Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação
E-ISSN: 1518-2924
bibli@ced.ufsc.br
Universidade Federal de Santa Catarina
Brasil

de Azevedo, Ryan Ribeiro; Ataíde Dias, Guilherme; Gonçalves de Freitas, Frederico Luiz; Campos Veras, Wendell; Rocha, Rodrigo

Um sistema autonômico baseado em ontologias e agentes inteligentes para uso em segurança da informação

Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação, vol. 17, núm. 35, septiembre-diciembre, 2012, pp. 167-184
Universidade Federal de Santa Catarina
Florianopolis, Brasil

ENSAIO

Recebido em:
22/04/2012

Aceito em:
01/12/2012

Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação, v. 17, n. 35, p 167-184, set./dez., 2012. ISSN 1518-2924. DOI: 10.5007/1518-2924.2012v17n35p167

Um sistema autonômico baseado em ontologias e agentes inteligentes para uso em segurança da informação[1]

An autonomic system based on ontologies and intelligent agents for use in information security

Ryan Ribeiro de AZEVEDO [2]
Guilherme Ataíde DIAS [3]
Frederico Luiz Gonçalves de FREITAS[4]
Wendell Campos VERAS[5]
Rodrigo ROCHA[6]

RESUMO

Este artigo apresenta um sistema autonômico baseado em ontologias e agentes Inteligentes para uso em Segurança da Informação, tendo como intuito resguardar a infraestrutura computacional e de tecnologia da informação protegidas de agentes maliciosos. Como suporte teórico para o desenvolvimento da pesquisa utilizou-se de conceitos da Ciência da Informação e Ciência da Computação. São apresentados resultados do uso do sistema proposto em ambiente simulado. Como estratégia de avaliação do sistema, foi realizada uma avaliação do uso do sistema em cenários simulados com intuito de verificar e analisar o potencial da ferramenta proposta e seu funcionamento autonômico nas atividades de segurança da informação. A avaliação consistiu da aplicação de ataques de negação de serviço (DoS - *Denial of Service*) e *SYN Flooding.* O AutoCore atingiu os objetivos desejados, os resultados apresentados demonstram que o AutoCore é uma ferramenta adequada para o tratamento e utilização da informação no que diz respeito à segurança da informação, possibilitando aos responsáveis pela Gestão de Riscos e Gestão de Segurança da Informação tomarem decisões estratégicas de alinhamento das Tecnologias de Informação e Comunicação e Segurança aos processos de negócios das organizações.

PALAVRAS-CHAVE: Representação do Conhecimento. Ontologias. Segurança da Informação. Tecnologia da Informação.

1 Pesquisa parcialmente financiada através do Edital MCT/CNPq/MEC/CAPES nº 02/2010
[2] Universidade Federal de Pernambuco - rra2@cin.ufpe.br
[3] Universidade Federal da Paraíba - guilhermeataide@gmail.com
[4] Universidade Federal de Pernambuco - fred@cin.ufpe.br
[5] Universidade Federal do Rio Grande do Norte - wendellweb@gmail.com
[6] Universidade Federal de Pernambuco - rgcrocha@gmail.com

v. 17, n. 35, 2012.
p. 167-184
ISSN 1518-2924

ABSTRACT

This paper presents an autonomic system based on ontologies and intelligent agents for use in information security, aiming to protect the computing infrastructure and information technology from malicious agents. Theoretical support for the research development was grounded on concepts from Information Science and Computer Science. The results of using the proposed system in a simulated environment are presented. A strategy for system evaluation was performed to check the system use in simulated scenarios to verify and analyze the potential of the proposed tool and its autonomic functioning in activities of information security. The evaluation consisted in the execution of denial of service attacks (DoS) and SYN Flooding. The AutoCore achieved the desired objectives. The results show that the AutoCore is a suitable tool for the treatment and utilization of information with regard to information security, enabling those responsible for Risk Management and Information Security Management to make strategic decisions alignment of Information and Communication Technologies Security with the business processes of organizations.

KEY-WORDS: Knowledge Representation. Ontologies. Information Security. Information Technology.

1 INTRODUÇÃO

Os diversos ambientes de atuação humana, sejam eles acadêmicos, corporativos ou militares, necessitam de meios transparentes para gerir as diversas situações relacionadas à segurança de seus ativos, especificamente à segurança da informação. Os problemas relacionados à segurança da informação passaram a ser uma questão estratégica para os negócios. A IBM (*International Business Machines*) apresentou um manifesto (HORN, 2001) alertando a indústria de Tecnologia da Informação e Comunicação (TIC) para uma nova crise.

Horn (2001) acrescenta que os sistemas computacionais têm evoluído consideravelmente desde que surgiram, ao mesmo tempo em que tornaram-se mais complexos de serem gerenciados. Alguns motivos que contribuem para esse fato são a segurança de sistemas de missão crítica, a necessidade crescente de interconectividade, a integração de tecnologias heterogêneas e a alocação satisfatória dos recursos computacionais (KEPHART & CHESS, 2003). Há,

portanto, um aumento significativo na complexidade de projetar e planejar segurança, necessitando que meios de manipulação, interpretação e tratamento de informações sejam adotados.

O conjunto de tecnologias que compõem a arquitetura da Web Semântica, juntamente com sistemas baseados em agentes inteligentes e computação autonômica, com perspectiva sensível ao contexto permite que sistemas computacionais manipulem e processem essas informações através de ontologias, aplicadas em diversas áreas que englobam acesso, organização, compartilhamento e uso de informação semântica. Ontologias têm contribuído para facilitar a comunicação e o processamento da informação semântica, tanto entre sistemas baseados em agentes quanto entre seres humanos, promovendo assim, interoperabilidade entre sistemas ao representarem dados compartilhados por diversas aplicações (USCHOLD & GRÜNINGER, 1996).

A partir da construção de bases de conhecimentos com foco em aspectos relacionados à segurança da informação, organizações poderão desenvolver e implantar mais facilmente mecanismos de proteção, correção e prevenção de acordo com os seus requisitos de segurança, requisitos estes expressos através de Acordos de Níveis de Serviços (SLA-*Service Level Agreement*) ou de políticas de segurança exigidas para que seus serviços estejam sempre disponíveis, íntegros e confiáveis. Portanto, um sistema computacional autonômico e baseado na computação ubíqua (WEISER, 1991) poderá realizar tarefas relacionadas à administração da segurança informacional de forma automática e transparente, mantendo a sua integridade, havendo assim, pouca ou nenhuma intervenção humana no processo.

Neste trabalho de pesquisa é apresentado o AutoCore, um sistema multiagente e pervasivo baseado em ontologias para apoiar as atividades realizadas pelos responsáveis pela segurança da informação em organizações. Os objetivos contemplados nesta pesquisa foram: formalizar e desenvolver uma ontologia para representação de informações sobre segurança da informação para especificar, tratar e mitigar riscos de segurança em diversos ambientes corporativos; Desenvolver uma arquitetura autonômica baseada em ontologias, com intuito de auxiliar os responsáveis pela segurança da informação na

proteção, cura, otimização e configuração de sistemas computacionais corporativos, com capacidade de determinar o diagnóstico mais próximo da situação atual e/ou aprender autonomamente um novo diagnóstico para casos sem semelhanças nas ontologias e capacidade de considerar contexto ao prover soluções.

As demais seções deste artigo estão organizadas conforme descrição a seguir: na seção 2 são apresentados os trabalhos existentes relacionados ao tema. O referencial teórico necessário para um entendimento geral das tecnologias acerca da pesquisa é apresentado na seção 3. Na seção 4 é detalhada a pesquisa. Na seção 5 são apresentados e discutidos os resultados. Por fim, as conclusões e os trabalhos futuros são delineados na seção 6.

2 TRABALHOS RELACIONADOS

Nos últimos anos, vários esforços foram despendidos com o intuito de se obter modelos autonômicos que permitissem auxiliar a redução da complexidade na proteção de sistemas computacionais e que contribuissem com a segurança da informação, tendo como objetivo precípuo o fornecimento de diferentes níveis de proteção aos ativos de informação associados às organizações. Embora com diferentes focos e níveis de detalhamento, encontra-se na literatura trabalhos que abordam os sistemas autonômicos no domínio de segurança a fim de gerenciar, avaliar e especificar segurança, seja da informação ou não, nos mais diversos ambientes. Dentre esses trabalhos que ajudaram no amadurecimento e suporte no desenvolvimento da pesquisa ora descrita, destacam-se:

O trabalho de (ALMEIDA et al, 2007) propõe o uso de uma ontologia para a gestão de segurança da informação em ambientes computacionais autonômicos denominada OSACA. Apesar de apresentar contribuições importantes, a OSACA é um trabalho que ainda pode ser desenvolvido. A ontologia não foi implementada utilizando linguagens da Web Semântica, como *Resource Description Framework* (RDF) ou *Web Ontology Language* (OWL), não possui axiomas nem restrições em DL (Lógica de Descrição) que são os padrões de fato estabelecidos. O Trabalho carece da apresentação de resultados, não garantindo desta forma a sua aplicação efetiva em atividades práticas.

No trabalho de (MARTIMIANO, 2006) foi desenvolvida uma ontologia denominada OntoSec, representando de maneira padronizada informações relativas a incidentes de segurança. Esta ontologia possibilita facilitar o compartilhamento e reuso de informações, o gerenciamento e geração de conhecimento sobre incidentes e provê uma ferramenta voltada para o auxílio dos responsáveis pela segurança na corporação, os *Chief Security Officers* (CSO). A principal diferença entre o trabalho de Martimiano (2006) e o aqui apresentado é que o sistema desenvolvido para auxiliar os CSO não é baseado em agentes, não possuindo autonomia para realizar correções preventivas nem analisar ataques em tempo real. Já o AutoCore é um sistema autonômico e possui características mais sofisticadas que OntoSec em seu sistema de auxílio.

Cita-se ainda o sistema *eAutomation* (Stojanovic, 2004) desenvolvido com foco em um sistema de correlação com mesmo nome, mas não trata especificamente a questão da segurança da informação, além de descrever um conjunto muito reduzido de classes na ontologia que propõe. A pesquisa apresentada trás melhorias significativas em relação aos trabalhos mencionados nesta seção, embora não podemos garantir que a mesma seja a solução final para os cenários destacados. Nas próximas seções são apresentados os conceitos (Computação autonômica, ubíqua e ontologias) necessários para a compreensão da proposta descrita.

3. TECNOLOGIAS UTILIZADAS

3.1 Computação Autonômica e Computação Ubíqua

Em 2001, a IBM (HORN, 2001) apresentou ao mercado uma proposta de solução para o problema da excessiva complexidade do *software* corporativo e a denominou de computação autonômica. A ideia segue uma abordagem comum dentro das diversas ramificações da engenharia que é a de buscar inspiração no sistema nervoso autonômico humano, o qual é capaz de manter as funções vitais a fim de compreender e resolver os mais diversos tipos de problemas sem qualquer ou pouca iniciativa, participação ou intervenção direta do ser humano.

Um sistema autonômico pode ser definido como sendo aquele capaz de se gerenciar de acordo com objetivos de alto nível definidos pelo administrador (KEPHART & CHESS, 2003). Uma das principais metas deste sistema é a capacidade de autogerenciamento, livrando o administrador de sistemas da preocupação com detalhes operacionais e possibilitando ao usuário empregar as máquinas com melhor desempenho durante todo o tempo. Aliado ao objetivo da computação ubíqua (Weiser, 1991) que propõe possibilitar ao usuário o acesso a ambientes computacionais, em qualquer lugar e a qualquer momento, mas de forma que o mesmo não perceba a interação com a máquina, sendo esta autônoma, interativa e relevante, surge um paradigma computacional com possível capacidade de auxiliar de forma menos intrusiva os seres humanos em suas atividades profissionais.

Para atingir os objetivos mencionados e ser considerado um sistema de computação autonômica é necessário atender a quatro serviços básicos de acordo com Parashar & Hariri (2007):

1. Auto-Configuração (*Self-Configuring*): capacidade do sistema de se configurar em tempo de execução, com quase ou nenhuma intervenção humana. Dessa forma, o ambiente de TIC pode se acomodar a novas configurações dinamicamente seja pela inclusão, alteração ou remoção de componentes;
2. Auto-Otimização (*Self-Optimizing*): capacidade do sistema em otimizar a alocação dos recursos existentes, de tal forma que as necessidades dos usuários sejam atendidas sem comprometimento dos demais recursos. A idéia é que, por exemplo, questões relacionadas com desempenho e qualidade de serviço possam ser tratadas de acordo com políticas de alto nível e de forma transparente;
3. Auto-Cura (*Self-Healing*): habilidade do sistema de identificar falhas e executar ações corretivas, sem que para isso seja necessário paralisar o funcionamento dos serviços. Sendo assim, o sistema poderá, após a ocorrência de uma falha, voltar a um estado estável sem a necessidade de intervenção do administrador;

4. Auto-Proteção (*Self-Protecting*): habilidade do sistema em detectar comportamentos maliciosos ou hostis e tomar decisões eficientes e eficazes de forma transparente sobre qual é a ação mais adequada para impedir ou minimizar um ataque.

3.2 Ontologias

Desde o século XVII, o termo "ontologia" tem sido utilizado para denominar a disciplina de metafísica geral, dentro da tradição da "primeira Filosofia" de Aristóteles, como sendo a ciência do ser no papel de ser (Freitas et al, 2009). Diversas definições têm surgido a fim de descrever o que é uma ontologia, tanto na área da Ciência da Computação (CC), como na área da Ciência da Informação (CI). Na área da CC a definição mais amplamente conhecida é de que a ontologia seria "uma especificação formal e explícita de uma conceitualização compartilhada" (Gruber, 1995), onde: - *formal* implica em ser declarativamente definida, portanto, compreensível para agentes e sistemas; - e*xplícita* significa que os elementos e suas restrições estão claramente definidos; - *conceitualização* trata de um modelo abstrato de uma área de conhecimento ou de um universo limitado de discurso; - *compartilhada*, indica um conhecimento consensual, seja uma terminologia comum da área modelada, ou acordada entre os desenvolvedores dos agentes que se comunicam.

Com relação às ontologias, Sales e Café (2009, p.101), autores da área da CI trazem o seguinte esclarecimento:

> As ontologias possibilitam compartilhar uma visão de determinado campo de conhecimento, compartilhar uma forma de pensar de determinado assunto, proporcionando um mapa semântico e uma estrutura conceitual de um domínio específico por meio de um vocabulário comum (SALES e CAFÉ, 2009, p.101).

Almeida e Bax (2003, p.9) trazem ainda que:

> A ontologia define as regras que regulam a combinação entre os termos e as relações. As relações entre os termos são criadas por especialistas, e os usuários formulam consultas usando os c onceitos especificados. Uma ontologia define assim uma "linguagem"

> (conjunto de termos) que será utilizada para formular consultas (ALMEIDA e BAX, 2003, p.9).

As definições das duas áreas mencionadas (CC e CI) com relação ao conceito de ontologias são convergentes, embora, eventuais diferenças terminológicas sejam possíveis.

A utilização de ontologias proporciona algumas vantagens, dentre elas indicamos como fundamental a oportunidade para os desenvolvedores reusarem ontologias e bases de conhecimento, mesmo com adaptações e extensões. Desta forma ressaltamos que o reuso de ontologias pode promover um ganho significativo em termos de esforços e de investimentos. A grande disponibilidade de "ontologias de prateleira", prontas para uso, reuso e comunicação entre agentes, podendo estas serem estendidas e complementadas com conceitos de domínios específicos (FREITAS, 2003). Outras vantagens conforme apresentado por Freitas (2003) seriam: O acesso *on-line* a servidores de ontologias, capazes de armazenar milhares de classes e instâncias, servindo a empresas ou grupos de pesquisa, funcionando como ferramentas para manter a integridade do conhecimento compartilhado entre elas e garantindo um vocabulário uniforme; Possibilidade de tradução entre diversas linguagens e formalismos de representação do conhecimento tais como: *CLIPS*, *Jess*, *Prolog*, *XML*, *RDF*, *OWL*, *OIL*, *DAML-OIL* e *FLogic*.

4 APRESENTAÇÃO DA PESQUISA

4.1 AutoCore

Nesta seção apresentamos o AutoCore, um sistema multiagente autonômico baseado em ontologias. O núcleo do AutoCore foi implementado utilizando o *framework* de *software* conhecido como JADE (*Java Agent Development Framework*). Os agentes foram organizados em *containers* de agentes, onde cada *container* é composto por um conjunto de agentes modelados de acordo com os serviços básicos inerentes a um sistema

autonômico (Auto-Otimização, Cura, Configuração e Proteção) explicados na Seção 3.1 deste artigo.

O AutoCore faz uso de uma ontologia de domínio como base de conhecimento. Esta ontologia é denominada CoreSec e modela elementos do domínio relacionado à segurança da informação. Parte da CoreSec é ilustrada na Figura 1.

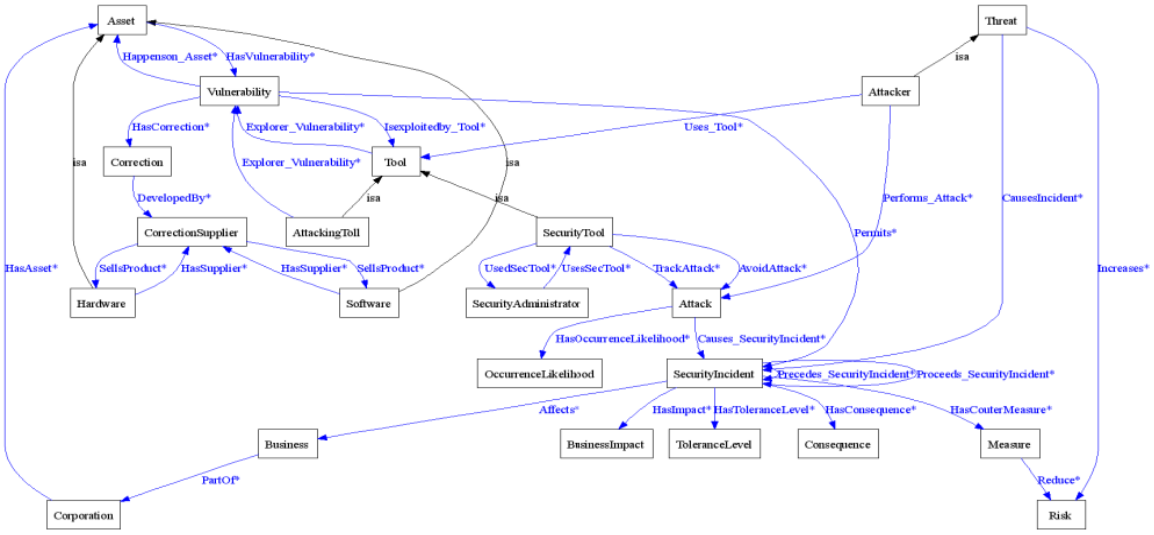

Figura 1 - **Parte das classes e relacionamentos que formam a CoreSec**

FONTE: Desenvolvimento nosso.

A Figura 2 ilustra a arquitetura do AutoCore, seu funcionamento e os componentes que a constituem. O sistema é composto por um Agente Gerenciador, responsável por gerenciar as colônias de agentes verificadores, avaliadores, acionadores e visualizadores, definindo os objetivos e as tarefas que deverão ser executadas por cada uma delas. O Agente Gerenciador possibilita a Auto-Configuração, visto propiciar às colônias, em tempo de execução, as regras a serem seguidas.

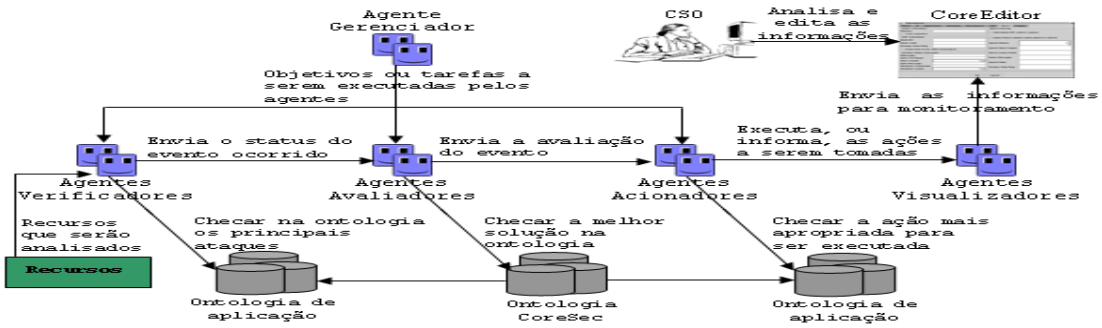

Figura 2 - **Arquitetura do AutoCore**

FONTE: Veras (2010, p.80)

Os Agentes Verificadores ficam em constante análise dos recursos a que estão ligados, sendo responsáveis pela coleta e análise do tráfego de rede, possuem o objetivo de detectar comportamentos hostis e classificá-los. Para cada categoria de ataque existe um conjunto de agentes especializados, possibilitando a detecção de possíveis atividades maliciosas. No momento em que são instanciados, os Agentes Verificadores consultam uma sub-ontologia, criada a partir da CoreSec, e mantêm em memória interna todas as configurações atuais para detecção de anomalias associadas aos recursos. Caso seja detectada alguma alteração não configurada, uma nova inserção é realizada à ontologia através do CoreEditor.

Os Agentes Verificadores têm a tarefa de enviar aos Agentes Avaliadores as informações de status obtidas. Juntos caracterizam a Auto-Proteção, pois enquanto um agente detecta um mau funcionamento, o outro toma decisões eficientes e eficazes de como solucionar determinado problema encontrado.

Após receberem as informações do estado do ambiente, os Agentes Avaliadores consultam a CoreSec para descobrir as melhores medidas a serem tomadas, enviando as novas informações aos Agentes Acionadores. As consultas realizadas pelos Agentes Avaliadores possibilitam uma segunda caracterização, a Auto-Otimização, visto possibilitarem a otimização dos recursos sem comprometerem os demais recursos. Os Agentes Avaliadores constroem diversas soluções baseadas em métricas para a correta resolução das vulnerabilidades detectadas, enviando aos Agentes Acionadores alguns indicadores como: Impacto nos Negócios, Tempo de Resolução, Eficácia e Eficiência, Qualidade e Custo das Resoluções.

Os Agentes Acionadores são os responsáveis em receber as avaliações e com base nestas informações executarem as melhores ações a serem tomadas. Toda ação realizada, mesmo que não seja de execução automática de alguma medida de prevenção, é enviada aos Agentes Visualizadores, e estes disponibilizam para o CSO através do CoreEditor. Caso seja necessária uma intervenção na decisão tomada pelos Agentes Acionadores, o CSO utiliza o CoreEditor para tal ação. Os Agentes Acionadores realizam também Auto-Cura,

visto que após a identificação das falhas, executam ações corretivas e preventivas sem paralisarem o funcionamento dos serviços, possibilitando retornar a um estado estável sem intervenções humanas. Esses agentes também consultam uma sub-ontologia, criada a partir da CoreSec, sempre antes de executar uma ação.

Os Agentes Visualizadores são responsáveis por exibir gráficos, relatórios gerais e de execução através do CoreEditor, de maneira a manter os responsáveis pela segurança da informação cientes a respeito da situação atual de todo o sistema, inclusive, facilitando as possíveis intervenções nas resoluções em momentos críticos de riscos aos sistemas.

Para ajudar na monitoração e atualização da ontologia (CoreSec) utilizada pelo AutoCore foi desenvolvida uma aplicação denominada CoreEditor para uso exclusivo do CSO, com intuito de auxiliar na transmissão, geração e distribuição do conhecimento, manipulação, avaliação e utilização da CoreSec. Um aspecto a ser considerado é que o CoreEditor não é apenas uma ferramenta para manipulação da CoreSec, mas também, um *framework* para criação e edição de ontologias. Desenvolvido utilizando o *framework Jena*, o CoreEditor é uma aplicação exclusiva para manipulação de ontologias, utilizada quando for necessária a intervenção humana na manutenção da ontologia.

Entre as principais características e funcionalidades do CoreEditor estão: uso da linguagem OWL para criação de ontologias (Conceitos, Propriedades e Indivíduos) e a máquina de inferência *Pellet* para a geração de hipóteses a partir das informações nas ontologias; manipulação e geração de código para aplicações da ontologia desenvolvida; geração do OntoDoc, documento similar ao *javadoc* da Linguagem *Java*; geração de diagramas de classes UML da ontologia desenvolvida e interfaces de consultas utilizando a linguagem *SPARQL*.

O CoreEditor está estruturado a partir dos seguintes módulos principais:

- Módulo de Inferência. Permite inferir informações acerca da CoreSec sobre o código OWL e RDF e com a utilização da máquina de inferência *Pellet*, utilizando a capacidade de representação da ontologia.

- Módulo de Consulta. Onde são realizadas as consultas pelos usuários da aplicação utilizando a CoreSec como base de conhecimento. O módulo de consulta interage com o módulo de inferência.

- Módulo de Manipulação. Responsável pela criação, edição e exclusão de classes e subclasses, propriedades (Tipo de Dados e Objetos), instâncias, tipos de relacionamentos (Transitivo, Simétrico, Funcional e Inversamente Funcional), criação de novas ontologias, geração de códigos RDF, OWL, UML e código *Java* referente à ontologia desenvolvida.

- Módulo de Visualização. Permite a visualização de gráficos com instâncias, heranças e outros tipos de relacionamentos, configurados de acordo com as necessidades dos usuários finais.

- Módulo de Engenharia de Ontologias. Permite a criação de ontologias com suporte à metodologia *Methontology* (FERNÁNDEZ et al, 1997), sendo a ontologia criada e documentada automaticamente. Os experimentos e resultados da utilização do AutoCore são apresentados na seção seguinte.

5 EXPERIMENTOS E RESULTADOS

Foram realizados experimentos controlados com intuito de verificar e analisar o potencial da ferramenta proposta e seu funcionamento autonômico, foram realizados ataques de negação de serviço, DoS (*Denial of Service*) e *SYN Flooding*, caracterizados pelo envio de múltiplas solicitações de abertura de conexão a um mesmo *host* em uma mesma porta, em um curto período de tempo. Desenvolvemos um simulador para esse tipo de ataque, no qual configuramos a quantidade de requisições ao servidor, a porta atacada e o IP atacado. A Figura 3 ilustra o simulador em funcionamento.

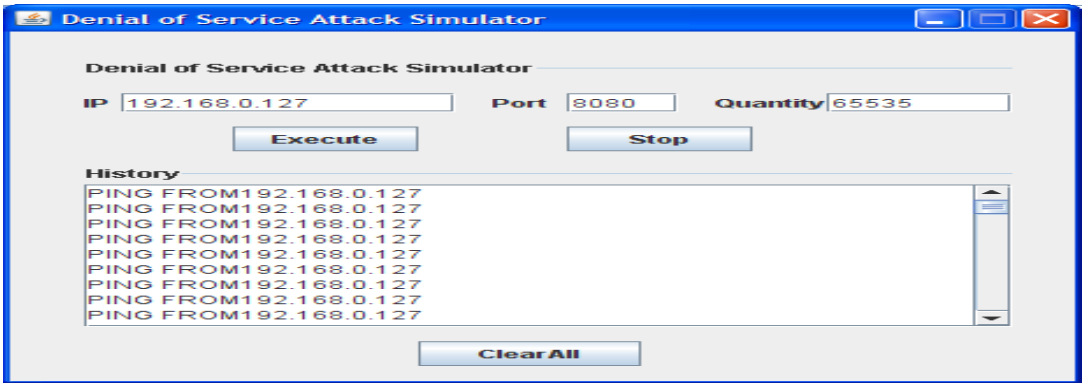

Figura 3 - **Simulador de Ataque DoS**

FONTE: Veras (2010, p.80)

Nos campos IP e *Port* são inseridos o endereço IP e a porta alvo do ataque DoS. O campo *Quantity* informa ao simulador quantas requisições deverão ser realizadas. No momento em que o AutoCore é inicializado o mesmo detecta o ataque DoS realizado (Ver Figura 4) e inicia o processo de criação dos agentes no momento da simulação do ataque (Ver Figura 5).

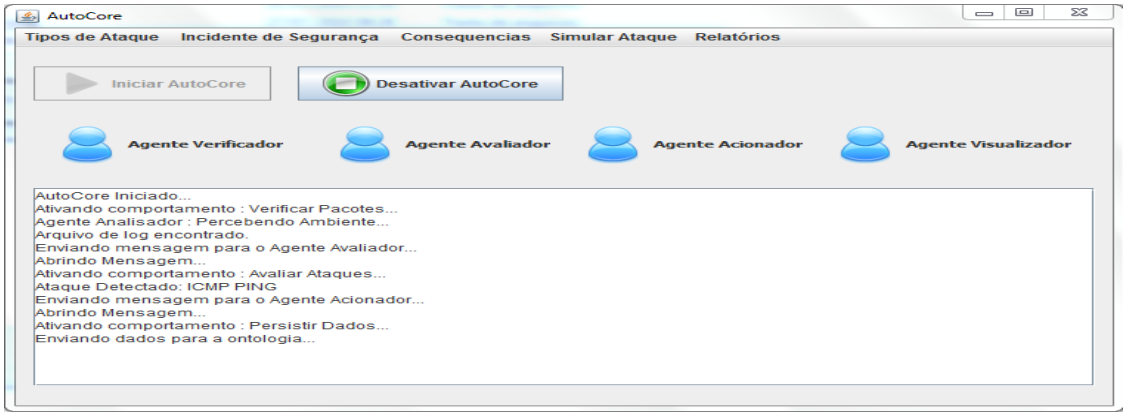

Figura 4 - **Tela inicial do AutoCore em Funcionamento Detectando um Ataque**

FONTE: Desenvolvimento nosso

Para explicar a Figura 5, utilizamos a ferramenta de monitoramento do processo de comunicação dos pacotes (*sniffer*) do JADE.

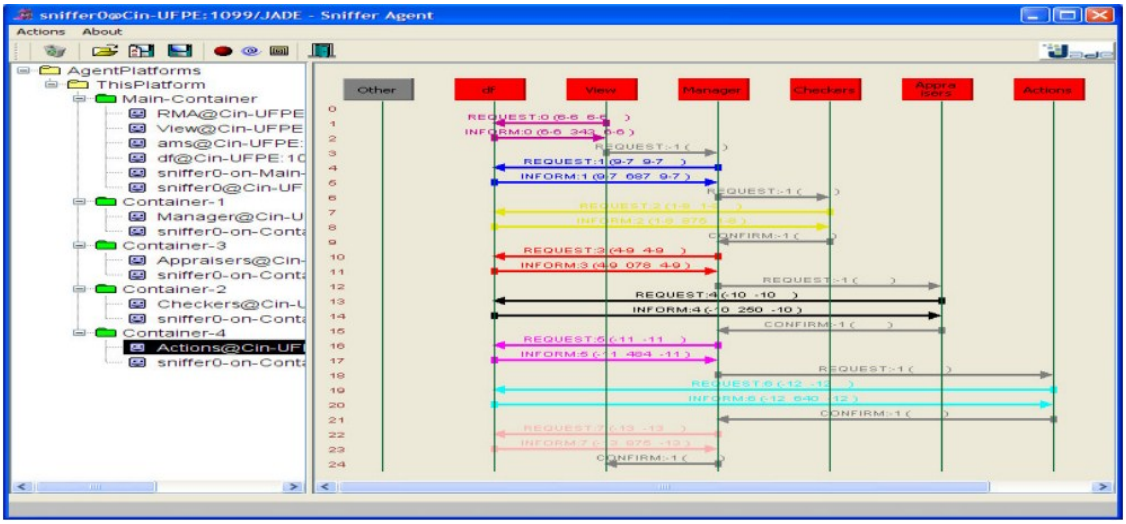

Figura 5 - **Inicialização e Criação dos Agentes**

FONTE: Veras (2010, p.85)

O funcionamento acontece da seguinte forma: O Agente Visualizador (*View*) processa o evento de requisição de inicialização do sistema, visto ser o responsável por monitorar os eventos da interface entre o sistema e o usuário. Logo, ele envia ao Agente Gerenciador (*Manager*) a solicitação de *start* do sistema e este por sua vez, inicializa os Agentes Verificadores (*Checkers*), Avaliadores (*Appraisers*) e Acionadores (*Actions*). É possível notar que antes da conexão com cada agente, é realizada uma consulta ao Facilitador de Diretórios (DF – *Directory Facilitator*), sendo este responsável por informar o identificador (ID) dos agentes, possibilitando, portanto, a comunicação e troca de mensagens. As mensagens são codificadas utilizando o protocolo de interação *contract-net*. Após receber a confirmação da inicialização de cada agente, o Agente *Manager* confirma essa informação ao Agente *View*, finalizando o processo de *start* do sistema. Os Agentes *Checkers* são os responsáveis por verificar o tráfego da rede. No momento em que percebem uma alta frequência de pacotes vindos de um mesmo *host*, considerando ainda o tamanho e o tipo do pacote em um dado período de tempo, detectam estar havendo algo anormal, baseado nas regras da base de conhecimento de uma instância da CoreSec carregadas durante a inicialização. Após consultarem o DF e saberem para quais agentes devem enviar essas informações, fazem uma requisição para os Agentes *Appraisers* enviando as informações de *status* da anomalia da rede.

Os Agentes *Appraisers*, baseados nas descrições informadas pelos Agentes *Checkers*, consultam a ontologia CoreSec com o objetivo de avaliar a anomalia. Após consultarem o DF, fazem uma requisição aos Agentes *Actions*. Estes devem tomar decisões para amenizar o problema e para isso consultam uma instância da CoreSec, procurando informações específicas para resolução da anomalia em questão. A Auto-Cura caracteriza-se, então, pela execução da melhor solução baseada nas métricas recebidas dos Agentes *Appraisers*. Após o AutoCore detectar um possível ataque DoS, as seguintes ações são tomadas pelos agentes: encaminhar os pacotes forjados para uma falsa aplicação, com o objetivo de alterar o fluxo do ataque liberando a aplicação real para uso e informar o CSO sobre a ocorrência do ataque, apresentando-lhe a origem do emissor dos pacotes forjados, facilitando a criação de regras concisas para configuração do *firewall*. Após consultarem o DF, os Agentes *Actions* enviam uma mensagem para os Agentes *Views*, que interagem com o CoreEditor informado a situação atual, o que foi realizado ou não e as possíveis outras correções ao CSO para, caso seja necessário, intervir no sistema.

Como o AutoCore é composto pelo CoreEditor na camada de visualização, pode-se então utilizá-lo para monitoramento e apoio à Gestão de Segurança e Riscos, mantendo assim, os serviços oferecidos pelas organizações disponíveis. Na Tabela 1, são apresentadas pelo módulo de relatórios do CoreEditor alguns dos resultados obtidos em forma de consultas baseadas em questões de competência que a CoreSec deve ser capaz de responder e criadas de acordo com as informação extraídas de acordo com o desejo dos CSO:

Tabela 1 - **Resultado dos relatórios emitidos pelo CoreEditor**

Quais ataques são realizados pelos agentes mal intencionados? Sua Probabilidade de Ocorrência? A Ferramenta Utilizada pelo agente mal intencionado? E os incidentes de segurança causados por estes ataques?				
Consulta 1: *SELECT DISTINCT * WHERE { ?Attacker :Performs_Attack ?Attack . { ?Attack :HasOccurrenceLikelihood ?OccurrenceLikelihood} . { ?Attacker :Uses_Tool ?Tool } . { ?Attack :Causes_SecurityIncident ?SecurityIncident} . {?SecurityIncident :HasToleranceLevel ?ToleranceLevel}}*				
Attacker	***Attack***	***Occurrence Likelihood***	***Tool***	***ToleranceLevel***
Terrorista	Eng. Social	Média	Persuasão	Baixo

SecurityIncident: Roubo de Informações sigilosas com intuito de obter aceso a recursos computacionais ou ganhos financeiros ilícitos.			
Quais as consequencias de um determinado ataque? Seu nível de tolerância? E o impacto nos negócios?			
Consulta 2: *SELECT DISTINCT * WHERE {?SecurityIncident :HasConsequence ?Consequence . { ?SecurityIncident :HasToleranceLevel ?ToleranceLeve} . { ?SecurityIncident :HasImpact ?BusinessImpact} . { ?SecurityIncident :HasCouterMeasure ?Measure}}*			
SecurityIncident	***Consequence***	***ToleranceLeve***	***BusinessImpact***
Roubo de Informações	Perda de Credibilidade	Baixo	Alto
Measure: Treinamento de Usuários			

FONTE: Desenvolvimento nosso

As consultas que respondem às questões de competências foram definidas de acordo com o que os especialistas necessitam responder para mitigar riscos reais em suas corporações. De acordo com o resultado das consultas, os responsáveis pela segurança tomarão decisões estratégicas para mitigar os riscos e manter os serviços oferecidos pelas organizações disponíveis sem perder competitividade no mercado.

6 CONCLUSÕES E TRABALHOS FUTUROS

A partir dos experimentos realizados, conclui-se que o AutoCore atinge os objetivos elencados na introdução deste trabalho, constituindo-se desta forma em uma ferramenta autonômica que poderá ser utilizada para o tratamento e utilização da informação a respeito de riscos e segurança da informação, possibilitando aos responsáveis pela Gestão de Riscos e Gestão de Segurança da Informação tomar decisões estratégicas de alinhamento de TIC e segurança aos processos de negócios das organizações fornecendo uma visão de alto nível entre os envolvidos (VERAS, 2010, p.92).

Dentre as limitações encontradas na pesquisa, é pertinente mencionar como dignos de nota, os seguintes aspectos: A ontologia utilizada no processo carece de melhorias no seu domínio de conhecimento, a resposta do AutoCore a eventuais ameaças à segurança da informação nos sistemas é dependente da expressividade e qualidade da ontologia utilizada como base de conhecimento.

O processo de atualização das informações contidas na ontologia abre espaço para a atuação de cientistas envolvidos com a respectiva área, isto é pertinente, pois permite um aumento das sinergias entre os cientistas da computação e os da Ciência da Informação; A interface do sistema pode ser melhorada, aspectos relacionados a usabilidade e a Arquitetura da Informação precisam ser revistos de maneira a possibilitar uma melhor interação do sistema com os seus usuários.

Como possibilidade de trabalhos futuros indica-se o desenvolvimento de trabalhos na detecção de outros tipos de ataques como o DDoS e variantes do DoS, SQL *Injection*, *Buffer Overflow* entre outros, ampliando o número de estudos de caso realizados, também poderá ser realizada a junção do AutoCore a *firewalls* e a utilização de Raciocínio Baseado em Casos – RBC em conjunto com ontologias. A partir da introdução dos aprimoramentos mencionados, existe a possibilidade de disponibilizar o AutoCore como um produto de mercado.

REFERÊNCIAS

ALMEIDA, M. J. S. C. et al. An Ontology about Information Security Management for Autonomic Computing Environments. In: *Proceedings of the 2nd Latin American Autonomic Computing Symposium*, 2007, Petrópolis - RJ.

ALMEIDA, M. B.; BAX, M. P. Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da Informação*, v. 32, n. 3, p. 7-20, set./dez. 2003.

FERNÁNDEZ, M. A. et al. Methontology: From ontological art towards ontological engineering. In: *Proceedings of the AAAI Spring Symposium Series*, 1997, p. 33-40.

FREITAS, F. et al. Survey of Current Terminologies and Ontologies in Biology and Medicine. *Revista eletrônica de comunicação, informação e inovação em saúde*. v. 3, p. 1-13. 2009.

FREITAS, F. Ontologias e a web semântica. In: VIEIRA, R; OSÓRIO, F. (Org.). *Anais do XXIII Congresso da Sociedade Brasileira de Computação*, 2003, Campinas: SBC. v. 8, p. 1-52, 2003.

GRUBER, T. R. Toward Principles for the Design of Ontologies used for Knowledgement Sharing. *International Journal of Human-Computer Studies*, v. 43, n. 5-6, p. 907-928, 1995.

HORN, P. 2001. *Autonomic Computing: IBM´s Perspective on the State of Information Technology*. Disponível em: <http://www1.ibm.com/industries/government/doc/contet/resource/thought /2786061 9.html>. Acesso em: 6 de fev. 2011.

KEPHART, J. O.; Chess, D. M. The Vision of Autonomic Computing. *IEEE Computer Society*, p.41-50, Jan. 2003.

MARTIMIANO, L. A. F. *Sobre a estruturação de informação de segurança computacional: o uso de ontologia*. 2006. 163 p. Tese (Doutorado em Ciências de Computação e Matemática Computacional) – ICMC - USP, São Carlos.

PARASHAR, M.; HARIRI S. *Autonomic Computing: Concepts, Infrastructure, and Applications*. CRC Press, USA, 2007.

SALES, R. de; CAFÉ, L. Diferenças entre tesauros e ontologias. *Perspectivas em Ciência da Informação*, v. 14, n. 1, p. 99-116, jan./abr. 2009. Disponível em: <http://portaldeperiodicos.eci.ufmg.br/index.php/pci/article/viewFile/646/54 1>. Acesso em: 8 de set. 2011.

STOJANOVIC, L. et al. The Role of Ontologies in Autonomic Computing Systems. *IBM Systems Journal*, v. 43, n. 3. p. 598, 2004.

USCHOLD, M.; GRÜNINGER, M. Ontologies: Principles, Methods and Applications. *Knowledge Engineering Review*, v. 11, n. 2. Jun.1996, 16 p.

VERAS, W. C. AutoCore: *Um Sistema Multiagente Autonômico Baseado em Ontologias para Segurança em Ambientes Computacionais*. 2010. 104f. Dissertação (Mestrado). Centro de Informática, Universidade Federal de Pernambuco, Recife, 2010.

WEISER, M. The computer for the 21st century. *Scientific American*, v. 265, n. 3, p. 66–75, 1991.