

# Técnica para Retenção e Recuperação de Conhecimento na Resolução de Incidentes de Segurança

Marcelo Colomé<sup>1</sup>, Raul Ceretta Nunes<sup>1</sup>, Luis Alvaro de Lima Silva<sup>1</sup>

<sup>1</sup> Programa de Pós-Graduação em Ciência da Computação (PPGCC)  
Universidade Federal de Santa Maria (UFSM) - Santa Maria - RS – Brasil

{marcelocolome, ceretta, luisalvaro}@inf.ufsm.br

**Abstract.** *This work explores the reasoning computing techniques application for knowledge management of cybersecurity incidents to offer a methodological approach for the retention and reuse of the specialist's knowledge aiming the resolution of new incidents. The information security specialist's knowledge is central for organizations because the effective resolution of incidents depends on their knowledge. However, organizations should not be totally dependent on their employees. Thus, the proposed methodology explores Cased-based Reasoning with weighted attributes from the IODEF pattern, aiming the retention of the specialist's knowledge on the incident's resolution. The solution allows other organization members to perform similar tasks, helping to decrease the security company dependency of its employees. The results demonstrate that with this methodology the knowledge is effectively retained in the case-base and that new employees can be benefited from the recommendations built and provided by the system, improving the knowledge retention in organizations.*

**Resumo.** *Este trabalho explora a aplicação de técnicas computacionais para a gestão do conhecimento de incidentes de segurança, oferecendo uma abordagem metodológica para a retenção e reutilização do conhecimento do especialista, visando a resolução de novos incidentes. O conhecimento do especialista em segurança da informação é fundamental para as organizações, pois a resolução eficaz de incidentes de segurança depende do conhecimento dos mesmos. Porém, as organizações não devem ser totalmente dependentes de seus funcionários. Desta forma, a metodologia proposta utiliza-se de Raciocínio Baseado em Casos com ponderação dos atributos sobre dados representados no padrão IODEF, visando a retenção do conhecimento do especialista na resolução de incidentes, possibilitando que outros membros da organização possam desempenhar tarefas similares, diminuindo a dependência de empresas de segurança em relação a seus funcionários. Os resultados demonstram que através desta metodologia o conhecimento fica efetivamente retido na base de casos e que novos funcionários podem se beneficiar de recomendações construídas e fornecidas pelo sistema, melhorando com isto a retenção do conhecimento nas organizações.*

## 1 Introdução

O conhecimento sobre segurança é um ativo importante para as empresas nesta área, pois nele estão contidas as lições aprendidas ao longo do tempo, as quais são amplamente exploradas na melhoria contínua de qualidade dessas empresas [Dalkir e Liebowitz, 2011]. A retenção deste conhecimento de segurança pode trazer benefícios significativos para funcionários e a empresa como um todo. Entre outras razões, muitos incidentes de segurança que aconteceram no passado podem ser iguais ou semelhantes aos incidentes atuais e futuros de uma organização. A gestão desse conhecimento de segurança é um processo que deve aplicar uma abordagem sistemática para a captura, estruturação, gestão e disseminação do conhecimento em cada parte da organização, objetivando um trabalho mais veloz, que reutilize as melhores práticas e reduza o

custo do retrabalho em novos projetos [Rahimli, 2012]. Neste contexto, a gestão de conhecimento sobre a resolução de incidentes de segurança é crítica visto que tais incidentes podem causar prejuízos para organizações caso eles não sejam sanados de forma eficaz. Impactos nas finanças, na publicidade, ou ainda perda de dados são alguns dos exemplos de como uma empresa pode ser afetada por um incidente de segurança [Hove e Tarnes, 2013].

O CERT.br reportou 676.514 incidentes em 2018 [CERT.br, 2019], demonstrando a importância de reter conhecimento a cerca dos tratamentos adotados. Apesar deste significativo número de incidentes, o gerenciamento destes incidentes em muitas organizações não é realizado de forma sistemática. Na solução deste tipo de problema, Sistemas de Resposta à Intrusão (IRS) [Inayat *et al.*, 2016] são utilizados para tratamento de incidentes de segurança que transpõem barreiras iniciais introduzidas por sistemas de Detecção de Intrusão. Um IRS é projetado para melhorar as políticas de segurança, tratando incidentes/intrusões detectadas e mitigando os seus efeitos. Porém, a capacidade de um IRS oferecer uma resposta adequada à determinada intrusão está intimamente ligada ao conhecimento da equipe de Resposta e Tratamento de Incidentes, uma vez que a equipe deve construir um plano de resposta aos incidentes. Mais ainda, diferentes incidentes podem exigir diferentes planos de resposta. Este processo faz com que exista uma grande dependência desta equipe (pessoas) para que tais incidentes possam ser resolvidos de forma adequada. O resultado são organizações dependentes do conhecimento de seus colaboradores para a resolução de incidentes. O problema é que esta dependência dificulta muito o reuso de soluções de segurança por diferentes colaboradores, devendo ser evitada.

Na Inteligência Artificial, Raciocínio Baseado em Casos [Richter e Weber, 2013] (RBC) é uma técnica capaz de auxiliar na resolução de novos incidentes, visto que ela focaliza a manutenção e reuso de conhecimento adquirido na resolução de incidentes passados. Este trabalho propõe uma metodologia para IRS que explora RBC para automatização de recomendações de planos de resposta à incidentes de segurança. A metodologia difere-se de outras por explorar: *i*) a padronização IODEF [Takahashi *et al.*, 2014] [Takahashi e Miyamoto, 2016] para representação de dados, o qual é um padrão voltado a informações relacionadas a incidentes de segurança e adotado por diferentes CERTs [Gascon *et al.*, 2017]; e *ii*) o conhecimento do especialista relacionado a importância relativa de atributos usados na representação de casos de incidentes de segurança, onde a importância está representada via um conjunto numérico de pesos usados em cálculos de similaridade. A partir da recuperação e reuso de casos concretos de resolução de incidentes de segurança, os resultados de testes realizados neste trabalho demonstram que o mecanismo proposto para IRS baseado em RBC potencializa o reuso de conhecimento dos especialistas na resolução de novos incidentes.

Este trabalho está organizado da seguinte forma: inicialmente são apresentados a fundamentação teórica na seção 2 e os trabalhos relacionados na seção 3. O modelo de solução e os resultados são apresentados nas seções 4 e 5 respectivamente. Por fim, a seção 6 apresenta as considerações finais.

## **2 Fundamentação Teórica**

### **2.1 Raciocínio Baseado em Casos**

A gestão do conhecimento compreende iniciativas que corroboram para uma alocação racional do conhecimento do ponto de vista organizacional e de sistemas computacionais baseados em conhecimento. Neste contexto, Raciocínio Baseado em Casos (RBC) [Richter e Weber, 2013] é um paradigma para a resolução de problemas e para o aprendizado com base em experiências passadas. Neste paradigma, a solução de novos problemas é realizada por meio da reutilização de soluções encontradas em problemas passados. Tais problemas são chamados de *casos*, os quais são mantidos em uma base de casos. Em geral, um caso representa a descrição de

uma situação problema juntamente com as experiências adquiridas durante a resolução deste problema, sendo percebido como uma associação de dois conjuntos de informações: problema e solução. O ciclo de funcionamento de um sistema de RBC é composto por etapas: 1) recuperar da base um conjunto de casos onde a descrição do problema é similar ao problema atual (utilizado como consulta); 2) reutilizar as soluções dos casos recuperados para solucionar o problema atual; 3) revisar a solução proposta considerando possíveis diferenças entre o problema e casos passados recuperados; 4) reter (aprender) a nova experiência para solucionar problemas futuros.

No contexto de incidentes de segurança, RBC pode ser utilizado para apoiar o processo de resolução dos incidentes. Segundo [Mansar *et al.*, 2003], existem muitos argumentos que suportam o uso de RBC para gestão do conhecimento. A representação de dados com exemplos concretos é mais compreensível e aplicável em vários contextos de resolução de problemas do que cadeias complexas de regras ou modelos. Outro benefício é a possibilidade de o RBC aprender de forma sistemática com experiências passadas, através dos passos de revisão e retenção, providenciando um *framework* para a aquisição do conhecimento neste problema de aplicação.

## 2.2 Tratamento de Incidentes de Segurança

Com o objetivo de tratar incidentes de segurança, as empresas investem na coleta de dados sobre intrusões, para analisa-los e entendê-los. Procuram assim aprender a lidar da melhor forma com este tipo de ameaça. De acordo com a ISO/IEC 27035 [ISO/IEC, 2016], os processos de tratamento de incidentes podem ser divididos em fases:

- Planejar e preparar: deve-se pensar na elaboração de planos de tratamento de incidentes para tipos específicos, *check-lists* de tarefas e rotinas para quando uma eventualidade acontecer e planos de comunicação que irão conter informações de como as entidades envolvidas irão se comunicar durante a calamidade;
- Detectar e reportar: [Metzger *et al.*, 2011] recomenda que sejam utilizados múltiplos meios para reportar um incidente. Eles podem ser reportados de forma automática por serviços ou outra entidade como CERTs, e também de forma manual, como por telefone e *e-mail*;
- Avaliar e decidir: deve-se verificar primeiramente se o incidente realmente ocorreu, posteriormente verificando sua magnitude e consequências, e então rastrear sua origem;
- Responder: respostas devem refletir as ações planejadas na etapa anterior (avaliar e decidir). Utiliza-se o plano de tratamento como base para a tomada de ações, o qual possui os passos recomendados para contornar o incidente. Deve-se tomar as medidas apropriadas, incluindo recuperar-se do incidente, documentação, e comunicação às partes interessadas;
- Registrar as lições aprendidas: deve ser iniciada tão logo o incidente foi encerrado e tem o intuito de analisar se a solução projetada pelo CSIRT teve sucesso. Aprender sobre os incidentes é importante, porém muitas organizações acham difícil colocar isso em prática [Line *et al.*, 2016]. Uma das tarefas realizadas neste passo é a documentação apropriada, garantindo que o método de tratamento do incidente seja preciso.

O tratamento de incidentes possui ainda alguns aspectos desafiadores, o que demanda o surgimento de novas ferramentas e recomendações específicas [Line *et al.*, 2016]. Segundo [Rajnovic, 2011], cada vez que um novo especialista é contratado, ele deve compreender os processos e procedimentos envolvidos no tratamento dos incidentes. Isto pode causar uma "perda de memória" do time, pois o conhecimento dos antigos membros, que também possuíam alguma experiência, não está mais presente. Esta "perda de memória" pode gerar uma grande perda de tempo, pois o time geralmente reinventa uma solução para um incidente. Assim, uma equipe de segurança deve ser capaz de organizar suas lições aprendidas de modo que elas possam ser

reutilizadas. Mesmo usando IRS, segundo [Anuar *et al.*, 2010] o conhecimento de causa/tratamento relacionado a respostas reativas e passivas pode fornecer informações de *feedback* para respostas proativas, habilitando melhora na reação à novos incidentes.

### 2.3 Incident Object Description Exchange Format - IODEF

A troca de dados entre sistemas e a padronização nas representações dos dados são elementos chave para interoperabilidade entre sistemas de segurança. Em geral padrões de representação de dados são compostos por diferentes tipos de informações, as quais são muitas vezes coletadas automaticamente. O IODEF [Takahashi e Miyamoto, 2016] é um padrão criado pela IETF que define uma representação de dados para informações relacionadas a incidentes de segurança, e que inclui dados relacionados a *hosts*, redes e serviços que são executados nestes sistemas; metodologia de ataques e evidências forenses; impacto da atividade; e abordagem para documentação do fluxo de trabalho. O padrão IODEF fornece um *framework* para o compartilhamento de informações comumente trocadas por CSIRT sobre incidentes.

Este padrão possui 34 classes no total, sendo o mesmo desenvolvido para ser adaptável às diferentes necessidades das organizações. Pode-se utilizar todas as classes ou apenas as que sejam necessárias. Duas das principais classes são a classe *IODEF-Document* e *Incident*. A classe *IODEF-Document* é a classe principal deste formato de dados, sendo todos os documentos IODEF uma instância desta classe. Esta classe possui informações sobre a versão do documento, linguagem e informações sobre o processamento do documento, bem como uma classe agregada *Incident*, a qual pode ter uma ou mais instâncias de incidentes (um documento pode se referir a um ou mais incidentes). A classe *Incident* oferece uma representação padrão para a troca de dados sobre incidentes comumente reportados. Ela possui informações sobre o motivo da criação do documento IODEF, linguagem do documento e restrições. Esta classe também possui quatorze classes agregadas para a representação das seguintes informações: identificador do incidente, identificadores alternativos, identificadores para incidentes relacionados, momento em que o incidente foi detectado, momento em que teve início o incidente, momento em que teve fim o incidente, momento em que o incidente foi reportado, descrição do incidente, técnicas usadas pelo invasor, informação de contato das partes envolvidas no incidente, descrição dos eventos que compreendem o incidente, histórico de eventos e ações que ocorreram durante contenção do incidente e dados adicionais que não cabem no modelo.

O *IODEF Extension for Structured Cybersecurity Information* [Takahashi *et al.*, 2014] [Takahashi e Miyamoto, 2016] é uma extensão que visa uma melhora na troca de informações feitas por meios automatizados, melhorando a leitura automática (por computador) das mensagens. As seguintes informações foram contempladas nesta extensão: padrão do ataque, informação da plataforma, vulnerabilidades e fraquezas, instruções para contra-medidas, *logs* de eventos computacionais e grau de severidade. Segundo os autores da extensão, apesar do IODEF permitir que essas informações fossem expressas, ele não definia formatos detalhados para especificar as informações.

## 3 Trabalhos Relacionados

A literatura traz alguns trabalhos que utilizam RBC para apoiar a resposta à incidentes de segurança [Jiang, 2014] [Kim, 2010] [Ping, 2010], principalmente para fins de detecção de intrusão.

Em [Jiang, 2014], é proposto um sistema de IRS com RBC que explora o uso de lógica descritiva para representação dos atributos de um incidente. Os autores usam uma hierarquia de atributos advindos dos possíveis ataques para representar o caso de consulta (problema) ao RBC. Uma descrição textual de contra-medidas é proposta como plano de tratamento, juntamente com

um atributo para expressar o grau de satisfação do usuário com a solução. Embora o trabalho apresente uma abordagem de resolução interessante, diferente deste ele não explora padrões para representação de dados sobre incidentes, tão pouco permite a reutilização de ações para expressar diferentes planos de tratamento.

Em [Kim, 2010], a metodologia RFM (*Recency, Frequency, Monetary*) é proposta para a redução de falsos alertas com o uso de RBC. RFM analisa arquivos de *log* levando em consideração a "recência", frequência e valor, em um processo estatístico que permite detectar anomalias e mau uso. A técnica de RBC é usada para encontrar a similaridade de padrões de ataque já conhecidos. Diferente deste, [Kim, 2010] foca na detecção de incidentes sem oferecer planos de resposta para o contingenciamento do incidente.

Em [Ping, 2010], é proposto o uso de ontologias e RBC para a construção de um sistema de decisão e resposta à incidentes de segurança. A ontologia é usada para a representação de incidentes. O sistema é alimentado por informações de sensores, juntamente com informações inseridas manualmente. Após a extração dos dados é criada a ontologia e então o caso é processado pelo RBC. A proposição de ontologias permite uma boa organização hierárquica dos tipos de ataques existentes, porém seu uso em IRS depende de um padrão de ontologia para incidentes. Enquanto em [Ping, 2010] foi proposto um padrão de ontologia com apenas 6 atributos, neste trabalho é proposto o uso do IODEF, que oferece atributos amplamente discutidos e aceitos.

Do conhecimento destes autores, a literatura ainda não explorou, tal como neste trabalho, o uso de RCB combinado com incidentes representados em uma padronização internacional de incidentes, tal como IODEF ou STIX [OASIS, 2019]. Tão pouco a ponderação de atributos foi objeto de investigação no âmbito de IRS.

#### **4 Modelo para um IRS baseado em RBC**

O conhecimento do especialista na resolução de incidentes de segurança computacionais deve ser reutilizado para o tratamento de novos incidentes. Neste trabalho, o conhecimento do especialista é retido em uma base de conhecimento, a qual utiliza a técnica de Raciocínio Baseado em Casos visando automatizar a recomendação de planos de respostas a novos incidentes. Na metodologia proposta, cada incidente de segurança é capturado como um *caso*, o qual é inserido em uma base de conhecimento denominada *base de casos*. Os incidentes são representados no padrão IODEF, o que mantém a metodologia em conformidade com os esforços para melhora de capacidades operacionais para CSIRTs [Takahashi e Miyamoto, 2016].

Para apoiar o emprego da metodologia proposta, uma *ferramenta de formulação de casos* auxilia a transformação de um novo incidente de segurança em um caso, bem como no correto preenchimento dos atributos e valores advindos de uma ocorrência de incidente de segurança. Embora tais incidentes possam ser registrados manualmente, eles também podem ser gerados automaticamente como resultados de sistemas de segurança. Quando gerados automaticamente, as representações em IODEF muitas vezes apresentam informações incompletas ou ambíguas, podendo sofrer ajustes manuais por especialistas de segurança. *Logs* recebidos juntamente com os incidentes podem assim ser também utilizados para a extração de informações que permitem complementar os dados do incidente.

A Figura 1 ilustra a arquitetura da metodologia proposta. No passo inicial, incidentes representados em IODEF são gerados ou recebidos pela equipe responsável. A equipe pode então fazer uso da ferramenta de formulação de casos para revisar ou preencher informações faltantes. Desta forma, a ferramenta transforma o incidente em um caso. Salienta-se que o novo caso gerado a partir do incidente ainda não possui solução. O RBC é então utilizado para pesquisar a base de casos por problemas similares ao problema atual. Como resultado da

aplicação do RBC, são recomendados um ou mais planos de tratamento de incidentes para o novo incidente. Pode-se utilizar a solução recomendada ou adaptá-la para que a mesma reflita as particularidades do problema em questão (incidente relatado). Após utilizado o(s) plano(s) sugerido(s), o novo caso é armazenado na base de casos, retendo o conhecimento/experiência do especialista e gerando aprendizado ao sistema.

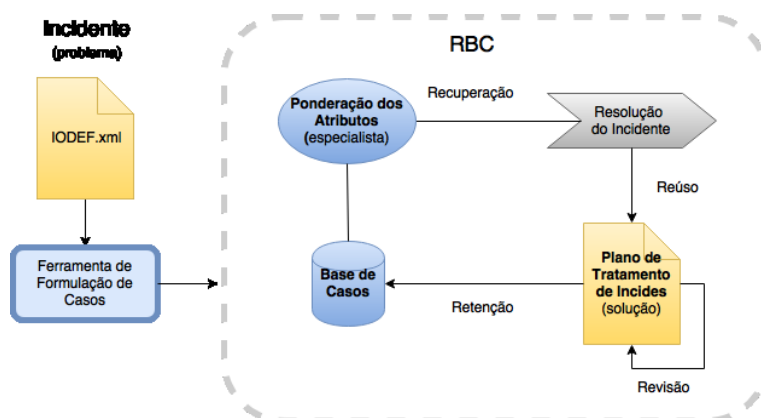


Figura 1: Arquitetura funcional para implantação da metodologia proposta.

#### 4.1 Modelagem da base de casos

A flexibilidade proporcionada pela adoção do padrão IODEF na descrição de incidentes faz com certos incidentes sejam documentados de forma incorreta. Logo, para que um incidente seja inserido na base de casos o mesmo deve passar por uma etapa de verificação/modelagem. No escopo deste trabalho, o problema a ser resolvido é o incidente de segurança gerado e a solução é um plano de tratamento para este incidente. Nestes *casos*, a representação do problema corresponde ao *conjunto de atributos do incidente* e a representação da solução corresponde a uma *sequência de passos do plano de tratamento* elaborada para a contenção do incidente. Note que esta relação problema/solução está inicialmente retida nos especialistas. Contudo, a partir da metodologia proposta, esta relação permanece retida no sistema baseado em conhecimento da organização.

#### 4.2 Representatividade dos atributos

As diferentes características contidas em incidentes de segurança podem ser representadas a partir de um conjunto de atributos. Em RBC, casos representando incidentes possuem atributos que são mais importantes que outros, assim permitindo melhor medir a similaridade entre dois casos. Na etapa de recuperação de casos em RBC, estes atributos podem ser ou não indexados. Os atributos indexados são aqueles utilizados para a recuperação de casos, pois são preditivos para encontrar a solução para um caso (incidente). Assim, cabe a um especialista do domínio de aplicação avaliar e decidir o que será ou não indexado.

Para representar casos de incidentes de segurança, os tipos de incidentes mais comuns foram identificados, bem como elencados os atributos necessários para sua representação. Os tipos de incidentes considerados são: Bot, DoS, Proxy, URL Maliciosa, Copyright, Spam, Scan, Tentativa de Login por força bruta, Phishing e Defacement.

Após esta categorização, foram selecionados para a representação de um incidente os atributos presentes nos trabalhos de [Jiang, 2014], [Kim, 2010] e [Ping, 2010], mas que eram comuns aos quatro trabalhos. Então avaliou-se se estes atributos estavam de acordo com os que estavam presentes nos casos reportados pelo Centro de Atendimento a Incidentes de Segurança -

CAIS/RNP, permitindo capturar com fidelidade uma situação de segurança real do mundo empresarial. Como resultado, foram derivados para a metodologia proposta atributos comuns e específicos por tipo de incidente identificado, dado que diferentes tipos de incidentes podem possuir atributos específicos e também atributos que podem ser compartilhados entre eles.

Na metodologia proposta, o conjunto padrão de atributos de um *problema*, presente em todos os incidentes, corresponde ao: identificador do incidente, tipo de incidente, descrição, dia e hora da detecção, endereço IP do *host* comprometido, *logs* do incidente, um campo para indicar a categoria do *host* (origem, destino, etc.) na hora do incidente e um identificador para a correlação de incidentes. A este “conjunto padrão” pode ou não ser agregados novos atributos, dependendo do interesse do especialista em segurança. Por exemplo, um incidente do tipo *Spam* pode possuir apenas os atributos do conjunto padrão. Os outros tipos de incidentes podem possuir outros atributos de interesse. Outros atributos que podem estar presentes nos incidentes são: Porta de Origem, nome do *host*, endereço IP do atacante da *botnet*, total de conexões efetuadas pelo atacante, protocolo pelo qual foi realizado o ataque, nome do *malware* que infectou o *host*, falha explorada no ataque, endereço IP de onde o ataque foi originado, sistema operacional do *host* em questão, tipo de configuração do *Proxy* utilizada, código criptográfico MD5 gerado sobre o *malware* em questão, aplicação utilizada para a troca de informações entre o *host* e o servidor responsável por receber estas informações, a URL de referência do incidente, título e tamanho do arquivo com direitos autorais, programa utilizado para compartilhamento de arquivo com direitos autorais, portas que foram escaneadas, serviço utilizado para tentativa de login e falha explorada na intrusão. Em geral, a ideia central aqui apresentada é armazenar estes incidentes de forma padronizada, produzindo um modelo de incidentes de segurança reusável.

### 4.3 Mapeamento do modelo para o padrão IODEF

Após a seleção inicial de atributos, o modelo criado necessita ser mapeado para o padrão IODEF. A Figura 2 ilustra o padrão IODEF adaptado ao modelo proposto.

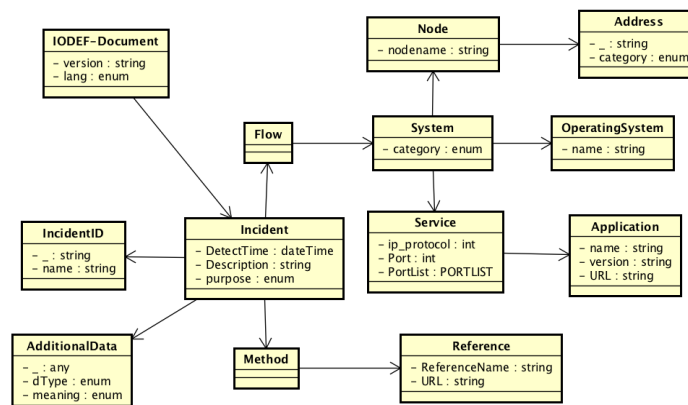


Figura 2: Padrão IODEF adaptado ao modelo proposto.

O padrão fornece a representação para alguns atributos em classes específicas do modelo e uma classe denominada *AdditionalData*, a qual pode ser utilizada para a representação de outras informações. Ainda, o IODEF obriga a criação de algumas classes e atributos os quais foram incorporados ao modelo. Deste modo, as classes que compõe o modelo proposto são: *IODEF-Document*, classe raiz do modelo IODEF, *Incident*, descrição padronizada para dados de incidentes, *Flow*, agrupa *hosts* de origem e destino relacionados, *System*, descreve sistema ou rede envolvido em um evento, *Node*, *Address*, *OperatingSystem*, *Method*, descreve o método de ataque, *Reference*, referencia vulnerabilidades associadas, *Service*, descreve serviço alvo, *Application*: descreve aplicação que provê o serviço, e *AdditionalData*, utilizada para

representação dos atributos considerados e não previstos nas classes do IODEF, tal como *Logs*, *HashDoMalware*, *Agente*, *Titulo*, *Tamanho*, *IpCC*, *IpOrigem*, *TtConexoes*, *TipoProxy*.

#### 4.4 Resolução de Incidentes de Segurança

Após a aquisição de um incidente reportado é realizada a resolução deste problema. Neste estágio, um novo caso representa um incidente que ainda não possui solução. Para que seja analisada uma solução para este novo problema, é realizada uma consulta na base de casos. Esta consulta visa buscar casos passados similares ao problema corrente.

##### Ponderação dos Atributos

Incidentes de segurança recuperados da base de casos devem ser os mais similares ao problema corrente. Dado um par de incidentes de segurança  $a$  e  $b$  (dois casos), a similaridade entre incidentes é indicada entre 0 e 1, de acordo com o menor ou maior grau de similaridade, respectivamente. A comparação de similaridade é realizada avaliando-se os  $n$  pares de atributos  $a_i$  e  $b_i$  dos casos individualmente, utilizando um peso  $W$  associado a relevância dos atributos considerados, conforme a Equação (1). O resultado da comparação indicará o quão semelhantes são os casos  $a$  e  $b$ .

$$sim(a, b) = \sum_{i=1}^n W_i \times sim_i(a_i, b_i) \quad (1)$$

O cálculo de distância entre dois casos é realizado da seguinte forma: cada caso é representado por um vetor de  $n$  atributos, o que permite a comparação atributo a atributo entre dois casos, medindo-se então a distância entre ambos. A ponderação de atributos permite descrever a importância de destes atributos no cálculo de similaridade. Desta forma, é realizado o cálculo de Distância Euclidiana Ponderada, conforme Equação (2),

$$d(a, b) = \sqrt{\sum_{i=1}^n W_i \times (a_i, b_i)^2} \quad (2)$$

onde é calculada a raiz quadrada do somatório dos pesos  $W$  multiplicados pela diferença quadrada entre os atributos  $i$  dos casos  $a$  e  $b$ . No modelo proposto, o conhecimento do especialista em relação a importância de determinado atributo pode ser retido no sistema. Uma vez retido, ele pode ser utilizado para melhorar o processo de recuperação dos casos, fazendo com que a metodologia proposta reflita esse conhecimento na resolução de incidentes.

##### Plano de Tratamento de Incidentes

Ao final do processo de resolução de um incidente, é gerado um documento que corresponde ao plano de tratamento para este incidente. A Figura 3 exemplifica um plano de tratamento de incidentes para um incidente do tipo *Bot* específico. Em muitos sentidos, este plano contém um script de solução a ser seguido visando contornar o problema.

O plano é representado por uma sequência de passos consecutivos, os quais detalham ações tomadas para tratar este incidente. Estas ações são definidas e armazenadas em uma Biblioteca de Ações voltada para a solução de incidentes de segurança, assim permitindo que diferentes tipos de ações sejam reusadas no tratamento de incidentes variados. A Figura 4 demonstra, no quadro da esquerda, três diferentes casos cujos passos para a solução encontram-se na biblioteca de ações, as quais estão representadas no quadro à direita. A biblioteca contém



todos os passos que foram utilizados para tratar diferentes incidentes. É importante notar que esta biblioteca de ações reflete o conhecimento do especialista sobre a resolução de incidentes de segurança e permite o reuso de passos.

#### Plano de Tratamento de Incidentes

##### Incidente #997164

1. Desabilitar rapidamente o acesso do host a rede de dados da Instituição
2. Abrir solicitação ao Centro de Apoio ao Usuário para enviar técnico ao local
3. Analisar evidências para comprovar e identificar o incidente recebido
4. Execute uma varredura completa com o programa antivírus
5. Se algum arquivo foi detectado, siga as instruções exibidas pelo programa de antivírus
6. Se o programa de antivírus não pode ser executado, reinicie o computador em "Modo Seguro" e repita os passos 4 e 5 deste plano e posteriormente reinicie em modo normal
7. Se o programa antivírus não pode ser executado em modo Normal e Seguro, use ferramenta específica para remoção dos arquivos
8. Caso o sistema de arquivos do Sistema Operacional tiver sido infectado, efetue uma reinstalação completa do mesmo
9. Assegurar-se de que o Sistema Operacional esteja atualizado e presencionalmente de forma automática
10. Habilitar o acesso do host a rede de dados da Instituição
11. Assegurar-se de que o firewall esteja instalado e ativo no computador
12. Assegurar-se de que o antivírus esteja instalado e com as últimas definições de vírus
13. Recomendar ao usuário a seguir as orientações da Cartilha de Segurança para Internet disponível em <https://cartilha.cert.br/>
14. Responder sobre a resolução do incidente ao CAIS

Figura 3: Plano de resposta para o tratamento de um incidente do tipo *Bot*.

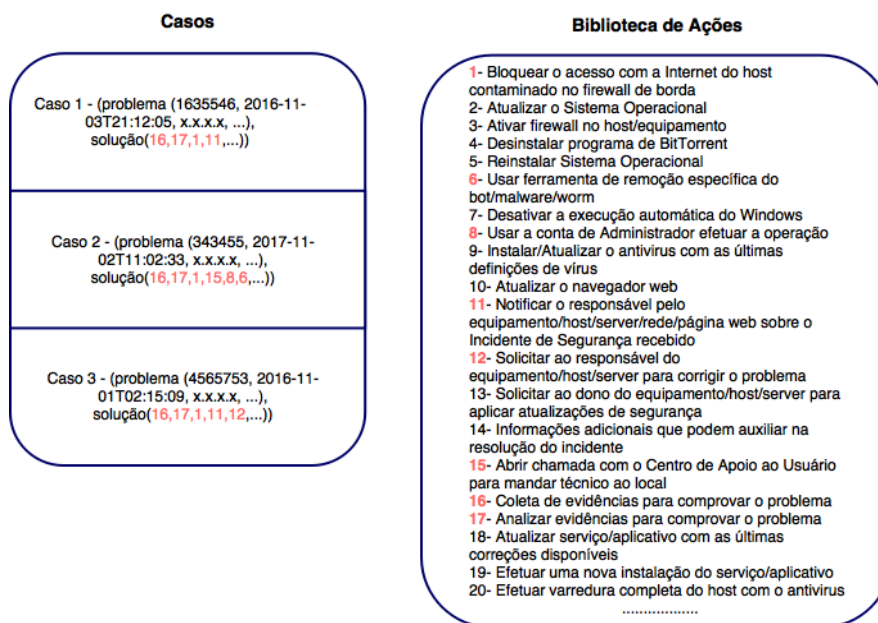


Figura 4: Compartilhamento de ações entre incidentes.

A Figura 5 demonstra como um caso RCB, problema do tipo violação de *Copyright*, é representado na base de casos. O incidente/problema está relacionado a uma solução (plano de tratamento), onde ações representam uma solução já dada para o problema.

PROBLEMA								
IdIncidente	Timestamp	IpDeOrigem	Logs	PortaOrigem	Hostname	Título	Tamanho	ProgramaCliente
1633546	2016-11-03T21:12:05	X.X.X.X	.....	55951	11	Toy Story 3	331098894	1

SOLUÇÃO								
Passo 1	Passo 2	Passo 3	Passo 4	Passo 5	Passo 6	Passo 7	Passo 8	Passo 9
16	17	1	15	33	31	21	22	23

Figura 5: Exemplificação de um caso.

## 5 Resultados

Para avaliar a metodologia proposta, dois experimentos foram realizados. O objetivo foi analisar os efeitos da ponderação de atributos em relação a precisão do sistema e avaliar a retenção do conhecimento do especialista em segurança na resolução dos incidentes.

Os incidentes utilizados foram coletados no setor de segurança de redes de um CPD empresarial. Os incidentes foram enviados pelo CAIS - Centro de Atendimento a Incidentes de Segurança - da RNP ao CPD através do Sistema de Gestão de Incidentes de Segurança (SGIS). Logo, estes incidentes foram originados dentro da instituição, devendo ser tratados. Na versão corrente do sistema, foram utilizados 258 casos advindos de diferentes tipos de incidentes. Foi utilizada a ferramenta FreeCBR [FreeCBR, 2019] para o desenvolvimento de um protótipo visando a validação deste trabalho, sendo usado o algoritmo *k-Nearest Neighbours* e distância Euclidiana ponderada na recuperação de casos da base de casos.


Para avaliar a retenção e reuso do conhecimento do especialista em resolver incidentes de segurança, um experimento preliminar foi desenvolvido. Neste contexto, cinco incidentes novos (não usados na construção do sistema) foram coletados e usados em testes. Para cada um deles, recuperou-se dois incidentes passados (mais similares) usando o sistema. Isso permitiu avaliar se as soluções dos problemas recuperados poderiam ser (re)utilizadas na solução dos problemas usados como consulta. Nestes testes, os incidentes 2102389, 2261674, 966062, 1746148 e 845538 foram usados como consulta. Em seguida, os incidentes consultados e recuperados, com os seus respectivos planos de solução definidos, foram apresentados para um especialista na área para que fosse possível realizar a comparação entre eles. Em geral, quando o plano de solução do incidente usado como consulta correspondia ao plano do incidente recuperado, o especialista entenderia que o mesmo poderia ser aplicado para a resolução do problema.

A Figura 6 apresenta o incidente 2102389, usado como consulta no sistema, e os incidentes 1483711 e 1510754, recuperados pelo sistema. O incidente consultado e ambos incidentes recuperados são do tipo *Bot*. Os dois incidentes recuperados foram gerados com quinze dias de diferença entre eles, e com alguns meses de diferença em relação ao incidente pesquisado, provavelmente pelo mesmo *botmaster*, pois os três possuem o mesmo endereço de IP de origem do controlador (*IpCC* do *botmaster*). Ambos incidentes recuperados possuem o mesmo plano de tratamento do incidente pesquisado. Portanto, o sistema obteve sucesso na recuperação de dois casos cujas soluções foram consideradas suficientemente similares, demonstrando que o conhecimento do especialista foi retido no sistema.

Para o incidente 2261674, os incidentes 1022675 e 1620589 foram recuperados pelo sistema. Ambos incidentes (consultado e recuperados) são do tipo *Copyright* representando o compartilhamento indevido de filmes por meio de um programa de *BitTorrent*. Em relação ao plano de tratamento, apenas o incidente 1022675 possuía o mesmo plano de tratamento usado no incidente de segurança usado na consulta. Portanto, a solução deste caso recuperado poderia ser diretamente utilizada para resolver o problema corrente. Porém, o incidente 1620589 possuía um plano de tratamento relativamente distinto do incidente pesquisado (2261674). A Figura 7 apresenta os planos de tratamento dos incidentes 2261674 (pesquisado) e 1620589 (recomendado) lado a lado. Percebe-se que o incidente pesquisado possui passos mais detalhados para resolver o problema em questão do que o incidente recomendado, onde os passos do

incidente recomendado demonstram como resolver o problema de maneira mais genérica. Desta forma, a solução recuperada pode não ser utilizada na solução do problema usado como consulta, principalmente pelo fato de ela não conter os detalhes necessários para resolver o incidente pesquisado, o qual trata do compartilhamento indevido de arquivos.

Incidente pesquisado			
ID 2102389			
Incidente	Solução / Passos		
HoraDetec	2017-05-09 14:44:30	1	16
IP	21	2	17
Logs	...	3	28
Descricao	...	4	29
Tipo	Bot	5	8
RefID	-	6	6
Categoria	Source	7	9
Porta	34934	8	2
Hostname	26	9	3
IpCC	65	10	31
TtConexoes	-	11	38
Protocolo	http	12	23
NomeMalware	Downadup	13	-
URLRef	36	14	-



Incidente recomendado 1			
ID 1483711			
Incidente	Solução / Passos		
HoraDetec	2016-09-01 12:21:31	1	16
IP	23	2	17
Logs	...	3	28
Descricao	...	4	29
Tipo	Bot	5	8
RefID	-	6	6
Categoria	Source	7	9
Porta	34590	8	2
Hostname	28	9	3
IpCC	65	10	31
TtConexoes	-	11	38
Protocolo	http	12	23
NomeMalware	Downadup	13	-
URLRef	36	14	-

Incidente recomendado 2			
ID 1510754			
Incidente	Solução / Passos		
HoraDetec	13-09-16 13:20:00	1	16
IP	23	2	17
Logs	...	3	28
Descricao	...	4	29
Tipo	Bot	5	8
RefID	-	6	6
Categoria	Source	7	9
Porta	42247	8	2
Hostname	28	9	3
IpCC	65	10	31
TtConexoes	-	11	38
Protocolo	http	12	23
NomeMalware	Downadup	13	-
URLRef	36	14	-

Figura 6: Incidentes 2102389, 1483711 e 1510754.

Para o incidente 966062, os incidentes 103483 e 103398 foram recuperados pelo sistema. Neste caso, tratam-se de tentativas de *login* em três *hosts* nos quais o serviço SSH estava disponível. Nesta situação o plano do incidente pesquisado difere-se dos planos dos incidentes recuperados. Assim, necessita-se de uma análise mais completa para que seja possível identificar se os diferentes planos são maneiras diferentes de resolver o mesmo problema, ou se um dos casos apresenta um plano inconsistente. No incidente 966062 os passos são realizados de forma mais detalhada do que nos incidentes 103483 e 103398. Apesar disso, os planos de incidente recuperados podem ser aplicados ao caso usado como consulta. De acordo com o especialista responsável pela resolução destes incidentes no CPD da instituição, a diferença entre os dois planos se deve ao fato de que o *host* do incidente 966062 encontra-se fisicamente alocado no próprio CPD, enquanto os *hosts* pertencentes aos outros dois incidentes encontram-se em outros prédios na mesma instituição. Assim, o primeiro incidente pôde ser resolvido pela equipe do próprio CPD, enquanto os incidentes 103483 e 103398 foram resolvidos pelo responsável pelo equipamento. Desta forma, entende-se que o conhecimento do especialista recuperado permitiu resolver o incidente pesquisado.

IdIncidente	2261674	1620589
Passo 1	Coleta de evidências para comprovar o problema	Coleta de evidências para comprovar o problema
Passo 2	Analisar evidências para comprovar o problema	Analisar evidências para comprovar o problema
Passo 3	Bloquear o acesso com a Internet do host contaminado no firewall de borda	Bloquear o acesso com a Internet do host contaminado no firewall de borda
Passo 4	Abriu chamada com o Centro de Apoio ao Usuário para enviar técnico ao local	Notificar o responsável pelo equipamento/host/server/rede/página web sobre o Incidente de Segurança recebido
Passo 5	Desinstalar o programa cliente do protocolo BitTorrent	Solicitar ao responsável do equipamento/host/server para imediatamente interromper a comunicação com a rede de dados.
Passo 6	Orientar o usuário a seguir as orientações da <a href="https://cartilha.cert.br/">Livro da Cartilha de Segurança para Internet https://cartilha.cert.br/</a>	Após as correções/atualizações/resolução do incidente de segurança, solicitar o desbloqueio no firewall de borda
Passo 7	Após as correções/atualizações/resolução do incidente de segurança, solicitar o desbloqueio no firewall de borda	Após as correções/atualizações/resolução do incidente de segurança, desbloquear no firewall de borda
Passo 8	Após as correções/atualizações/resolução do incidente de segurança, desbloquear no firewall de borda	Responder sobre a resolução do incidente ao CAIS
Passo 9	Responder sobre a resolução do incidente ao CAIS	

Figura 7: Detalhamento do plano de tratamento dos incidentes 2261674 e 1620589.

Por fim, para o incidente 1746148, os incidentes 1744804 e 1818260 foram recuperados pelo sistema. Os três incidentes dizem respeito a uma tentativa de ataque *DDoS* realizada no mesmo *host* com um dia de diferença. Neste caso, eles exploram uma falha de DNS recursivo "aberto". Ambos incidentes recuperados possuem o mesmo plano de tratamento que o incidente usado como consulta. Para o incidente 845538, os incidentes 808204 e 542693 foram recuperados pelo sistema. Os três incidentes dizem respeito a ataques do tipo *Bot* que foram realizados entre setembro e outubro pelo *ip*. Ambos incidentes recuperados possuem o mesmo plano de tratamento que o incidente pesquisado. Portanto, observa-se que o conhecimento do especialista pode ser reutilizado para resolver satisfatoriamente estes dois incidentes teste.

Complementando esta avaliação, testes automatizados foram realizados visando analisar a precisão do sistema RBC. Para isso, dividiu-se aleatoriamente os  $n$  casos da base de casos em  $p$  partições de tamanho  $t$ . A cada rodada de testes uma das  $p$  partições assume a posição de grupo de testes enquanto as demais são consideradas parte do grupo de decisão. Cada um dos casos presentes no grupo de testes tem sua representatividade aferida quando busca-se então, no grupo de decisão, outros casos semelhantes que possuem um plano de tratamento também semelhante. No caso de dois incidentes semelhantes possuírem planos semelhantes, soma-se um acerto pelo sistema de RBC. Caso contrário, soma-se um erro. Ao final de cada rodada de testes, calcula-se a subtração de acertos e erros e divide-se pelo número total de testes realizados. A acurácia em todos os experimentos foi medida utilizando-se limiares de similaridade (*Sim*) de 50% à 100% em intervalos de 5%. Dois métodos para a estimativa de precisão foram utilizados nos experimentos: *Leave-one-out and Test Cross Validation* (LOOCV) e *K-Fold Cross Validation*, ambos considerando de um a cinco vizinhos mais próximos. O *Leave-one-out and Test Cross Validation* consiste em retirar um caso por vez do grupo de testes comparando-o com os demais, fazendo assim  $n-1$  comparações, sendo então  $k=n$ . O método *K-Fold Cross Validation* consiste em dividir os dados em  $k$  subconjuntos mutuamente exclusivos de mesmo tamanho, e então cada subconjunto é utilizado como conjunto de teste e o restante dos dados como conjunto de treinamento. Assim, este processo é realizado  $k$  vezes com alternância circular dos subconjuntos de teste, sendo neste experimento utilizado  $k=10$ .

Para realizar os testes, a ponderação de atributos está diretamente relacionada com ajustes no cálculo de similaridade entre casos. Nos testes, foram realizadas duas avaliações para investigar a influência da ponderação na precisão do sistema. A primeira foi realizada com todos os atributos dos casos com peso  $w = 1$ , onde todos os atributos têm a mesma importância no cálculo de similaridade. A segunda foi realizada com ajuste de pesos realizado por um especialista em segurança da informação.

Nos testes automatizados dois casos são similares se suas "soluções" são idênticas e a parte do "problema" usada na representação destes casos possuir similaridade maior ou igual a um determinado limiar (50% a 100%). Porém, foram também considerados os vizinhos mais próximos (*k-Nearest Neighbors*) para avaliar o aproveitamento de soluções possíveis de serem reaproveitadas. Em geral, existem situações reais onde a solução recomendada pode não obter um completo sucesso na resolução do problema, fazendo necessário requerer outras recomendações de um especialista.

A Tabela 1 demonstra os resultados de precisão quando utilizado o método *K-Fold Cross Validation* sem e com ponderação de atributos. O número de partições  $p$  utilizados é 10 e o limiar de similaridade (*Sim*) avaliado é entre 50% e 100%. Foram considerados até 5 vizinhos mais próximos. Observa-se que, sem ponderação, para os limiares de similaridade compreendidos entre 100% e 75%, não há resultados precisos, ou seja, não há casos na base de casos que possuam um grau de similaridade que esteja nestes limiares. Observa-se também que embora os casos possuam similaridade abaixo de 70% entre si, o grau de precisão para limiares de 65% a 50% é considerado satisfatório. Por outro lado, observa-se resultados satisfatórios

quando utiliza-se os vizinhos para o cálculo. Quando considerado a ponderação dos atributos pelo especialista, observa-se um aumento substancial na similaridade entre os casos comparando-se com quando não utiliza-se ponderação, chegando a um limiar máximo de 95%. Os valores de precisão mantêm-se em um bom nível inclusive quando utiliza-se para o cálculo mais de um vizinho próximo. Este resultado demonstra a importância da ponderação no cálculo de similaridade, fortalecendo a metodologia proposta.

A Tabela 2 demonstra os resultados de precisão quando utilizado o método *Leave-one-out Cross Validation* sem e com ponderação de atributos e sob as mesmas condições ( $p = 10$  e  $k = 5$ ). Observa-se que entre 95% e 70% são encontrados poucos casos que satisfazem estes limiares, mas que os valores de precisão são inferiores aos calculados pelo método K-fold, se considerado os vizinhos. Porém, se considerado a ponderação do especialista, nota-se um aumento discreto na similaridade entre os casos quando comparado com resultados obtidos pelo mesmo método sem ponderação. Este resultado reforça que a ponderação de atributos pelo especialista é fundamental para capturar mais fielmente essas medidas de similaridade entre casos neste problema de aplicação e, conseqüentemente, aumentar a efetividade do sistema.

Tabela 1: Resultado de Precisão do Método K-fold.

k-NN	1	2	3	4	5
Sim (%)	Precisão (%)				
Pesos $w=1$ (sem ponderação)					
100	NaN	NaN	NaN	NaN	NaN
95	0,00	NaN	NaN	NaN	NaN
90	0,00	0,00	0,00	0,00	0,00
85	0,00	0,00	0,00	0,00	0,00
80	0,00	0,00	0,00	0,00	0,00
75	0,00	0,00	0,00	0,00	0,00
70	50,00	50,00	50,00	50,00	50,00
65	87,50	87,50	83,33	81,25	80,00
60	87,50	84,38	88,89	83,33	80,00
55	82,35	79,41	78,43	73,53	70,59
50	82,35	79,41	78,43	73,53	70,59
Pesos ponderados por especialista					
100	NaN	NaN	NaN	NaN	NaN
95	93,33	90,00	95,24	91,67	90,00
90	82,35	79,41	83,33	78,57	77,14
85	82,35	79,41	83,33	78,13	75,00
80	82,35	79,41	83,33	78,13	75,00
75	82,35	79,41	78,43	73,53	70,59
70	75,00	70,00	68,33	63,75	61,00
65	75,00	70,00	68,33	63,75	61,00
60	75,00	70,00	68,33	63,75	61,00
55	75,00	70,00	68,33	63,75	61,00
50	75,00	70,00	68,33	63,75	61,00

Tabela 2: Resultado de Precisão do Método LOOCV

k-NN	1	2	3	4	5
Sim (%)	Precisão (%)				
Pesos $w=1$ (sem ponderação)					
100	NaN	NaN	NaN	NaN	NaN
95	66,67	33,33	50,00	0,00	0,00
90	66,67	33,33	33,33	25,00	0,00
85	50,00	33,33	33,33	25,00	20,00
80	50,00	33,33	33,33	25,00	20,00
75	50,00	33,33	33,33	25,00	20,00
70	59,26	63,89	71,11	76,92	75,38
65	78,74	78,93	78,45	76,96	75,83
60	78,18	75,59	74,16	73,90	73,73
55	77,55	75,11	72,82	71,79	71,45
50	75,20	72,11	69,87	68,45	68,52
Pesos ponderados por especialista					
100	100,00	NaN	NaN	NaN	NaN
95	85,44	88,42	90,45	93,37	94,93
90	79,34	77,49	77,48	77,49	77,78
85	78,31	75,52	73,95	74,67	76,42
80	78,40	75,52	73,78	73,40	75,57
75	77,47	75,00	73,33	71,72	73,25
70	77,34	74,31	72,18	70,37	71,03
65	77,04	74,02	71,90	70,10	70,35
60	76,74	73,84	71,71	69,96	70,08
55	76,74	73,84	71,71	69,96	70,08
50	76,74	73,84	71,71	69,96	70,08

## 6 Conclusão

Este trabalho apresentou uma metodologia para organizar uma memória de resolução de incidentes de segurança com o auxílio da técnica de RBC e que potencializa o reuso do conhecimento, mitigando a perda do conhecimento mantido por especialistas em segurança. A solução explora o padrão de representação de incidentes IODEF, possibilitando a comunicação entre ambientes heterogêneos, e demonstra que, a partir de um conjunto de atributos para a representação dos incidentes de segurança, é possível recuperar e reusar o conhecimento prévio sobre resoluções de incidentes. A solução também permite explorar a ponderação de atributos, permitindo reusar o conhecimento do especialista em relação à importância dos atributos. O resultado é uma solução que possibilita reusar planos de tratamento de incidentes de segurança, os quais são sugeridos de forma automatizada via respostas para consultas de RBC.

Para avaliação do sistema construído, o qual apresenta uma implementação concreta da solução proposta neste trabalho, dois experimentos foram realizados. O primeiro buscou explorar

a avaliação da proposta por um especialista, o qual foi envolvido na solução de novos problemas de segurança usando o sistema desenvolvido. O segundo foi uma análise via dois métodos de validação cruzada. Os resultados obtidos nestes experimentos demonstraram a efetividade da metodologia no que diz respeito a manutenção e reuso de conhecimento sobre resolução de incidentes de segurança.

## Referências

- Dalkir, K. e Liebowitz, J. (2011). *Knowledge Management in Theory and Practice*. MIT Press.
- Rahimli, A. (2012). Knowledge management and competitive advantage. In *Information and Knowledge Management*, pp. 37-43.
- Hove, C. e Tarnes, M. (2013). *Information Security Incident Management: An Empirical Study of Current Practice*. Norwegian University of Science and Technology, Trondheim, Norway.
- CERT.br. Estatísticas dos Incidentes Reportados ao CERT.br. Disponível em <https://www.cert.br/stats/incidentes/>. Acessado em: 24/07/2019.
- Inayat, Z., Gani, A., Anuar, N. B., Khan, M. K. e Anwar, S. (2016). Intrusion response systems: Foundations, design, and challenges. *Journal of Network and Computer Applications*, vol. 62, pp. 53.
- Richter, M. M. e Weber, R. O. (2013). *Case-Based Reasoning: A Textbook*. Springer.
- Takahashi, T., Landfield, k. e Kadobayashi, Y. (2014). An Incident Object Description Exchange Format (IODEF) Extension for Structured Cybersecurity Information, RFC 7203, April.
- Takahashi, T. e Miyamoto, D. (2016). Structured cybersecurity information exchange for streamlining incident response operations. In *Network Operations and Management Symposium*, pp. 949-954.
- Gascon, H. et al, (2017). Mining attributed graphs for threat intelligence. In *Proc of the 7th ACM Conference on Data and Application Security and Privacy*, Arizona, USA, pp. 15-22.
- Mansar, S. L., Marir, F. e Reijers, H. A. (2003). Case-based reasoning as a technique for knowledge management in business process redesign. *Electronic Journal on Knowledge Management*, vol. 1, (2), pp. 113-124.
- ISO/IEC 27035. (2016). Information technology - security techniques - information security incident management. *International Organization for Standardization*, Geneva, CH, November.
- Metzger, S., Hommel, W. e Reiser, H. (2011). Integrated security incident management – concepts and real-world experiences. In *Proc of the 2011 Sixth International Conference on IT Security Incident Management and IT Forensics*, pp. 107-121.
- Line, M. B., Tøndel, I. A. e Jaatun, M. G. (2016). Current practices and challenges in industrial control organizations regarding information security incident management – Does size matter? Information security incident management in large and small industrial control organizations. *International Journal of Critical Infrastructure Protection*, vol. 12, (Supplement C), pp. 12.
- Rajnovic, D. (2011). *Computer Incident Response and Product Security*. Indianapolis, USA: Cisco Press.
- Anuar, N. B., Papadaki, M., Furnell, S. e Clarke, N. (2010). An investigation and survey of response options for intrusion response systems (IRSs). In *Information Security for South Africa*, pp. 1-8.
- Jiang, F., Gu, T., Chang, L. e Xu, Z. (2014). Case Retrieval for Network Security Emergency Response Based on Description Logic. In *Int. Conf. on Intelligent Information Processing*, China, pp. 284-293.
- Kim, H. K., Im, K. H. e Park, S. (2010). DSS for computer security incident response applying CBR and collaborative response. *Expert Systems with Applications*, vol. 37, (1), pp. 852-870.
- Ping, L., Haifeng, Y. e Guoqing, M. (2010). An incident response decision support system based on CBR and ontology. In *Int Conf on Computer Application and System Modeling*, pp. 11-337-11-340.
- OASIS. (2019). Introduction to STIX. Disponível em: <https://oasis-open.github.io/cti-documentation/stix/intro>. Acessado em: 24/07/2019.
- Software FreeCBR. (2019). Disponível em: <http://freecbr.sourceforge.net/>. Acessado em: 23/07/2019.