostos, além de habilitar inferências por modelos computacionais e simplificar o processo de transferência de informações e conhecimento dentro de um contexto de ciberdefesa colaborativa.

Dentre as contribuições deste trabalho, podemos destacar a criação da *Computer Security Incident Handling Ontology* (CSIHO), em formato OWL, com a utilização de um *dataset* criado com base num estudo de caso de um incidente com o *ransomware* WannaCry, bem como a elaboração de outra ontologia OWL, a partir do modelo VERIS, onde foi realizada a conversão e carga do *dataset* original. Para demonstrar a aplicabilidade das ontologias foram criadas consultas SPARQL baseadas em perguntas de competência em três diferentes cenários, incluindo consultas que correlacionam dados entre as ontologias.

ABSTRACT

The fast evolution of information technology in the last decades led the society to a growing process of dependency in computer systems and Internet-based services. This complex and dynamic scenario implies in more challenging cyberdefense initiatives and, although the industry applies many efforts to ensure the Information Security, considerable growth in volume, frequency and severity of incidents is still observed. With attacks motivated by financial and political gain, each time using more resources and, many times, financed by States, the occurrence of a security incident is practically inevitable. Proper preparation is needed to respond quickly and efficiently to the cyber incidents, but the traditional information security approaches are more focused on prevention than on incident response.

The objective of this work is to present a new model for incident handling, described as an ontology, which is easily extensible and integrable with other models, and enables inferences by computational models and simplifies the knowledge transfer within a collaborative cyberdefense context.

Among the contributions of this work, we can highlight the creation of the Computer Security Incident Handling Ontology (CSIHO), in OWL format, with the use of a dataset created based on a case study of an incident with the WannaCry ransomware, as well as the creation of another OWL ontology, based on the VERIS model, where the original dataset was converted and loaded into. To demonstrate the applicability of the ontologies, SPARQL queries were created based on competency questions of three different scenarios, including examples of queries that correlate data between both ontologies.

1 INTRODUÇÃO

A evolução da tecnologia da informação, nas últimas décadas, permitiu à sociedade a realização de atividades antes inviáveis ou até mesmo impensadas, tendo melhorado significativamente a eficiência em diversos campos da ciência e modificado profundamente a economia e os meios de comunicação, impactando diretamente as relações humanas, as profissões e as organizações.

Estas mudanças podem ser facilmente observadas em aspectos do cotidiano, como o uso massivo de *smartphones* de pequeno tamanho, porém imenso poder computacional, mais poderosos que computadores de mesa de poucos anos atrás, rodando aplicações que fazem uso intenso de Inteligência Artificial e Aprendizado de Máquina em seu estado da arte. Outros exemplos notáveis são o enorme volume de dados compartilhado e armazenado diariamente na Internet, os aplicativos de transporte e aluguel por temporada, as criptomoedas, como o Bitcoin, a tecnologia *blockchain* e tudo o que o mercado chama agora de "disruptivo", bem como os processos de automação industrial, com infraestrutura crítica como usinas de energia e plantas nucleares controladas totalmente por sistemas computacionais, entre muitos outros.

A discussão sobre benefícios e problemas associados a essas transformações é bastante ampla e complexa, porém é inquestionável que elas trouxeram novos desafios e novas ameaças de Segurança Cibernética para as pessoas e principalmente para as instituições. Com o significativo aumento da dependência nos sistemas computacionais e nos serviços baseados na Internet, além do crescimento exponencial dos recursos computacionais, houve um aumento proporcional na complexidade dos sistemas, e consequentemente, na sua gestão e proteção contra ameaças.

Este cenário dinâmico e complexo exige ações cada vez mais eficientes no campo de Defesa Cibernética. Abordagens colaborativas vêm sendo aplicadas neste domínio visando compartilhar conhecimento globalmente com o intuito de combater rapidamente novas ameaças. Inspirados nos conceitos da web semântica, que buscam definir padrões para a representação formal de conhecimento na web, surgiram propostas de modelos utilizando ontologias, uma das principais ferramentas para se definir conceitos e relações distribuídas que são compreensíveis tanto por humanos quanto por máquinas, e que também fundamentam este trabalho.

1.1 MOTIVAÇÃO

Um relatório da consultoria PwC Brasil (O GLOBO, 2015) indica que o investimento em Segurança da Informação no Brasil tem crescido a um ritmo anual de 30% a 40%, atingindo cifras de até US$ 8 bilhões. O mesmo artigo informa ainda que, de acordo com relatório da empresa de segurança McAfee, ataques cibernéticos causaram perda estimada na economia brasileira de R$ 15 bilhões a R$ 20 bilhões ao ano. Embora as companhias tenham aumentado o seu investimento em Segurança da Informação, o número de incidentes continua aumentando, como pode ser observado na Figura 1.1.

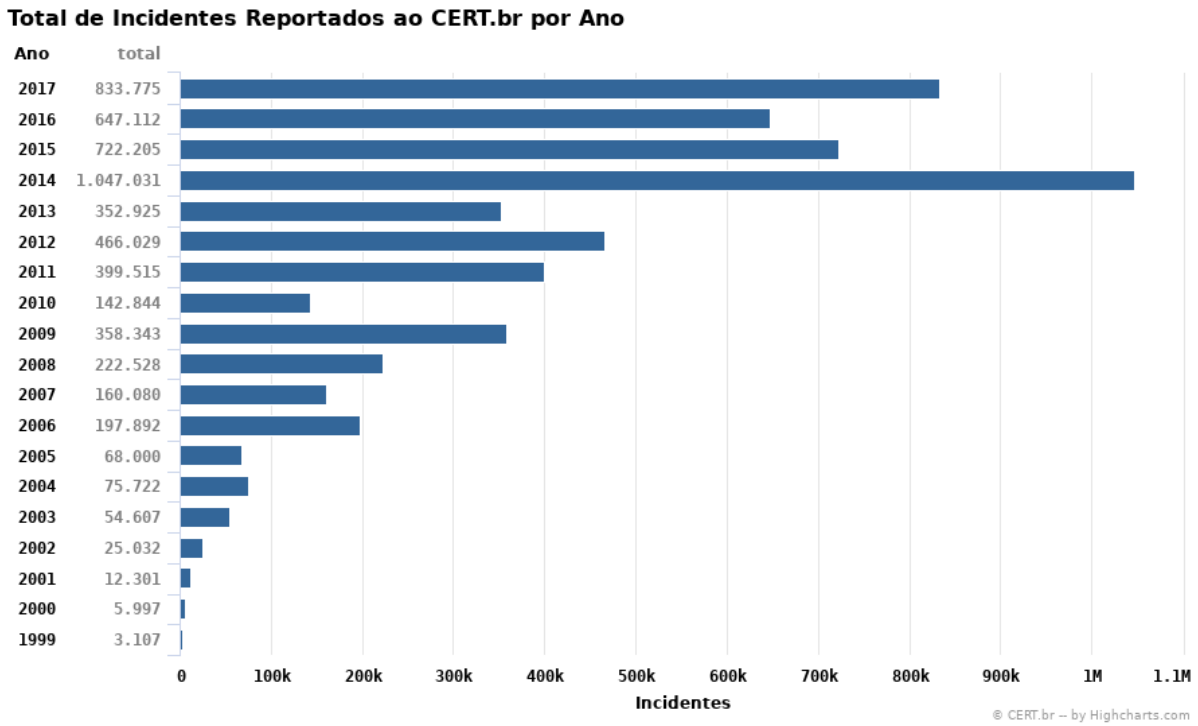

FIG. 1.1: Estatísticas dos Incidentes Reportados em CERT.br (2018)

Inúmeras notícias sobre ataques cibernéticos pelo mundo, como algumas referenciadas por este trabalho, reforçam a percepção global de que o problema continua aumentando em frequência e gravidade. Segundo Kharraz et al. (2015), a maioria dos ciberataques atuais possuem motivações políticas ou financeiras. Healey (2016) afirma ainda que muitos deles são financiados por Estados como parte de ofensivas de uma verdadeira guerra cibernética.

De acordo com a Computer World (2017), a maioria das empresas brasileiras não possui um plano de resposta a incidentes, e, ao agirem no improviso, levam mais tempo para solucionar os problemas causados por ciberataques. O relatório de segurança da F-Secure (2017) afirma que "os defensores ignoram que por trás de cada ameaça cibernética existem pessoas 100% focadas em contornar um mecanismo de prevenção para atingir

suas vítimas, e eventualmente conseguirão. Os defensores precisam, portanto, de soluções para quando seus planos (preventivos) falharem". Por outro lado, as abordagens de segurança tradicionais estão muito mais voltadas para a prevenção do que para a resposta a incidentes (BASKERVILLE et al., 2014).

Ainda que o "mercado" de Segurança Cibernética ofereça diversas soluções com foco na prevenção e no combate às últimas ameaças, eventualmente elas falharão perante um atacante com maior proficiência, recurso ou motivação. Além disso, nenhuma solução é perfeita ou à prova de falhas; portanto é fundamental a preparação para se tratar adequadamente e tempestivamente os incidentes cibernéticos.

1.2 CARACTERIZAÇÃO DO PROBLEMA

Embora a gestão de Segurança da Informação seja um tema relativamente maduro devido ao número de padrões internacionais, diretrizes e publicações acadêmicas, a literatura é inconsistente quando descreve gestão, tratamento e resposta a incidentes (AB RAHMAN; CHOO, 2015). Parte desta inconsistência pode ser observada na Tabela 1.1, que mostra um comparativo entre as etapas dos processos de resposta a incidentes nos trabalhos revisados por Grispos (2016).

Dois estudos de caso realizados com empresas listadas na Fortune 500[1] abordam, na prática, os problemas relacionados com a falta de padronização na resposta a incidentes.

TAB. 1.1: Comparativo entre processos de resposta a incidentes (GRISPOS, 2016)

Modelos	*Preparação*	*Proteção*	*Detecção*	*Resposta Inicial*	*Triagem*	*Análise*	*Coleta de Dados*	*Resposta*	*Contenção*	*Erradicação*	*Recuperação*	*Acompanhamento*
CERT/CC Incident Response	✓	✓	✓		✓			✓				
Padrão ISO/IEC 27035	✓		✓		✓			✓	✓	✓	✓	✓
Modelo NIST 800-61	✓		✓			✓			✓	✓	✓	✓
Guia de Boas Práticas da ENISA			✓		✓	✓		✓				
SANS Incident Response	✓		✓						✓	✓	✓	✓
Mandia, Prosise e Pepe (2003)	✓		✓	✓	✓	✓	✓				✓	✓
Mitropoulos, Patsos e Douligeris (2006)	✓		✓						✓	✓	✓	✓
Werlinger e Botta (2007)			✓			✓		✓				
Ryba et al. (2009)	✓											✓
Vangelos (2011)	✓		✓						✓	✓	✓	✓

[1]Lista compilada e publicada anualmente pela revista Fortune com a classificação das 500 maiores corporações em todo o mundo, conforme sua receita declarada.

Grispos (2016) identificou falhas no tocante à documentação de incidentes dentro do processo de tratamento de incidentes da companhia. Entre 188 registros, somente um deles possuía todos os campos preenchidos. A prática geral entre os analistas era obter informações com foco principalmente na erradicação e recuperação de um incidente, e não necessariamente garantir a retenção de informações relevantes para consulta e aprendizado futuro. Devido à ausência de uma taxonomia clara e objetiva, foi observado também um alto índice de incidentes com classificação inadequada ou inconsistente, o que gerava desvios nas estatísticas do processo. Como resultado, a empresa tinha dificuldade para identificar tendências e definir estratégias de melhoria.

Moreira et al. (2017) relataram, em um estudo de caso sobre a resposta ao incidente com o *ransomware* WannaCry, problemas similares aos observados por Grispos (2016). O grande potencial destrutivo do *ransomware* e sua rápida propagação exigiam ações energéticas e tempestivas. Esta noção de urgência aliada à ausência de uma ferramenta adequada para o acompanhamento de incidentes e a falta de processos padronizados levou a uma documentação errática e desestruturada do incidente. A geração de um relatório *post mortem* levou dias para ser concluída, demandando horas preciosas de praticamente todos os analistas envolvidos, em longas reuniões, com a consulta a informações de fontes diversas como e-mail, anotações pessoais, fotos de quadros brancos, papéis, entre outras. Estes fatos levaram os autores a inferirem que muitos incidentes sequer são registrados.

Em ambos os casos, a adoção de um modelo estruturado para o tratamento de incidentes proveria consistência na documentação bem como clareza nos requisitos, permitindo maior foco e eficiência nos processos operacionais. A padronização também facilitaria a aplicação de algoritmos para inferência de conhecimento, enriquecendo o onipresente e frequentemente ignorado processo de "lições aprendidas", e habilitando a interoperabilidade dos dados num cenário de defesa colaborativa.

Outro problema recorrente no processo de tratamento de incidentes, que requer o uso de processos e ferramentas adequados, é a falta de visibilidade plena do ambiente defendido. Este quesito é particularmente afetado por dois aspectos: é extremamente desafiador manter um inventário atualizado e fidedigno de todos os ativos da rede, especialmente em infraestruturas grandes ou complexas; neste mesmo cenário, o grande volume de registros (*logs*) e transações dificulta sensivelmente o correlacionamento destes dados e a extração de conhecimento.

1.3 OBJETIVO

O objetivo do trabalho é propor um modelo de tratamento de incidentes, descrito utilizando-se de uma ontologia, que seja facilmente extensível e integrável com outros modelos propostos. Esta ontologia deve prover fundamento para o processo de tratamento de incidentes, simplificar a transferência de informação e conhecimento dentro de um contexto de trabalho colaborativo, além de habilitar a realização de inferências lógicas com o uso da linguagem OWL.

Como resultado do desenvolvimento deste trabalho espera-se também atingir os seguintes objetivos específicos:

- Construir uma ontologia baseada no modelo proposto;
- Identificar as perguntas de competência que se pretendem responder através do uso desta ontologia;
- Implementar consultas (*queries*) que permitam obter respostas para as perguntas de competência;
- Construir uma ontologia baseada no modelo VERIS (2017) - *Vocabulary for Event Recording and Incident Sharing*;
- Demonstrar a aplicabilidade das ontologias utilizando exemplos reais do estudo de caso com o WannaCry;
- Demonstrar exemplos de integração entre as ontologias.

1.4 JUSTIFICATIVA

A etapa de "lições aprendidas" é uma das mais importantes no processo de resposta a incidentes, porém é a mais frequentemente omitida (CICHONSKI et al., 2012). A literatura demonstra também que os incidentes são normalmente mal documentados, o que agrava ainda mais o problema.

A necessidade de se documentar adequadamente os incidentes objetivando a extração de conhecimento pode ser considerada um problema clássico de representação formal de conhecimento. Nesta linha, o conceito de Web Semântica propõe abordagens para se padronizar a representação de conhecimento na Web, tornando possível o seu "entendimento" e manipulação pela máquina, habilitando a conexão lógica entre conceitos e permitindo a interoperabilidade entre sistemas. Uma das ferramentas propostas pela Web Semântica são as ontologias. A *Web Ontology Language* (ou OWL) permite descrever com grande expressividade os objetos e suas relações, inclusive de forma distribuída entre sistemas,

suportando processos decisórios. A linguagem OWL ainda oferece suporte a vários tipos de inferência, como classificação e subsunção (categorização) (SHADBOLT et al., 2006).

Tendo em vista o cenário cada vez mais dinâmico, complexo e distribuído dos ciberincidentes e a demanda por uma representação eficiente destes eventos, com foco em extração de conhecimento, as características propostas pelas ontologias no conceito de Web Semântica justificam a sua aplicabilidade no contexto do tratamento de incidentes.

1.5 METODOLOGIA DE TRABALHO

Uma das primeiras atividades realizadas consistiu em uma pesquisa de campo sobre incidentes emblemáticos. Em dezembro de 2016, com o objetivo de enriquecer a dissertação, foi realizada uma pesquisa informal aberta, com aproximadamente 350 pessoas, em quatro fóruns de discussão cujos membros são profissionais de Segurança da Informação ou de TI. A seguinte pergunta foi enviada:

"Qual incidente de segurança da informação, publicamente conhecido, você considera mais relevante ou emblemático na área?"

Treze profissionais responderam à pesquisa. Algumas respostas citaram incidentes específicos, outras, tipos de ataques atualmente comuns, citando exemplos de incidentes. Uma das respostas citou uma vulnerabilidade que afetou amplamente sistemas de computação, sem informar exemplos de incidentes específicos. Um resumo deste resultado pode ser visto na Tabela 1.2 e os incidentes mais relevantes são discutidos no capítulo 2.

Foi realizada, em seguida, uma extensa revisão bibliográfica de publicações relacionadas aos temas estudados, encontrada no capítulo 4 da dissertação.

Em 12/05/2017, a irrupção do *ransomware* WannaCry ocorreu em escala mundial. A observação participante pelo autor em ações de resposta ao incidente motivou a escrita de um artigo com estudo do caso, que foi publicado no XII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (MOREIRA et al., 2017).

O estudo e modelo desenvolvidos evoluíram para a proposta de uma ontologia, o que culminou em uma nova revisão bibliográfica sobre tratamento de incidentes, ontologias e metodologias de desenvolvimento de ontologias, bem como a revisão de conceitos e ferramentas relacionadas. Foi desenvolvido um metamodelo inicial, definidos os axiomas e finalmente modelada uma nova ontologia "do zero". Em paralelo, devido ao seu representativo *dataset* com mais de sete mil registros de incidentes, foi estudado o *framework* VERIS (2017) - *Vocabulary for Event Recording and Incident Sharing*, um modelo colaborativo para registro de incidentes cibernéticos amplamente utilizado pela comunidade

TAB. 1.2: Pesquisa sobre incidentes

Ameaça/Incidente	**Nº cit.**	**Descrição/Tipo**	**Exemplos de incidentes citados**
Ransomware	5	Sequestro de dados	Ataques ao metrô de Sâo Francisco e hospitais Hollywood Presbiterian e Kansas Heart, nos EUA, sequestro de bancos de dados de prefeituras e autarquias no Brasil
Stuxnet	3	Ataque direcionado ao Irã, com fins políticos/militares	n/a
Mirai	2	Negação de serviço distribuído utilizando dispositivos embarcados	Dyn DNS
Ataque à Saudi Aramco (Shamoon)	1	Ataque direcionado, com fins políticos	n/a
Ataque à infraestrutura de energia na Ucrânia	1	Ataque direcionado, com fins políticos	n/a
Vazamento de dados do site Ashley Madison	1	Ataque direcionado	n/a
Vazamento de dados da Sony	1	Ataque direcionado, com fins políticos	n/a
Heartbleed	1	Vulnerabilidade no OpenSSL	Nenhum exemplo citado
Total de respostas	**13**	**Universo de aproximadamente 350 pessoas**	

de Defesa Cibernética. As primeiras dificuldades em se "explorar" o conjunto de dados, codificado originalmente no formato JSON, posteriormente revelaram um potencial de melhoria do modelo com a sua extensão para uma ontologia, cuja abordagem está fortemente ligada à filosofia de compartilhamento de dados, tornando assim o modelo mais universal e acessível. Estes experimentos culminaram no desenvolvimento de uma segunda ontologia, criada de forma sistemática a partir do *framework* VERIS e transformada em *knowledge base* através da conversão de seu conjunto de dados original.

1.6 ORGANIZAÇÃO DA DISSERTAÇÃO

A dissertação está organizada da seguinte forma: após esta introdução que descreve contexto, motivação, caracterização do problema, objetivo, justificativa e metodologia, os capítulos 2 e 3 trazem o referencial teórico necessário para o pleno entendimento do trabalho, apresentando conceitos de Segurança da Informação e Defesa Cibernética, bem como Representação de Conhecimento e Web Semântica. No capítulo 4, é abordada a revisão bibliográfica dos trabalhos relacionados. O Capítulo 5 apresenta o modelo proposto e as metodologias utilizadas. Já no Capítulo 6, por sua vez, são apresentados os experimentos e resultados na avaliação do modelo proposto. E, finalmente, o Capítulo 7 discute as conclusões e trabalhos futuros.

2 INCIDENTES CIBERNÉTICOS

Para possibilitar o pleno entendimento dos problemas relacionados aos incidentes de segurança no contexto apresentado pelo trabalho, este capítulo aborda os conceitos fundamentais de Segurança da Informação e Defesa Cibernética.

2.1 SEGURANÇA DA INFORMAÇÃO

A norma ISO/IEC 27001:2005 (2005) define Segurança da Informação como a **preservação da confidencialidade, integridade e disponibilidade da informação**. Confidencialidade consiste na propriedade de que a informação não seja divulgada ou disponibilizada para indivíduos, entidades ou processos não autorizados, ou seja, busca preservar e restringir o seu acesso e divulgação, incluindo meios de proteger a sua privacidade e informação de propriedade. Integridade é a propriedade de salvaguardar a exatidão e completude da informação, isto é, protegê-la de modificação ou destruição não autorizada. Já a disponibilidade é a propriedade de estar "acessível e utilizável" assim que demandada por um agente autorizado, o que significa garantir acesso confiável, apropriado e tempestivo à informação para entidades autorizadas (ISO/IEC 27001:2005, 2005). Estes três termos são considerados os pilares da Segurança da Informação (HARRIS, 2013).

Outras propriedades como autenticidade, responsabilidade, não repúdio e confiabilidade também estão comumente associadas à Segurança da Informação (ISO/IEC 27001:2005, 2005). Os termos possuem definições e propósitos muito similares neste contexto. São necessários mecanismos para garantir que algo é de fato aquilo que diz ou aparenta ser (autenticidade), para identificar indivíduos, grupos ou organizações responsáveis pela salvaguarda da informação e ações relacionadas (responsabilização ou *accountability*), bem como garantir que uma vez identificado o autor de uma ação, como o envio de uma mensagem, por exemplo, o seu emitente não possa negá-la (não repúdio ou irretratabilidade). Finalmente, o sucesso em prover processos e controles que garantam as características descritas anteriormente aumenta o nível de **confiabilidade** da informação.

Várias organizações, com ou sem fins lucrativos, desenvolveram as suas próprias abordagens para a gestão, controle, gerenciamento de processos e desenvolvimento de Segurança da Informação (HARRIS, 2013). É muito comum na literatura relacionada encon-

trar referências a boas práticas, padrões, diretivas e ao termo *framework*. Pela definição clássica de engenharia de *software*, um *framework* é "um conjunto integrado de artefatos de *software* (como classes, objetos e componentes) que colaboram para fornecer uma arquitetura reutilizável para uma família de aplicações relacionadas" (SOMMERVILLE, 2011), no entanto, no contexto de Segurança da Informação, o conceito é usado de forma mais abrangente. Ainda segundo Sommerville (2011), *frameworks* são também modelos gerais de processos a partir de uma perspectiva de sua arquitetura, ou uma estrutura genérica estendida para se criar uma aplicação ou subsistema mais específico. Ab Rahman e Choo (2015) observam que os termos "modelo" e "*framework*" são utilizados de forma indiferente na produção de trabalhos acadêmicos.

Frameworks de Segurança da Informação muitas vezes estão relacionados a Governança, Risco e Conformidade (GRC). A ISACA (2012) define GRC como um termo "guarda chuva", em uso crescente, que cobre três áreas progressivamente alinhadas e integradas com o intuito de melhorar a performance corporativa para entregar as necessidades das partes interessadas (*stakeholders*). A Governança busca garantir o exercício da autoridade, do controle e da gestão. A Gestão de Risco lida com o controle da exposição a perdas e danos. A Conformidade, por sua vez, está preocupada com a aderência às regras internas e externas (ex: leis e regulamentações).

2.2 DEFESA CIBERNÉTICA

Pela definição do Ministério da Defesa brasileiro (BARROS et al., 2011), Defesa Cibernética consiste no "conjunto de ações defensivas, exploratórias e ofensivas, no contexto de um planejamento militar, realizadas no espaço cibernético, com as finalidades de proteger os nossos sistemas de informação, obter dados para a produção de conhecimento de inteligência e causar prejuízos aos sistemas de informação do oponente. No contexto do preparo e emprego operacional, tais ações caracterizam a Guerra Cibernética."

Segundo a OTAN (2017), Defesa Cibernética é "uma medida pró-ativa para detectar ou obter informações sobre uma invasão cibernética, ataque cibernético ou operação cibernética iminente ou para determinar a origem de uma operação que envolve o lançamento de uma contra-operação preventiva ou preemptiva contra o atacante de origem". Uma vez que a atividade de detecção de violação (invasão, ataque ou ações relacionadas) está inserida no processo de tratamento de incidentes de segurança da informação, pode-se considerar que **este processo é fundamental nas iniciativas de Defesa Cibernética**.

2.3 RISCO, VULNERABILIDADE, AMEAÇA E CONTRAMEDIDA

A ISO/IEC 27001:2005 (2005) define o termo **ativo** (*asset*) como "qualquer coisa que possua valor para a organização". Um ativo pode ser de valor tangível, como por exemplo um projeto, ou intangível, como a reputação de uma empresa ou indivíduo. A informação é, portanto, considerada um ativo, cujo valor depende do seu conteúdo e contexto.

O conceito de **risco** consiste, normalmente, na relação entre a perda de valor potencial, caso um ativo seja afetado, e a probabilidade deste ativo ser comprometido. Deve ser mensurável e, portanto, representável numericamente. Uma **vulnerabilidade** é a ausência ou fraqueza de um controle, que pode aumentar a exposição de um ativo ao risco. Uma **ameaça** consiste em qualquer possibilidade de se explorar uma vulnerabilidade, intencionalmente ou acidentalmente, e causar dano a um ativo. **Agentes de ameaça** são indivíduos ou organizações dispostos a explorar as vulnerabilidades com algum propósito. Finalmente, **contramedida, controle ou salvaguarda**, são as ações administrativas, técnicas ou físicas que tratam as vulnerabilidades e portanto mitigam o risco. A Figura 2.1 mostra a relação entre estes conceitos.

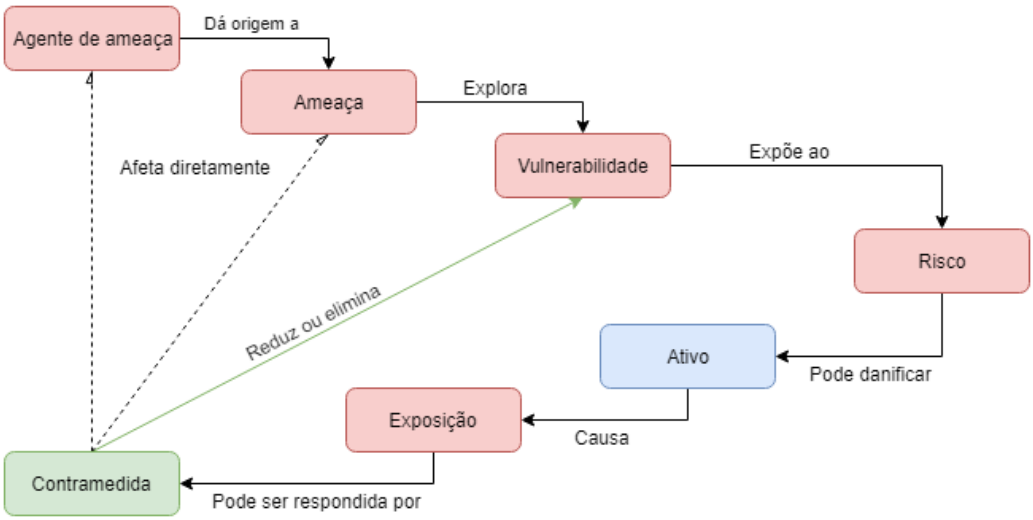

FIG. 2.1: Relação entre ameaça, risco e contramedida. Adaptado de Harris (2013)

2.4 MODELAGEM DE AMEAÇAS (*THREAT MODELING*)

Modelagem de ameaças (*Threat Modeling*) é uma abordagem estruturada para analisar a segurança de aplicações e sistemas que é parte da fase de design do Ciclo de Desenvolvimento de *Software* – *Software Development Life Cycle* ou SDLC (HARRIS, 2013). Ela

consiste em identificar, quantificar e endereçar os riscos associados a uma aplicação, e é dividida nas fases: avaliação do escopo, identificação dos agentes de ameaça e possíveis ataques, entendimento das contramedidas já aplicadas, identificação das vulnerabilidades, priorização dos riscos, identificação de contramedidas para as vulnerabilidades identificadas. Alguns exemplos de ataques em potencial incluem:

- Engenharia Social: o atacante usa interação social para obter informações sobre uma pessoa, organização ou sistema, normalmente utilizando combinações de outras técnicas, bem como conversas pessoais ou ligações telefônicas se passando por uma pessoal real ou fictícia;
- Pretexto: envolve a criação de um cenário imaginário para envolver a vítima, de maneira que ela tome atitudes que não tomaria em circunstâncias normais;
- *Phishing*: um tipo de engenharia social que usa email ou sites maliciosos para solicitar informações pessoais, se passando como uma organização confiável.
- *Vishing* (*Phone Phishing*): Usa um sistema de voz interativo para se fazer passar por uma instituição. Normalmente a vítima é levada a ligar para o número após receber um phishing, e fornece informações pessoais de "validação".
- *Baiting*: consiste em deixar uma isca para a vítima, um *pen drive* ou link para um componente malicioso, por exemplo;
- Tailgating: consiste em esperar que uma pessoa autorizada acesse (fisicamente) uma área restrita e simplesmente acompanhar sua entrada. Muitas pessoas seguram gentilmente a porta para o atacante.

2.5 INTELIGÊNCIA DE AMEAÇAS (*THREAT INTELLIGENCE*)

Segundo Mavroeidis e Bromander (2017) a inteligência de ameaças refere-se a tarefa de reunir conhecimento, baseado em evidências, sobre uma ameaça ou risco emergentes para os ativos, de maneira a utilizá-lo para otimizar as ações de resposta a esta ameaça. As evidências obtidas incluem contexto, técnicas, mecanismos, indicadores, implicações e advertências. O alto volume de informações sobre ameaças relatadas e compartilhadas entre equipes de segurança torna difícil a sua absorção e correlação com o conhecimento armazenado existente. Como resultado, fornecedores nesta "área de negócio" estão cada vez mais buscando formas de automatizar este processo e fazer com que a análise de ameaças torne-se uma tarefa viável.

Mavroeidis e Bromander (2017) afirmam que um time de resposta a incidentes com baixas habilidades e maturidade eventualmente será capaz de detectar ataques com obser-

vações de baixo nível técnico da rede, sem necessariamente entender o significado dessas observações. Por outro lado, assumem que um time com alta habilidade e maturidade será capaz de interpretar as observações técnicas da rede, no sentido de identificar o tipo de ataque, os métodos utilizados e possivelmente o atacante.

Dentre as iniciativas de Inteligência de Ameaças pode-se destacar o STIX, *Structured Threat Information Expression* (OASIS, 2017), uma formalização de conceitos, em XML, que permite o compartilhamento de informações sobre inteligência de ameaças cibernéticas entre organizações com o objetivo de realizar análises de forma colaborativa. Segundo Syed et al. (2016), é considerado o esforço mais abrangente para unificar o compartilhamento de informações de segurança cibernética e permite extensões através da incorporação de vocabulários de outros padrões.

Conforme referências apresentadas nesta dissertação, muitas organizações ainda têm problemas com seus processos de tratamento de incidentes. Pode-se concluir que, ainda que possuam pessoal altamente qualificado, as organizações que ainda enfrentam desafios nos seus processos de resposta a incidentes, especialmente no tocante a documentação e avaliação de lições aprendidas, dificilmente terão um nível de maturidade suficiente em seus processos que permita realizar adequadamente a Inteligência de Ameaças, o que corrobora a importância desta proposta.

2.6 TRATAMENTO DE INCIDENTES

No contexto de tratamento de incidentes duas definições se destacam como as mais relevantes: "evento" e "incidente" de segurança da informação. A norma ISO/IEC 27035:2011 (2011) define um **evento de segurança da informação** como a "identificação de uma ocorrência em um sistema, serviço ou estado de rede, indicando uma possível violação na política de segurança da informação e seus controles, ou uma situação previamente desconhecida que pode ser relevante no âmbito da segurança da informação". Este é **o conceito mais importante no contexto de tratamento de incidentes e defesa cibernética**, uma vez que sem a efetiva detecção dos eventos de segurança é improvável que um incidente seja identificado e respondido no tempo adequado.

Já **incidente de segurança da informação** consiste em "um ou mais eventos de segurança da informação, indesejados ou inesperados, que possuem probabilidade significativa de comprometer as operações do negócio e ameaçar a segurança da informação"; observa ainda que "a ocorrência de um evento de segurança da informação não necessariamente indica que uma tentativa (de violação) foi bem sucedida ou que impactou a

confidencialidade, integridade ou disponibilidade. Isto é, nem todos os eventos de segurança da informação são classificados como incidentes" (ISO/IEC 27035:2011, 2011).

Uma extensa revisão bibliográfica de publicações em inglês relacionadas ao tratamento de incidentes de Segurança da Informação foi realizada por Ab Rahman e Choo (2015), a mais abrangente sobre o assunto encontrada na literatura, de acordo com os autores.

Segundo Ab Rahman e Choo (2015), os processos de tratamento e resposta a incidentes são um subconjunto do macroprocesso de gestão de incidentes, que também inclui o tratamento de vulnerabilidades, tratamento de artefatos, gestão de eventos, anúncios e alertas (Figura 2.2).

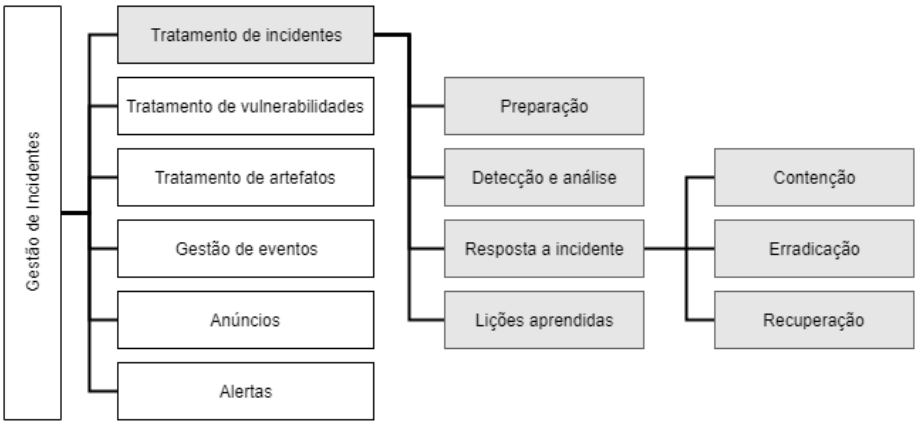

FIG. 2.2: Macroprocesso de gestão de incidentes. Adaptado de Ab Rahman e Choo (2015)

De maneira geral, o cerne de um processo de tratamento de incidentes de Segurança da Informação consiste nas seguintes etapas:

(1) Um evento, ou conjunto de eventos, é detectado, onde há suspeita de violação da confidencialidade, integridade ou disponibilidade;

(2) A situação é investigada e caso seja confirmada a violação caracteriza-se então a ocorrência de um incidente;

(3) Aplicam-se ações (imediatas) para tentar interromper, ou conter, a violação, ainda que não se conheçam todos os detalhes sobre a sua origem, extensão ou gravidade;

(4) Aplicam-se ações para evitar a recorrência da violação na situação investigada e também no contexto global. Isto requer um conhecimento mais amplo sobre a origem da violação;

(5) Busca-se restaurar o ambiente para o status anterior à ocorrência da violação.

Sabe-se que uma violação de confidencialidade, integridade ou disponibilidade tem potencial de gerar dano. Com base neste cenário pode-se postular que:

(i) Quanto maior o nível de conhecimento sobre a violação, maior será a efetividade das ações de resposta. Logo, é desejável maximizar o conhecimento sobre causas, consequências e origem do incidente desde o momento da sua confirmação;

(ii) Quanto menor o tempo desde a detecção do evento até a total recuperação do ambiente, menor a probabilidade de dano. Logo, é desejável minimizar o tempo entre a primeira e a última etapa;

(iii) Quanto maior o tempo entre a efetiva ocorrência de um evento e a sua detecção, maior será a probabilidade de dano. Logo, é desejável minimizar o tempo entre a ocorrência e a detecção de um evento;

2.7 TAXONOMIA ADOTADA

Com base na revisão da literatura e inspirada no ciclo de vida de reposta a incidentes do padrão NIST SP 800-61 (CICHONSKI et al., 2012) a seguinte taxonomia foi adotada para os macroprocessos de tratamento de incidentes: PREPARAÇÃO, IDENTIFICAÇÃO, RESPOSTA e ACOMPANHAMENTO.

As etapas 1 e 2 da generalização apresentada na seção anterior correspondem ao macroprocesso IDENTIFICAÇÃO. O macroprocesso RESPOSTA possui os processos de CONTENÇÃO, ERRADICAÇÃO e RECUPERAÇÃO, que correspondem, respectivamente, às etapas 3, 4 e 5 da generalização, conforme apresentado na Figura 2.3.

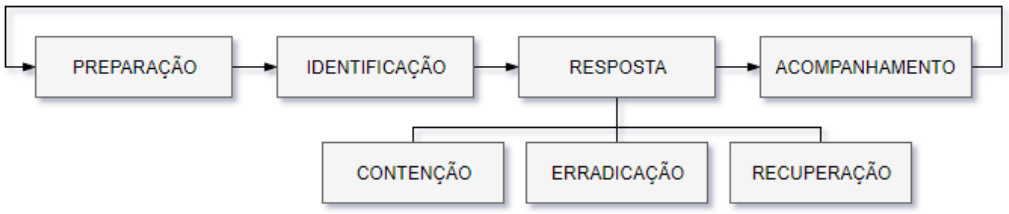

FIG. 2.3: Taxonomia adotada nas etapas do macroprocesso de tratamento de incidentes

O macroprocesso PREPARAÇÃO consiste em prover todos os recursos necessários para a perfeita execução dos demais macroprocessos, ou seja: pessoal, processos, ferramentas, bem como sua manutenção e configuração, etc. Este macroprocesso não faz parte do escopo deste trabalho.

O macroprocesso ACOMPANHAMENTO consiste em todos os processos relativos ao acompanhamento geral e registro das etapas que constituem o tratamento de incidentes, bem como na extração de conhecimento relacionado a estas etapas, seja sobre os processos em si ou sobre as particularidades de cada evento, incidente e violação tratada. Além disso, essa etapa trata do processo comumente conhecido como "lições aprendidas".

Conforme pode ser observado na Figura 2.3, o macroprocesso ACOMPANHAMENTO retroalimenta o macroprocesso PREPARAÇÃO, fechando um ciclo. Isto significa que **o conhecimento extraído no ACOMPANHAMENTO subsidia decisões em toda a cadeia de macroprocessos nas iterações futuras**.

A detecção e análise de eventos de Segurança da Informação é um dos aspectos mais críticos de todo o processo de tratamento de incidentes.

2.8 EXEMPLOS DE INCIDENTES EMBLEMÁTICOS

Inspirada pela pesquisa sobre incidentes referenciada na introdução, esta seção apresenta uma revisão de incidentes relevantes no contexto deste trabalho, com o intuito de aumentar o entendimento sobre o problema apresentado.

2.8.1 WANNACRY E PETYA (2017)

Tratam-se de dois ataques com características comuns que utilizaram software malicioso do tipo *ransomware*, ou "sequestro de dados". Pela definição da empresa de segurança Trend Micro (2016), *ransomware* é um tipo de *malware* que impede ou limita o usuário de acessar seu sistema, bloqueando a tela do sistema ou seus arquivos, até que um "resgate" seja pago. Famílias mais modernas do *malware*, conhecidas como *crypto-ransomware*, criptografam certos tipos de arquivos em sistemas infectados e forçam o usuário a pagar um "resgate" para obter a chave utilizada na criptografia (TREND MICRO, 2016). A motivação principal do *ransomware* é o retorno financeiro, portanto este tipo de ameaça normalmente utiliza mecanismos para maximizar o número de máquinas infectadas.

Diversos incidentes relacionados à infecção por *ransomware* vieram a público nos últimos anos e o WannaCry é, possivelmente, um dos mais emblemáticos, devido a suas características peculiares (MOREIRA et al., 2017). Detectado em maio de 2017, o incidente é visto como um dos maiores da atualidade (REUTERS, 2017), tendo infectado mais de 200 mil computadores em 150 países e interrompido a operação de diversos hospitais e empresas. O *ransomware* utilizou código roubado da NSA, e trouxe polêmica sobre o tema de agências de defesa armazenarem código malicioso de vulnerabilidades não di-

vulgadas. Ele também levou a Microsoft a desenvolver, extraordinariamente, atualizações para o Windows XP, sistema operacional cujo suporte foi descontinuado em 8 de abril de 2014. A descoberta de um mecanismo que permitia a desativação do *ransomware* em sua versão original elevou um jovem pesquisador ao status de "herói", pois permitiu a contenção global do ataque inicial. Apesar de todo o impacto e características bem particulares, o *software* malicioso possui baixa complexidade, reaproveita código de ameaças anteriores e não é considerado inovador (MOREIRA et al., 2017).

O *ransomware* Petya utiliza o mesmo *exploit* roubado da NSA, o *Eternal Blue*, e é considerado um sucessor do WannaCry. Atacada em junho de 2016 por uma variante do Petya, a Maersk, auto intitulada a maior empresa de transporte de *containers* do mundo, pode ser uma das maiores vítimas deste *ransomware*. A companhia alega ter sofrido um prejuízo aproximado de US$ 300 milhões com interrupções do negócio, tendo sido forçada a praticamente reconstruir toda a sua infraestrutura de TI. Para livrar-se da variante "NonPetya" a Maersk precisou reinstalar 4.000 servidores, 45.000 mil estações de trabalho e 2.500 aplicações (OSBORNE, 2018).

Outros casos relevantes de ataques com *ransomware* incluem: ataque à Agência Municipal de Transportes (MTA, na sigla em inglês) de São Francisco, CA, nos EUA (GIBBS, 2016); Hollywood Presbyterian Medical Center, em Los Angeles, CA, nos EUA (WINTON, 2016); Kansas Heart Hospital, em Wichita, KS, EUA (SMITH, 2016); Prefeitura do Município de Pratânia, SP (TECNOBLOG, 2015); Sequestro de computadores da Agência Nacional de Telecomunicações (Anatel) pelo grupo "hackativista" *Anonymous* (TECMUNDO, 2016).

2.8.2 ATAQUE À INFRAESTRUTURA CRÍTICA NA UCRÂNIA (2015)

Segundo Lee et al. (2016), tal ataque foi o primeiro "apagão", publicamente conhecido, causado por um ciberataque. Perpetrado em 23/12/2015, o ataque afetou três companhias elétricas, impactando 225.000 clientes. Outras três organizações do setor de infraestrutura crítica foram comprometidas, porém não tiveram impacto operacional. Sistemas de monitoração foram comprometidos para que não fossem detectados os cortes de energia, em seguida, atuadores que controlam disjuntores foram utilizados para desligar a energia em diversos pontos. Para dificultar ainda mais a detecção, foi lançado um ataque de negação de serviço nos centros de atendimento telefônico, de maneira a prevenir que clientes reportassem o problema.

Não se sabe exatamente como a rede foi invadida, porém dois *malwares* foram iden-

tificados nos sistemas: variantes do BlackEnergy e o Killdisk. O primeiro é um *trojan* que abre um *backdoor* e possui natureza modular, permitindo o download e instalação de funcionalidades adicionais. O segundo sobrescreve arquivos essenciais do sistema, inclusive o *Master Boot Record* (MBR), causando o travamento do Sistema Operacional e inviabilizando a reinicialização da máquina.

Lee et al. (2016) afirma que o *malware* BlackEnergy possivelmente foi utilizado para acessar remotamente máquinas de automação que controlavam as estações de distribuição elétrica, permitindo aos atacantes desligá-las como se fossem operadores autorizados. O mesmo mecanismo pode ter sido usado para disseminar o Killdisk, e assim prolongar ainda mais os efeitos do ataque, dificultando a retomada de controle da rede.

O caso do ciberataque à Ucrânia é muitas vezes associado, de forma simplista, somente ao *malware* BlackEnergy, o que é tecnicamente incorreto (LEE et al., 2016). Trata-se de um ataque multifacetado e extremamente bem orquestrado, onde várias abordagens e ferramentas diferentes foram utilizadas, sendo o BlackEnergy apenas uma delas.

2.8.3 VAZAMENTO DE DADOS DO SITE ASHLEY MADISON (2015)

O site Ashley Madison é uma espécie de rede social paga voltada para relacionamentos extraconjugais, ou seja, possui um modelo de negócio onde o sigilo é uma peça chave para sua sobrevivência.

Em julho de 2015, foi realizado um ataque bem sucedido ao site e foram divulgados publicamente detalhes sobre 32 milhões de membros. Autointitulado "The Impact Team", o grupo atacante alegou duas motivações: criticaram a missão central da empresa de organizar encontros entre pessoas casadas e atacaram as práticas da empresa, em particular a exigência de que os usuários paguem US$ 19 pelo privilégio de excluir todos os seus dados do site (FORTUNE, 2015).

2.8.4 VAZAMENTO DE DADOS DA SONY PICTURES (2014)

Em 21 de novembro de 2014, vários executivos da Sony Pictures receberam mensagens de um grupo chamado "God'sApstls", ameaçando um ataque à Sony caso não houvesse "compensação financeira". Em 24 de novembro de 2014, diversos computadores da Sony Pictures tiveram seus discos rígidos apagados e exibiram mensagens de alerta do grupo "Guardians of Peace", informando sobre alguns dados confidenciais roubados. Nas semanas seguintes ao ataque, pouco a pouco o grupo liberou na Internet diversos conteúdos confidenciais da companhia, como filmes ainda não lançados, e-mails internos de executi-

vos, relatórios financeiros, contratos de filmagem, informações pessoais de celebridades e registros médicos de funcionários. Esta estratégia de vazamento manteve o caso na mídia por várias semanas (HAGGARD; LINDSAY, 2015).

O grupo continuou enviando e-mails para funcionários da Sony com diversos tipos de ameaças, inclusive físicas e, em 8 de dezembro, fez demandas explícitas à Sony para que não lançasse o filme "A entrevista", a primeira associação do ataque com a Coreia do Norte (o filme é uma comédia sobre uma tentativa de assassinato ao ditador norte coreano Kim Jong-un pelos EUA).

O ataque culminou numa reação sem precedentes do governo dos EUA, resultando na primeira atribuição de um ciberataque a uma nação feita por um presidente norte americano, bem como as respectivas medidas de retaliação (HAGGARD; LINDSAY, 2015).

2.8.5 SHAMOON – ATAQUE À SAUDI ARAMCO (2012)

A petrolífera saudita Saudi Aramco é considerada pela Fortune (2016) a companhia de maior valor do mundo, responsável pela produção de aproximadamente 9,5 milhões de barris de óleo por dia, o que corresponde a quase 10% da demanda mundial de petróleo (RASHID, 2015).

O ataque à empresa, em 15/08/2012, durou apenas algumas horas mas destruiu parcialmente ou totalmente os dados dos discos rígidos de 35.000 computadores da companhia (RASHID, 2015). A equipe de TI precisou desconectar todos os sistemas e centros de dados para impedir o *malware* de se propagar pela rede. Ainda segundo Rashid (2015), sistemas de pagamento foram afetados, causando quilômetros de engarrafamento de caminhões tanque que não podiam ser pagos.

Um documento vazado da NSA sugere que o ataque à Saudi Aramco foi uma retaliação a um ciberataque similar, perpetrado contra a indústria de petróleo do Irã no mesmo ano (HEALEY, 2016).

2.8.6 STUXNET – ATAQUE À INFRAESTRUTURA CRÍTICA NO IRÃ (2008)

Considerado o primeiro ciberataque realmente destrutivo (HEALEY, 2016), o Stuxnet foi detectado em meados de 2008, tendo destruído ao menos 1000 centrífugas de enriquecimento de urânio no Irã. Devido a inúmeras evidências identificadas por estudos do malware, a literatura sugere que o Stuxnet foi um ataque orquestrado por órgãos governamentais, num esforço conjunto para atrasar o programa nuclear do Irã (NOURIAN; MADNICK, 2015) (HEALEY, 2016).

Embora tenha obtido inquestionável sucesso operacional, segundo Healey (2016), é impossível afirmar se os objetivos estratégicos do ataque foram de fato plenamente atingidos. Além disso, o incidente forçou os iranianos a redobrar suas capacidades cibernéticas, ignoradas até o ataque. Como retaliação, o Irã, aparentemente, conduziu ciberataques massivos contra os EUA em 2012 "sem precedentes em escala, escopo e efetividade", visto como "o primeiro ataque digital significativo deste tipo empreendido contra a indústria americana por um adversário estrangeiro" (HEALEY, 2016).

3 REPRESENTAÇÃO DE CONHECIMENTO E WEB SEMÂNTICA

Com o crescimento exponencial dos dados publicados na Internet e nos sistemas computacionais, a habilidade de se extrair informações úteis relacionadas a um tema específico vem diminuindo rapidamente. Uma das limitações encontradas nos sistemas tradicionais de busca está na inabilidade de se reconhecer ambiguidades nos termos e de realizar consultas eficientes em ambientes com formato não controlado (JAKUS et al., 2013).

A Web Semântica se propõe justamente a contornar estas dificuldades. Ela é descrita como uma evolução da rede tradicional, que consistia em documentos legíveis apenas por humanos, para uma rede com dados e informações manipuláveis pelo computador, onde a semântica provê "significado" para os conteúdos, habilitando a conexão lógica entre conceitos, estabelecendo interoperabilidade entre sistemas e permitindo a obtenção dos dados certos dentro de um contexto de uso particular (SHADBOLT et al., 2006).

Esta seção descreve os principais conceitos que fundamentam a Web Semântica, bem como as tecnologias que a tornaram possível.

3.1 LINKED DATA

A *World Wide Web*, ou simplesmente Web, alterou radicalmente a maneira como o conhecimento é compartilhado, removendo barreiras para a publicação e o acesso à documentos em um espaço de informação global. Os *links* em hipertexto, os motores de busca e indexação de documentos, bem como a natureza genérica, aberta e extensível da Web foram recursos chave para o seu crescimento e popularização (BIZER et al., 2009).

À medida que a Web crescia, esta mesma flexibilidade que permitiu a sua ascensão apresentou-se também como um desafio para a interligação de dados e documentos. Tradicionalmente, os dados publicados na Web utilizavam formatos primitivos de codificação, sacrificando muito da sua estrutura e semântica. Na Web convencional, baseada em hipertexto, a natureza da relação entre dois documentos conectados (por um *link*) é implícita, portanto não é expressiva o suficiente para definir os tipos de ligação entre entidades (BIZER et al., 2009).

Como uma resposta a estes problemas, Tim Berners-Lee, referenciado pela literatura como o criador da *World Wide Web*, definiu em 2006 uma "extensão" da Web tradicional

que permitisse a conexão explícita de dados entre diferentes domínios de conhecimento, criando assim uma "Web de dados". O conceito chamado de "*Linked Data*", ou dados conectados, em tradução livre, consiste em um conjunto de melhores práticas para publicar e conectar dados de forma estruturada na Web (DUCHARME, 2013). Seus quatro princípios fundamentais são:

(1) Use URIs para dar nome às coisas;
(2) Use HTTP URIs, de maneira que as pessoas possam possam procurar esse nomes;
(3) Quando alguém procurar uma URI, forneça informações úteis, usando os padrões;
(4) Inclua links para outras URIs, de maneira que outras coisas possam sem descobertas.

Assim como os URLs (*Uniform Resource Locators*) são utilizados para referenciar páginas na Web, os URIs (*Uniform Resource Identifiers*) são identificadores mais genéricos, que podem ser utilizados para referenciar qualquer tipo de entidade existente, e são um dos aspectos fundamentais no conceito de *Linked Data*. Eles são descritos utilizando o mesmo esquema "http://" dos URLs, exemplo: http://www.w3.org/People/Berners-Lee/card#i.

O uso de URIs também ajudou a resolver outro problema importante: um mesmo termo pode ter definições diferentes em contextos ou domínios de conhecimento distintos, gerando ambiguidade de informações. O termo *namespace* vem sendo usado pela computação há anos como referência ao grupo de nomes utilizados para um propósito particular (DUCHARME, 2013). Como os URIs permitem especificar a origem de cada definição, eles funcionam como um limitador de *namespace*, evitando assim ambiguidades.

Assim como a "semântica" define "o significado das palavras", o uso de URIs provê um pouco de semântica ao contextualizar um termo. Os URIs e o protocolo HTTP são suplementados por uma tecnologia que é crítica para a "Web de dados", o *Resource Description Framework* (RDF), descrito a seguir.

3.2 RESOURCE DESCRIPTION FRAMEWORK (RDF)

A primeira definição formal de um padrão para a Web Semântica aconteceu em 1997, quando o *World Wide Web Consortium*, ou W3C, introduziu a especificação do *Resource Description Framework* (RDF), um modelo de dados de representação simples, porém poderoso, baseado em triplas compostas por sujeito, predicado e objeto (Figura 3.1). Da mesma maneira que a Web tradicional projetou o hipertexto a uma escala global (com o padrão HTML), a visão do RDF era prover representação de conhecimento minimalista para a Web (SHADBOLT et al., 2006).

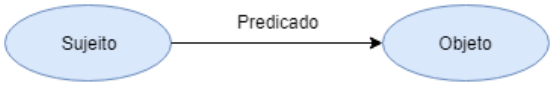

FIG. 3.1: Grafo ilustrando a tripla sujeito, predicado, objeto

Fazendo uma analogia das triplas RDF com outros modelos de representação de dados, o "sujeito" seria equivalente ao identificador de um recurso, já o "predicado" um atributo ou propriedade deste recurso, e o "objeto" um outro recurso ou uma definição de valor para a propriedade/atributo. Praticamente qualquer tipo de dado pode ser representado por um conjunto de triplas. O código 3.1 exemplifica a transformação parcial da tabela 3.1 para um conjunto de triplas, onde as linhas viram objetos, as colunas predicados e os campos valores.

TAB. 3.1: Tabela com registros exemplo

cod	**nome**	**sobrenome**	**cargo**	**dtContratacao**	**dtOrientacao**
emp1	Joana	Sancho	Gerente	29-03-2017	
emp2	Roberto	Victor	Analista	30-07-2017	18-08-2017
emp3	Armando	Santos	Engenheiro	18-11-2017	05-12-2017
emp4	Clara	Dias	Diretora	15-08-2016	

```
emp1 nome "Joana"
emp1 sobrenome "Sancho"
emp1 cargo "Gerente"
emp1 dtContratacao "29-03-2017"

emp2 nome "Roberto"
emp2 sobrenome "Victor"
emp2 cargo "Analista"
emp2 dtContratacao "30-07-2017"
emp2 dtOrientacao "18-08-2017"
```

Código 3.1: Exemplo de conversão (parcial) da tabela 3.1 em triplas

Para que não haja ambiguidade em uma informação definida por uma tripla, seu sujeito e predicado devem ser URIs, ou seja, devem pertencer à *namespaces* específicos. Isto evitará que uma pessoa ou um processo computacional confundam estes termos com outros similares, especialmente se os dados forem combinados de diferentes fontes. Os valores que não são URIs são chamados "literais" e eles normalmente possuem uma declaração de qual o tipo de dado informado, de maneira que a aplicação saiba, por exemplo, que pode realizar operações matemáticas, caso trate-se de um número, ou que deve formatá-lo de

maneira específica, caso seja uma data. Isto permite aumentar as possibilidades de uso do dado.

O objeto de uma tripla também pode ser um URI, isto significa que um recurso pode ser ao mesmo tempo o "objeto" de uma tripla e o "sujeito" de outra, o que permite conectar triplas em redes de dados, como pode ser visto na Figura 3.2. Por esta razão dados modelados em RDF são comumente chamados de grafos RDF.

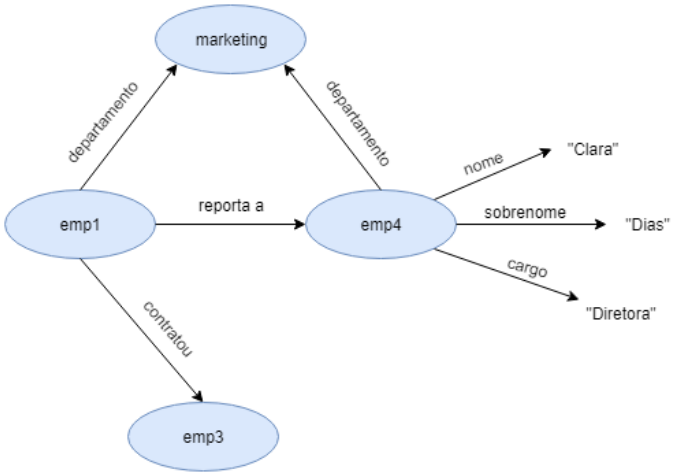

FIG. 3.2: Grafo ilustrando diversas triplas conectadas

O W3C define o RDF como "uma linguagem de uso geral para representação de informação na Web", entretanto a codificação dos dados ocorre, efetivamente, através de uma das diversas sintaxes propostas para o modelo, que serão apresentadas a seguir.

O formato mais antigo, **RDF/XML**, é parte da especificação original do RDF de 1999, e guarda uma característica extremamente importante que foi adotada em sintaxes mais modernas: a possibilidade de abreviar URIs através do uso de prefixos. O código 3.2 exemplifica o uso de prefixos (rdf, dc, v).

O formato mais simples para representação de triplas RDF é o ***N-Triples*** (na verdade trata-se de uma versão simplificada do N3, detalhado a seguir). Nele, os URIs são escritos entre os símbolos "<" e ">" e as cadeias de caracteres entre aspas duplas. Cada tripla deve estar em sua própria linha, com um ponto ao final. O *N-Triples* é utilizado apenas para fins didáticos, uma vez que não permite o uso de prefixos tampouco quebra de linhas.

O **Notation 3 (N3)** é um projeto pessoal de Tim Berners-Lee que reúne a simplicidade do *N-Triples* com a possibilidade do uso de prefixos do RDF/XML. O N3 nunca se tornou um padrão, mas serviu de inspiração para a criação do formato ***Turtle***, que se popularizou rapidamente (DUCHARME, 2013). Em 25/02/2014 o W3C transformou

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dc="http://purl.org/dc/elements/1.1/"
         xmlns:v="http://www.w3.org/2006/vcard/">

  <rdf:Description rdf:about="urn:isbn:006251587X">
    <dc:title>Weaving the Web</dc:title>
    <dc:creator rdf:resource="http://www.w3.org/People/Berners-Lee/card#i"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://www.w3.org/People/Berners-Lee/card#i">
    <v:title>Director</v:title>
  </rdf:Description>

</rdf:RDF>
```

Código 3.2: Exemplo da sintaxe RDF/XML (DUCHARME, 2013)

o *Turtle* em uma recomendação. O código 3.3 apresenta um exemplo utilizando esta sintaxe.

```
@prefix ex: <http://ds.gb.moreira.nom.br/dados_exemplo#>

ex:emp1 ex:departamento "marketing"
ex:emp1 ex:reporta_a ex:emp4
ex:emp1 ex:contratou ex:emp3

ex:emp4 ex:departamento "marketing"
ex:emp4 ex:nome "Clara"
ex:emp4 ex:sobrenome "Dias"
ex:emp4 ex:cargo "Diretora"
```

Código 3.3: Exemplo da sintaxe *Turtle* utilizando os dados da Figura 3.2

Graças ao seu poder de representação de conhecimento e à sua flexibilidade, o modelo de dados RDF vem sendo utilizado em projetos que não estão relacionados diretamente à web semântica (DUCHARME, 2013), como é o caso deste trabalho.

3.3 ARMAZENAMENTO DE DADOS RDF

A medida que o número de triplas aumenta e questões como performance, disponibilidade, integridade e confidencialidade entram em cena, o uso de simples arquivos torna-se inadequado e passa a ser necessário um mecanismo mais robusto para o armazenamento dos dados. Existem sistemas que permitem o uso de RDF com bancos de dados relacionais tradicionais, porém a melhor maneira de armazena-los é em bancos de dados otimizados para trabalhar com triplas RDF, os chamados *triplestores*.

Os *triplestores*, ou "repositórios de triplas", em tradução livre, são ferramentas especializadas no armazenamento de conteúdo em formato RDF. Alguns são focados em

prover recursos ricos para o processo de *reasoning* (inferência), outros são voltados para o armazenamento de grandes quantidades de dados, alguns operam como *plugins* e outros funcionam como *middleware* para armazenamento de triplas em bancos de dados tradicionais.

3.4 LINGUAGEM SPARQL

O SPARQL está para a Web Semântica assim como o SQL está para os bancos de dados relacionais. Seu nome é um acrônimo recursivo que significa *"SPARQL Protocol And RDF Query Language"*, ou seja, é um protocolo e linguagem que define padrões e sintaxe para a realização de consultas (*queries*) em repositórios RDF (*triplestores*).

Uma consulta SPARQL consiste em duas partes básicas: *Pattern Matching* - a descrição dos padrões, filtros e restrições dos dados pesquisados; *Solution Modifiers* - modifica os padrões de saída usando comandos como *projection, distinct, order by, limit* e *offset*.

Normalmente, inicia-se a construção da *query* pela cláusula WHERE, pois ela descreve quais triplas se deseja obter do *data set* que esta sendo consultado.

Assim como alguns formatos de serialização do RDF, o SPARQL permite definir prefixos, evitando a necessidade de se reescrever os URIs em cada parâmetro da consulta. O código 3.4 mostra um exemplo de consulta SPARQL ao código 3.1, apresentado anteriormente neste capítulo, e a Tabela 3.2 apresenta o seu resultado.

```
PREFIX ex: <http://ds.gb.moreira.nom.br/dados_exemplo#>

SELECT ?empregado ?nome ?sobrenome
WHERE {
        ?empregado ex:departamento "marketing" .
        OPTIONAL ( ?empregado ex:nome ?nome ) .
        OPTIONAL ( ?empregado ex:sobrenome ?nome ) .
}
```

Código 3.4: Exemplo de pesquisa SPARQL

TAB. 3.2: Resultado da consulta exemplo do código 3.4

empregado	nome	sobrenome
emp1		
emp4	Clara	Dias

3.5 VOCABULÁRIOS E ONTOLOGIAS

Ontologias são definições formais de vocabulários que permitem definir estruturas complexas bem como as relações entre os termos e os membros das classes que foram definidas. Elas normalmente descrevem domínios específicos, como áreas de pesquisa científica, permitindo que cientistas de diferentes instituições possam compartilhar dados mais facilmente. No contexto de sistemas e computação, as ontologias propõem-se em prover a especificação formal de conceitos e relações entre entidades dentro de um domínio de maneira interpretável (facilmente) por uma máquina (PING et al., 2010).

Em sua pesquisa sobre representação de conhecimento, Jakus et al. (2013) concluíram que a maioria dos autores concordam que a representação uniforme do conhecimento deve ser alcançada pelo uso de ontologias, e que o agrupamento de conceitos relacionados em ontologias provou ser uma maneira muito eficiente de capturar e estruturar significado em linguagens naturais.

Shadbolt et al. (2006) afirmam que ontologias são a racionalização da prática real de compartilhamento de dados, já que são um meio de se fazer um compromisso explícito de compartilhamento de conhecimento entre uma comunidade de interesse comum, ao mesmo tempo habilitando qualquer um a utilizar estas ontologias para descrever seus próprios dados, estendê-las ou reutilizar elementos livremente.

3.6 LÓGICAS DE DESCRIÇÃO (*DESCRIPTION LOGICS*)

As Lógicas de Descrição (LDs) são uma família de linguagens formais que podem ser utilizadas para representação de conhecimento em um domínio de aplicação, de maneira clara e estruturada (BAADER et al., 2017). Elas fornecem uma definição explícita de conceitos (classes) interpretados como grupos de objetos e regras em relações binárias entre objetos (Jakus et al. (2013) apud Nakabasami (2002)).

Segundo Baader et al. (2017), o domínio de conhecimento é tipicamente separado em *TBox*, o componente terminológico, e *ABox*, o componente assertivo. A combinação de ambos é designada como a base de conhecimento (*knowledge base*).

Pelo fato de agregar uma semântica baseada em lógica, as LDs permitem a realização de inferência (*reasoning*) em toda a base de conhecimento e, particularmente, no domínio capturado pela TBox. Esta capacidade, porém, apresenta um alto custo computacional e uma das áreas mais importantes de pesquisa neste contexto tem sido o compromisso entre o poder de representação da linguagem e a complexidade computacional das suas tarefas de inferência (BAADER et al., 2017).

A Tabela 3.3 apresenta alguns dos construtores usados em Lógica de Descrição. Em notação abstrata, usam-se as letras A e B para conceitos atômicos, a letra R para papéis atômicos e as letras C e D para descrições de conceito (BAADER et al., 2003).

TAB. 3.3: Sintaxe e semântica dos construtores de LD. Adaptado de Syed et al. (2016)

Nome	**Sintaxe**	**Semântica**	**Símbolo**
Top	$\top$	Δ^I	AL
Bottom	$\bot$	ϕ	AL
Intersection	C $\sqcap$ D	$C^I \cap D^I$	AL
Union	C $\sqcup$ D	$C^I \cup D^I$	U
Negation	$\neg$C	$\Delta^I \backslash D^I$	C
Value restriction	$\forall$R.C	$\{a \in \Delta^I \mid \forall b.\ (a,b) \in R^I \rightarrow b \in C^I\}$	AL
Existential quantification	$\exists$R.C	$\{a \in \Delta^I \mid \forall b.\ (a,b) \in R^I \rightarrow b \wedge C^I\}$	E
Nominal	I	$I^I \subseteq \Delta^I$ with $\lvert I^I \rvert = 1$	O
Qualified num.restriction	$\leq nR.C$	$\{a \in \Delta^I \mid \mid \{\forall b \in \Delta^I \mid (a,b) \in R^I \wedge b \in C^I\} \mid \leq n\}$	Q
Qualified num.restriction	$= nR.C$	$\{a \in \Delta^I \mid \mid \{\forall b \in \Delta^I \mid (a,b) \in R^I \wedge b \in C^I\} \mid = n\}$	Q
Qualified num.restriction	$\geq nR.C$	$\{a \in \Delta^I \mid \mid \{\forall b \in \Delta^I \mid (a,b) \in R^I \wedge b \in C^I\} \mid \geq n\}$	Q
Role Hierarchy	$R_1 \sqsubseteq R_2$	$\{(a,b) \in \Delta^I \times \Delta^I \mid (a,b) \in R_1^I \rightarrow (a,b) \in R_2^I\}$	H
Role Inverse	R^-	$\{(b,a) \in \Delta^I \times \Delta^I \mid (a,b) \in R^I\}$	I
Role Composition	$R_1 \circ R_2$	$\{(a,c) \mid \exists b.\ (a,b) \in R_1^I \wedge (b,c) \in R_2^I\}$	R

As Lógicas de Descrição têm desempenhado um papel central na Web Semântica, onde têm sido adotadas como fundamento para várias linguagens de construção de ontologias, como a OWL (BAADER et al., 2017).

3.7 WEB ONTOLOGY LANGUAGE (OWL)

O W3C introduziu a *Web Ontology Language* (abreviada OWL, por ser mais facilmente pronunciável, na língua inglesa, do que WOL) como uma linguagem para definir ontologias. Segundo DuCharme (2013) o OWL é o padrão mais importante definido pelo W3C para de fato "trazer a semântica para a web semântica". Já em sua segunda versão, a especificação estende o modelo RDF com maior expressividade nas descrições dos objetos e suas relações. O OWL permite ainda que as ontologias referenciem conceitos em diferentes sistemas, de forma distribuída, através de suas URIs.

Outra característica importante da linguagem OWL é o fato dela oferecer suporte a alguns tipos de inferência, tipicamente classificação e subsunção (categorização). Diversas ferramentas de *reasoning* estão disponíveis, como por exemplo o HermiT Reasoner (KRR, 2017), que utiliza um algoritmo estado da arte.

A linguagem OWL foi dividida em três diferentes níveis de expressividade, permitindo a implementação de ontologias de forma "escalável". São elas OWL Lite, a versão mais simples para implementação rápida e simplificada, OWL DL (*Description Logics*), a versão

com nível intermediário de robustez e finalmente a OWL *Full*, que possui o maior poder de expressividade, porém com maior custo computacional.

3.8 METODOLOGIA PARA CRIAÇÃO DE ONTOLOGIAS

A literatura descreve melhores práticas para o processo de desenvolvimento de ontologias com características bastante comuns, por isto não há um "guia definitivo." No escopo desta dissertação foi adotada uma metodologia chamada de *Ontology Development 101* (NOY; MCGUINNESS, 2000), disponível no Wiki da ferramenta Protégé. Embora se proponha a ser "um guia para o desenvolvimento da sua primeira ontologia", o documento fornece orientações bastante detalhadas sobre todo o processo de desenvolvimento, e postula que é apenas uma das possíveis abordagens.

O processo é dividido em sete etapas, detalhadas a seguir.

Etapa 1) Determinar o domínio e o escopo da ontologia. Para isto é preciso responder a perguntas como: Qual o domínio que a ontologia irá cobrir? Para quê ela será utilizada? Quem usará e manterá a ontologia?

Finalmente, o questionamento mais importante é: quais perguntas pretende-se responder com o uso desta ontologia? A metodologia chama essas perguntas específicas de "questões de competência".

Etapa 2) Reuso de ontologias existentes. Deve-se realizar uma pesquisa por ontologias com o mesmo propósito ou com temas relacionados para que se aproveite o máximo possível conceitos existentes, ou seja, não se deve reinventar a roda.

Etapa 3) Enumerar termos importantes na ontologia. Isto significa levantar todos os termos relevantes que precisam ser definidos. Inicialmente a lista pode ser extensiva, sem uma grande preocupação sobre o nível de pertinência, como uma espécie de *brain storm*.

Etapa 4) Definir as classes e sua hierarquia. Nesta etapa os termos listados são avaliados e classificados quanto a sua importância para a ontologia, e os principais termos são elencados e então relacionados entre si.

Etapa 5) Definir as propriedades das classes. Exemplo: Classe "pessoa" poderia ter as propriedades nome, sobrenome, data de nascimento, identificação, etc.

Etapa 6) Definir as "facetas" das propriedades, isto é, sua cardinalidade, domínio (qual ou quais classes possuem esta propriedade) e escopo (*range*, o tipo de dado ou valores possíveis).

Etapa 7) Popular a ontologia com os seus indivíduos, ou seja, "instanciar" os objetos.

4 REVISÃO DA LITERATURA E ESTADO DA ARTE

Este capítulo apresenta os padrões, modelos e abordagens tradicionalmente relacionados à Segurança da Informação, bem como os trabalhos acadêmicos, bases de dados e ontologias relacionadas ao tratamento de incidentes.

4.1 SEGURANÇA DA INFORMAÇÃO

Esta seção foi dividida em duas subseções. Em "padrões de mercado" são descritas as referências a normas e padrões globalmente reconhecidos e consolidados. A subseção "artigos relacionados" apresenta os estudos mais recentes na área e o seu estado da arte.

4.1.1 PADRÕES DE MERCADO

ISO/IEC 27000. A família ISO 27000 constitui um conjunto de padrões internacionais que apresentam melhores práticas da indústria para a gestão de controles de segurança, de uma maneira holística, nas organizações. As normas utilizam o ciclo PDCA – Planejar (plan), Fazer (do), Verificar (check), Atuar (act) – um processo iterativo de melhoria contínua que é comumente usado em programas de controle de qualidade (HARRIS, 2013).

Podem-se citar alguns exemplos destes padrões:

(i) ISO/IEC 27000: Visão geral e vocabulário.

(ii) ISO/IEC 27001: Requisitos para um Sistema de Gestão de Segurança da Informação. (Também existe uma certificação ISO 27001, onde uma organização neutra avalia o cumprimento dos requisitos da norma e atesta um nível de conformidade.)

(iii) ISO/IEC 27002: Código de práticas para a gestão da segurança da informação.

(iv) ISO/IEC 27005: Diretriz para gerenciamento de riscos de segurança da informação.

(v) ISO/IEC 27035: Diretriz para gerenciamento de incidentes de segurança.

***Sherwood Applied Business Security Architecture* (SABSA)**. É um modelo e metodologia "multicamada" para o desenvolvimento de arquiteturas de Segurança da Informação. Cada camada diminui o nível de abstração e aumenta o nível de detalhes,

evoluindo, por exemplo, do nível de políticas em direção à implementação prática de tecnologias e soluções. O SABSA utiliza uma abordagem de ciclo de vida, permitindo monitoração e melhoria sistemática da arquitetura ao longo do tempo (HARRIS, 2013).

COBIT (*Control Objectives for Information and Related Technologies*). Desenvolvido pela *Information Systems Audit and Control Association* (ISACA) e o IT *Governance Institute* (ITGI), o COBIT é um *framework* que define objetivos e metas de controles, e provê direcionamento para a gestão adequada de recursos de TI, garantindo que as necessidades do negócio sejam atendidas. Ele é dividido em quatro domínios: (i) Planejar e Organizar; (ii) Adquirir e Implementar; (iii) Entregar e Suportar; (iv) Monitorar e Avaliar. Cada domínio possui subcategorias com orientações específicas, totalizando 34 objetivos de controle (HARRIS, 2013).

Embora não seja focado em Segurança da Informação, o COBIT trata de temas como aquisição e manutenção de software e infraestrutura, desenvolvimento e manutenção de processos, instalação e acreditação de sistemas e também gestão de mudanças. Estes aspectos impactam diretamente na gestão da Segurança da Informação, e, por isso, o *framework* é comumente associado a este tema na literatura.

NIST SP 800-53. Ao passo em que o CobiT contém objetivos de controle usados no setor privado, o Instituto Nacional de Padrões e Tecnologia (NIST) norte americano desenvolveu a *Special Publication* (SP) 800-53, um conjunto de controles de Segurança da Informação cujo principal objetivo é proteger os sistemas federais. O *framework* é composto por controles divididos entre 18 "famílias", classificadas como técnicas, operacionais ou de gestão. CobiT e SP 800-53 possuem *checklists* de controles bastante diferentes, porém há grande interseção, já que sistemas e redes devem ser protegidos de forma similar, não importando o tipo de organização onde estão inseridos (NIST, 2013).

Embora o NIST seja uma organização não regulatória, as agências de governo precisam seguir o SP 800-53 para estarem em conformidade com a lei *Federal Information Security Management Act*, de 2002 (HARRIS, 2013).

NIST SP 800-61. Esta publicação provê diretrizes para auxiliar as organizações a estabelecer os recursos necessários para responder e tratar adequadamente os incidentes de segurança cibernética. Os tópicos abordados incluem a organização dos recursos para resposta a incidentes, o seu tratamento desde a preparação inicial até a fase de lições aprendidas, bem como os tipos específicos de incidentes. O documento também aborda como os incidentes devem ser comunicados dentro e fora das organizações e como deve ser realizado o compartilhamento de informações de inteligência entre times num cenário de trabalho colaborativo (CICHONSKI et al., 2012).

COSO. Um modelo de governança corporativo que oferece controles internos para auxiliar na redução do risco de fraude financeira. Foi desenvolvido pelo *Committee of Sponsoring Organizations of the Treadway Commission* (COSO), uma organização privada, sem fins lucrativos, criada nos EUA em 1985 e que dá nome ao padrão. O CobiT é derivado do COSO, porém com foco em gestão de TI (mais ligado ao nível operacional, enquanto o COSO é mais estratégico). A lei norte-americana Sarbanes–Oxley (*Sarbanes–Oxley Act*, normalmente abreviada em SOX ou Sarbox), de 2002, também é baseada no modelo COSO (HARRIS, 2013).

***Information Technology Infrastructure Library* (ITIL)**. Devido à crescente dependência em Tecnologia da Informação para se atender requisitos de negócio, a *Central Computer and Telecommunications Agency* do Reino Unido desenvolveu, no final dos anos 1980, um conjunto de boas práticas de gestão de serviços de TI. O ITIL é um *framework* personalizável que fornece as metas, atividades necessárias para atingi-las e valores de entrada e saída para cada processo necessário para alcançar os objetivos determinados. Embora o ITIL possua um componente que lida com segurança, seu foco principal está nos acordos de nível de serviço entre a área de TI e seus clientes, normalmente departamentos internos (HARRIS, 2013).

Assim como o CobiT, não está ligado diretamente à Segurança, porém é sempre referenciado por oferecer mecanismos de gestão que impactam positivamente a Segurança da Informação ou lhe fornece algum tipo de subsídio fundamental (ex.: controle de ativos).

***Capability Maturity Model Integration* (CMMI)**. Um modelo para melhoria de processos desenvolvido pela universidade Carnegie Mellon na década de 1980. O CMMI define cinco "níveis de maturidade", oferecendo às organizações um método para medir e comparar a sua capacidade com seus concorrentes, sua indústria e consigo mesma ao longo do tempo. O modelo está disponível com foco em quatro diferentes disciplinas: Engenharia/Desenvolvimento de produtos e serviços, Aquisição de produtos e serviços, Provimento de serviços e Desenvolvimento de pessoas (CMMI INSTITUTE, 2017).

O cerne do CMMI é desenvolver etapas estruturadas que podem ser seguidas pelas organizações, permitindo-as evoluir de nível e melhorar continuamente seus processos de segurança (até um nível de maturidade desejado, permitindo mantê-lo). Um programa de segurança contém muitos elementos e não se espera que todos estejam perfeitamente implementados em seu primeiro ano de existência. Além disso, alguns componentes realmente não podem ser colocados em prática até que algumas peças rudimentares sejam estabelecidas, como no gerenciamento de incidentes (HARRIS, 2013).

A Tabela 4.1 apresenta uma comparação entre os padrões e *frameworks* citados.

TAB. 4.1: Abordagens tradicionalmente relacionadas à Segurança da Informação

Padrões	**Características**	**Limitações**	**Foco em SI?**
SABSA	Ciclo de vida	Alto nível	**sim**
CMMI	Níveis de maturidade	Depende de uma v0	não
ITIL v3	Boas práticas em gestão de TI	Focado em serviços	não
CobIT v5	Controles para governança de TI (setor privado)	Diz o que fazer, mas não como fazer.	não
COSO	Redução de risco (fraude financeira)	Muito estratégico, pouco operacional	**sim**
NIST SP 800-53 rev.4	Diretrizes para tratamento e resposta a ciberincidentes	Foco em sistemas Federais; Alto nível	**sim**
NIST SP 800-61 rev.2	Controles para governança de TI (sistemas federais)	Não provê um *framework* para documentação de incidentes	**sim**
ISO/IEC 27000	Melhores práticas em gestão de segurança	Muito abrangente, dificulta a seleção do que é estritamente necessário	**sim**

4.1.2 ARTIGOS RELACIONADOS

Eloff e von Solms (2000) - Information Security Management: A Hierarchical Framework for Various Approaches. Segundo o autor, a literatura faz referência a diversas abordagens de Segurança da Informação (padrões, diretrizes, *frameworks*, etc), porém, identificar quais elementos devem ser adotados constitui um dilema para as organizações. O objetivo do trabalho é assistir os gestores na interpretação e na aplicação de abordagens aceitas internacionalmente na gestão de Segurança da Informação. Realiza uma pesquisa bibliográfica tentando elucidar termos e conceitos gerais de gestão de Segurança da Informação e desenha um *framework* hierárquico correlacionando estes termos e posicionando as abordagens citadas. Divide a Segurança da Informação em diversas subáreas e aponta aonde se aplica cada uma das "abordagens tradicionais" referenciadas. Estritamente teórico, não há experimentação.

Gonzalez e Sawicka (2002) - A Framework for Human Factors in Information Security. Afirma que embora os sistemas de segurança sejam bem desenhados e planejados, sempre haverá dependência do fator humano. O trabalho aplica Dinâmica de Sistemas em um exemplo hipotético, baseado em trabalho referenciado, com o objetivo de melhorar o entendimento sobre o papel do fator humano nos sistemas de Segurança

da Informação. Levanta diversas hipóteses, porém é inconclusivo. Solicita a colaboração de organizações para que compartilhem dados reais que possam aprofundar a pesquisa e torná-la mais relevante.

Rees et al. (2003) - PFIRES: A Policy Framework for Information Security. Segundo o autor, as diretrizes tradicionais para criação de políticas de segurança não conseguem acompanhar a crescente taxa de mudança de tecnologias e aplicações, e tampouco consideram como manter as políticas consistentes e alinhadas com os objetivos organizacionais. Propõe então um *framework* iterativo, de alto nível, que permite auxiliar na avaliação, planejamento, entrega e operação de políticas de segurança. Estritamente teórico, o trabalho utiliza os conceitos de ciclo de vida do desenvolvimento de produto e ciclo de vida do desenvolvimento de sistemas, claramente inspirado nos conceitos de engenharia de software e de melhoria de processos.

Veiga e Eloff (2007) - An Information Security Governance Framework. Em sua contextualização, os autores dividem as abordagens de Segurança da Informação em quatro fases: 1) estritamente técnica; 2) incorporada às estruturas organizacionais; 3) incorporada à cultura organizacional (destaca o fator humano); e finalmente a fase 4) com o desenvolvimento da governança de segurança da informação e os papéis relacionados (destaca a prevenção de riscos). Avalia então quatro diferentes abordagens de governança de Segurança de Informação: ISO/IEC 17799 e ISO/IEC 27001, PROTECT, Capability Maturity Model e Information Security Architecture (ISA), compila seus principais componentes, identificando interseções e complementações e propõe um novo *framework* com base nesta avaliação, considerando aspectos técnicos, procedurais e o fator humano. Estritamente teórico, sem experimentação ou estudo de caso.

Da Veiga e Eloff (2010) - A framework and assessment instrument for information security culture. Cita os prejuízos das organizações devido aos incidentes de segurança da informação e pesquisas que defendem que o comportamento dos funcionários deve ser considerado nas ações de proteção aos ativos de informação, ponderando que uma cultura de conscientização é tão importante quanto os controles e contramedidas técnicos. Propõe um *framework* para cultivar a cultura de Segurança da Informação numa organização e ilustra como utilizá-lo. Esta abordagem incorpora o fator do comportamento humano como ameaça à proteção dos ativos de informação. Utiliza exemplos hipotéticos e realiza um estudo empírico para validar o modelo proposto. O estudo consiste em uma pesquisa com questionário (www.surveytracker.com), aplicada em uma empresa sul-africana com aproximadamente 3000 funcionários, com posterior análise estatística dos resultados. A pesquisa foi respondida por 1085 empregados.

Baskerville et al. (2014) - Incident-centered information security: Managing a strategic balance between prevention and response. Identifica as abordagens tradicionais como universais em escopo e com fundamentos baseados em princípios de controle de qualidade, como o PDCA. Definem os paradigmas de Prevenção (centrados em controles) e Resposta (centrados em incidentes), seus contrastes e defendem que deve haver um equilíbrio entre eles. Propõe um *framework* teórico centrado em incidentes e realiza um estudo de caso em três organizações, inseridas em contextos diferentes, para avaliar o equilíbrio entre os dois paradigmas: empresa postal, transporte ferroviário e polícia. O trabalho é muito focado na questão da identificação das características de cada paradigma (prevenção/resposta), utilizando o modelo proposto muito mais como um instrumento de avaliação nos estudos de caso do que como uma ferramenta de implementação de mudanças.

NIST (2014) - *Framework* for Improving Critical Infrastructure Cybersecurity. A criação do *framework*, em colaboração com a indústria, foi motivada por uma Ordem Executiva emitida pelo presidente Obama em fevereiro de 2013. Esta Ordem Executiva apela ao desenvolvimento de um *framework* de cibersegurança voluntário que forneça uma abordagem "priorizada, flexível, repetível, baseada em desempenho e custo-efetiva" para gerenciar os riscos em processos, informações e sistemas diretamente envolvidos na prestação de serviços de infraestrutura crítica. O trabalho apresenta um *framework* que fornece uma taxonomia comum e um mecanismo para que as organizações possam: 1) Descrever sua atual postura de segurança cibernética; 2) Descrever o nível de segurança cibernética desejado; 3) Identificar e priorizar oportunidades de melhoria no contexto de um processo contínuo e repetitivo; 4) Avaliar o progresso em direção ao nível desejado; 5) Comunicar as partes interessadas (*stakeholders*) sobre os riscos de segurança cibernética.

Utiliza uma abordagem baseada em risco, e referencia diversos padrões citados anteriormente, como ISO/IEC 27001, NIST SP 800-53 e COBIT.

Nourian e Madnick (2015) - A Systems Theoretic Approach to the Security Threats in Cyber Physical Systems Applied to Stuxnet. Segundo os autores, as abordagens tradicionais de proteção adotadas em Sistemas Ciber-físicos (CPS, na sigla em inglês, sistemas que conectam elementos computacionais colaborativos com o intuito de controlar entidades físicas) consideram os componentes individuais do sistema, mas não suas relações. Propõe um *framework* que permite avaliar e melhorar a segurança dos sistemas CPS, contemplando as lacunas identificadas nas abordagens tradicionais. Utiliza análise causal baseada em STAMP (*System Theoretical Accident Model and Pro-*

cess) e demonstra a aplicabilidade do modelo através de um estudo de caso extremamente detalhado com o ataque do *malware* Stuxnet.

A Tabela 4.2 apresenta uma comparação entre os trabalhos acadêmicos citados.

TAB. 4.2: Trabalhos acadêmicos relacionados

Autores	**Paradigmas**	**Características**	**Limitações**
Eloff e von Solms	Prevenção	Terminologia e linguagem comum entre abordagens	Apenas definições, não explora os métodos
Rees et al.	Prevenção	Voltado à análise de risco	Não subsidia a resposta a incidentes
Da Veiga e Ellof (2007)	Prevenção	Análise de quatro abordagens tradicionais de SI	Não subsidia a resposta à incidentes
Da Veiga e Ellof (2010)	Prevenção	Cultura de Segurança da Informação	Não subsidia a resposta a incidentes
Baskerville et al.	Prevenção e Resposta	Define bem os paradigmas de Prevenção e Resposta	Não oferece ferramenta ou metodologia para aplicação
NIST	Prevenção	Voltado para infraestrutura crítica	Orientações de muito alto nível
Nourian e Madnick	Prevenção e Resposta	Voltado a redes industriais	Alto nível de abstração
Moreira, G.B. (este trabalho)	Resposta	Foco em tratamento de incidentes; Propõe uma ontologia;	Não foi testado em ambiente real

4.2 BASES DE DADOS SOBRE INCIDENTES

Bases de dados sobre Incidentes de Segurança da Informação são uma rica ferramenta para estudos diversos e principalmente para a extração de conhecimento relacionado a estes eventos. O próprio Departamento de Segurança Interna dos EUA (DHS, 2015) sugere a criação de um repositório de dados compartilhado entre grupos de trabalho relacionados a incidentes cibernéticos. É provável, porém, que justamente pelo seu valor, exista uma barreira na manutenção de tal repositório público, já que as consultorias especializadas querem tirar proveito das informações para benefício próprio (por exemplo, possuir estratégias exclusivas de resposta e defesa como parte de seu portfólio de serviços), as empresas vítimas dos ataques querem preservar ao máximo sua imagem e reputação e também há questões relacionadas à própria soberania nacional.

Por estas razões, existem poucas iniciativas de bases de dados públicas com informações sobre incidentes. A seguir, são apresentadas algumas delas:

Repository of Industrial Security Incidents - RISI (EXIDA LLC, 2017)

É um banco de dados de incidentes de natureza cibernética que afetaram processos de controle, automação industrial ou sistemas SCADA (*Supervisory Control and Data Acquisition*). O RISI nasceu de um projeto acadêmico, em 2001, com o objetivo de rastrear padrões e tendências nos incidentes de segurança em ambiente industrial, tendo se tornado um produto comercial em 2008, com o objetivo de prover informações relevantes para a indústria através de uma assinatura.

O RISI se propõe a obter dados de incidentes de três fontes distintas: (1) Relatórios de incidentes privados enviados por membros da organização; (2) Pesquisas por incidentes em fontes públicas de informações, realizadas por analistas do RISI; e (3) Acordos de compartilhamento de dados com parceiros estratégicos.

Embora um vídeo promocional publicado no site informe sobre a disponibilidade de uma série de detalhes e estatísticas sobre os incidentes, a base de dados disponível para pesquisa *on-line* é muito precária em informações, desatualizada (última alteração em 28 de janeiro de 2015) e não possui indicações sobre como comprar/contratar o serviço. Não está claro se o produto foi abandonado, descontinuado ou se há uma forma alternativa de se acessar os dados, porém, a falta de informações e de referências são indícios de que a iniciativa tenha fracassado.

Web-Hacking-Incident-Database WHID (WEB APPLICATION SECURITY CONSORTIUM, 2017)

O objetivo do WHID é manter uma lista de incidentes de segurança relacionados a aplicações web, fornecer informações para a análise estatística destes incidentes e servir como uma ferramenta para aumentar a conscientização sobre os problemas de segurança de aplicações web.

O site do projeto não explica exatamente como as informações são obtidas, mas há um formulário para submissão de incidentes. A página também oferece estatísticas sobre os principais métodos de ataque, fraquezas em aplicações e consequências, bem como uma ferramenta de pesquisa de incidentes.

DataLossDB (KANNAN et al., 2011)

O DataLossDB foi fundado em 2005 como um projeto de rastreamento de dados, baseado em um conceito discutido em 2001. De acordo com sua descrição, "fornece dados imparciais, de alta qualidade sobre eventos de perda de dados". Gerido originalmente por voluntários da *Open Security Foundation*, os dados são fornecidos desde 2011 pela

empresa Risk Based Security (RBS). É, na verdade, um blog que referencia documentos publicados pela RBS, porém não recebe atualizações desde fevereiro de 2016.

Centro de Tratamento de Incidentes de Segurança de Redes de Computadores da Administração Pública Federal - CTIR Gov (2017)

Sua finalidade é atender os incidentes em redes de computadores da Administração Pública Federal (APF). Subordinado ao Departamento de Segurança de Informação e Comunicações (DSIC) do Gabinete de Segurança Institucional da Presidência da República (GSI/PR), o CTIR Gov foi criado em 2004, após recomendação dos grupos de trabalhos instituídos pelo Decreto nº 3505, de 13 de junho de 2000, que estabeleceu a Política de Segurança da Informação (em âmbito federal).

O CTIR Gov publica alertas sobre temas relacionados à Segurança da Informação e Comunicações, julgados críticos e que afetam os órgãos e entidades da AFP. O Centro utiliza como fontes de consulta o CERT.br, CERT/CC e bases de vulnerabilidades dos principais fornecedores de soluções de TI e Segurança da Informação (exemplo: NVD).

Vocabulary for Event Recording and Incident Sharing - VERIS (2017)

O VERIS tem sua origem nos relatórios de vazamentos de dados da Verizon Enterprise (Verizon's Data Breach Investigations Reports - DBIR), largamente referenciado em publicações sobre Segurança da Informação. De acordo com informações disponíveis no site do projeto, seu objetivo é estabelecer um banco de dados cooperativo e colaborativo de incidentes de Segurança da Informação, ajudando as organizações a coletar informações úteis relacionadas a incidentes e a compartilhá-las de forma "anônima e responsável" com uma comunidade de pesquisa. Além das bases de dados públicas, o site oferece um conjunto de métricas e uma linguagem comum para descrever incidentes de segurança de forma estruturada e repetitiva.

O banco de dados comunitário do VERIS possui mais de 7000 registros e sua estrutura consiste em 136 campos diversos. O esquema é bem detalhado e documentado na página web do projeto, e está estruturado de maneira a simplificar o compartilhamento anônimo de dados, segregando informações privadas em campos que são ignorados na base de dados pública.

Segundo depoimento de especialistas envolvidos no projeto, a principal razão para criação do VERIS foi permitir a identificação de dados específicos relacionados aos incidentes e também para fornecer alguma estrutura em torno desses dados para que pudessem ser analisados e compartilhados. Na sua visão a documentação de incidentes é quase sempre ignorada, porém, quando ocorre, raramente é realizada de uma forma que incentive a organização a identificar causas raiz, lições aprendidas, etc. A proposta do VERIS é

justamente oferecer subsídio para esta análise estratégica pós-incidente.

Um dos problemas enfrentados pelo projeto é, justamente, a falta de colaboração entre a comunidade: a maioria do trabalho de coleta de informações é realizado pelo próprio time responsável, utilizando informações publicamente disponíveis. Algumas informações, no entanto, são submetidas por organizações parceiras. Isto acaba limitando o nível de detalhamento e qualidade dos dados. Apesar do modelo ser extremamente bem definido e documentado, a maioria dos registros utiliza apenas uma pequena fração dos campos disponíveis, e possuem poucos detalhes técnicos.

É importante observar que os responsáveis pelo VERIS apresentaram as mesmas características colaborativas de seu projeto, tendo esclarecido diversos questionamentos técnicos sobre o modelo bem como seu histórico de desenvolvimento. Aliado às demais características apresentadas aqui, isto levou a um maior aprofundamento do modelo e seu posterior uso em experimentos, o que aumentou a sua importância para a dissertação.

A Tabela 4.3 apresenta um comparativo entre as bases de dados pesquisadas.

TAB. 4.3: Bases de dados sobre incidentes

Base	**Fontes de informação**	**Status**	***Database* estruturado?**
CTIR Gov	CERT.br, CERT/CC e bases de vulnerabilidades públicas	Ativo	não
DataLossDB	Empresa RBS e pesquisas públicas	Descontinuado	não
RISI	Membros, pesquisas públicas e parceiros estratégicos	Descontinuado	sim
WHID	Comunidade e pesquisas públicas	Descontinuado	sim
VERIS	Verizon Enterprise, parceiros e pesquisas públicas	Ativo	sim

4.3 ONTOLOGIAS RELACIONADAS A GESTÃO DE INCIDENTES

Blackwell (2010) propõe "uma ontologia de segurança para análise de incidente", mas a proposta é mais uma taxonomia do que uma ontologia. O artigo define alguns conceitos de incidente, ciberdefesa e ataque, mas não os formaliza como uma ontologia. O uso do OWL está listado como melhoria futura.

Mundie et al. (2014) propõe uma ontologia com base em um trabalho anterior que define um meta-modelo com processos essenciais da Gestão de Incidentes. O objetivo é documentar, comparar e analisar times de resposta a incidentes (CSIRTs – *Computer Security Incident Response Teams*). Os processos modelados são de alto nível, mais voltados para as relações e papéis entre os times, sem entrar no detalhe técnico do incidente. O trabalho está ligado ao CERT/CC – Computer Emergency Response Team Coordination Center (www.cert.org) – o primeiro e mais conhecido centro de resposta a incidentes do mundo, ligado aos departamentos de defesa e segurança interna dos EUA e gerido pela Universidade Carnegie Mellon. Alguns conceitos úteis (em tradução literal):

- **componentes-do-incidente**: os elementos que constituem o modelo conceitual de um evento;
- **artefatos**: qualquer entidade deixada depois da ocorrência de um incidente. Exemplo: código malicioso ou arquivos de log;
- **eventos**: qualquer ocorrência que pode ter consequências negativas para a segurança;
- **incidentes**: eventos que tiveram consequências de segurança negativas confirmadas;
- **causa-raiz**: a primeira ocorrência na cadeia causal que levou a um incidente;
- **vulnerabilidades**: as fraquezas nos sistemas que foram exploradas por um incidente (ou potencialmente exploráveis).

No contexto de Tratamento de Incidentes (um subconjunto da gestão de incidentes), porém, nem todos os conceitos são pertinentes. O documento não detalha as propriedades de cada classe, pois o foco está nas conexões.

Silva e Fagundes (2014) propõe uma ontologia para gestão de incidentes de Segurança da informação baseada na norma ISO/IEC 27035:2011 e objetiva auxiliar na capacitação de equipes de resposta a incidentes. Compara outros trabalhos utilizando como critérios: a classificação (vocabulário, tipo de ontologia, taxonomia etc), as referências, o registro da metodologia para construção da ontologia e a disponibilidade para a comunidade (publicação). Apresenta oito superclasses e as propriedades da classe *Incident*, porém não vai além e não apresenta nenhum caso de uso. Não define o conceito de evento de segurança, embora seja parte fundamental da ISO 27035. A ontologia não está mais disponível no local indicado pelo artigo (repositório do Protégé), porém pode ser reproduzida parcialmente com base nas ilustrações.

O'Sullivan e Turnbull (2015) postulam sobre a dependência em tecnologia no contexto militar e sobre o paradoxo de cada vez mais utilizar ataques cibernéticos e,

ao mesmo tempo, depender tanto da infraestrutura cibernética. Introduz então uma ontologia "de código aberto" (disponível no github) que representa ativos cibernéticos com o objetivo de modelar cenários para simulação de defesa de redes de computadores, aumentando a sua resiliência. Divide a ontologia em duas: Cyber Simulation Terrain (CST), que define conceitos como computadores, conectividade de rede, usuários, software, vulnerabilidades e exploits; Cyber Effects Simulation Ontology (CSEO), que modela os impactos de um ciberataque em sistemas que fazem parte de uma rede complexa. É o trabalho mais completo entre os revisados, porém o foco maior dos casos de uso é a identificação de vulnerabilidades e o impacto relacionado a elas. Não considera os eventos de segurança, nem os aspectos temporais. Os conceitos mais relevantes são:

- <**Incident**, **'Incident caused by'**, **Exploit**> – Tripla que define a relação entre o incidente e a exploração utilizada para perpetrá-lo;
- <**Incident**, **Victim**, **Node**> – Consiste na tripla que define a ligação entre o incidente e o nó afetado (incidente vitima nó - vitima, do verbo vitimar);
- <**Incident**, **'Remedial Course of Action'**, **'Course of Action'**> – Consiste na tripla que define a relação entre o incidente e a ação para remediá-lo;
- **CoursesOfAction** – são medidas específicas a serem tomadas para enfrentar a ameaça, sejam elas corretivas ou preventivas, para tratar os *ExploitTargets* (alvos de explooração), ou seja objetiva mitigar os impactos potenciais de Incidentes.

O trabalho, entretanto, possui oportunidades de melhoria. Embora a primeira tripla preveja a identificação do *exploit* utilizado, o que pressupõe a prévia identificação de um ou mais eventos causadores do incidente, a ontologia não define eventos de segurança. Isto se deve, provavelmente, à natureza da proposta muito voltada para simulação e análise de vulnerabilidades. Na segunda tripla o uso do verbo "vitimar" pode causar confusão, já que outros vocabulários e ontologias adotam o termo "vítima" como a instituição afetada pelo incidente – as palavras em inglês são idênticas e em português só podem ser diferenciadas pelo uso do acento.

Syed et al. (2016) propõem uma ontologia aberta (disponível no github) com o objetivo de "incorporar e integrar dados heterogêneos e esquemas de conhecimento de diferentes sistemas de cibersegurança e padrões comumente usados para compartilhamento de informações" (exemplo: VERIS), unificando, inclusive, ontologias relacionadas propostas anteriormente. Embora tenha o objetivo grandioso e seja apresentada como "a

primeira ontologia de cibersegurança a suportar uma ampla gama de casos de uso", o trabalho apresenta apenas 4 pequenos casos de uso, todos relacionados à identificação de vulnerabilidades de software, um resultado que parece um subconjunto do resultado apresentado pelo trabalho australiano (O'SULLIVAN; TURNBULL, 2015). O trabalho é coerente e possui conceitos bem definidos, alguns deles reutilizáveis no contexto deste trabalho. Não responde, entretanto, às perguntas de competência propostas aqui, não define "evento" e tampouco considera os aspectos temporais (quando algo aconteceu), mas propõe este último ponto como melhoria futura.

Mavroeidis e Bromander (2017) consolidam um modelo com definições comuns de *Cyber Threat Intelligence* e realizam uma avaliação das taxonomias, padrões e ontologias relacionadas ao conceito. O modelo também se propõe a apoiar as organizações na medição do nível de maturidade de suas capacidades de realizar a *Threat Intelligence*.

Os seguintes termos são definidos pelo modelo:

- **Identidade** - a referência a uma pessoa, organização ou Estado a que se atribui uma ameaça, ou ainda a relação com outros ataques;
- **Motivação** - as forças que habilitam as ações em busca de objetivos específicos. Segundo os autores, ainda que os objetivos do atacante mudem, em geral as motivações permanecem as mesmas;
- **Objetivos** - "Uma representação cognitiva de um ponto final desejado que impacta avaliações, emoções e comportamento". Segundo os autores, normalmente é descrito por uma tupla (ação, objeto), mas ainda é necessária a criação de uma taxonomia com nível de detalhes adequado. Exemplos de objetivos são "roubar propriedade intelectual", "causar dano à infraestrutura" e "atrapalhar um concorrente";
- **Estratégia** - Uma descrição não técnica, de alto nível, sobre o ataque planejado, normalmente descrita em prosa.
- **Táticas, técnicas e procedimentos (TTPs)** - Caracterizam o quê o adversário está fazendo e como. As subcategorias são "padrões de ataque", "*malware*", "infraestrutura";
- **Ferramentas** - as ferramentas para na rede da vítima, normalmente modificadas para evitar a sua detecção. *Malware* é uma subcategoria, e muitas vezes ferramentas não maliciosas são usadas para razões maliciosas, ex: scanners de vulnerabilidade e de topologia de rede.
- **Indicadores de Compromisso (IOCs)** - Detectivos por natureza, descrevem como reconhecer comportamento malicioso ou suspeito que diretamente detectam

campanhas, TTPs, padrões de ataque, *malware*, ferramentas e atores de ameaça. Um bom IOC consiste em uma combinação de diferentes informações;

- **Alvo** - Podem ser organizações, companhias, setores, nações ou indivíduos;
- **Cursos de ação (*Courses of Action*)** - Referem-se às medidas que podem ser tomadas para prevenir ou responder ataques.

A Tabela 4.4 sintetiza o comparativo entre as ontologias apresentadas nesta seção. O trabalho de Mavroeidis e Bromander (2017) não é listado por se tratar também de um comparativo e não de uma ontologia em si.

Com exceção do trabalho de Syed et al. (2016) (*Unified Cyber Ontology*), que tem uma proposta de uso abrangente, nenhum dos outros trabalhos referenciados nesta dissertação aparece na comparação publicada por Mavroeidis e Bromander (2017), já que seu objetivo é específico em *Cyber Threat Intelligence*.

Assim como identificado nesta dissertação, os autores também observam que muitas das ontologias pesquisadas não disponibilizam os arquivos RDF/OWL relevantes, mesmo alguns que se autointitulam "de código aberto".

TAB. 4.4: Comparativo entre as ontologias relacionadas a tratamento de incidentes

Autores	**Características**	**Limitações**	**Casos de uso**	**Perguntas de competência**	**Disponível publicamente?**
Blackwell (2010)	Define conceitos de incidente, ciberdefesa e ataque	Mais uma taxonomia do que uma ontologia	0	0	Não
Mundie et al. (2014)	Define relações e papéis entre times de resposta a incidentes	Não entra no detalhe técnico do incidente; Mais uma taxonomia de processos organizacionais do que uma ontologia	0	0	Não
Silva e Fagundes (2014)	Propõe uma ontologia para gestão de incidentes baseada na ISO/IEC 27035:2011	Não define evento	0	0	Não
O'Sullivan e Turnbull (2015)	Define uma ontologia que modela cenários para simulação de defesa de redes de computadores	Não considera os eventos de segurança, nem os aspectos temporais; Não utiliza o formato OWL	2	45	Sim (Github)
Syed et al. (2016)	Propõe uma nova ontologia unificando diversos trabalhos anteriores	Não considera os eventos de segurança, nem os aspectos temporais	4	4	Sim (Github)
"Moreira, G.B. (2018)"	Foco em tratamento de incidentes; Introduz o conceito fundamental de evento e considera aspectos temporais do incidente; Disponível no formato OWL	n/a	3	20	Sim (Github)

5 ONTOLOGIAS PARA TRATAMENTO DE INCIDENTES

Neste capítulo são apresentadas a modelagem das ontologias propostas, as tecnologias utilizadas, bem como as metodologias de desenvolvimento empregadas.

5.1 DEFINIÇÕES INICIAIS

A partir do referencial teórico apresentado nos capítulos 2 e 3 e da revisão bibliográfica foram definidos alguns requisitos fundamentais para o modelo: um incidente só pode ser caracterizado caso exista **ao menos um evento relacionado**. Embora um incidente possa ser identificado de forma automática, comumente ele é **validado por um analista**, portanto, um incidente sempre estará relacionado a um analista responsável. O tratamento de um incidente é **dividido em fases**, um analista deve definir a fase atual do incidente. Um incidente precisa de **ações de resposta**, estas ações são definidas e aplicadas por um ou mais analistas. O analista também deve ser responsável por informar o **estado do incidente**, por exemplo, se está em andamento ou já foi concluído.

Também foi apresentando no capítulo 2 que é desejável **maximizar o conhecimento** sobre causas, consequências e origem do incidente desde o momento da sua confirmação, além de **minimizar o tempo** entre a ocorrência e a detecção de um evento e entre a descoberta e a conclusão do incidente. Para que seja possível abordar estes aspectos, o modelo deve implementar uma "linha do tempo" com registro histórico de todas as **ocorrências** relacionadas ao incidente, isto é, não apenas os eventos associados e ações aplicadas, mas também planos de ação, observações sobre os resultados das ações, informações relevantes obtidas de fontes externas, comunicações entre a equipe, etc.

A partir destas definições preliminares foram identificadas as primeiras classes do modelo, além de algumas de suas propriedades principais, conforme descrições a seguir.

AÇÃO (de resposta): são as atividades realizadas para se responder ao incidente. Este conceito aparece em mais de uma das ontologias relacionadas, com o nome de *Course of Action* (curso de ação, em tradução livre). Os principais atributos são: data/hora da ação, objetivo, descrição, analista responsável.

ANALISTA: consiste em uma pessoa desempenhando esta função. Os atributos são os dados pessoais do profissional designado, como seu nome, sobrenome, matrícula, cargo,

função, telefone de contato, entre outros. Por ser um tipo de representação trivial em sistemas, esta é uma classe com forte potencial de ser "reutilizada" de outro modelo.

EVENTO (de segurança): consiste na "identificação de uma ocorrência em um sistema, serviço ou estado de rede, indicando uma possível violação na política de segurança da informação e seus controles, ou uma situação previamente desconhecida que pode ser relevante no âmbito da segurança da informação", conforme a definição da norma ISO/IEC 27035:2011 (2011), apresentada no capítulo 2. Na prática são registros (*logs*) de ferramentas de segurança, como por exemplo *firewalls*, sistemas de prevenção/detecção de intrusão (IPS/IDS) de *host* ou de rede, antivírus, servidores Web, serviços de diretório (ex: *Active Directory*), DNS, DHCP, entre outros. **As propriedades dos objetos desta classe dependerão da ferramenta que gerou o registro**. Alguns exemplos de atributos são: data/hora de detecção do evento, identificação do sistema de origem, endereço IP que originou o evento, endereço IP alvo do evento, nome dos *hosts*, porta, tipo de evento, resposta do sistema, etc.

Embora seja um dos conceitos mais importantes no contexto de resposta a incidentes, praticamente nenhuma das ontologias relacionadas na revisão bibliográfica (capítulo 4) apresenta a definição de EVENTO.

FASE: conforme definido no capítulo 2, as possíveis fases de um incidente serão CONTENÇÃO, ERRADICAÇÃO e RECUPERAÇÃO. A fase não é necessariamente uma classe do modelo, porém deseja-se realizar o registro histórico de cada transição de fase, ou seja, data/hora em que foi definida, bem como o analista que a definiu.

INCIDENTE: caracteriza-se quando identifica-se "um ou mais eventos de segurança da informação, indesejados ou inesperados, que possuem probabilidade significativa de comprometer as operações do negócio e ameaçar a segurança da informação", conforme a definição da norma ISO/IEC 27035:2011 (2011), apresentada no capítulo 2. Uma vez confirmado o incidente, ele deve ser documentando de forma apropriada e associado aos eventos relacionados. Dentre seus atributos podemos destacar data/hora de confirmação, descrição, tipo, analista responsável pela confirmação, status e data/hora de conclusão.

OCORRÊNCIA: permite documentar o registro histórico de todas as ocorrências relacionadas ao incidente numa linha do tempo, sejam elas aplicações das ações de resposta definidas ou qualquer outro tipo de informação relevante. Esta é uma das contribuições mais importantes do modelo, já que nenhuma outra ontologia relacionada aborda aspectos temporais. Cada ocorrência é registrada por um analista e deve conter ao menos os atributos data/hora, fase do incidente e descrição.

A Figura 5.1 apresenta um diagrama de classes construído a partir destas definições.

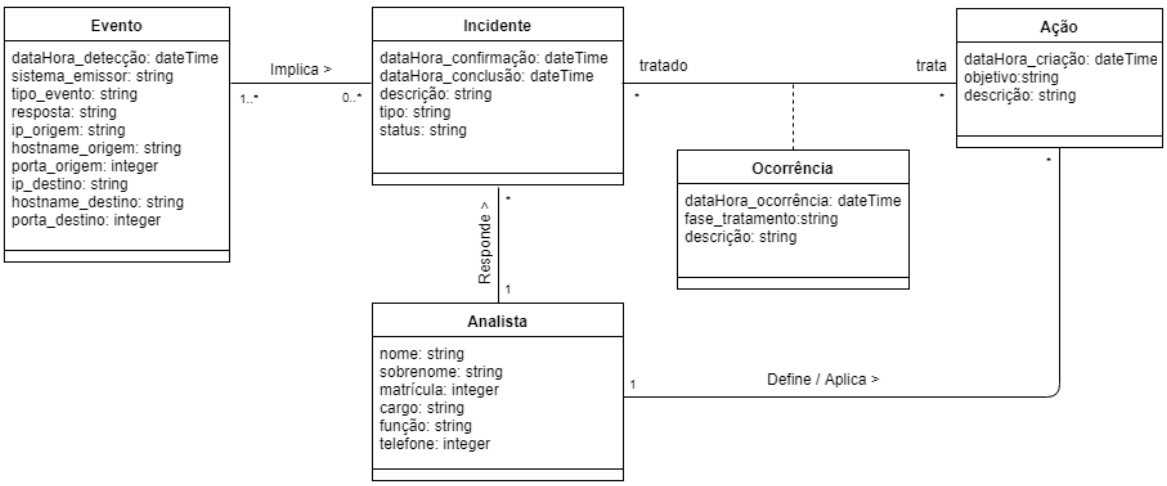

FIG. 5.1: Diagrama de classes preliminar com propriedades exemplo

À medida que novas iterações da modelagem ocorriam, surgiu-se a necessidade da criação de novas propriedades e características. Por se tratar de um processo que busca otimizar a documentação com objetivo de se extrair conhecimento, um dos aspectos desejáveis mais importantes identificados foi a necessidade de se realizar o registro histórico das ações relacionadas aos incidentes, bem como guardar a sua característica de temporalidade, ou seja, quando cada ação ocorreu.

O aumento da complexidade nesta modelagem aliado às características comuns com problemas clássicos de representação de conhecimento e a expansão da pesquisa com trabalhos relacionados ligados às tecnologias da Web Semântica direcionaram esta dissertação para o uso de ontologias.

Como visto no capítulo 3, a *Web Ontology Language* (OWL) permite descrever com grande expressividade os objetos e suas relações, inclusive de forma distribuída entre sistemas, oferecendo suporte a processos decisórios e recursos para inferência. Como o aspecto principal na criação de uma ontologia é o modelo, mudanças estruturais podem acontecer todo o tempo sem prejuízo para os dados, o que torna seu uso particularmente bom para exploração e experimentação (FERGUSON, 2005).

5.2 CRIAÇÃO DAS ONTOLOGIAS

Ao longo do desenvolvimento do trabalho foi prevista a criação de duas ontologias. A primeira delas, a que se deu o nome de *Computer Security Incident Handling Ontology* (CSIHO), foi criada "do zero" baseada nos conceitos aqui apresentados, com a motivação inicial de responder às perguntas de competência que foram desafiadoras no estudo de

caso do WannaCry e prover, assim, um modelo para a melhoria na documentação de incidentes futuros.

O modelo VERIS (2017), *Vocabulary for Event Recording and Incident Sharing*, além de referenciado por diversos trabalhos relacionados é o único projeto ativamente mantido que oferece um *dataset* público com registro de incidentes e dados representativos, ainda que muitas vezes incompletos, bem como um modelo com elevado grau de detalhamento. Estas razões motivaram a criação de uma segunda ontologia baseada no modelo, com o objetivo de demonstrar o seu potencial de uso quando estendido para esta forma mais completa de representação.

Um terceiro cenário foi previsto, demonstrando a correlação de dados entre as duas ontologias desenvolvidas, buscando-se obter um diferente nível de informações.

5.2.1 TECNOLOGIAS UTILIZADAS

Para o desenvolvimento da CSIHO, a principal ferramenta adotada foi o Protégé, um software gratuito desenvolvido na Universidade de Stanford, largamente utilizado pela comunidade acadêmica (MUNDIE et al., 2014). O Protégé encontra-se na versão 5.2, é um ambiente altamente extensível, de código aberto, que permite rápida prototipagem e desenvolvimento, é ativamente suportado por uma comunidade de usuários e desenvolvedores, há mais de 17 anos, e atende a especificação mais atual do W3C, a Web Ontology Language (OWL) 2.

Embora o Protégé seja uma ferramenta completa para criação e manipulação de ontologias, foi observado sensível impacto de performance ao se realizar pesquisas SPARQL em grandes *datasets*. Para contornar este problema foi utilizado o Apache Jena Fuseki v3.6. O Fuseki é um servidor SPARQL *standalone* com uma interface web para monitoração e administração que oferece segurança e uma "robusta camada de armazenamento persistente transacional", além de suportar os protocolos SPARQL 1.1 e Graph Store (THE APACHE SOFTWARE FOUNDATION, 2017). Ele é parte do *framework* de código aberto "Apache Jena", voltado para construção de aplicações para Web Semântica e *Linked Data*.

Para a criação da ontologia VERIS foi necessária a leitura e conversão do esquema e *dataset* originais no formato JSON. Para a realização sistemática destas tarefas utilizou-se a linguagem de programação Python 3, de sintaxe simples e fácil prototipagem, porém bastante poderosa e flexível. Para a manipulação das ontologias foi utilizado o módulo Owlready2 v0.5 (LAMY, 2017). O owlready é um projeto de código aberto, oriundo

da área biomédica, que permite manipular classes, instâncias e propriedades de uma ontologia como se fossem objetos na linguagem Python. Ele suporta os formatos N-Triples, RDF/XML e OWL/XML, além de incluir uma adaptação do HermiT *reasoner*.

5.2.2 DESENVOLVIMENTO DA CSIHO

A CSIHO foi criada "do zero" reutilizando definições pertinentes de outras ontologias, conforme apresentado no capítulo 4, porém introduzindo novos conceitos. O *dataset* utilizado para popular a ontologia foi construído a partir do estudo de caso de um incidente com o *rasomware* Wannacry (MOREIRA et al., 2017).

As seguintes premissas foram estabelecidas para o desenvolvimento da ontologia:

- A documentação dos incidentes deve conter registros detalhados (*logs*) de detecção, não apenas descrições textuais;
- Deve-se prever um modo de trabalho colaborativo incentivando a pesquisa em entidades externas;
- Temporalidade é importante – deve-se registrar em que momento eventos e ações ocorreram ou foram registrados;

A ontologia foi implementada através da ferramenta Protégé, no formato OWL 2.0 RDF/XML, utilizando a metodologia *Ontology Development 101* (NOY; MCGUINNESS, 2000), conforme apresentado no capítulo 3. Os detalhes do desenvolvimento de cada uma das sete etapas da metodologia são descritos a seguir.

Etapa 1 - Determinar o domínio e o escopo da ontologia.

A ontologia CSIHO cobre o domínio abordado pelo macroprocesso de tratamento de incidentes (*incident handling*, na definição em inglês), com o propósito de melhorar os processos de resposta a incidentes de Segurança da Informação nas organizações. Ela deve prover fundamento para a documentação adequada de incidentes, e ser capaz de apoiar na definição de ações de resposta em diferentes cenários. Espera-se que a ontologia seja utilizada e mantida por pesquisadores na área de tratamento de incidentes, desenvolvedores de soluções para reposta a incidentes e analistas que trabalham com resposta a incidentes. As seguintes **perguntas de competência** foram definidas:

1) Quais os tipos de eventos relacionados a um dado incidente?
2) Quais os ativos afetados relacionados a um dado incidente?
3) Quais os incidentes relacionados a um determinado ativo?
4) Quais os eventos relacionados a um dado incidente?

5) Quais as ações de resposta relacionadas a um dado incidente?
6) Dadas características de um evento, quais incidentes estiveram relacionados?
7) Dado um novo incidente, quais ações previas poderiam ser reaplicadas?
8) Quantas ações foram aplicadas em cada fase de um determinado incidente?
9) Qual o tempo desde a detecção e a conclusão de um determinado incidente?

Etapa 2 - Reuso de ontologias existentes.

Os seguintes conceitos foram incorporados dos trabalhos relacionadas:

- **Eventos**: qualquer ocorrência que pode ter consequências negativas para a segurança (MUNDIE et al., 2014);
- **Incidentes**: eventos que tiveram consequências de segurança negativas confirmadas (MUNDIE et al., 2014);
- ***Courses of Action***: as ações responsivas ou preventivas para mitigar um ataque (O'SULLIVAN; TURNBULL, 2015) (MAVROEIDIS; BROMANDER, 2017).

Etapa 3 - Enumerar termos importantes na ontologia.

Os seguintes termos foram listados com base nas modelagens preliminares: incidente, evento, ação, fase, status, ameaça, vulnerabilidade, preparação, identificação, resposta, acompanhamento, erradicação, contenção, recuperação.

Etapa 4 - Definir as classes e sua hierarquia.

Para tornar a ontologia mais universal e se manter a consistência com os termos incorporados de outras ontologias, foram adotados nomes no idioma inglês. A seguir são apresentadas as definições das classes bem como sua hierarquia, na Figura 5.2.

- **Computer_Asset**: são os equipamentos nós na rede defendida. Na CSIHO estão divididos em duas categorias, *Workstation* (estação de trabalho) e *Server* (servidor).
- **Course_of_Action**: são as ações de resposta ao incidente.
- **Incident**: é a classe principal associada a todos os demais elementos do incidente.
- **Person**: classe que define os analistas relacionados ao incidente. Eles são classificados em três diferentes níveis, Tier1, Tier2 e Tier3.
- **Security_Event**: classe que define os eventos de segurança associados ao incidente. Ela é dividida em subclasses de acordo com a natureza do evento: Antivirus_Event, Firewall_Event, IPS_Event, Proxy_Event e Sysmon_Event. Para incluir *logs* provenientes de novas ferramentas basta criar uma nova subclasse relacionada, garantindo assim uma fácil escalabilidade da classe.

- **Timeline_Occurrence**: é a classe que define todas as ocorrências históricas relacionadas ao incidente (quem, quando, por quê). Está classificada em duas subclasses:
  - COA_Timeline_Occurrence: registro de aplicação das ações de resposta;
  - Generic_Timeline_Occurrence: qualquer outro tipo de registro relacionado ao incidente. Por exemplo: definição de um plano de ação, observações sobre o resultado de uma ação e informações relevantes obtidas de fontes externas.

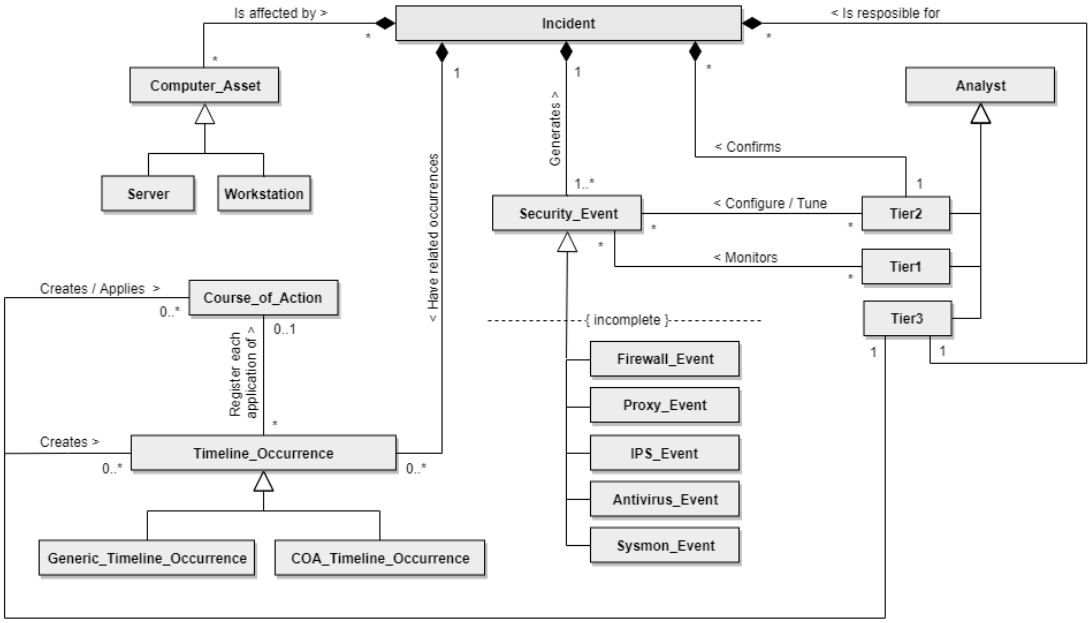

FIG. 5.2: Metamodelo da ontologia CSIHO

Etapa 5 - Definir as propriedades das classes.

As propriedades das classes são apresentadas na Tabela 5.1. Como a classe "Security_Event" possui uma subclasse para cada tipo de sistema de segurança, propriedades adicionais são incluídas de acordo com a natureza do evento.

Etapa 6 - Definir as "facetas" de cada propriedade.

É como a metodologia se refere às definições de cardinalidades, tipos de dados (ou *range*) e domínio (uma ou mais classes a que a propriedade está relacionada). Com base nas definições preliminares, os seguintes axiomas foram definidos:

- Um **Security_Event** pode estar associado a 0 ou 1 **Incident**;
- Um **Incident** possui ao menos 1 **Security_Event** associado;
- **Incident** é confirmado por algum **Tier3_Analyst**;
- **Course_of_Action** é criada por algum **Tier3_Analyst**;

TAB. 5.1: Propriedades das classes (etapa 5) e tipos de dados (etapa 6)

Classe (domínio)	**Propriedades**	**Tipo de dado (*range*)**
Computer_Asset	nome do host	string
	endereço IP	string
Course_of_Action	data/hora de criação	dateTime
	objetivo	string
	descrição	string
Person	nome	string
	sobrenome	string
	matrícula	string
	cargo	string
	função	string
	telefone	string
Security_Event	data/hora de detecção	dateTime
	sistema emissor	string
	tipo do evento	string
	resposta	string
	IP de origem	string
	hostname de origem	string
	porta de origem	integer
	IP de destino	string
	hostname de destino	string
	porta de destino	integer
Incident	data/hora de confirmação	dateTime
	data/hora de conclusão	dateTime
	descrição	string
	tipo	string
	status	string
Timeline_Occurrence	data/hora da ocorrência	dateTime
	fase do tratamento	'Contenção', 'Erradicação' ou 'Recuperação'
	descrição	string

- Uma **Timeline_Occurrence** deve estar associada a um **Incident**;
- Uma **Timeline_Occurrence** deve ser informada por um **Tier3_Analyst**;
- Uma **Timeline_Occurrence** pode estar relacionada a 0 ou 1 **Course_of_Action**.

Etapa 7 - Criar as instâncias (indivíduos).

Os dados para popular a ontologia foram criados a partir do estudo de caso sobre um incidente com o *ransomware* Wannacry descrito por Moreira et al. (2017). O histórico do caso foi organizado cronologicamente em uma tabela com identificadores e em seguida transportados para a ontologia, conforme exemplos na Tabela 5.2.

TAB. 5.2: Exemplos de dados instanciados na classe Timeline_Occurrence (etapa 7). Adaptado de Moreira et al. (2017)

Identificador	**COA relacionado**	**Data/hora**	**Fase**	**Descrição da ocorrência**
occurrence_001	coa_001	2017-05-12 10:35:00	Contenção	Aplicada coa_001
occurrence_002	coa_002	2017-05-12 10:38:00	Contenção	Aplicada coa_002
occurrence_003	-	2017-05-12 14:00:00	Contenção	Instaurada sala de crise
occurrence_004	coa_003	2017-05-12 10:51:00	Contenção	Aplicado coa_003
occurrence_010	coa_004	2017-05-12 12:11:00	Contenção	Bloqueio da porta 445 na conexão com a Filial RJ1
occurrence_011	coa_004	2017-05-12 12:40:00	Contenção	Bloqueio da porta 445 na conexão com a Filial RJ2
occurrence_015	coa_005	2017-05-12 14:00:00	Contenção	Bloqueados binários...
occurrence_016	coa_006	2017-05-12 15:33:00	Contenção	Solicitada nova distribuição do patch que corrige a vulnerabilidade no SMB do Windows (boletim MS17-010), com boot obrigatório.
occurrence_017	coa_007	2017-05-12 19:40:00	Contenção	Distribuida nova "vacina" para o WannaCry liberada pelo fornecedor de antivírus.
occurrence_018	coa_008	2017-05-13 11:00:00	Contenção	Realizado levantamento de hosts: (i) Sem os patches do SMB; (ii) Com antivírus desatualizado (sem a nova vacina).
occurrence_019	coa_009	2017-05-13 18:00:00	Contenção	Desconectadas (ou desligadas) estações com potencial de contaminação, identificadas pelo coa_008.
occurrence_020	coa_010	2017-05-14 19:00:00	Contenção	Implementado servidor web interno respondendo às URLs do Kill Switch do WannaCry (honney pot).
occurrence_021	coa_011	2017-05-15 09:45:00	Erradicação	Bloqueadas extensões utilizadas pelo WannaCry nos servidores de arquivos.
occurrence_022	coa_012	2017-05-15 11:05:00	Erradicação	Definido e divulgado fluxo de tratamento de máquinas suspeitas/vulneráveis e comunicar as equipes.
occurrence_023	coa_013	2017-05-15 11:45:00	Erradicação	Enviada orientação para equipes de suporte local sobre procedimentos necessários para reconectar máquinas suspeitas desconectadas
occurrence_024	coa_014	2017-05-16 14:00:00	Recuperação	Reestabelecida conexão com a filial RJ1, de acordo com o fluxo padrão estabelecido pelo coa_014.
occurrence_025	coa_014	2017-05-16 15:00:00	Recuperação	Reestabelecida conexão com a filial RJ2, de acordo com o fluxo padrão estabelecido pelo coa_014.

As estatísticas, *metrics* do Inglês, da ontologia, segundo o Protégé, são apresentadas na Tabela 5.3. A linha "expressividade DL" indica os tipos de representação em Lógica de Descrição utilizados na ontologia, conforme exemplos da Tabela 3.3.

TAB. 5.3: Estatísticas da ontologia CSIHO

Estatística	**Contagem**
Axiomas	585
Axiomas lógicos	406
Axiomas declarativos	153
Classes	20
Propriedades de objetos	11
Propriedades de dados	66
Indivíduos (instâncias)	58
Anotações	2
Expressividade DL	ALCHQ(D)

5.2.2.1 EXEMPLO DE INFERÊNCIA LÓGICA NO OWL

Na etapa 6 da construção da ontologia foi definido um axioma que diz que um incidente (classe "Incident") é confirmado por um analista da classe "Tier3". Durante a criação dos indivíduos, na etapa 7, foram criados dois objetos no Protégé chamados "analyst_005" e "incident_444", porém eles não foram associados à nenhuma classe. Posteriormente foi criada uma relação do objeto "analyst_005" como analista que confirmou o objeto "incident_444" (associação has_confirmation_analyst). Ao sincronizar o HermiT *reasoner* no Protégé, os axiomas da ontologia são analisados e o sistema infere que o objeto "analyst_005" faz parte da classe "Tier3", bem como o objeto "incident_444" faz parte da classe "Incident", ainda que estas declarações não existam explicitamente.

5.2.3 DESENVOLVIMENTO DA ONTOLOGIA VERIS

O modelo VERIS está, originalmente, dividido em dois elementos:

(i) Esquema - disponível em **https://github.com/vz-risk/veris/**
(ii) Dataset - disponível em **https://github.com/vz-risk/VCDB/tree/master/data**

Tanto o esquema quanto o *dataset* originais do modelo VERIS são representados no formato JSON, conforme exemplo apresentado pelo código 5.1. O JSON é uma notação de formatação leve para troca de dados (JSON.ORG, 2017) largamente utilizada na Web e suportada pela maioria das linguagens de programação modernas. O principal desafio foi

interpretar estes elementos em sua representação original e convertê-los sistematicamente para a representação desejada.

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "additionalProperties": false,
  "description": "VERIS Community Schema 1.3.2",
  "properties": {
    "action": {
      "minProperties": 1,
      "properties": {
        "environmental": {
          "additionalProperties": false,
          "properties": {
            "notes": {
              "minLength": 1,
              "type": "string"
            },
            "variety": {
              "items": {
                "type": "string"
              },
              "minItems": 1,
              "type": "array",
              "uniqueItems": true
            }
          },
          "required": [
            "variety"
          ],
          "type": "object"
        },
    (...)
  },
  "required": [
    "actor",
    "action",
    "discovery_method",
    "schema_version",
    "asset",
    "timeline",
    "incident_id",
    "security_incident"
  ],
  "type": "object"
}
```

Código 5.1: Extrato do esquema original do modelo VERIS em JSON

O objetivo inicial nesta etapa era desenvolver um algoritmo capaz de converter o esquema JSON original do VERIS diretamente para uma ontologia OWL com mínima intervenção ou tratamento de exceções. Yao et al. (2014) e Miličić (2014) propõem metodologias para converter dados em JSON para ontologias RDF/OWL, mas devido à

natureza flexível do JSON não há uma forma de conversão direta e o processo passa, invariavelmente, pela análise cuidadosa de como a informação foi representada e estruturada.

Diversos códigos preliminares foram desenvolvidos com o objetivo de "explorar" o esquema do VERIS até que finalmente fosse possível chegar ao código final para a construção de uma "ontologia VERIS". Este processo levou à identificação de pequenas inconsistências no modelo original, que foram reportadas aos seus mantenedores. O código final para construção da ontologia em OWL está disponível no apêndice da dissertação.

O processo seguinte foi a criação de um algoritmo para leitura do *dataset* em JSON e a sua conversão e escrita no formato OWL. A criação deste código também foi um processo iterativo que envolveu, inclusive, ajustes no algoritmo de criação da ontologia. O código final para carga do *dataset* também está disponível no apêndice.

Um aspecto importante na construção da ontologia está no fato de que os diferentes tipos de propriedade viraram classes individuais. Isto significa que um incidente passou a ser um grupo de instâncias conectadas ao invés de um único registro linear, conforme ilustrado pela Figura 5.3, permitindo assim que um objeto único com descrição comum seja representado uma única vez e associado a vários incidentes. Por exemplo, uma empresa vítima de mais de um incidente, um ataque específico que tenha vitimado mais de uma instituição, como o WannaCry, ou ainda um mesmo grupo *hacker* que tenha sido o agente de ameaça de vários incidentes reportados.

Como os dados foram migrados do *dataset* original de forma sistemática, esta mudança estrutural não representou ganhos imediatos para o modelo, porém, a inclusão de novos incidentes ou a adaptação mais profunda dos antigos permitiria uma representação mais eficiente dos dados, como será mostrado pelos experimentos.

Uma vez validado o algoritmo de carga com um conjunto amostral de dados, foram importados 8.127 registros individuais VERIS em formato JSON. Devido à modelagem das classes independentes para cada tipo de característica do incidente, os registros convertidos resultaram em 71.790 indivíduos (instâncias). A consistência da ontologia foi também validada através do HermiT Reasoner no Protégé 5.2. Algumas estatísticas da ontologia VERIS podem ser observadas na Tabela 5.4.

Com este maior volume de dados foi percebido sensível impacto de performance na interface com o usuário do Protégé, mesmo após a customização da memória no ambiente Java. Na tentativa de contornar estes problemas, foram realizados testes utilizando o Apache Jena Fuseki como *triplestore* no modo *Standalone Server*. Esta ferramenta se demonstrou mais estável e apresentou melhor desempenho nas pesquisas com a ontologia VERIS, sendo assim adotada em todos os demais experimentos deste trabalho.

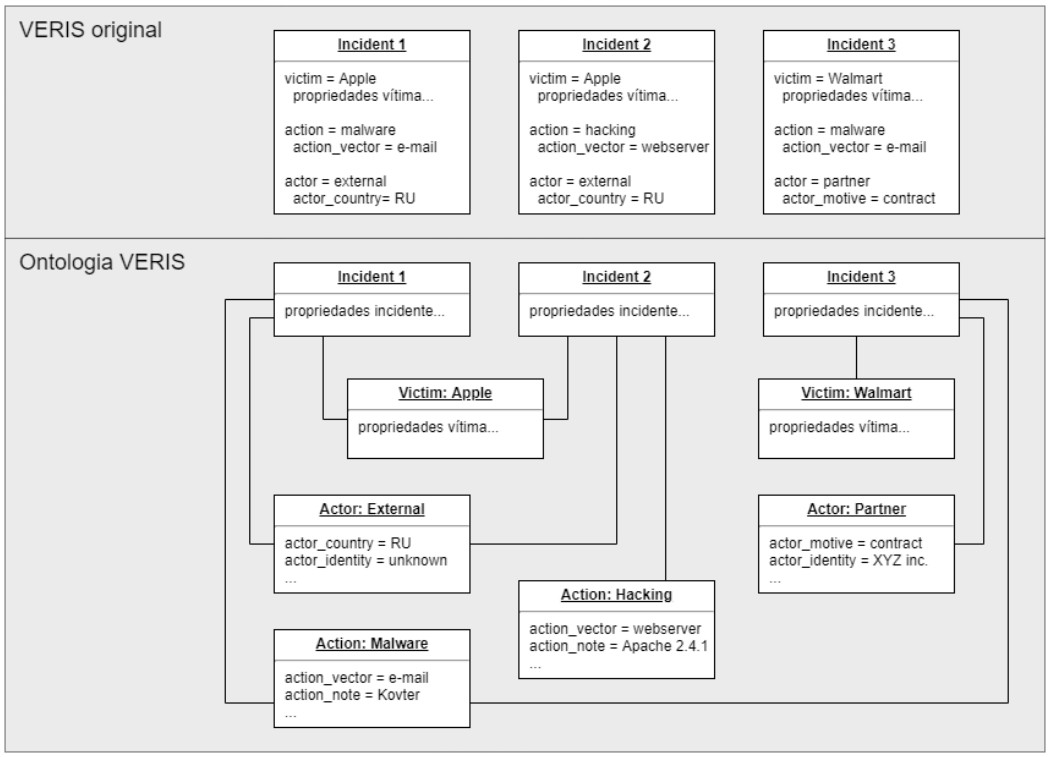

FIG. 5.3: Comparativo entre as estruturas do VERIS original e a ontologia VERIS

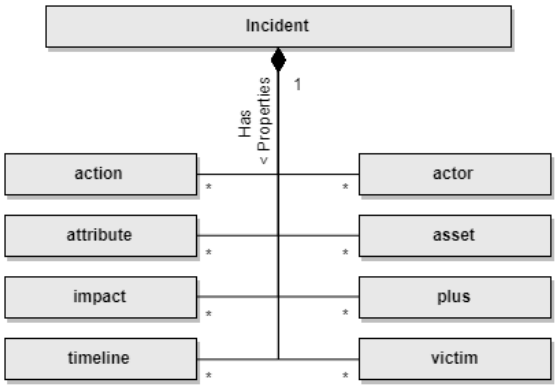

FIG. 5.4: Metamodelo da ontologia VERIS

As seguintes "perguntas de competência" foram definidas para os experimentos com a ontologia VERIS:

1) Qual o número de incidentes por ano, país e organização?
2) Quais os tipos de motivação registrados?
3) Qual a média geral de perda financeira considerando todos os incidentes e por país?
4) Quais os tipos de ação registrados e sua frequência?
5) Qual a contagem de ativos afetados?

TAB. 5.4: Estatísticas da ontologia VERIS com 8127 registros importados

Estatística	**Contagem**
Axiomas	526.576
Axiomas lógicos	454.210
Axiomas declarativos	72.030
Classes	12
Propriedades de objetos	9
Propriedades de dados	220
Indivíduos (instâncias)	71.790
Anotações	1
Expressividade DL	ALF(D)

6) Qual o status dos incidentes registrados?
7) Em quantos casos houve vazamento de dados?
8) Quais os tipos de dados vazados?

5.2.4 CORRELAÇÃO DE DADOS ENTRE AS ONTOLOGIAS VERIS E CSIHO

A criação das duas ontologias permitiu a construção de um terceiro cenário de experimentos buscando demonstrar o potencial de correlacionamento de dados entre os dois modelos e também o uso geral de consultas cruzadas entre ontologias.

É importante observar que embora os dois modelos possuam interseções em seu objetivo e representação, a natureza dos dados registrados no *dataset* do VERIS é bastante diferente daquela que se pretende aplicar ao CSIHO, o que limita significativamente este grupo de experimentos.

As seguintes perguntas de competência foram definidas:

1) Quais registros do VERIS descrevem um incidente do mesmo tipo que um registro específico encontrado na CSIHO?
2) Quais os tipos de ameaças reportadas no VERIS no mesmo ano e mês de um incidente específico do CSIHO?
3) Como um incidente registrado na CSIHO pode ser publicado na ontologia VERIS?

6 AVALIAÇÃO DE COMPETÊNCIAS DAS ONTOLOGIAS

Neste capítulo são apresentados os experimentos realizados, bem como a avaliação dos resultados das perguntas de competência das ontologias propostas. Eles foram divididos em três diferentes grupos com objetivos específicos. O primeiro grupo consiste nos experimentos realizados com a ontologia CSIHO demonstrando a eficácia do modelo ao responder as perguntas de competência que foram desafiadoras no estudo de caso do WannaCry, bem como alguns cenários hipotéticos. O segundo grupo consiste nos experimentos com a ontologia VERIS, demonstrando os benefícios de se estender o modelo original VERIS para uma ontologia e identificando as contribuições obtidas com esta mudança. No terceiro grupo são realizados experimentos com as duas ontologias, VERIS e CSIHO, demonstrando maneiras de se correlacionar dados entre elas, o tipo de informação que pode ser obtida no contexto apresentado e melhorias que enriqueceriam potencialmente a relevância destas informações.

As perguntas de competência são respondidas através de consultas SPARQL, fazendo uso de prefixos que referenciam, principalmente, as URIs das ontologias apresentadas. Conforme explicado no capítulo 3, o uso de prefixos permite definir um espaço de nomes (*namespace*) para cada ontologia específica, permitindo assim referenciar seus elementos em uma pesquisa SPARQL sem a necessidade de repetir a URI completa a cada *pattern*, tornando assim a construção das consultas mais simples e facilitando a sua leitura.

6.1 COMPUTER SECURITY INCIDENT HANDLING ONTOLOGY (CSIHO)

Para cada consulta, parte-se da premissa de que o usuário realizou uma consulta prévia, mais abrangente, para identificar o seu objeto de interesse, ou seja, obteve uma lista de todos os incidentes registrados, ou pesquisou por uma palavra chave específica, para identificar o nome designado a um determinado incidente.

É importante destacar que as perguntas de competência apresentadas, bem como as consultas SPARQL construídas para respondê-las, são generalizáveis para qualquer incidente. Para ilustrar a sua aplicação, porém, foram utilizados neste trabalho registros exemplo inspirados, principalmente, no estudo de caso do incidente com o *ransomware* WannaCry (MOREIRA et al., 2017).

Pergunta de competência 1 - Quais os tipos de eventos relacionados a um dado incidente?

Com base no nome do incidente informado, "WannaCry 2017-05", é localizado o código identificador dos eventos relacionados e utilizando o predicado "rdf:type" é possível identificar a qual classe pertence cada um dos objetos. Na ontologia CSIHO foi definida uma classe para cada tipo de evento, portanto esta consulta permite identificar o tipo de evento. Como todo objeto instanciado em OWL também faz parte da classe "owl:NamedIndividual", aplicou-se um filtro para que esta informação fosse inibida. A consulta SPARQL é apresentada no código 6.1 e o resultado na Tabela 6.1.

Esta informação permitiria ao time de resposta a incidentes identificar os principais sistemas de segurança responsáveis pela identificação de um incidente e também aqueles que não apoiaram no processo, subsidiando decisões técnicas ou estratégicas a respeito da infraestrutura relacionada.

Os prefixos "rdf" e "owl" referenciam URIs que representam definições padrões de termos pelo W3C, permitindo o uso das classes "rdf:type" e "owl:NamedIndividual".

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>

SELECT ?event_type (COUNT(?event_type) AS ?count)
WHERE {
    ?incident_id csiho:incident_name "WannaCry 2017-05" ;
                 csiho:has_event ?event_id .
    ?event_id rdf:type ?event_type .
    FILTER (?event_type != owl:NamedIndividual) .
}
GROUP BY ?event_type
```

Código 6.1: CSIHO - Consulta SPARQL - Pergunta de competência 1

TAB. 6.1: CSIHO - Resultado da consulta SPARQL 1

event_type	count
csiho:Sysmon_Event	2
csiho:IPS_Event	2

Pergunta de competência 2 - Quais os ativos afetados relacionados a um dado incidente?

A consulta, apresentada no código 6.2, obtém o código identificador do incidente "WannaCry 2017-05", identifica os eventos relacionados e em seguida quais os ativos afetados por cada um destes eventos. Os atributos desejados são então extraídos de cada ativo e exibidos. O resultado é apresentado na Tabela 6.2.

Esta informação pode apoiar o time de resposta nas atividades durante o tratamento de um incidente e também nas etapas de lições aprendidas pós incidente, como estudo de causa raiz, análise de vulnerabilidades, etc.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT ?hostname ?ip
WHERE {
    ?incident_id csiho:incident_name "WannaCry 2017-05" ;
                 csiho:has_event ?event_id .
    ?event csiho:has_affected_asset ?affected_assets .
    ?affected_assets csiho:ip_address ?ip ;
                     csiho:hostname ?hostname .
}
```

Código 6.2: CSIHO - Consulta SPARQL - Pergunta de competência 2

TAB. 6.2: CSIHO - Resultado da consulta SPARQL 2

hostname	ip
newton	10.0.51.110
euler	10.0.19.247
gauss	10.0.70.112
fermat	10.0.90.156
einstein	10.0.87.91
turing	10.0.20.125
neumann	10.0.74.188
pascal	10.0.48.254
fibonacci	10.0.16.80
hardy	10.0.9.108
nash	10.0.42.67
lovelace	10.0.2.102
hawking	10.0.92.190

Pergunta de competência 3 - Quais os incidentes relacionados a um determinado ativo?

A consulta, apresentada no código 6.3, identifica o ativo que possui o *hostname* "gauss", quais eventos estão relacionados a este ativo e em seguida quais incidentes estão relacionados a estes eventos. Obtém então a data de confirmação de cada incidente, bem como seu nome, tipo e descrição. O resultado é apresentado na Tabela 6.3.

Esta informação auxilia na identificação de alvos frequentes de ataques e pode subsidiar decisões técnicas ou estratégicas sobre configurações de sistemas, topologia de rede, gestão de vulnerabilidades e atualizações, processos de negócio, entre outras.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT ?incident_dt ?name ?type ?description
WHERE {
    ?asset csiho:hostname "gauss" .
    ?event csiho:has_affected_asset ?asset .
    ?incident csiho:has_event ?event ;
              csiho:incident_name ?name ;
              csiho:incident_type ?type ;
              csiho:incident_desc ?description ;
              csiho:incident_confirmation_dateTime ?incident_dt .
}
```

Código 6.3: CSIHO - Consulta SPARQL - Pergunta de competência 3

TAB. 6.3: CSIHO - Resultado da consulta SPARQL 3

incident_dt	name	type	description
2016-08-23T11:00:00	Vazamento 2016-07	data leak	Vazamento de dados de usuários do sistema de compras
2017-05-12T09:30:00	WannaCry 2017-05	ransomware	Ataque do ransomware WannaCry
2017-06-19T09:00:00	Petya 2017-06	ransomware	Ataque do ransomware Petya

Pergunta de competência 4 - Quais os eventos relacionados a um dado incidente?

Esta é uma extensão da primeira consulta, com a inclusão dos atributos data de detecção, descrição, rede de origem e endereço de origem para cada evento listado, como pode ser observado no código 6.4.

Conforme visto no capítulo 2, as declarações WHERE formam um padrão (*design pattern*) onde todas as condições precisam ser cumpridas para que se retorne alguma informação, ou seja, devem existir dados em todos os campos. Nos casos onde um campo pode ou não conter dados, como é o caso desta consulta, é preciso utilizar a cláusula "OPTIONAL". O resultado é apresentado na Tabela 6.4.

Esta consulta oferece um maior detalhamento técnico sobre os eventos relacionados a um incidente, apoiando o entendimento sobre as técnicas e estratégias de ataque e, consequentemente, subsidiando as ações de resposta.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT (STR(?det_dt) AS ?detection_dt) ?event_type ?event_description ?source_network
 ↪  ?source_ip
WHERE {
    ?incident_id csiho:incident_name "WannaCry 2017-05" ;
                 csiho:has_event ?event_id .
    ?event_id csiho:event_detection_dateTime ?det_dt ;
              rdf:type ?event_type .
    FILTER (?event_type != owl:NamedIndividual) .

    OPTIONAL { ?event_id csiho:event_description ?event_description . }
    OPTIONAL { ?event_id csiho:event_src_net ?source_network . }
    OPTIONAL { ?event_id csiho:event_src_address ?source_ip . }
}
ORDER BY ?detection_dt
```

Código 6.4: CSIHO - Consulta SPARQL - Pergunta de competência 4

TAB. 6.4: CSIHO - Resultado da consulta SPARQL 4

detection_dt	event_type	event_description	source_network	source_ip
2017-05-12T09:00:00	csiho:Sysmon_Event	Service STOPPED "h6hy63y2uhs"		10.0.1.51
2017-05-12T10:30:00	csiho:Sysmon_Event			
2017-05-12T11:45:00	csiho:IPS_Event		10.7.0.0/16	
2017-05-12T11:46:00	csiho:IPS_Event		10.12.0.0/16	

Pergunta de competência 5 - Quais as ações de resposta relacionadas a um dado incidente?

Esta consulta, apresentada pelo código 6.5, localiza o identificador do incidente "WannaCry 2017-05" e obtém do objeto "TimeLine_Occurrence" as relações com ações de resposta (*Courses of Action*) aplicadas, trazendo então a sua data de aplicação e descrições da ação e sua aplicação. De maneira a limitar o volume de dados retornados e demonstrar mais uma competência da ontologia, foram consideradas apenas as ações aplicadas na fase de CONTENÇÃO.

Para que as datas não sejam exibidas com a informação do tipo de dado, a variável "oc_dt" é convertida para uma *string*. O resultado da consulta está na Tabela 6.5.

Esta informação pode ser utilizada durante o processo de tratamento de incidentes, para acompanhar as ações já aplicadas, ou num momento posterior, para emissão de relatórios, exercícios de lições aprendidas ou a identificação de ações que podem ser reaplicadas em um novo cenário.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT (STR(?oc_dt) AS ?occurrence_dt) ?occurrence_notes ?coa_description
WHERE {
    ?incident_id csiho:incident_name "WannaCry 2017-05" .
    ?occurrence csiho:tl_occurrence_has_incident ?incident_id ;
                csiho:tl_occurrence_has_coa ?coa ;
                csiho:tl_occurrence_datetime ?oc_dt ;
                csiho:tl_occurrence_notes ?occurrence_notes ;
                csiho:tl_occurrence_incident_phase "Contenção" .
    ?coa csiho:coa_description ?coa_description .
}
ORDER BY ?occurrence_dt
```

Código 6.5: CSIHO - Consulta SPARQL - Pergunta de competência 5

TAB. 6.5: CSIHO - Resultado da consulta SPARQL 5

occurrence_dt	occurrence_notes	coa_description
2017-05-12T10:35:00	Aplicado coa_001	Isolar o segmento de rede 10.0.1.0/24
2017-05-12T10:38:00	Aplicado coa_002	Solicitar ação do fornecedor de antivirus (vacina/assinatura)
2017-05-12T10:51:00	Amostras enviadas para o fornencedor de antivírus	Enviar amostras dos binários de máquinas infectadas para o fornecedor de antivírus.
2017-05-12T10:51:00	Aplicado coa_003	Enviar amostras dos binários de máquinas infectadas para o fornecedor de antivírus.
2017-05-12T12:11:00	Bloqueio da porta 445 na conexão com a Filial RJ1	Bloquear a porta 445 (SMB) nos firewalls de borda nas conexões com redes de subsidiárias, caso apresentem tráfego suspeito.
2017-05-12T12:40:00	Bloqueio da porta 445 na conexão com a Filial RJ2	Bloquear a porta 445 (SMB) nos firewalls de borda nas conexões com redes de subsidiárias, caso apresentem tráfego suspeito.
2017-05-12T14:00:00	Bloqueada execução dos binários conhecidos do ransomware, com base nas amostras (hashes) coletadas e outras obtidas pelo fornecedor de antivírus.	Bloquear execução dos binários conhecidos do ransomware, com base nas amostras (hashes) coletadas e outras obtidas pelo fornecedor de antivírus;
2017-05-12T15:33:00	Solicitada nova distribuição do patch que corrige a vulnerabilidade no SMB do Windows (boletim MS17-010), com boot obrigatório.	Solicitar nova distribuição do patch que corrige a vulnerabilidade no SMB do Windows (boletim MS17-010), com boot obrigatório.
2017-05-12T19:40:00	Distribuida nova "vacina" para o WannaCry liberada pelo fornecedor de antivírus.	Distribuir nova "vacina" para o WannaCry liberada pelo fornecedor de antivírus.
2017-05-13T11:00:00	Realizado levantamento de hosts: (i) Sem os patches do SMB; (ii) Com antivírus desatualizado (sem a nova vacina).	Realizar levantamento de hosts: (i) Sem os patches do SMB; (ii) Com antivírus desatualizado (sem a nova vacina).
2017-05-13T18:00:00	Desconectadas (ou desligadas) estações com potencial de contaminação, identificadas pelo coa_008.	Desconectar ou desligar estações com potencial de contaminação, identificadas pelo coa_008.
2017-05-14T19:00:00	Implementado servidor web interno respondendo as URLs do Kill Switch do WannaCry (honeypot).	Implementar servidor web interno respondendo às URLs do Kill Switch do WannaCry (honeypot).

Pergunta de competência 6 - Dadas as características de um evento, quais incidentes estiveram relacionados?

Foi utilizado como exemplo para esta consulta a pesquisa de eventos de *Port Scan* nas portas 445 e 446, utilizadas pelo *ransomware* WannaCry na sua fase de disseminação. A primeira condição nesta consulta, apresentada pelo código 6.6, define que os dados de interesse estão na classe "IPS_Event". Em seguida são obtidos os campos que indicam a porta de destino bem como o tipo de resposta do IPS, e estes dados são filtrados de acordo com os critérios especificados. Finalmente, buscam-se os incidentes relacionados à estes eventos e seus respectivos nomes. Os resultados são apresentados na Tabela 6.6.

Trata-se de mais um caso de levantamento de informações técnicas para aprofundar o entendimento sobre um ataque e subsidiar as ações de resposta ao incidente.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT  ?incident (STR(?det_dt) AS ?detection_dt) ?resp (STR(?pt) AS ?port)
 ↪  ?source_net
WHERE {
    ?event a csiho:IPS_Event ;
             csiho:event_detection_dateTime ?det_dt ;
             csiho:event_src_net ?source_net ;
             csiho:event_dst_port ?pt ;
             csiho:event_ips_response ?resp .
    FILTER ((?pt = 446 || 445) && (?resp = "port scan"))
    ?incident_id csiho:has_event ?event ;
                 csiho:incident_name ?incident
}
ORDER BY ?det_dt
```

Código 6.6: CSIHO - Consulta SPARQL - Pergunta de competência 6

TAB. 6.6: CSIHO - Resultado da consulta SPARQL 6

incident	detection_dt	resp	port	source_net
WannaCry 2017-05	2017-05-12T11:45:00	port scan	445	10.7.0.0/16
WannaCry 2017-05	2017-05-12T11:46:00	port scan	445	10.12.0.0/16
Petya 2017-06	2017-06-19T14:35:00	port scan	445	10.3.0.0/16

Pergunta de competência 7 - Dado um novo incidente, quais ações prévias poderiam ser reaplicadas?

Esta consulta considera como critério de busca incidentes classificados como do tipo "*ransomware*". Apresentada pelo código 6.7, ela é equivalente à consulta 5, porém busca incidentes pelo seu tipo, e não por um nome de incidente específico, além de exibir alguns campos adicionais e filtrar ações aplicadas apenas nas fases de ERRADICAÇÃO e RECUPERAÇÃO. O resultado está na Tabela 6.7.

As informações fornecidas por esta consulta podem subsidiar a definição de um plano de ação em um novo incidente com características similares. A ideia é que o time de resposta a incidentes tenha acesso extenso e imediato ao conhecimento prévio, ainda que não tenha participado das ações anteriores.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT ?incident_name (STR(?dt) AS ?occurrence_dt) ?phase ?description ?notes
 ↪  ?analyst_name
{
    ?incident csiho:incident_type "ransomware" ;
              csiho:incident_name ?incident_name ;
             ^csiho:tl_occurrence_has_incident ?oc .
    ?oc csiho:tl_occurrence_has_coa ?coa ;
        csiho:tl_occurrence_datetime ?dt ;
        csiho:tl_occurrence_notes ?notes ;
        csiho:tl_occurrence_incident_phase ?phase .
    ?coa csiho:coa_description ?description ;
         csiho:has_coa_creation_analyst ?analyst_id .
    ?analyst_id csiho:analyst_name ?analyst_name .
    FILTER ((?phase = "Erradicação") || (?phase = "Recuperação"))
}
```

Código 6.7: CSIHO - Consulta SPARQL - Pergunta de competência 7

TAB. 6.7: CSIHO - Resultado da consulta SPARQL 7

incident_name	occurrence_dt	phase	description	notes	analyst_name
WannaCry 2017-05	2017-05-15T09:45:00	Erradicação	Bloquear extensões utilizadas pelo WannaCry nos servidores de arquivos.	Bloqueadas extensões utilizadas pelo WannaCry nos servidores de arquivos.	Luke
WannaCry 2017-05	2017-05-15T11:05:00	Erradicação	Definir e divulgar fluxo de tratamento de máquinas suspeitas/vulneráveis e comunicar as equipes.	Definido e divulgado fluxo de tratamento de máquinas suspeitas/vulneráveis e comunicar as equipes.	Luke
WannaCry 2017-05	2017-05-15T11:45:00	Erradicação	Orientar equipes de suporte local sobre procedimentos necessários para reconectar máquinas suspeitas desconectadas.	Enviada orientação para equipes de suporte local sobre procedimentos necessários para reconectar máquinas suspeitas desconectadas	Luke
WannaCry 2017-05	2017-05-16T14:00:00	Recuperação	Establecer fluxo padrão para autorização de desbloqueio de portas em conexões com filiais.	Reestabelecida conexão com a filial RJ1, de acordo com o fluxo padrão estabelecido pelo coa_014.	Lea
WannaCry 2017-05	2017-05-16T15:00:00	Recuperação	Establecer fluxo padrão para autorização de desbloqueio de portas em conexões com filiais.	Reestabelecida conexão com a filial RJ2, de acordo com o fluxo padrão estabelecido pelo coa_014.	Lea

Pergunta de competência 8 - Quantas ações foram aplicadas em cada fase de um determinado incidente?

As consultas 5 e 7 juntas listam todas as ações de resposta aplicadas ao incidente "WannaCry 2017-05". Esta consulta, apresentada pelo código 6.9, consolida a lista de ações aplicadas ao incidente e faz a sua contagem de acordo com cada fase do tratamento. O resultado está na Tabela 6.9.

Esta consulta fornece métricas para avaliação posterior sobre a efetividade das ações de resposta, bem como para exercícios de lições aprendidas e comparação entre incidentes.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT ?phase (COUNT(?phase) AS ?count)
WHERE {
    ?incident_id csiho:incident_name "WannaCry 2017-05" .
    ?occurrence csiho:tl_occurrence_has_incident ?incident_id ;
                    csiho:tl_occurrence_has_coa ?coa ;
                    csiho:tl_occurrence_incident_phase ?phase .
}
GROUP BY ?phase
ORDER BY DESC(?count)
```

Código 6.8: CSIHO - Consulta SPARQL - Pergunta de competência 8

TAB. 6.8: CSIHO - Consulta SPARQL - Pergunta de competência 8

phase	count
Contenção	11
Erradicação	3
Recuperação	2

Pergunta de competência 9 - Qual o tempo desde a detecção e a conclusão de um determinado incidente?

Esta pesquisa, apresentada pelo código 6.9, simplesmente obtém as datas de início e fim do incidente informado "WannaCry 2017-05" e realiza o cálculo de tempo.

Esta operação possui limitações no SPARQL: é preciso calcular a diferença de tempo entre cada componente da data, isto é, dia, mês, ano e horas, depois somá-las. Como não existe uma função que contenha dados de calendário, é necessário "arredondar" o total de dias dos meses e anos, como pode ser observado nas linhas contendo os comandos BIND. Como o objetivo final é consumir este resultado em algum sistema, isto pode ser facilmente resolvido com o uso de outras linguagens modernas, implementando-se diretamente na interface de visualização dos dados.

O resultado desta pesquisa está na Tabela 6.9 e fornece uma métrica para avaliação de impacto, efetividade e evolução de maturidade do processo de tratamento de incidentes.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>

SELECT ?description (STR(?conf_date) AS ?confirmation_dt) (STR(?end_date) AS ?end_dt)
 ↪  ?total_days
WHERE {
    ?incident_id csiho:incident_name "WannaCry 2017-05" ;
                 csiho:incident_confirmation_dateTime ?conf_date ;
                 csiho:incident_end_dateTime ?end_date ;
                 csiho:incident_desc ?description .

    BIND((hours(?end_date)-hours(?conf_date))/24 AS ?hd)
    BIND(day(?end_date)-day(?conf_date) AS ?dd)
    BIND(((month(?end_date)-month(?conf_date))*30) AS ?md)
    BIND(((year(?end_date)-year(?conf_date))*365) AS ?yd)
    BIND((?hd + ?dd + ?md + ?yd) AS ?total_days)
}
```

Código 6.9: CSIHO - Consulta SPARQL - Pergunta de competência 9

TAB. 6.9: CSIHO - Resultado da consulta SPARQL 9

description	confirmation_date	end_date	total_days
Ataque do ransomware WannaCry	2017-05-12T09:30:00	2017-05-21T09:00:00	9.0

6.2 ONTOLOGIA CRIADA A PARTIR DO VERIS

A primeira consulta realizada (código 6.10) lista todas as organizações vítimas dos incidentes registrados no *dataset* VERIS. Esta *query*, com 7.864 resultados, não possui grande utilidade prática, porém uma revisão visual da lista resultante revela um padrão que destaca alguns aspectos importantes sobre o modelo original:

a) Há inconsistência na descrição do campo "vítima". Por exemplo, as amostras apresentadas pelas Tabelas 6.10, 6.11 e 6.12 sugerem que uma mesma organização foi descrita com nomes diferentes;

b) Por utilizar originalmente o formato JSON para descrever os incidentes, na implementação original do VERIS é necessário repetir as descrições das organizações vítimas em cada um dos registros de incidentes. Na ontologia proposta, as organizações vítimas são objetos relacionados ao incidente, portanto um único indivíduo "vítima" poderia estar relacionado a diversos incidentes, tornando o registro mais eficiente e menos sujeito à inconsistências na descrição.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?incident_id ?victim_id
WHERE {
  ?incident veris:has_victim ?victim .
  ?incident veris:incident_id ?incident_id .
  ?victim veris:victim_victim_id ?victim_id .
}
ORDER BY ASC(?victim_id)
```

Código 6.10: VERIS - Consutla SPARQL 1

TAB. 6.10: Amostra de organizações vítimas (*keyword*: 7-ele)

victim_id	count
7-Eleven	8
7-ELEVEN, INC.	3
7-Elevan	1

A seguir são apresentadas as consultas que respondem às perguntas de competência definidas para a ontologia VERIS. Devido à natureza dos dados do *dataset* original VERIS, todos os resultados são voltados para pesquisas estatísticas sobre incidentes.

TAB. 6.11: Amostra de organizações vítimas (*keyword*: apple)

victim_id	count
Apple Inc	5
Apple Inc.	2
Apple	1

TAB. 6.12: Amostra de organizações vítimas (*keyword*: veteran)

victim_id	count
United States Department of Veterans Affairs	870
Department of Veterans Affairs	11
Department of Veterans' Affairs	2
U.S. Department of Veterans Affairs	2
United States Department of Veteran's Affairs	2
Veterans Affairs	2
Department of Veteran Affairs	1

Pergunta de competência 1 - Qual o número de incidentes por ano, país e organização?

A consulta 1(a), representada pelo código 6.11, obtém uma contagem de registros agrupados por ano. A lista completa possui 22 diferentes anos, e o incidente mais antigo data de 1971, por isto o resultado foi limitado em apenas 10 anos (Tabela 6.13).

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?year (COUNT (?year) AS ?count)
WHERE { ?timeline veris:timeline_incident_year ?year }
GROUP BY ?year
ORDER BY DESC(?year)
LIMIT 10
```

Código 6.11: VERIS - Consulta SPARQL 1(a)

TAB. 6.13: Resultado consulta 1(a) - Número de incidentes nos últimos dez anos

year	count
2016	1162
2015	1307
2014	958
2013	1943
2012	1282
2011	557
2010	588
2009	100
2008	85
2007	52

A consulta apresentada pelo código 6.12 obtém a contagem de registros agrupados por país, cujo resultado está na Tabela 6.14. A lista completa inclui 133 países, mas a *query* foi limitada a 10 resultados. O Brasil aparece em 41º na lista completa, com 7 incidentes reportados.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?country (COUNT (?country) AS ?count)
WHERE {
  ?victim veris:victim_country ?country
}
GROUP BY ?country
ORDER BY DESC(?count)
LIMIT 10
```

Código 6.12: VERIS - Consulta SPARQL 1(b)

TAB. 6.14: Resultado consulta 1(b) - Os dez países com mais incidentes

country	count
US	5889
GB	511
CA	325
Unknown	208
AU	124
NZ	95
IN	81
IE	59
JP	48
CN	47

A consulta apresentada pelo código 6.13 obtém a contagem de registros agrupados por organização. O *dataset* contém 4,998 nomes únicos de organizações e 72 incidentes informam organização "desconhecida" (*unknown*). Estes registros foram omitidos da consulta, e o resultado, apresentado na Tabela 6.15 foi limitado às 10 organizações com maior contagem.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?victim_id (COUNT(?victim_id) AS ?count)
WHERE {
  ?incident veris:has_victim ?victim .
  ?victim veris:victim_victim_id ?victim_id .
}
GROUP BY (?victim_id)
ORDER BY DESC(?count)
LIMIT 10
```

Código 6.13: VERIS - Consulta SPARQL 1(c)

TAB. 6.15: Resultado consulta 1(c) - As dez organizações com mais incidentes

victim_id	count
United States Department of Veterans Affairs	870
US National Security Agency (NSA)	14
Alabama Police Department	12
Department of Veterans Affairs	11
Experian	10
Internal Revenue Service	10
Alberta Health Services	9
Circle K	9
7-Eleven	8
Chase Bank	8

Pergunta de competência 2 - Quais os tipos de motivação registrados?

A consulta apresentada pelo código 6.14 contabiliza os motivos reportados em incidentes com agentes de ameaça externos (classe "actor_external_motive"). O resultado da pesquisa é mostrado na Tabela 6.16 e expõe mais uma possível inconsistência no *dataset* original, já que não existe diferença clara entre as definições *unknown*, NA e *other*.

A Tabela 6.17 apresenta o resultado para agentes de ameaça internos na organização, utilizando a classe "actor_internal_motive". Outras opções para esta *query* seriam "actor_partner_motive" e "actor_unknown_motive".

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?motive (COUNT (?motive) AS ?count)
WHERE { ?victim veris:actor_external_motive ?motive }
GROUP BY ?motive
ORDER BY DESC(?count)
```

Código 6.14: VERIS - Consulta SPARQL 2

TAB. 6.16: Resultado consulta 2(a) - Motivos para os ataques por agentes externos

motive	count
Financial	1800
Unknown	1463
Ideology	378
Espionage	270
Fun	233
Grudge	108
NA	32
Other	20
Secondary	9
Convenience	3
Fear	3

TAB. 6.17: Resultado consulta 2(b) - Motivos para os ataques por agentes internos

motive	count
NA	1935
Unknown	1038
Financial	747
Fun	256
Espionage	80
Convenience	74
Grudge	58
Other	45
Ideology	12
Fear	4

Pergunta de competência 3 - Qual a média geral de perda financeira considerando todos os incidentes e a média por país?

A consulta do código 6.15 obtém a média geral do impacto financeiro em todos os registros, cujo resultado é **8.160.747,54** (US$).

O código 6.16 apresenta a pesquisa da média por país. É importante observar que nem todos os registros possuem a informação de impacto financeiro, e por esta razão foi incluído um campo com a contagem dos registros que foram considerados. O resultado está na Tabela 6.18, onde é possível notar que uma minoria de registros possui esta informação.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT (AVG (?amount) AS ?average)
WHERE {
  ?impact veris:impact_overall_amount ?amount
}
```

Código 6.15: VERIS - Consulta SPARQL 3(a) - Média geral

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?country (COUNT (?country) AS ?count) (AVG (?amount) AS ?average_impact)
WHERE {
    ?incident veris:has_victim ?victim ;
              veris:has_impact ?impact .
    ?victim veris:victim_country ?country .
    FILTER (?country != "Unknown")
    ?impact veris:impact_overall_amount ?amount
}
GROUP BY ?country
ORDER BY DESC(?average_impact)
```

Código 6.16: VERIS - Consulta SPARQL 3(b) - Média por país

TAB. 6.18: Resultado consulta 3(b) - Média de perdas financeiras por país

country	count	average_impact
PK	1	72.000.000,00
OM	1	40.000.000,00
ZA	1	12.700.000,00
US	38	9.184.381,00
EC	1	9.000.000,00
AE	1	5.000.000,00
GB	11	2.085.460,54
TW	2	1.067.000,00
CA	2	320.000,00
CN	1	170.000,00
IL	1	100.000,00

Pergunta de competência 4 - Quais os tipos de ação registrados e sua frequência? ("ação" é equivalente a "ameaça" no contexto do VERIS)

Esta é uma consulta típica para "explorar" possíveis predicados em uma ontologia. O código 6.17 identifica todos os objetos do tipo "action" e em seguida obtém os predicados listados pelas triplas deste objeto, no caso específico, os tipos de ação registrados e suas propriedades agrupados pelo número de ocorrências, conforme resultado na Tabela 6.19.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?p (COUNT(?p) AS ?count)
WHERE {
  ?incident veris:has_action ?action .
  ?action ?p ?o .
  FILTER (?p != rdf:type) .
}
GROUP BY (?p)
ORDER BY DESC (?count)
```

Código 6.17: VERIS - Consulta SPARQL 4

TAB. 6.19: Resultado consulta 4 - Contagem dos tipos de "ação" e suas propriedades

p	count
veris:action_error_variety	2408
veris:action_hacking_variety	2405
veris:action_error_vector	2387
veris:action_hacking_vector	2185
veris:action_misuse_variety	2062
veris:action_physical_variety	1989
veris:action_malware_variety	1934
veris:action_misuse_vector	1831
veris:action_physical_vector	1568
veris:action_malware_vector	656
veris:action_social_vector	551
veris:action_social_variety	549
veris:action_social_target	540
veris:action_error_notes	344
veris:action_hacking_notes	243
veris:action_malware_name	227
veris:action_hacking_cve	197
veris:action_malware_cve	183
veris:action_misuse_notes	162
veris:action_malware_notes	123
veris:action_social_notes	69
veris:action_physical_notes	35
veris:action_environmental_variety	7
veris:action_unknown_notes	3
veris:action_environmental_notes	1

Pergunta de competência 5 - Qual a contagem de ativos afetados?

A consulta apresentada pelo código 6.18 agrupa a lista com os dez ativos mais afetados em todos os incidentes do *dataset*, conforme resultado na Tabela 6.20. Um total de 654 ativos de tipo "desconhecido" (*unknown*) foram filtrados.

A Tabela 10.1 (nos anexos) apresenta uma legenda com todas as possíveis descrições de *assets* que poderiam ser listados por esta consulta.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?assets (COUNT(?assets) as ?count)
WHERE {
  ?asset veris:asset_assets ?assets .
  FILTER (!regex(?assets, "unknown", "i"))
}
GROUP BY (?assets)
ORDER BY DESC(?count)
LIMIT 10
```

Código 6.18: VERIS - Consulta SPARQL 5

TAB. 6.20: Resultado consulta 5 - Os dez tipos de ativos mais afetados

assets	count
{'variety': 'M - Documents'}	1729
{'variety': 'S - Database'}	1563
{'variety': 'S - Web application'}	1409
{'variety': 'U - Desktop'}	676
{'variety': 'U - Laptop'}	586
{'variety': 'S - Mail'}	233
{'variety': 'T - ATM'}	212
{'variety': 'M - Flash drive'}	173
{'variety': 'U - Mobile phone'}	122
{'variety': 'S - File'}	105

Pergunta de competência 6 - Qual o status dos incidentes registrados?

A pesquisa apresentada pelo código 6.19 agrupa os incidentes pelo seu status, conforme resultado na Tabela 6.21.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?status (COUNT(?status) AS ?count)
WHERE {
  ?plus veris:plus_analysis_status ?status .
}
GROUP BY (?status)
ORDER BY DESC(?count)
```

Código 6.19: VERIS - Consulta SPARQL 6

TAB. 6.21: Resultado consulta 6 - Status dos incidentes

status	count
First pass	4849
Finalized	2823
Validated	135
Needs review	50

Pergunta de competência 7 - Em quantos casos houve vazamento de dados?

A consulta do código 6.20 obtém os detalhes sobre vazamento de dados dos incidentes, conforme resultado apresentado na Tabela 6.22.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?data_disclosure (COUNT(?data_disclosure) as ?count)
WHERE {
  ?attrib veris:attribute_confidentiality_data_disclosure ?data_disclosure
  }
GROUP BY (?data_disclosure)
ORDER BY DESC(?count)
```

Código 6.20: VERIS - Consulta SPARQL 7

TAB. 6.22: Resultado consulta 7 - Incidentes com vazamento de dados

data_disclosure	count
Yes	5261
Potentially	1984
Unknown	198
No	70

Pergunta de competência 8 - Quais os tipos de dados vazados?

O código 6.21 obtém os tipos de dados que foram vazados, filtrando a consulta por incidentes onde houve vazamento potencial ou confirmado, excluindo os registros de tipo desconhecido e limitando o resultado em 10 linhas, conforme apresentado na Tabela 6.23.

Alguns dos registros do *dataset* incluem também a informação da quantidade de dados vazados, como é o caso da última linha da Tabela.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?data (COUNT(?data) as ?count)
WHERE {
  ?attrib veris:attribute_confidentiality_data_disclosure ?disc .
  ?attrib veris:attribute_confidentiality_data ?data .
  FILTER ((?disc = "Yes" || ?disc ="Potentially") && !regex(?data, "unknown", "i"))
  }
GROUP BY (?data)
ORDER BY DESC(?count)
LIMIT 10
```

Código 6.21: VERIS - Consulta SPARQL 8

TAB. 6.23: Resultado consulta 8 - Os dez tipos mais frequentes de dados vazados

data	count
{'variety': 'Personal'}	1558
{'variety': 'Medical'}	676
{'variety': 'Payment'}	559
{'variety': 'Internal'}	404
{'variety': 'Credentials'}	305
{'variety': 'Secrets'}	247
{'variety': 'Bank'}	198
{'variety': 'System'}	189
{'variety': 'Classified'}	162
{'variety': 'Medical', 'amount': 1}	141

Enriquecimento de dados com uso de consultas externas.

O código 6.22 utiliza uma estratégia comum em consultas SPARQL que demonstra um dos grandes potenciais da linguagem OWL e dos conceitos de *linked data*, realizando a consulta a uma fonte externa para agregação de dados.

No cenário apresentado, o código de país padrão ISO 3166-1, com dois caracteres, de cada registro da classe "victim" da ontologia VERIS é enviado em uma consulta externa à ontologia da wikidata.org[2] para obter o nome do país por extenso. A informação é então inserida no *dataset* VERIS, através da criação de um novo atributo que foi chamado de "victim_country_name". Para se manter a consistência do idioma original foram obtidos os nomes em inglês, mas diversos idiomas estão disponíveis.

Os códigos 6.23 e 6.24 apresentam, respectivamente, as consultas 1(b) e 3(b) desta seção, modificadas para utilizarem o atributo "victim_country_name", gerando então resultados com os nomes dos países por extenso, conforme Tabelas 6.24 e 6.25.

É importante observar que as informações externas poderiam ter sido obtidas e agregadas na saída de cada consulta individual em tempo de execução. A opção de obter os dados previamente e retê-los localmente é uma forma de garantir a sua disponibilidade e melhor performance das *queries* SPARQL, mas esta é uma decisão que dependerá de cada cenário e caso de uso.

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX wdt:<http://www.wikidata.org/prop/direct/>

INSERT { ?victim veris:victim_country_name ?country_name . }

WHERE {  ?victim veris:victim_country ?country .

  SERVICE <https://query.wikidata.org/bigdata/namespace/wdq/sparql> {
    ?country_id wdt:P297 ?country ;
                rdfs:label ?country_name .
    FILTER(LANG(?country_name) = "en")
  }
}
```

Código 6.22: Inclusão dos nomes de países por extenso na ontologia VERIS a partir de consulta externa à wikidata.org

[2]A Wikidata é uma base de conhecimento gratuita e livre "que pode ser lida e editada tanto por humanos quanto por máquinas" e também é o repositório central de dados estruturados de projetos da Wikimedia, incluíndo Wikipedia, Wikivoyage, Wikisource, entre outros (WIKIDATA, 2018).

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?country (COUNT (?country) AS ?count)
WHERE { ?victim veris:victim_country_name ?country }
GROUP BY ?country
ORDER BY DESC(?count)
LIMIT 10
```

Código 6.23: Consulta VERIS 1(b) usando o novo atributo "victim_country_name"

TAB. 6.24: Resultado da consulta VERIS 1(b) com nomes dos países por extenso

country	count
United States of America	5889
United Kingdom	511
Canada	325
Australia	124
New Zealand	95
India	81
Ireland	59
Japan	48
People's Republic of China	47
South Korea	46

```
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?country (COUNT (?country) AS ?count) (AVG (?amount) AS ?average_impact)
WHERE {
    ?incident veris:has_victim ?victim ;
              veris:has_impact ?impact .
    ?victim veris:victim_country_name ?country .
    FILTER (?country != "Unknown")
    ?impact veris:impact_overall_amount ?amount
}
GROUP BY ?country
ORDER BY DESC(?average_impact)
```

Código 6.24: Consulta VERIS 3(b) usando o novo atributo "victim_country_name"

TAB. 6.25: Resultado da consulta VERIS 3(b) com nomes dos países por extenso

country	count	average_impact
Pakistan	1	72.000.000,00
Oman	1	40.000.000,00
South Africa	1	12.700.000.00
United States of America	38	9.184.381,00
Ecuador	1	9.000.000,00
United Arab Emirates	1	5.000.000,00
United Kingdom	11	2.085.460,54
Taiwan	2	1.067.000,00
Canada	2	320.000,00
People's Republic of China	1	170.000,00
Israel	1	100.000,00

Pergunta de competência 1 - Quais registros do VERIS descrevem um incidente do mesmo tipo que um registro específico encontrado na CSIHO?

A consulta apresentada no código 6.25 obtém a propriedade sobre o tipo de incidente (*ransomware*) do registro "WannaCry 2017-05" na ontologia CSIHO e em seguida lista os incidentes da ontologia VERIS que possuem em sua descrição a palavra chave obtida na primeira consulta. O resultado é apresentado pela Tabela 6.26, limitado na própria *query* em cinco registros apenas.

Pode-se observar mais uma possível inconsistência no *dataset* VERIS, com incidentes diferentes apresentando exatamente a mesma descrição.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?type ?year ?incident_id ?desc
WHERE {

  SERVICE <http://localhost:3030/csiho/sparql>
  { SELECT ?type
    WHERE {
      ?incident_id csiho:incident_name "WannaCry 2017-05" ;
                   csiho:incident_type ?type .
    }
  }

  SERVICE <http://localhost:3030/veris/sparql>
  { SELECT ?incident_id ?year ?desc
    WHERE {
      ?incident veris:incident_id ?incident_id ;
                veris:summary ?desc .

      OPTIONAL { ?incident veris:has_timeline ?timeline . }
      OPTIONAL { ?timeline veris:timeline_incident_year ?year . }
   }
  }

  # Filtra os registros cuja descrição possui a palavra chave na variável "?type"
  FILTER (regex(?desc, ?type, "i"))
}
ORDER BY DESC(?year)
LIMIT 5
```

Código 6.25: Consulta 1 correlacionando as ontologias VERIS e CSIHO

TAB. 6.26: Resultado da consulta correlacionando as ontologias VERIS e CSIHO

type	y	incident_id	desc
ransomware	2016	AFECA470-56B5-4D3C-83F6-3915ADF30ED8	first ever recorded incident of ransomware attack in India
ransomware	2016	C5CDEB24-ECD6-4861-90B8-FA879CF0BAE6	first ever recorded incident of ransomware attack in India
ransomware	2016	BBE90679-5E5A-43FD-B19C-0CEA3D7D0F37	A Los Angeles hospital paid a ransom in bitcoins equivalent to about $17,000 to hackers who infiltrated and disabled its computer network, the medical center's chief executive said Wednesday. It was in its best interest of Hollywood Presbyterian Medical Center to pay the ransom of 40 bitcoins currently worth $16,664 dollars after the network infiltration that began Feb. 5, CEO Allen Stefanek said in a statement. In the attacks, often known as ransomware, the hackers encrypt a network's data and provide a digital decryption key to unlock it for a price. The quickest and most efficient way to restore our systems and administrative functions was to pay the ransom and obtain the decryption key, Stefanek said. In the best interest of restoring normal operations, we did this. Hospital employees noticed the network problems on Feb. 5, and it became clear there was a malware infiltration that was disabling the network. Computer experts and law enforcement were immediately informed, Stefanek said. On Monday, 10 days after the attack, the network was in full operation again, he said. FBI spokeswoman Laura Eimiller said the agency is investigating the extortion plot, but she could not immediately provide further details. Neither law enforcement nor the hospital gave any indication of who might have been behind the attack or whether there are any suspects. Patient care was not affected by the hacking, and there is no evidence any patient data was compromised, Stefanek said. The 434-bed hospital in the Los Feliz area of Los Angeles was founded in 1924. It was sold to CHA Medical Center of South Korea in 2004. It offers a range of services including emergency care, maternity services, cancer care, physical therapy, and specialized operations such as fetal and orthopedic surgeries.
ransomware	2016	4ADC04AB-6B96-4FE1-BF07-2BD41CAFF250	Allergy clinic in Colorado recently found evidence of healthcare ransomware on its computer systems, causing the facility to shut down its server.
ransomware	2016	A9BA74BA-306D-46B5-AD23-290D5A3EF43B	Allergy clinic in Colorado recently found evidence of healthcare ransomware on its computer systems, causing the facility to shut down its server.

Pergunta de competência 2 - Quais os tipos de ameaças reportadas no VERIS no mesmo ano e mês de um incidente específico do CSIHO?

Este tipo de pesquisa poderia ser útil na identificação de similaridades com outros ataques para identificar, por exemplo, uma campanha de ataque global, buscar subsídios para análise da ameaça, ou ainda obter dados estatísticos para comparação de performance (*benchmarking*) entre outras "vítimas", em exercícios de lições aprendidas.

A consulta apresentada pelo código 6.26 obtém a data de confirmação do incidente "LizardStresser 2016-06" e extrai ano e mês. Em seguida obtém todos os objetos do tipo "action" do VERIS, neste mesmo período, e os agrupa contando o numero de ocorrências, limitando a saída em 10 resultados (Tabela 6.27).

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?p (COUNT(?p) AS ?count)
WHERE {
    ?csiho_incident_id csiho:incident_name "LizardStresser 2016-06" ;
                       csiho:incident_confirmation_dateTime ?csiho_dt .
    BIND(year(?csiho_dt) AS ?csiho_year)
    BIND(month(?csiho_dt) AS ?csiho_month)

    SERVICE <http://localhost:3030/veris/sparql>  {
              ?incident veris:has_timeline ?timeline ;
                  veris:has_action ?action .
              ?timeline veris:timeline_incident_year ?csiho_year ;
                  veris:timeline_incident_month ?csiho_month .
              ?action ?p ?o .
              FILTER (?p != rdf:type)
    }
}
GROUP BY ?p
ORDER BY DESC(?count)
LIMIT 10
```

Código 6.26: Consulta 2 correlacionando as ontologias VERIS e CSIHO

TAB. 6.27: As dez maiores ameaças, segundo a VERIS, no mesmo mês do incidente "LizardStresser 2016-06", registrado na CSIHO

p	count
veris:action_physical_variety	37
veris:action_error_variety	34
veris:action_error_vector	34
veris:action_hacking_variety	31
veris:action_hacking_vector	31
veris:action_physical_vector	31
veris:action_misuse_variety	21
veris:action_misuse_vector	21
veris:action_hacking_notes	9
veris:action_malware_variety	6

Pergunta de competência 3 - Como um incidente registrado na CSIHO pode ser publicado na ontologia VERIS?

Tendo em vista que a CSIHO fundamenta os processos de resposta a incidente dentro de uma organização e que o VERIS tem o objetivo de compartilhar estatísticas globalmente, em um cenário hipotético onde o projeto tenha adotado a extensão para a ontologia aqui proposta, seria relevante um mecanismo para adicionar anonimamente registros da CSIHO na ontologia VERIS.

A consulta apresentada pelo código 6.27 (na verdade um *update request*) adiciona um novo incidente na ontologia VERIS a partir de informações obtidas da ontologia CSIHO.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>
PREFIX afn: <http://jena.hpl.hp.com/ARQ/function#>

INSERT {
  ?newURI_incident a veris:Incident ;
                   veris:summary ?desc ;
                   veris:incident_id ?incident_id ;
                   veris:has_timeline ?newURI_timeline ;
                   veris:has_victim ?newURI_victim .
  ?newURI_timeline a veris:timeline ;
                   veris:timeline_incident_day ?day ;
                   veris:timeline_incident_month ?month ;
                   veris:timeline_incident_year ?year .
  ?newURI_victim a veris:victim ;
                 veris:victim_country "BR" .
}
WHERE {
  BIND(afn:struuid() as ?incident_id )
  BIND(CONCAT(?incident_id, "_incident") as ?veris_incident)
  BIND(URI(CONCAT("veris:", ENCODE_FOR_URI(?veris_incident))) as ?newURI_incident)
  BIND(CONCAT(?incident_id, "_timeline") as ?veris_timeline)
  BIND(URI(CONCAT("veris:", ENCODE_FOR_URI(?veris_timeline))) as ?newURI_timeline)
  BIND(CONCAT(?incident_id, "_victim") as ?veris_victim)
  BIND(URI(CONCAT("veris:", ENCODE_FOR_URI(?veris_victim))) as ?newURI_victim)

  SERVICE <http://localhost:3030/csiho/sparql>
  { SELECT ?csiho_inc_id ?type ?desc ?day ?month ?year
    WHERE {
      ?csiho_inc_id csiho:incident_name "WannaCry 2017-05" ;
                    csiho:incident_type ?type ;
                    csiho:incident_desc ?desc ;
                    csiho:incident_confirmation_dateTime ?dt .
      BIND(DAY(?dt) as ?day)
      BIND(MONTH(?dt) as ?month)
      BIND(YEAR(?dt) as ?year)
    }
  }
}
```

Código 6.27: Inclusão de um incidente da CSIHO na ontologia VERIS

A atualização obtém um identificador universal único (*Universally Unique identifier*, ou UUID), conforme padrão de identificação de incidentes adotado pelo VERIS, e cria os objetos necessários seguindo a notação definida para a ontologia VERIS. A geração de UUIDs diretamente no código SPARQL é possível graças a uma extensão oferecida pelo projeto Apache Jena (THE APACHE SOFTWARE FOUNDATION, 2017).

O código 6.28 apresenta a consulta ao incidente "WannaCry" diretamente na ontologia VERIS, cujo resultado pode ser observado na Tabela 6.28. Pelo fato de ser um exemplo e para garantir a apresentação adequada do código, apenas um pequeno grupo de atributos foi selecionado para a criação do novo registro. Na prática, porém, todas as informações pertinentes poderiam ter sido migradas sem dificuldades.

```
PREFIX csiho: <http://gb.moreira.nom.br/csiho.owl#>
PREFIX veris:<http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl#>

SELECT ?incident_id ?description ?year ?month ?day ?country
WHERE {
  ?incident veris:incident_id ?incident_id ;
            veris:summary ?description ;
            veris:has_timeline ?timeline ;
            veris:has_victim ?victim .
  ?timeline veris:timeline_incident_year ?year ;
            veris:timeline_incident_month ?month ;
            veris:timeline_incident_day ?day .
  ?victim veris:victim_country ?country .
  FILTER (regex(?description, "wannacry", "i"))
}
```

Código 6.28: Consulta ao incidente "WannaCry" na ontologia VERIS

TAB. 6.28: Resultado da consulta ao incidente "WannaCry" na ontologia VERIS

incident_id	description	year	month	day	country
752eb481-ad5e-4899-9313-256ad9a0e7f4	Ataque do ransomware WannaCry	2017	5	12	BR

7 CONCLUSÃO

Os avanços da tecnologia da informação proporcionaram novas oportunidades de realizações para a sociedade, tendo modificado profundamente as relações humanas, as profissões, os meios de comunicação, as organizações e a economia. Estas mudanças, porém, tornam a humanidade cada vez mais dependente, em diversos aspectos, dos sistemas de informação e da Internet. Esta dependência aliada à expansão dos recursos computacionais vêm culminando em sistemas cada vez mais complexos e interconectados, aumentando a sua exposição a riscos e tornando a sua proteção uma tarefa cada dia mais desafiadora.

As estatísticas expõem que apesar dos crescentes investimentos em Segurança Cibernética, o número de ciberincidentes continua aumentando em frequência e gravidade, com ataques motivados por questões políticas e financeiras, muitas vezes financiados por Estados como parte de ofensivas de uma verdadeira guerra cibernética.

Neste contexto, embora a percepção global seja de que a ocorrência de incidentes é praticamente inevitável, a literatura mostra que as iniciativas de Segurança Cibernética são, normalmente, mais voltadas para a prevenção do que para a resposta a incidentes, com muitas organizações mal preparadas e frequentemente ignorando processos fundamentais de tratamento de ciberincidentes.

Como uma maneira de oferecer recursos para contornar estes problemas o objetivo deste trabalho foi propor um modelo para tratamento de incidentes, que inspirado pelos ideais da Web semântica foi descrito na forma de uma ontologia, facilmente extensível e integrável com outros modelos. Esta ontologia provê fundamento para o processo de tratamento de incidentes, além de habilitar inferências lógicas e simplificar o processo de transferência de informação e conhecimento dentro de um contexto de trabalho colaborativo.

Com base em princípios simples, porém com sólidas referências, foi construída a *Computer Security Incident Handling Ontology*, ou CSIHO, e criado um *knowledge base* exemplo com registros inspirados em um estudo de caso sobre o incidente com o *ransomware* WannaCry. Seguindo a sua filosofia colaborativa, a ontologia foi publicada no site GitHub (https://github.com/moreiragb/csiho) para uso pela comunidade profissional ou desenvolvimento de trabalhos futuros pela academia.

Devido à representatividade do modelo VERIS, *Vocabulary for Event Recording and Incident Sharing*, um vocabulário com definições para o registro colaborativo de dados

sobre incidentes, e seu extenso *dataset* com informações estatísticas, originalmente codific-codificado em formato JSON, optou-se também por construir uma ontologia e *knowledge base* em formato OWL baseados no modelo. Devido à filosofia colaborativa proveniente dos ideais da Web Semântica, o objetivo foi demonstrar que uma representação em OWL seria mais adequada para o VERIS, tornando-o ainda mais colaborativo e acessível.

Para subsidiar a avaliação das ontologias foram definidas "perguntas de competência", de uso geral, para a CSIHO e a ontologia VERIS. De maneira a demonstrar as contribuições do trabalho, bem como o cumprimento dos requisitos definidos, foram estabelecidos três grupos de experimentos com cenários específicos, descritos a seguir.

O primeiro grupo de experimentos mostrou como a ontologia CSIHO permitiria um acompanhamento completo e eficiente de um incidente. Utilizando simples consultas SPARQL, foi possível responder todas as perguntas de competência definidas utilizando menos recursos e tempo do que aqueles informados pelo estudo de caso referenciado, isto é, partindo do princípio de que o modelo apoiaria a documentação adequada do incidente, o tempo para obtenção dos dados seria da ordem de minutos (criação das queries) contra dias no caso real. A CSIHO é, ainda, a única ontologia que define e implementa o conceito de "evento de segurança", além de permitir o registro histórico de ocorrências relacionadas ao incidente, permitindo construir uma linha do tempo.

O segundo grupo de experimentos demonstrou como a extensão do projeto VERIS para uma ontologia tornou a manipulação das informações mais simples e a representação de dados mais eficiente, respondendo às perguntas de competência e trazendo também novas possibilidades de uso para o modelo. O seu desenvolvimento permitiu ainda identificar inconsistências no modelo original, que foram devidamente reportadas aos seus mantenedores. O terceiro grupo demonstrou novos casos de uso através da correlação de dados entre ontologias, explorando ainda mais o potencial de recursos introduzidos pelo uso das linguagens OWL e SPARQL, e demonstrando o princípio de integrabilidade definido como requisito para a CSIHO.

Como sugestão de trabalhos futuros, propõe-se a expansão das subclasses de "eventos" e o desenvolvimento de conectores para ingestão de *logs* de ferramentas de segurança em tempo real, objetivando explorar cenários de inferência lógica (*reasoning*) e a aplicação de técnicas de IA para detecção automática de incidentes. Isto poderia motivar, ainda, a criação de uma ontologia dedicada ao tratamento de eventos, conectada à CSIHO.

Como oportunidade de desenvolvimento e expansão dos modelos, é sugerida a construção de uma ferramenta protótipo, implementando o CSIHO, com o intuito de realizar provas de conceito em ambientes reais, permitindo a validação do modelo e o seu amadu-

recimento, com a identificação de melhorias. Além disso, a expansão da ontologia VERIS, com a proposta de atualização do modelo corrente para o novo formato desenvolvido, permitiria oferecer os benefícios identificados por este trabalho para toda a comunidade atual do VERIS.

No âmbito da estruturação formal das ontologias, sugere-se como trabalho futuro a adequação das ontologias desenvolvidas à alguma ontologia de fundamentação. As ontologias de fundamentação buscam alinhar definições essenciais e globais desassociadas de domínios específicos.

Finalmente, é sugerida a avaliação dos aspectos de segurança dos *triplestores* e do protocolo SPARQL, como por exemplo uso de criptografia para garantia de confidencialidade, uso de autenticação para garantir o acesso e modificação apenas por agentes autorizados e identificação de possíveis fraquezas no protocolo, com o intuito de medir o nível de confiabilidade atual no uso destas tecnologias e propor eventuais melhorias.

8 REFERÊNCIAS BIBLIOGRÁFICAS

AB RAHMAN, N. H.; CHOO. A survey of information security incident handling in the cloud. **Computers & Security**, v. 49, p. 45–69, 2015.

BAADER, F.; CALVANESE, D.; MCGUINNESS, D. L.; NARDI, D. ; PATEL-SCHNEIDER, P. F., editores. **The Description Logic Handbook: Theory, Implementation, and Applications**. New York, NY, USA: Cambridge University Press, 2003. ISBN 0-521-78176-0.

BAADER, F.; HORROCS, I.; LUTZ, C. ; SATTLER, U. **An Introduction to Description Logic**. 1. ed. [S.l.]: Cambridge University Press, 2017. ISBN 978-0521873611.

BARROS, O. S. R.; GOMES, U. D. M. ; FREITAS, W. L. D. **Desafios estratégicos para segurança e defesa cibernética**. Brasília: Secretaria de Assuntos Estratégicos da Presidência da República, 2011. 216 p. ISBN 978-85-85142-32-2.

BASKERVILLE, R.; SPAGNOLETTI, P. ; KIM, J. Incident-centered information security: Managing a strategic balance between prevention and response. **Information & management**, v. 51, n. 1, p. 138–151, 2014.

BIZER, C.; HEATH, T. ; BERNERS-LEE, T. Linked data-the story so far. **International journal on semantic web and information systems**, v. 5, n. 3, p. 1–22, 2009.

BLACKWELL, C. A security ontology for incident analysis. In: PROCEEDINGS OF THE SIXTH ANNUAL WORKSHOP ON CYBER SECURITY AND INFORMATION INTELLIGENCE RESEARCH, 6., 2010. **Anais**... [S.l.: s.n.], 2010, p. 46.

CERT.BR. Estatísticas dos Incidentes Reportados ao CERT.br - Valores acumulados de 1999 a 2017. Disponível em: <https://www.cert.br/stats/incidentes/>. Acesso em: 03 abr. de 2018.

CICHONSKI, P.; MILLAR, T.; GRANCE, T. ; SCARFONE, K. **Computer Security Incident Handling Guide: Recommendations of the National Institute of Standards and Technology**. [S.l.: s.n.], 2012. 79 p. (Relatório Técnico, NIST SP 800-61r2).

CMMI INSTITUTE. What Is Capability Maturity Model Integration (CMMI)?. Disponível em: <http://cmmiinstitute.com/capability-maturity-model-integration>. Acesso em: 03 mar. de 2017.

COMPUTER WORLD. Maioria das empresas brasileiras não tem plano de resposta a incidentes. Disponível em: <http://computerworld.com.br/maioria-das-empresas-brasileiras-nao-tem-plano-de-resposta-incidentes>. Acesso em: 27 mar. de 2017.

CTIR GOV. Centro de Tratamento de Incidentes de Segurança de Redes de Computadores da Administração Pública Federal. Disponível em: <http://www.ctir.gov.br/>. Acesso em: 05 abr. de 2017.

DA VEIGA, A.; ELOFF, J. A framework and assessment instrument for information security culture. **Computers & Security**, v. 29, n. 2, p. 196–207, 2010. Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/S0167404809000923>. Acesso em: 08 fev. de 2017.

DHS. **Establishing Community-Relevant Data Categories in Support of a Cyber Incident Data Repository**. Washington, DC: USA Department of Homeland Security (DHS), 2015. 51 p. (Relatório Técnico).

DUCHARME, B. **Learning SPARQL: querying and updating with SPARQL 1.1**. 2. ed. [S.l.]: O'Reilly Media, 2013. ISBN 978-1-449-37143-2.

ELOFF, M.; VON SOLMS, S. Information Security Management: A Hierarchical Framework for Various Approaches. **Computers & Security**, v. 19, n. 3, p. 243–256, 2000. Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/S0167404800886137>. Acesso em: 08 fev. de 2017.

EXIDA LLC. The Repository of Industrial Security Incidents: RISI. Disponível em: <http://www.risidata.com/>. Acesso em: 27 mar. de 2017.

F-SECURE. F-Secure State of Cyber Security 2017. Disponível em: <https://www.f-secure.com/documents/996508/1030743/cyber-security-report-2017>. Acesso em: 17 fev. de 2017.

RAY FERGUSON. Protégé Short Course - Protege vs. Databases. Disponível em: <https://www.youtube.com/watch?v=pg1Y9RszKFE>. Acesso em: 05 nov. de 2017.

FORTUNE. What to know about the Ashley Madison hack. Disponível em: <http://fortune.com/2015/08/26/ashley-madison-hack/>. Acesso em: 19 fev. de 2017.

FORTUNE. One of the World's Most Valuable Companies Is Considering an IPO. Disponível em: <http://fortune.com/2016/01/07/saudi-arabia-aramco-ipo/>. Acesso em: 12 fev. de 2017.

GIBBS, SAMUEL. Ransomware attack on San Francisco public transit gives everyone a free ride. Disponível em: <https://www.theguardian.com/technology/2016/nov/28/passengers-free-ride-san-francisco-muni-ransomeware>. Acesso em: 12 fev. de 2017.

GONZALEZ, J. J.; SAWICKA, A. A framework for human factors in information security. In: WSEAS INTERNATIONAL CONFERENCE ON INFORMATION SECURITY, RIO DE JANEIRO, 1., 2002. **Anais**... [S.l.: s.n.], 2002, p. 448–187.

GRISPOS, G. **On the enhancement of data quality in security incident response investigations**. 2016. Tese (Phd in Computer Science) – University of Glasgow, Glasgow, 2016.

HAGGARD, S.; LINDSAY, J. R. North korea and the sony hack: exporting instability through cyberspace. **AsiaPacific Issues**, v. 117, p. 1–8, 2015.

HARRIS, S. **CISSP all-in-one exam guide**. 6. ed. New York: McGraw-Hill, 2013. 1377 p. ISBN 978-0-07-178173-2.

HEALEY, J. Winning and losing in cyberspace. In: INTERNATIONAL CONFERENCE ON CYBER CONFLICT (CYCON), 8., 2016. **Anais**... [S.l.]: IEEE, 2016, p. 37–49. Disponível em: <http://ieeexplore.ieee.org/document/7529425/>. Acesso em: 08 fev. de 2017.

ISACA. COBIT 5 and GRC. Disponível em: <https://www.isaca.org/COBIT/Documents/COBIT5-and-GRC.ppt>. Acesso em: 03 mar. de 2018.

ISO/IEC 27001:2005. **Information technology – Security techniques – Information security management systems – Requirements**. [S.l.]: ISO/IEC, Genebra, Suíça, 2005.

ISO/IEC 27035:2011. **Information technology – Security techniques – Information security incident management**. [S.l.]: ISO/IEC, Genebra, Suíça, 2011.

JAKUS, G.; MILUTINOVIĆ, V.; OMEROVIĆ, S. ; TOMAŽIČ, S. **Concepts, ontologies, and knowledge representation**. [S.l.]: Springer, 2013.

JSON.ORG. Introdução ao JSON. Disponível em: <https://www.json.org/json-pt.html>. Acesso em: 05 nov. de 2017.

KANNAN, J.; MANIATIS, P. ; CHUN, B.-G. Secure data preservers for web services. In: PROCEEDINGS OF THE 2ND USENIX CONFERENCE ON WEB APPLICATION DEVELOPMENT, 11., 2011. **Anais**... Berkeley, CA, USA: USENIX Association, 2011, p. 3–3. Disponível em: <https://dl.acm.org/citation.cfmid=2002168>. Acesso em: 19 mar. de 2018.

KHARRAZ, A.; ROBERTSON, W.; BALZAROTTI, D.; BILGE, L. ; KIRDA, E. Cutting the Gordian Knot: A Look Under the Hood of Ransomware Attacks. In: ALMGREN, M.; GULISANO, V. ; MAGGI, F. (Org.). **Detection of Intrusions and Malware, and Vulnerability Assessment**. Milan: Springer International Publishing, 2015. p. 3–24. ISBN 978-3-319-20549-6 978-3-319-20550-2.

KRR. Knowledge Representation and Reasoning Group - HermiT OWL Reasoner. Disponível em: <http://www.hermit-reasoner.com/>. Acesso em: 05 nov. de 2017.

LAMY, J.-B. Owlready: Ontology-oriented programming in python with automatic classification and high level constructs for biomedical ontologies. **Artificial intelligence in medicine**, v. 80, p. 11–28, 2017.

LEE, R. M.; ASSANTE, M. J. ; CONWAY, T. **Analysis of the Cyber Attack on the Ukrainian Power Grid - Defense Use Case**. Boston, MA: SANS Institute, 2016. 29 p. (Relatório Técnico).

MAVROEIDIS, V.; BROMANDER, S. Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In: EUROPEAN INTELLIGENCE AND SECURITY INFORMATICS CONFERENCE (EISIC), 17., 2017. **Anais**... Athens, Greece: IEEE, 2017. Disponível em: <http://ieeexplore.ieee.org/document/8240774/>. Acesso em: 03 mar. de 2018.

VUK MILIČIĆ. Can JSON and RDF be friends?. Disponível em: <http://milicicvuk.com/blog/2014/08/26/can-json-and-rdf-be-friends/>. Acesso em: 05 nov. de 2017.

MOREIRA, G. B.; CALEGARIO, V. M.; DUARTE, J. C. ; DOS SANTOS, A. F. P. A era dos crypto ransomwares: um estudo de caso sobre o wannacry. In: SIMPÓSIO BRASILEIRO EM SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS, 17., 2017. **Anais eletrônicos**... Brasília: Sociedade Brasileira de Computação, 2017, p. 509–516. Disponível em: <https://sbseg2017.redes.unb.br/wp-content/uploads/2017/04/20171109_ANAIS_SBSEG_2017_FINAL_E-BOOK.pdf>. Acesso em: 19 nov. de 2017.

MUNDIE, D. A.; RUEFLE, R.; DOROFEE, A. J.; PERL, S. J.; MCCLOUD, J. ; COLLINS, M. An incident management ontology. In: STIDS, 0., 2014. **Anais**... [S.l.: s.n.], 2014, p. 62–71.

NIST. **Security and Privacy Controls for Federal Information Systems and Organizations**. [S.l.]: National Institute of Standards and Technology, 2013. 462 p. (Relatório Técnico, NIST SP 800-53r4).

NIST. **Framework for Improving Critical Infrastructure Cybersecurity**. [S.l.]: National Institute of Standards and Technology, 2014. 41 p. (Relatório Técnico).

NOURIAN, A.; MADNICK, S. A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet. **IEEE Transactions on Dependable and Secure Computing**, v. 99, 2015. Disponível em: <http://ieeexplore.ieee.org/document/7360168/>. Acesso em: 08 fev. de 2017.

NATALYA F. NOY AND DEBORAH L. MCGUINNESS. Ontology Development 101: A Guide to Creating Your First Ontology. Disponível em: <https://protegewiki.stanford.edu/wiki/Ontology101>. Acesso em: 05 nov. de 2017.

O GLOBO. Investimento em Segurança da Informação cresce mais no país - 2015. Disponível em: <http://oglobo.globo.com/economia/negocios/investimento-em-seguranca-da-informacao-cresce-mais-no-pais-17645471>. Acesso em: 08 fev. de 2017.

OASIS. Structured Threat Information Expression (STIX™). Disponível em: <https://oasis-open.github.io/cti-documentation/>. Acesso em: 15 out. de 2017.

CHARLIE OSBORNE. ANALYSIS: WannaCry attack shows trend toward 'economic' cyber threats, rising regulatory risk. Disponível em: <http://www.zdnet.com/article/maersk-forced-to-reinstall-4000-servers-45000-pcs-due-to-notpetya-attack/>. Acesso em: 03 mar. de 2018.

O'SULLIVAN, K.; TURNBULL, B. The cyber simulation terrain: Towards an open source cyber effects simulation ontology. In: AUSTRALIAN INFORMATION WARFARE CONFERENCE, 16., 2015. **Anais**... [S.l.]: Security Research Institute, Edith Cowan University, 2015, p. 14–23. Disponível em: <http://ro.ecu.edu.au/isw/60/>. Acesso em: 05 nov. de 2017.

OTAN. NATO Cooperative Cyber Defence Centre of Excellence - Cyber Definitions. Disponível em: <https://ccdcoe.org/cyber-definitions.html>. Acesso em: 05 nov. de 2017.

PING, L.; HAIFENG, Y. ; GUOQING, M. An incident response decision support system based on cbr and ontology. In: COMPUTER APPLICATION AND SYSTEM MODELING (ICCASM), 2010 INTERNATIONAL CONFERENCE ON, 9., 2010. **Anais**... [S.l.: s.n.], 2010, p. V11–337.

RASHID, FAHMIDA Y. Inside The Aftermath Of The Saudi Aramco Breach - 2015. Disponível em: <http://www.darkreading.com/attacks-breaches/inside-the-aftermath-of-the-saudi-aramco-breach/d/d-id/1321676>. Acesso em: 08 fev. de 2017.

REES, J.; BANDYOPADHYAY, S. ; SPAFFORD, E. H. PFIRES: a policy framework for information security. **Communications of the ACM**, v. 46, n. 7, p. 101–106, 2003. Disponível em: <http://portal.acm.org/citation.cfm?doid=792704.792706>. Acesso em: 08 fev. de 2017.

REUTERS. ANALYSIS: WannaCry attack shows trend toward 'economic' cyber threats, rising regulatory risk. Disponível em: <http://www.reuters.com/article/bc-finreg-cyber-threats-wannacry-idUSKBN19C2RU>. Acesso em: 28 ago. de 2017.

SHADBOLT, N.; BERNERS-LEE, T. ; HALL, W. The semantic web revisited. **IEEE intelligent systems**, v. 21, n. 3, p. 96–101, 2006.

SILVA, P. C. D.; FAGUNDES, L. L. Simo: Security incident management ontology. In: SIMPÓSIO BRASILEIRO EM SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS, 14., 2014. **Anais eletrônicos**... Brasília: Sociedade Brasileira de Computação, 2014, p. 302–305. Disponível em: <http://www.lbd.dcc.ufmg.br/colecoes/sbseg/2014/0023.pdf>. Acesso em: 05 nov. de 2017.

SMITH, MS. Kansas Heart Hospital hit with ransomware; attackers demand two ransoms. Disponível em: <http://www.networkworld.com/article/3073495/security/kansas-heart-hospital-hit-with-ransomware-paid-but-attackers-demanded-2nd-ransom.html>. Acesso em: 12 fev. de 2017.

SOMMERVILLE, I. **Engenharia de software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011. 530 p. ISBN 978-85-7936-108-1.

SYED, Z.; PADIA, A.; FININ, T.; MATHEWS, M. L. ; JOSHI, A. Uco: A unified cybersecurity ontology.. In: AAAI WORKSHOP: ARTIFICIAL INTELLIGENCE FOR CYBER SECURITY, 2016., 2016. **Anais**... [S.l.: s.n.], 2016. Disponível em: <https://www.aaai.org/ocs/index.php/WS/AAAIW16/paper/view/12574>. Acesso em: 05 nov. de 2017.

TECMUNDO. Anonymous sequestra PCs da Anatel e exige ação contra internet limitada. Disponível em: <http://www.tecmundo.com.br/anatel/106803-anonymous-sequestra-pcs-anatel-exige-acao-internet-limitada.htm>. Acesso em: 12 fev. de 2017.

TECNOBLOG. O curioso caso da prefeitura que teve seu sistema bloqueado por hackers. Disponível em: <https://tecnoblog.net/184550/prefeitura-sistema-bloqueado-pratania/>. Acesso em: 12 fev. de 2017.

THE APACHE SOFTWARE FOUNDATION. Apache Jena Fuseki. Disponível em: <https://jena.apache.org/>. Acesso em: 06 nov. de 2017.

TREND MICRO. Trend Micro's Definition of Ransomware. Disponível em: <https://www.trendmicro.com/vinfo/us/security/definition/ransomware>. Acesso em: 12 fev. de 2017.

VEIGA, A. D.; ELOFF, J. H. P. An Information Security Governance Framework. **Information Systems Management**, v. 24, n. 4, p. 361–372, 2007. Disponível em: <http://www.tandfonline.com/doi/abs/10.1080/10580530701586136>. Acesso em: 08 fev. de 2017.

VERIS. Vocabulary for Event Recording and Incident Sharing (VERIS). Disponível em: <http://veriscommunity.net/>. Acesso em: 27 mar. de 2017.

WEB APPLICATION SECURITY CONSORTIUM. Web-Hacking-Incident-Database. Disponível em: <http://projects.webappsec.org/w/page/13246995/Web-Hacking-Incident-Database>. Acesso em: 27 mar. de 2017.

WIKIDATA. . Disponível em: <https://www.wikidata.org/wiki/Wikidata:Main_Page>. Acesso em: 30 mar. de 2018.

WINTON, RICHARD. Hollywood hospital pays $17,000 in bitcoin to hackers; FBI investigating. Disponível em: <http://www.latimes.com/business/technology/la-me-ln-hollywood-hospital-bitcoin-20160217-story.html>. Acesso em: 12 fev. de 2017.

YAO, Y.; WU, R. ; LIU, H. Jtowl: A json to owl converto. In: PROCEEDINGS OF THE 5TH INTERNATIONAL WORKSHOP ON WEB-SCALE KNOWLEDGE REPRESENTATION RETRIEVAL & REASONING, '14., WEB-KR '14, 2., 2014. **Anais**... New York, NY, USA: ACM, 2014, p. 13–14.

9 APÊNDICES

APÊNDICE 1: CÓDIGO PARA CRIAÇÃO DA ONTOLOGIA VERIS

```
#!/usr/bin/python3
# coding=utf-8

import json, types, os
from owlready2 import *

def main():
    #output_file = os.environ.get('userprofile') + '\\Desktop\\work\\saida_v008.txt'
    json_file = os.environ.get('userprofile') + '\\Desktop\\work\\vcdb-merged.json'
    json_data = json.load(open(json_file))

    onto_path.append(os.environ.get('userprofile') +
    ↪   '\\OneDrive\\Documents\\CloudStation\\Mestrado IME\\Dissertação\\onto')
    #onto_path.append('E:\\Documents\\Syncd\\Mestrado IME\\Dissertação\\onto')
    onto = get_ontology("http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl")

    # Cria classes principais
    add_class('Incident', 'Thing', onto)
    add_class('IncidentProperty', 'Thing', onto)
    #add_class('IncidentResponseAction', 'Thing', onto)

    # VERIS: carrega classes principais e propriedades
    recurse_classes(json_data['properties'], 'IncidentProperty', onto)

    # Define propriedades que são funcionais
    with onto:
        class incident_id(DataProperty, FunctionalProperty): pass

    # Checar por que não vieram do SCHEMA (versão?) - informar Gabe
    add_data_prop('plus_sub_source', 'plus', 'str', onto)
    add_data_prop('action_notes', 'action', 'str', onto)

    # Cria relações
    add_class_prop('has_action', 'Incident', 'action', onto)
    add_class_prop('has_actor', 'Incident', 'actor', onto)
    add_class_prop('has_asset', 'Incident', 'asset', onto)
    #add_class_prop('has_asset', 'plus', 'asset', onto)              #observar
```

```
    add_class_prop('has_attribute', 'Incident', 'attribute', onto)
    #add_class_prop('has_attribute', 'plus', 'attribute', onto)      #observar
    add_class_prop('has_impact', 'Incident', 'impact', onto)
    add_class_prop('has_plus', 'Incident', 'plus', onto)
    add_class_prop('has_subsets', 'Incident', 'subsets', onto)
    add_class_prop('has_timeline', 'Incident', 'timeline', onto)
    #add_class_prop('has_timeline', 'plus', 'timeline', onto)        #observar
    add_class_prop('has_victim', 'Incident', 'victim', onto)

    # Salva a ontologia
    onto.save()


def recurse_classes(data, parent, onto):
    for L1 in data:
        if data[L1]['type'] == 'object':
            add_class(L1, 'onto.' + parent, onto)
            for L2 in data[L1]:
                if L2 == 'properties':
                    recurse_props(data[L1][L2], L1, onto)
                    recurse_comments(data[L1][L2], L1, onto)
        elif data[L1]['type'] == 'string':
            add_data_prop(L1, 'Incident', 'str', onto)

def recurse_props(data, parent, onto):
    for L1 in data:
        if type (data[L1]) is dict:
            if (L1 == 'properties') or (L1 == 'items'):
                recurse_props(data[L1], parent, onto)
            else:
                recurse_props(data[L1], parent + '_' + L1, onto)

        else:   # Cria Data Propery
            domain = parent.split('_')[0]
            if L1 == 'type':
                #print('*{} ( type = {} )'.format(parent, data[L1]))
                if (data[L1] == 'string') or (data[L1] == 'array'):
                    add_data_prop(parent, domain, 'str', onto)
                    #print('{}\t\t\t{}\t{}'.format(parent, domain, data[L1]))
                elif data[L1] == 'boolean':
                    add_data_prop(parent, domain, 'bool', onto)
                    #print('{}\t\t\t{}\t{}'.format(parent, domain, data[L1]))
```

```
                elif data[L1] == 'number':
                    add_data_prop(parent, domain, 'float', onto)
                    #print('{}\t\t\t{}\t{}'.format(parent, domain, data[L1]))
                    getattr(onto, parent).comment.en.append('{} = {}'.format(L1,
                     ↪  str(data[L1])))
                elif data[L1] == 'integer':
                    add_data_prop(parent, domain, 'int', onto)
                    #print('{}\t\t\t{}\t{}'.format(parent, domain, data[L1]))
                elif data[L1] == 'object':
                    # Exceção devido à inconsistência no VERIS !!!!
                    if parent not in
                     ↪  {'impact_loss','asset_assets','attribute_confidentiality_data','plus_event_c
                        add_data_prop(parent, domain, 'str', onto)
                        getattr(onto, parent).comment.en.append('{} = {}'.format(L1,
                         ↪  data[L1]))
                    #print('{}\t\t\t{}\t{}'.format(parent, domain, data[L1]))

            elif L1 in {'pattern', 'required', 'minProperties', 'minLength',
             ↪  'maxLength', 'minItems', 'maxItems',
                        'maximum', 'minimum', 'uniqueItems', 'additionalProperties',
                         ↪  'enum'}:
                continue

            else:
                # Nenhum outro tipo localizado no schema 1.3.2
                print(' **** {}'.format(L1))


def recurse_comments(data, parent, onto):
    for L1 in data:
        if type (data[L1]) is dict:
            if (L1 == 'properties') or (L1 == 'items'):
                recurse_comments(data[L1], parent, onto)
            else:
                recurse_comments(data[L1], parent + '_' + L1, onto)
        else:   # Adiciona comentários
            domain = parent.split('_')[0]
            if L1 == 'type':
                continue
            elif L1 in {'pattern', 'required', 'minProperties', 'minLength',
             ↪  'maxLength', 'minItems', 'maxItems',
```

```
                        'maximum', 'minimum', 'uniqueItems', 'additionalProperties',
                        ↪  'enum'}:
                    # ENUM *deveria* virar DATA RANGE expression
                    try:
                        getattr(onto, parent).comment.en.append('{} = {}'.format(L1,
                        ↪  str(data[L1])))
                    except:
                        print('  <ERRO> verifique: {} ( {} = {} {} )'.format(parent, L1,
                        ↪  data[L1], type(data[L1])))
                else:
                    # Nenhum outro tipo localizado no schema 1.3.2
                    print(' **** {}'.format(L1))


def add_class(name,parent,onto):
    with onto:
        exec('class {}({}): pass'.format(name, parent))
    #print('class {}({}): pass'.format(name, parent))

def add_data_prop(name, domain, type, onto):
    with onto:
        exec('class {}(onto.{} >> {}): pass'.format(name, domain, type))
    #print('class {}(onto.{} >> {}): pass'.format(name, domain, type))

def add_class_prop(name, domain, range, onto):
    with onto:
        exec('class {}(onto.{} >> onto.{}): pass'.format(name, domain, range))
    #print('class {}(onto.{} >> onto.{}): pass'.format(name, domain, range))

if __name__ == '__main__':
    main()
```

APÊNDICE 2: CÓDIGO PARA CARGA DO *DATASET* VERIS

```
#!/usr/bin/python3
# coding=utf-8

import json, types, os
from owlready2 import *

err = 0

def main():
    json_schema_file = os.environ.get('userprofile') +
    ↪  '\\Desktop\\work\\vcdb-merged.json'
    json_data_dir = os.environ.get('userprofile') + '\\Desktop\\work\\json'
    #json_data_dir = os.environ.get('userprofile') +
    ↪  '\\Desktop\\work\\json\\65fdbcfb6bab5170d5c1e39a631e07d1'
    json_schema = json.load(open(json_schema_file))

    onto_path.append(os.environ.get('userprofile') +
    ↪  '\\OneDrive\\Documents\\CloudStation\\Mestrado IME\\Dissertação\\onto')
    #onto_path.append('E:\\Documents\\Syncd\\Mestrado IME\\Dissertação\\onto')
    onto =
    ↪  get_ontology("http://gb.moreira.nom.br/VERIS_20171218_v002_auto.owl").load()

    i = 0

    for file in os.listdir(json_data_dir):
        fullpath = os.path.join(json_data_dir, file)
        if os.path.isfile(fullpath) and file[-5:] == '.json':
            json_data = json.load(open(fullpath))
            id = file[:-5]
            print ('({}) Incident_ID: {}'.format(i, id))
            incident = onto.Incident(id + '_incident', incident_id = id)
            recurse_data_load_pass1(id, json_data, 'incident', incident, onto)
            i += 1

    print('\n<Incidentes carregados: {}> <erros: {}>'.format(i,err))
    onto.save()
```

```
def recurse_data_print(data, parent, inc_obj, onto):
    for item in data:
        #print(item,data[item])
        if type(data[item]) is dict:
            recurse_data_print(data[item], parent + '_' + item, inc_obj, onto)
        else:
            print(parent + '_' + item, data[item], type(data[item]))

def recurse_data_print_pass1(id, data, parent, inc_obj, onto):
    for item in data:
        #print(item,type(data[item]),data[item])
        if type(data[item]) is dict:
            print('objeto = onto.{}("{}_{}")'.format(item, item, id))
            objeto = item
            recurse_data_print(data[item], parent + '_' + item, objeto, onto)
        else:
            print(parent + '_' + item, data[item], type(data[item]))


def recurse_data_load_pass1(id, data, parent, inc_obj, onto):
    global err
    for item in data:
        if type(data[item]) is dict:
            objeto = 'mock'
            if item == 'action':
                objeto = onto.action(id + '_action')
                inc_obj.has_action = [objeto]

            if item == 'actor':
                objeto = onto.actor(id + '_actor')
                inc_obj.has_actor = [objeto]

            if item == 'asset':
                objeto = onto.asset(id + '_asset')
                inc_obj.has_asset = [objeto]

            if item == 'attribute':
                objeto = onto.attribute(id + '_attribute')
                inc_obj.has_attribute = [objeto]

            if item == 'impact':
```

```
            objeto = onto.impact(id + '_impact')
            inc_obj.has_impact = [objeto]

        if item == 'plus':
            objeto = onto.plus(id + '_plus')
            inc_obj.has_plus = [objeto]

        if item == 'subsets':
            objeto = onto.subsets(id + '_subsets')
            inc_obj.has_subsets = [objeto]

        if item == 'timeline':
            objeto = onto.timeline(id + '_timeline')
            inc_obj.has_timeline = [objeto]

        if item == 'victim':
            objeto = onto.victim(id + '_victim')
            inc_obj.has_victim = [objeto]

        #print('    - objeto criado: {}_{}'.format(id, item))
        recurse_data_load(data[item], parent + '_' + item, objeto, onto)
    else:
        if item == 'incident_id': continue
        attrib = parent + '_' + item
        if (type(data[item]) is str) or (type(data[item]) is int):
            try:
                getattr(inc_obj, attrib[9:]).append(data[item])
                #print('{}.{} = {} {} (SAVED)'.format(inc_obj, attrib[9:],
                 ↪  data[item], type(data[item])))
            except (AttributeError, TypeError):
                print('   <ERRO> verifique: {}.{} = {}'.format(inc_obj,
                 ↪  attrib[9:], data[item]))
                err += 1

        elif type(data[item]) is float:
            try:
                getattr(inc_obj, attrib[9:]).append(int(data[item]))
                #print('{}.{} = {} {} (SAVED)'.format(inc_obj, attrib[9:],
                 ↪  data[item], type(data[item])))
            except (AttributeError, TypeError):
                print('   <ERRO> verifique: {}.{} = {}'.format(inc_obj,
                 ↪  attrib[9:], data[item]))
```

```
                err += 1

            elif (type(data[item]) is list):
                try:
                    for x in data[item]:
                        getattr(inc_obj, attrib[9:]).append(str(x))
                    #print('{}.{} = {} {} (SAVED)'.format(inc_obj, attrib[9:],
                     ↪  data[item], type(data[item])))
                except (AttributeError, TypeError):
                    print('   <ERRO> verifique: {}.{} = {}'.format(inc_obj,
                     ↪  attrib[9:], data[item]))
                    err += 1

            else:
                print('{}.{} = {} (NOT SAVED: {})'.format(inc_obj, attrib[9:],
                 ↪  data[item],type(data[item])))



def recurse_data_load(data, parent, inc_obj, onto):
    global err
    for item in data:
        if type(data[item]) is dict:
            recurse_data_load(data[item], parent + '_' + item, inc_obj, onto)
        else:
            if item == 'incident_id': continue
            attrib = parent + '_' + item
            try:
                datatype = getattr(onto, attrib[9:]).range
            except (AttributeError):
                print('   <ERRO> verifique: {}'.format(attrib[9:]))
                err += 1
                continue
            #print('{} {}'.format(attrib, datatype))

            # Converte o tipo do dado, caso esteja diferente do declarado na
             ↪  propriedade

            if type(data[item]) is list:
                try:
                    for x in data[item]:
```

```
                        conv_data = data_conv(datatype, x)          # Converte o tipo
                         ↪  do dado, caso necessário
                        getattr(inc_obj, attrib[9:]).append(conv_data)
                        #print('{}.{} = {} {} {}(SAVED)'.format(inc_obj, attrib[9:],
                         ↪  conv_data, type(conv_data), datatype))
                except (AttributeError, TypeError):
                    print('   <ERRO> verifique: {}.{} = {}'.format(inc_obj,
                     ↪  attrib[9:], data[item]))
                    err += 1
            else:
                try:
                    conv_data = data_conv(datatype, data[item])     # Converte o tipo
                     ↪  do dado, caso necessário
                    getattr(inc_obj, attrib[9:]).append(conv_data)
                    #print('{}.{} = {} {} {}(SAVED)'.format(inc_obj, attrib[9:],
                     ↪  conv_data, type(conv_data), datatype))
                except (AttributeError, TypeError):
                    print('   <ERRO> verifique: {}.{} = {}'.format(inc_obj,
                     ↪  attrib[9:], data[item]))
                    err += 1



def data_conv(datatype, data): # Converte o tipo do dado, caso esteja diferente do
 ↪  declarado na propriedade
    if (datatype == "[<class 'int'>]") and type(data) is not int:
        conv_data = int(data)
    elif (datatype == "[<class 'float'>]") and type(data) is not float:
        conv_data = float(data)
    elif (datatype == "[<class 'str'>]") and type(data) is not str:
        conv_data = str(data)
    elif (type(data) is list) or (type(data) is dict):
        conv_data = str(data)
    else:
        conv_data = data
    return(conv_data)



if __name__ == '__main__':
    main()
```

10 ANEXOS

ANEXO 1: ENUMERAÇÕES DO MODELO VERIS

TAB. 10.1: Enumeração da variedade de ativos descrita pelo VERIS (2017)

Tipo de ativo	**Descrição**
S - Authentication	Server - Authentication
S - Backup	Server - Backup
S - Database	Server - Database
S - DHCP	Server - DHCP
S - Directory	Server - Directory (LDAP, AD)
S - DCS	Server - Distributed control system (DCS)
S - DNS	Server - DNS
S - File	Server - File
S - Log	Server - Log or event management
S - Mail	Server - Mail
S - Mainframe	Server - Mainframe
S - Payment switch	Server - Payment switch or gateway
S - POS controller	Server - POS controller
S - Print	Server - Print
S - Proxy	Server - Proxy
S - Remote access	Server - Remote access
S - SCADA	Server - SCADA system
S - Web application	Server - Web application
S - Code repository	Server - Code repository
S - VM host	Server - Virtual Host
S - Other	Server - Other/Unknown
N - Access reader	Network - Access control reader (e.g., badge, biometric)
N - Camera	Network - Camera or surveillance system
N - Firewall	Network - Firewall
N - HSM	Network - Hardware security module (HSM)
N - IDS	Network - IDS or IPs
N - Broadband	Network - Mobile broadband network
N - PBX	Network - Private branch exchange (PBX)
N - Private WAN	Network - Private WAN
N - PLC	Network - Programmable logic controller (PLC)
N - Public WAN	Network - Public WAN
N - RTU	Network - Remote terminal unit (RTU)
N - Router or switch	Network - Router or switch
N - SAN	Network - Storage area network (SAN)
N - Telephone	Network - Telephone
N - VoIP adapter	Network - VoIP adapter
N - LAN	Network - Wired LAN
N - WLAN	Network - Wireless LAN
N - Other	Network - Other/Unknown

U - Auth token	User Device - Authentication token or device
U - Desktop	User Device - Desktop or workstation
U - Laptop	User Device - Laptop
U - Media	User Device - Media player or recorder
U - Mobile phone	User Device - Mobile phone or smartphone
U - Peripheral	User Device - Peripheral (e.g., printer, copier, fax)
U - POS terminal	User Device - POS terminal
U - Tablet	User Device - Tablet
U - Telephone	User Device - Telephone
U - VoIP phone	User Device - VoIP phone
U - Other	User Device - Other/Unknown
T - ATM	Public Terminal - Automated Teller Machine (ATM)
T - PED pad	Public Terminal - Detached PIN pad or card reader
T - Gas terminal	Public Terminal - Gas "pay-at-the-pump" terminal
T - Kiosk	Public Terminal - Self-service kiosk
T - Other	Public Terminal - Other/Unknown
M - Tapes	Media - Backup tapes
M - Disk media	Media - Disk media (e.g., CDs, DVDs)
M - Documents	Media - Documents
M - Flash drive	Media - Flash drive or card
M - Disk drive	Media - Hard disk drive
M - Smart card	Media - Identity smart card
M - Payment card	Media - Payment card (e.g., magstripe, EMV)
M - Other	Media - Other/Unknown
P - System admin	People - Administrator
P - Auditor	People - Auditor
P - Call center	People - Call center
P - Cashier	People - Cashier
P - Customer	People - Customer
P - Developer	People - Developer
P - End-user	People - End-user
P - Executive	People - Executive
P - Finance	People - Finance
P - Former employee	People - Former employee
P - Guard	People - Guard
P - Helpdesk	People - Helpdesk
P - Human resources	People - Human resources
P - Maintenance	People - Maintenance
P - Manager	People - Manager
P - Partner	People - Manager
P - Other	People - Other/Unknown
Unknown	Unknown