Universidade Federal de Pernambuco
Centro de Informática
Graduação em Ciência da Computação

Um Sistema de Apoio à Recuperação de Informação na Web voltado à Segurança de Redes e Sistemas

Thiago Gomes Rodrigues

Recife, Dezembro de 2009.

Universidade Federal de Pernambuco
Centro de Informática
Graduação em Ciência da Computação

Um Sistema de Apoio à Recuperação de Informação na Web voltado à Segurança de Redes e Sistemas

Thiago Gomes Rodrigues

Monografia apresentada ao Centro de Informática da Universidade Federal de Pernambuco, para obtenção do Grau de Bacharel em Ciência da Computação.

Orientador: Djamel Fawzi Hadj Sadok

Co-orientador: Eduardo Luzeiro Feitosa

Recife, Dezembro de 2009.

Agradecimentos

Primeiramente a Deus que por ter me agraciado com mais esta conquista e por ter me dado a oportunidade de passar momentos bons e ruins sempre me iluminando para conseguir aprender com cada situação vivida e poder dizer que nada como um dia após o outro.

Aos meus pais que sempre primaram pela minha educação, sempre tentando me entender e me direcionar para o caminho correto.

Ao orientador o professor Djamel Sadok e ao co-orientador Eduardo Feitosa por terem me orientado, direcionado e liberado a infra-estrutura do GRPT para que eu construísse este trabalho de graduação, sem a ajuda deles eu não teria conseguido.

A todas as pessoas que convivem comigo minha irmã Tatianne, minha namorada Janaise, amigos de trabalho Alysson (Alyss), Bruno (Pigmeu), Thiago (Cheroso), Rodrigo (Digão), Eduardo (Duda), Ademir (do Janga), Cirdes (Cidão), Felipe (Urubuzinho), Fernando (Furão), Josias, Leonardo (Léo), Joseane (Josy), Arthur, Thiago (Mouse) e de curso Armando (Potter), Ícaro (Segurança), Allan (Galego do Caldinho), Maria Carolina (Carol), Inocêncio (Inó), Guilherme (Guila), Thiago (tavl) pela paciência que tiveram comigo ao longo do curso sem vocês esta caminhada seria mais difícil.

A todos os outros não mencionados e que torceram pelo meu sucesso, obrigado.

Resumo

A área de segurança em redes de computadores e sistemas apresenta-se como uma das maiores preocupações das empresas, atualmente. Com o aumento do número de usuários de computadores ocorreu também o crescimento no número de incidentes de segurança.

A falta de segurança tem causado enormes prejuízos em todos os países . Como uma das soluções desse problema, a exposição (divulgação) de vulnerabilidades permite que administradores de redes e sistemas obtenham informações relevantes e possam minimizar o impacto que uma exploração pode acarretar a uma determinada entidade (empresa, universidades, entre outras).

Apesar da importância das informações divulgadas, normalmente, elas encontram-se espalhadas em diferentes sítios web, o que dificulta o trabalho das equipes de administração de redes e sistemas, tornando lento o processo de busca das informações necessárias para solucionar os problemas. Além disso, a simples divulgação da informação não garante sua relevância para a solução dos problemas. Baseado neste cenário, este trabalho de graduação se propõe a criar um sistema de apoio à recuperação de informação de segurança de redes e sistemas.

Sumário

Índice de Figuras

Índice de Tabelas

Lista de Abreviaturas

CVE	Common Vulnerabilities and Exposures
SA	Security Alerts
SB	Security Bulletins
TA	Technical Cyber Security Alerts

1. Introdução

Nos últimos dez anos, administradores de rede, gerentes de TI, especialistas de segurança e até mesmo usuários finais têm notado o aumento do tráfego Internet não desejado, não solicitado e freqüentemente ilegítimo. Grande parte deste problema está relacionada diretamente com violações de segurança por vulnerabilidades em software, sistemas e serviços, *spam* e ataques de negação de serviço.

As perdas financeiras ao redor do mundo, não somente no Brasil, confirmam que este tipo de tráfego aumenta ano após ano e apresenta potencial para tornar esses problemas globais. Em 2006, vulnerabilidades foram responsáveis por perdas de aproximadamente US$ 245 milhões somente entre os provedores de Internet dos USA [**1**]. Em 2007, o CSI (*Computer Security Institute*) entrevistou 194 empresas americanas e estimou perdas financeiras superiores a US$ 66 milhões [**2**]. No Brasil, o CERT.br (Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil) [**3**] contabilizou o número de incidentes relacionados às tentativas de fraude em 45.298 em 2007 enquanto que, no mesmo período, o CAIS (Centro de Atendimento a Incidentes de Segurança) da RNP (Rede Nacional de Pesquisa) [**4**] registrou cerca de 4000 tentativas de fraudes através de *spam* e *phishing*[1].

Uma vez que garantir a não existência de vulnerabilidades em software, sistemas e serviços é praticamente impossível e que a quantidade de vulnerabilidades conhecidas cresce todos os dias, a melhor solução é manter todo o pessoal neste tipo de atividade atualizado sobre estas questões. Neste contexto, bases de informação e sítios Web sobre vulnerabilidades, anomalias e informações de segurança apresentam-se como a mais comum e prática forma para divulgar dados sobre essas questões e vêm sendo empregadas para construir ferramentas de detecção de intrusão e buscadores de vulnerabilidades mais precisas.

A relevância deste tipo de soluções é facilmente comprovada pela existência de dezenas de bases de dados e sítios Web, tanto de acesso público quanto privado, sobre vulnerabilidades, anomalias e ataques tais como VulDa [**5**], Cisco Security Center [**6**], National Vulnerability Database (NVD) [**7**], Secunia Advisories [**8**] e Open Source Vulnerability Database (OSVDB) [**9**]. Tais bases divulgam periodicamente boletins de segurança, alertas de vulnerabilidades, relatórios técnicos e até estatísticas envolvendo vulnerabilidades, *spams*, ataques, vírus, entre outras.

[1] *Phishing* é um tipo de fraude eletrônica caracterizada pela tentativa de obter informações pessoais privilegiadas através de sites falsos ou mensagens eletrônicas forjadas.

Contudo, ainda existem algumas questões ou limitações. Primeiro, o processo de aceitação pode demorar muito tempo, uma vez que qualquer relatório de vulnerabilidade deve passar por uma série de etapas para comprovar sua veracidade. Este processo de aceitação tem influencia na operação de atualização, um processo tipicamente manual e dependente do ser humano. Embora existam padrões para descrever vulnerabilidade, a interoperabilidade de informações entre diferentes bases de dados e sítios Web é quase inexistente (tipicamente visualizada por simples *links* Web). Como resultado, existe informações duplicadas, repetidas e divergentes.

Diante do exposto, este trabalho de graduação propõe um sistema de apoio à recuperação de informação na Web (do inglês *Web-based Information Retrieval Support System* - WIRSS), chamada **ARAPONGA** (uma citação aos antigos detetives), capaz de integrar as mais relevantes classes de informação sobre vulnerabilidade e anomalias da Internet e fornecendo uma única e direta fonte a este tipo de informação. Mais especificamente, ARAPONGA fornece características para lidar com questões de interoperabilidade e o uso integrado de recuperação de informação e ferramentas de tomada de decisão.

1.1 Objetivo

Este trabalho propõe uma solução que concentre o máximo de informações divulgadas e disponíveis na Internet nas áreas de vulnerabilidades, anomalias e estatísticas do tráfego Internet.

A idéia central é ter um sistema de apoio a recuperação de informação na Web, especificamente para informações de segurança, voltada ao auxilio das atividades de gerenciamento e administração da segurança em redes de computadores e sistemas.

Especificamente pretende-se:

- Estudar e definir uma ferramenta de coleta (*crawler*) para buscar as informações disponíveis na Internet;
- Estudar e definir uma ferramenta para indexação do conteúdo coletado;
- Projetar e desenvolver um mecanismo capaz de melhorar a eficiência da indexação e, conseqüentemente, fornecendo opções de consulta mais detalhadas e focadas nos aspectos de segurança.

1.2 Organização do trabalho

O restante deste trabalho está organizado da seguinte forma.

O segundo capítulo apresenta alguns exemplos de sítios Web e bases de dados relacionados a vulnerabilidades e estatísticas do tráfego Internet, bem como de ferramentas para busca de vulnerabilidades em redes e sistemas. O terceiro capítulo

descreve os conceitos básicos da área de recuperação da informação, incluindo classificação e uma breve discussão sobre as tendências nessa área.

O quarto capítulo apresenta o projeto e a implementação da solução proposta, descrevendo seus requisitos, características e os passos de seu desenvolvimento. O capítulo cinco traz algumas avaliações sobre a implementação, em termos de desempenho e completude das consultas, exibindo os resultados encontrados.

Por fim, o sexto capítulo apresenta as conclusões do trabalho, incluindo as dificuldades encontradas na elaboração e os possíveis trabalhos futuros.

2. Conceitos Básicos

Uma vez que manter-se informado é a solução mais eficiente para lidar com as inúmeras vulnerabilidades e tentativas de ataques, existe, atualmente, uma grande necessidade de armazenar e compartilhar informação sobre tais problemas e suas possíveis soluções. Visando servir de "ponto de encontro" para administradores de rede, gerentes de TI e até mesmo usuários interessados, padrões, sítios web (incluindo bases de dados) e ferramentas de busca têm sido desenvolvidos e disponibilizados nos últimos anos.

Tipicamente, essas soluções são mantidas por organizações privadas, companhias e autoridades nacionais que disponibilizam bases públicas ou privadas sobre informações de vulnerabilidades e possíveis soluções. A idéia por trás deste tipo de solução é simples: documentar e registrar o problema e suas soluções, e após confirmação de sua corretude, divulgá-las. Esta necessidade de exibição destas informações fez com que surgissem padrões de divulgação e novas formas de obtenção do conteúdo. Estes padrões permitiram que as informações fossem trocadas facilmente de forma precisa e permitindo a interoperabilidade entre os vários sítios.

Este capítulo descreve alguns dos principais sítios web e ferramentas de busca existentes, visando aumentar o entendimento sobre a relevância deste trabalho.

2.1 Sites de Vulnerabilidades

Apesar de haverem vários sítios que divulgam informações sobre vulnerabilidades, este conteúdo exposto segue um padrão de exibição do conteúdo. Pode-se falar que um padrão é formado por um conjunto de características que descrevem algo e assim são as páginas que expõem este tipo de conteúdo.

2.1.1 OSVDB

O OSVDB (*Open Source Vulnerability Database*) [**9**] é uma base independente, de acesso gratuito, criada em agosto de 2002, cujo intuito é prover a comunidade de segurança, informações precisas, atualizadas, detalhadas e imparciais sobre vulnerabilidades.

Atualmente, o OSVDB mantém mais de 58.000 relatos de vulnerabilidades, além de efetuar atualizações constantes e lançamentos diários de novos relatos. Para identificar as vulnerabilidades, utiliza um padrão proprietário chamado OSVDB ID, que consiste apenas de números, incrementados um a um à medida que novas vulnerabilidades são adicionadas à base. O repositório do OSVDB pode ser acessado diretamente via web (http://www.osvdb.org), utilizando-se diferentes termos de consulta como, por exemplo, conteúdo da página, título, identificador OSVDB (ID), criador do

produto, entre outros. Além disso, o repositório pode ser copiado para consulta local em quatro diferentes formatos: XML, CSV, MySQL e SQlite.

Contudo, embora existam várias pessoas da comunidade de segurança colaborando na atualização e adição de novos relatos, o OSVDB apresenta páginas (vulnerabilidades) que não estão completas. O trabalho de Borba [**10**] detalha este problema e apresenta uma solução.

2.1.2 Secunia Advisores

Com o objetivo de publicar informações sobre vulnerabilidades, a Secunia [**8**] desenvolveu uma base de informações, chamada Secunia Advisories, focada em coletar, avaliar, verificar e analisar informações sobre vulnerabilidades e apresentar possíveis soluções. Criada em 2002, a Secunia Advisories registra mais de 35.000 relatos de vulnerabilidades e recomendações.

Essas informações são adicionadas e/ou modificadas periodicamente e grande parte do conteúdo é aberto para qualquer usuário, atraindo mais de cinco milhões de novos visitantes por ano. O acesso às informações é feito apenas pelo sítio da empresa (http://secunia.com/advisories/search/). A busca pode ser feita utilizando informações chaves como *headline*, *Software/OS*, *Body Text* e *CVE Reference*, e os resultados podem ser filtrados por impacto, nível de severidade e localização.

Com os resultados obtidos e divulgados na Secunia Advisores, a Secunia desenvolve sistemas que procuram vulnerabilidades em programas de computadores e constrói soluções que resolvem as vulnerabilidades encontradas por estes programas.

2.1.3 US-CERT

O *United States Computer Emergency Readiness Team* (US-CERT) é uma organização governamental dos Estados Unidos que publica periodicamente informações sobre vulnerabilidades, *exploits* e práticas de segurança. As publicações do US-CERT são divididas em três categorias: alertas técnicos (*Technical Alerts*) [**11**], boletins (*Bulletin*s) [**12**] e alertas não técnicos (*Alert*) [**13**]. O conteúdo de todas as páginas é gratuito e pode ser acessado via web.

O US-CERT mantém uma base de vulnerabilidades [**14**], criada em 2000, que contém mais de 2.500 relatos de vulnerabilidades. A base do US-CERT é pública e qualquer pessoa pode relatar uma vulnerabilidade. Contudo, somente após a verificação da veracidade das informações é que ela será publicada.

2.1.4 NVD

O *National Vulnerability Database* (NDV) [**7**] é um repositório de informação sobre vulnerabilidade do governo americano mantido pelo *National Institute of Standards and*

Technologies (NIST) que contém mais de 39.000 publicações sobre vulnerabilidades, com média diária de 20 novas inserções.

O NVD é baseado e sincronizado com o CVE e utiliza o mesmo padrão de nomes adotados pelo CVE. Além disso, fornece sumários para todas as vulnerabilidades contidas no CVE, ligações para outros sistemas, correções (*patches*) e outras fontes de informações relacionadas a vulnerabilidades. O NVD também provê um escore da periculosidade de cada vulnerabilidade chamado de *Common Vulnerability Scoring System* (CVSS). Os dados presentes em seus registros provêem mecanismos de análise quantitativa de vulnerabilidades com baixo nível de risco [**10**]**Erro! Fonte de referência não encontrada.**.

2.1.5 ATLAS

O ATLAS (*Active Threat Level Analysis Network System*) [**15**] é um sistema, desenvolvido pela Arbor Networks, que disponibiliza uma grande variedade de informações de ameaças a sistemas de computação. Tais informações contêm: resumos de ameaças, *ranking* de ataques da Internet, índice de riscos de vulnerabilidades, índice de ameaças e um mapa global de ameaças, atualizado em tempo real. Estas informações são atualizadas a cada 24 horas com completude e sempre estão ligadas a uma referência CVE. A Figura 2.1 ilustra os pontos na Internet onde os dados divulgados pelo ATLAS são capturados.

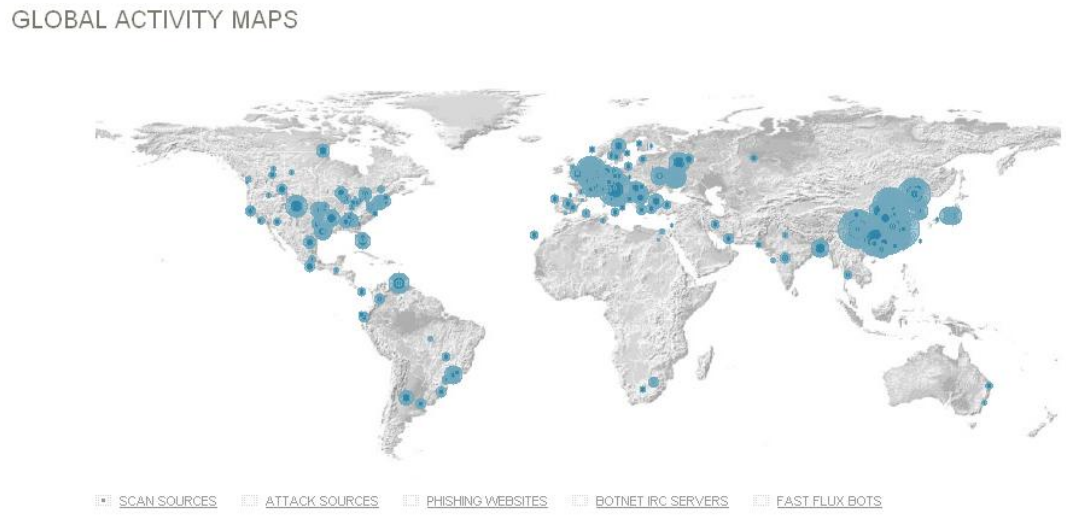

Figura 2.1: Mapa de atividades global fornecido pelo ATLAS.

Contudo, parte do conteúdo do ATLAS é restrito, somente pode ter acesso os usuários cadastrados. Para confecção deste trabalho, uma conta de acesso foi concedida gentilmente e gratuitamente.

2.1.6 Discussão

Visando melhorar a compreensão sobre as soluções apresentadas, a Tabela 2.1 exibe um comparativo entre elas. Para tanto, algumas características interessantes e comuns às soluções apresentadas neste trabalho serão tomadas como base:

- **Quantidade de informações registradas**: permite mensurar a quantidade de URLs que poderão ser visitadas por um sistema de busca de informações e também ajuda a mensurar o espaço necessário para indexar o conteúdo.
- **Tipo de acesso**: indica se o acesso as informações é simples (todo conteúdo disponível na página web) e direto (sem necessidade de autenticação) ou se é realizado por diferentes modos (via requisição HTTP ou Banco de Dados). Para um sistema de busca esta métrica é interessante porque indica a necessidade da criação ou utilização de mecanismo de autenticação para acesso ao conteúdo.
- **Atualização**: indica a periodicidade de atualização da base de informações. Esta métrica é importante porque permite mensurar o intervalo de tempo no qual um sistema de busca deve visitar as páginas de um referido domínio.
- **Completude**: indica se as informações registradas estão completas ou não. Esta métrica pode servir de indicador do grau de confiabilidade e usabilidade das informações contidas nessa base.
- **Uso de padrões**: permite identificar se os conteúdos estão seguindo algum tipo de padrão para divulgação das informações. Esta métrica é interessante permitindo que sites com o mesmo "perfil" sejam tratados de forma similar.

Tabela 2.1: Comparação entre os sítios Web avaliados.

	OSVDB	Secunia	US-CERT	NVD	Atlas
Número de publicações	> 58000	> 35000	> 2500	>45000	N/I
Acesso gratuito	Total (Web/BD)	Parcial (WEB)	Total (Web)	Total (Web)	Parcial (Web)
Taxa de atualização	Diária	N/I	N/I	Diária	Diária
Completude	Parcial	Total	Total	Total	Total
Uso de padrões	Próprio (osvdb ID)	Próprio	Próprio	CVE	Próprio

Em uma avaliação rápida e preliminar dos quatro sítios focados em vulnerabilidades (OSVDB, Secunia, US-CERT e NVD) pode-se afirmar que o OSVDB é o melhor representante entre os sítios sobre vulnerabilidades uma vez que disponibiliza suas informações de forma gratuita, em diferentes formatos, com atualizações diárias. Contudo, apesar do OSVDB realmente conter o maior o número de publicações (quantidade de vulnerabilidades) registradas, o aspecto completude das

informações ainda é um fator limitante. As análises feitas por Borba [**10**] mostram que, em 05 de Junho de 2009, a base do OSVDB continha 54.004 registros dos quais apenas 12.407 estavam completos (restando 41.597 registros incompletos), o que sem dúvida é considerada uma taxa muito alta de informações incompletas.

Embora apresentem um menor número de publicações, as bases da Secunia, US-CERT e NVD possuem seus atrativos. Tanto US-CERT quanto NVD permitem acesso gratuito as suas bases, cuja completude é visível em qualquer consulta. Contudo, o Secunia e US-CERT não indicam claramente qual o período de atualização de suas bases. Além disso, o Secunia é o único entre os quatro cujo acesso é limitado, somente para usuários cadastrados.

Entre os sítios descritos, o único voltado para divulgação de resultados e estatísticas de anomalias ocorridas na Internet é o ATLAS. Assim como o Secunia, grande parte de suas informações só podem ser acessadas por usuários cadastrados. Em compensação suas informações são atualizadas a cada 24 horas. A completude das informações disponibilizadas é um dos pontos fortes do ATLAS.

2.2 Buscadores de Vulnerabilidades

Os buscadores de vulnerabilidades, também conhecidos como *Vulnerability Scanners*, são programas de computadores que executam varreduras nas máquinas a fim de encontrar falhas em aplicativos e serviços. Basicamente, estas ferramentas dividem–se em três categorias: *Port Scanner*, *Network Scanner* e *Web Application Security Scanner*.

2.2.1 Port Scanner

Port scanners são software que vasculham (“varrem”) a rede ou um computador a procura de portas de comunicação abertas com a finalidade de precaver-se contra ataques ou tentativas de intrusão (por parte administradores de redes) ou como meios para invadir redes e sistemas e/ou provocar ataques (por parte dos hackers).

Dentre os diversos exemplos de *Port Scanners* encontrados no mercado pode-se citar o Nmap [**16**] e o UnicornScan [**17**]. O Nmap é uma ferramenta gratuita, de código aberto, bastante utilizada por administradores de redes para o monitoramento de redes, serviços ou máquinas. Já o UnicornScan é uma ferramenta gratuita, de código aberto, que mapeia a pilha TCP/IP, fornecendo uma interface de administração que mede as respostas ou as entradas dadas pelos dispositivos de rede ativos.

2.2.2 Network Scanner

Também conhecido como *Network Enumerator*, os *network scanners* tem a função de encontrar nomes de usuários, informações sobre arquivos ou serviços de computadores presentes na rede analisada.

Como exemplo de *network scanners*, pode-se citar o Nessus [**18**], o LanGuard [**19**] e o Retina [**20**]. O Nessus é uma ferramenta "fechada", construída pela Tenable, que pode ser usado dentro de uma zona desmilitarizada (DMZ), dentro de um empreendimento ou em redes separadas fisicamente. O LanGuard também é uma ferramenta "fechada" capaz de detectar, avaliar e corrigir qualquer potencial risco em uma rede. O Retina é uma ferramenta usada para conhecer e identificar vulnerabilidades recém-encontradas ("*zero-day"*), permitindo melhores práticas de segurança, fiscalização das políticas e auditoria. Também é uma ferramenta de código fechado.

2.2.3 Web Application Security Scanner

Web application security scanners são programas com interface Web (*front-end* Web) que buscam no computador do usuário aplicativos presentes em listas de vulnerabilidades.

Dentre os exemplos mais usados pode-se citar o Acunetix [**21**] e N-Stalker [**22**]. O Acunetix é uma ferramenta de código fechado que verifica automaticamente as aplicações web para evitar ataques de "*SQL injection*", "*XSS*" e outras vulnerabilidades web. O N-Stalker é uma ferramenta de código fechado que visa proteger empresas e indivíduos de ameaças digitais. Sua base encontra-se sempre atualizada com uma lista de assinaturas de ataques que somam mais de 39.000 publicações.

3. Recuperação de Informação na Web

Utilizando o contexto apresentado no capítulo anterior (informações sobre vulnerabilidades e anomalias Internet), percebe-se uma real necessidade por novos sistemas que explorem os conteúdos disponíveis na Internet na busca por informações úteis e que permitam a criação de bases de conhecimento. É neste contexto que a área de recuperação de informações se apresenta como solução para alguns dos problemas já descritos.

Recuperação de Informação (do inglês *Information Retrieval*) é uma área de pesquisa dedicada às tecnologias para manipulação e recuperação de grandes coleções de informação em diferentes formatos de apresentação. Tipicamente, RI investiga formas de representação, armazenamento, organização e acesso a itens de informação de modo a permitir ao usuário fácil acesso à informação na qual ele está interessado através de consultas. Entretanto, quando a necessidade por informações é aplicada na realização de atividades como aprendizado, tomada de decisão, e outras atividades mentais complexas que ocorrem ao longo do tempo, a recuperação é necessária, mas não suficiente [**23**]. A solução é mudar a pesquisa de informações dos motores de busca que fornecem itens discretos como respostas as consultas para ferramentas e serviços que suportem pesquisas interativas e reflexivas ao longo do tempo e explorando o modo colaborativo.

Este capítulo faz uma caracterização da área de recuperação de informação, apresentando os conceitos envolvidos, discutindo os desafios presentes e, por fim, descrevendo alguns trabalhos cujos resultados são encorajadores para a pesquisa nessa área.

3.1 Web Crawlers

Os Web Crawlers, também conhecido como *robots, ant, spider, wanderers, walkers, knownbots* ou *bot*, são programas responsáveis por percorrer a Web e baixar (*download)* de páginas para serem usadas por sistema de busca [**24**].

Normalmente, um *web crawler* inicia o processo de navegação na Internet com um grupo inicial de URLs armazenado em uma estrutura de dados chamada *seeds*. À medida que acessa a URL, o *web crawler* faz o download da página pertencente a essa URL e analisa todas as URLs encontradas nessa página afim de selecionar e armazenar essas novas URLs na lista de páginas a visitar. Esse processo é repetido até satisfazer a condição de parada do *web crawler*.

Contudo, o funcionamento de um *web crawler* enfrenta três importantes problemas: o grande número de páginas, a velocidade com que estas páginas são

atualizadas e, com o advento da Web 2.0, a geração de páginas dinâmicas. Uma vez que a Web apresenta um grande volume de páginas, um *web crawler* pode apenas baixar uma pequena porção, o que faz com que seu funcionamento seja norteado pelo estabelecimento de prioridades relativas na seleção das páginas a serem baixadas. Já a rápida atualização de conteúdo aumenta a probabilidade do *web crawler* baixar conteúdo desatualizado. Por fim, com a geração dinâmica de páginas, o número de possíveis de páginas que podem ser baixadas aumenta consideravelmente, influenciando no processo de coleta. Além disso, páginas dinâmicas tipicamente não têm HTML como conteúdo e sim referências a uma estrutura dinâmica.

O comportamento de um *web crawler* é baseado em uma série de políticas de implementação que visam melhorar o seu rendimento. Estas políticas estão relacionadas ao comportamento do *web crawler* quando está em ação como, por exemplo, que *links* visitar primeiro, o que fazer quando encontrar uma página já baixada, se vai executar em paralelo ou se vai seguir as políticas criadas pelo robots.txt[2] de cada domínio [**25**]. O uso de políticas permite a classificação dos *web crawlers* em três tipos: *Retricting Followed Links*, *Path-ascending crawling* e *Focused Crawler*.

3.1.1 Restricting Followed Links

Restricting Followed Links é um tipo de *web crawler* que busca somente *links* nas páginas HTML. Basicamente, este tipo de *web crawler* tenta encontrar o máximo de referências possíveis usando estratégias como procurar apenas por URLs que terminam com .html, .htm, .asp, .aspx, .php ou com "/".

Existem muitos empecilhos neste tipo de abordagem, uma vez que uma escolha errada na estratégia de mapeamento dos *links* das páginas pode levar o *web crawler* a requisições infinitas de páginas. Um exemplo, são as URLs que têm em seu nome o símbolo "?", um claro indicativo de que o conteúdo é construído dinamicamente.

Este tipo de *web crawler* é bastante usado para verificar se os *links* das páginas continuam funcionando.

3.1.2 Path-Ascending Crawling

Path-Ascending Crawling é um tipo de *web crawler* que busca encontrar todos os recursos de um determinado sítio. Basicamente, utiliza um *link* inicial passado como referência e tenta extrair o máximo de páginas navegando pelos diretórios da URL. Supondo que a URL http://www.cin.ufpe.br/~tgr/arquivos/tg/index.html seja passada, o *web crawler* procurará arquivos no index.html, www.cin.ufpe.br/~tgr/arquivos/tg/, www.cin.ufpe.br/~tgr/arquivos/, www.cin.ufpe.br/~tgr/ e, por fim, www.cin.ufpe.br/.

[2] Robots.txt são arquivos criados em sítios Web para controlar as ações de dos robôs (*robot*) de busca, ditando seu comportamento no domínio.

Este tipo de *web crawler* pode ser usado quando se deseja transferir todo o conteúdo de um sítio.

3.1.3 Focused Crawling

Focused Crawling é um tipo de *web crawler* que busca páginas que tenham conteúdo inserido dentro de um tópico ou vários tópicos previamente determinados. Em seu funcionamento podem ser usadas abordagens que usam apenas os nomes dos *links* para decidir se vão baixar a página ou não, bem como abordagens que usam uma medida de similaridade entre o conteúdo do HTML das páginas baixadas com os conteúdos das páginas ainda não visitadas para decidir se irá baixar ou não a página.

3.2 Mecanismos de Buscas

Mecanismos de Busca, também chamados de *Search Engines*, são aplicações utilizadas para buscar grande quantidade de informações na Web. Tipicamente, buscas na Web são realizadas via mecanismos de busca acessíveis via Web browsers. Após a requisição inicial, o mecanismo busca as informações, utilizando técnicas particulares, e retorna aos usuários as referências a documentos que melhor satisfazem a consulta.

Esta seção apresenta os conceitos e técnicas de recuperação utilizadas por mecanismos de busca e de recuperação de informação na Web. Para tanto, tais conceitos e técnicas serão exemplificados.

3.2.1 Módulos componentes

Mecanismos de busca tradicionais são projetados de forma modular visando isolar atividades e funções específicas. De modo geral, o primeiro módulo a ser ativado em um mecanismo de busca é o módulo de coleta de páginas (*crawler*), responsável por navegar pela Web e montar um repositório com as páginas visitadas e selecionadas.

Em seguida, tipicamente após a finalização da coleta pelo *crawler*, o módulo de indexação analisa o conteúdo de cada página armazenada no repositório, cria um conjunto de palavras-chave (índice) que identifica o conteúdo da página e associa, em um banco de dados, a URL na qual cada palavra-chave ocorre. Os métodos de indexação variam de acordo com a utilidade e as técnicas aplicadas por cada mecanismo de busca.

Finalmente, os módulos de consulta e *ranking* (ordenação) recebem as requisições de usuários e as processam para retornar, de maneira ordenada, os documentos que melhor satisfazem essas requisições pelas consultas que foram processadas pelo módulo de consulta.

A subseção a seguir apresenta os módulos básicos de um mecanismo de busca, exceto o módulo de *crawler* que já foi apresentado e discutido anteriormente.

Módulo de Indexação

Índices são descritos como palavras cuja semântica representa o principal assunto do documento. Sendo assim, a indexação de informação realizada neste módulo corresponde à representação de informações de páginas Web em termos de índice.

Entre as técnicas de indexação mais utilizadas em mecanismos de busca na Web encontram-se:

a) *Inverted files* - um mecanismo de indexação orientado a palavras o qual armazena as diferentes palavras encontradas no texto e suas ocorrências;

b) *Suffix arrays* - tratam o conteúdo textual dos documentos como uma única cadeia de caracteres (*string*) e cada posição da palavra como um termo de índice;

c) *Signature files* - um mecanismo de indexação orientado a palavras manipuladas em tabelas de tipo *hash*. O texto é dividido em blocos de palavras e a cada bloco é aplicado uma função *hash* cujo resultado será o identificador desse bloco (*signature*).

Em [**26**] Kobayashi e Takeda apresentam algumas das principais características e funcionalidades de módulos de indexação utilizadas pelos mecanismos de busca. São elas:

- *Indexação manual ou humana*: especialistas no conteúdo a ser indexado organizam e compilam os diretórios e os índices da maneira que facilite as consultas. Por essa razão esse tipo de indexação é ainda considerado o mais preciso de todos os métodos;
- *Indexação inteligente ou baseada em agentes*: são compostas por "agentes" computacionais que selecionam páginas, as indexam, criam índices e armazenam as informações importantes para posterior recuperação da informação;
- *Indexação baseada em metadados, RDF e anotação*: a indexação é feita considerando exclusivamente metadados.

Módulo de Busca e Ordenação

O módulo de busca e ordenação está extremamente relacionado com o modo com que as páginas foram indexadas, uma vez que nem todos os tipos de busca podem ser usados em qualquer sistema. Uma consulta passada a um mecanismo de busca é conhecida como *query* e representa a necessidade de informação do usuário. Uma consulta pode ser Baseada em Palavras-Chave, Casamento de Padrão ou Estruturada.

Consulta baseada em Palavras-Chave

Consulta baseada em palavras-chave permite o ordenamento das respostas segundo a função de relevância adotada pelo mecanismo de busca. Pode ser construída baseada em palavras isoladas, baseada no contexto ou com junções *booleanas*. Seu objetivo é recuperar todos os documentos que contêm ao menos uma das palavras da consulta e em seguida, os documentos recuperados são ordenados e retornados ao usuário.

Alguns mecanismos de busca são capazes de realizar consultas de palavras dentro de algum contexto. Para este tipo de consulta, palavras que aparecem juntas são mais relevantes do que àquelas que aparecem separadas como, por exemplo, as palavras "redes" e "computadores" no contexto de documentos sobre o tema de Redes de Computadores.

As consultas com junções booleanas são aquelas que combinam as palavras com operadores booleanos OR, AND, BUT. Em geral, mecanismos de busca não usam o operador NOT, pois o resultado retornaria um número muito grande de documentos. Já o operador BUT pode ser usado para restringir este universo.

Consulta baseada no Casamento de Padrão

Mecanismos de busca que permitem esse tipo de consulta realizam o "casamento" com *strings* ao invés de apenas palavras isoladas. Estas consultas podem ter um padrão simples (quando é apenas uma palavra, um prefixo, um sufixo, substring ou intervalo) ou um padrão complexo (que pode ser uma expressão regular). O objetivo deste tipo de consulta é encontrar documentos que contêm segmentos de texto que casam com o padrão da consulta e, para realizar tal tipo de busca, a lista de índices invertidos não é suficiente para uma recuperação eficiente.

Consulta baseada na Estrutura

Este tipo de consulta permite ao usuário realizar buscas a campos específicos das páginas. Por exemplo, um usuário que deseja procurar por páginas que no título aparece "Vulnerabilidade" recebe do mecanismo de busca somente páginas que contêm a *string* "Vulnerabilidade" em seu título.

Módulo de Armazenamento

Os vários módulos de um mecanismo de busca utilizam repositórios para armazenar as páginas manipuladas. Segundo Arasu et al. [**27**], esses repositórios devem possuir as seguintes características:

- *Método duplo de acesso às informações armazenadas*: acesso randômico, para ser usado rapidamente pelo módulo de busca e acesso por fluxo, para ser usado por indexadores para processar e analisar as páginas em volume;

- *Manipulação de grande volume de atualizações*, pois esses repositórios devem ser capazes de adicionar, de atualizar, e de reorganizar facilmente informações enviadas por *crawlers*;
- *Controle de páginas obsoletas*, pois um repositório deve ser capaz de detectar e remover páginas que não são utilizadas;
- *Escalabilidade*, durante a distribuição de repositórios através de *clusters* de computadores e de unidades de armazenamento distintas.

3.3 Novas abordagens para Web

Uma vez que o ser humano sempre procura aprimorar o seu conhecimento e sabedoria, a busca (pesquisa) de informações em bases e repositórios com grande volume de dados vem se tornando cada vez mais complexa. O pressuposto básico de uma pesquisa é que o usuário sabe exatamente o que quer. Entretanto, o efeito desta hipótese torna-se cada vez pior quando o usuário tem de enfrentar o crescimento diário da Web. Apesar dos muitos avanços nesta área, o foco principal de trabalho ainda é centrado na busca e recuperação da informação.

Estudos recentes demonstram que existe uma tendência emergente: a mudança dos atuais sistemas centrados na busca e recuperação para os sistemas centrados no apoio aos usuários. Um desses estudos introduz o conceito dos sistemas de apoio a recuperação de informação (do inglês *Information Retrieval Support Systems* - IRSS). Yao [**28**] prevê que "tais sistemas são concebidos com o objetivo de fornecer os serviços públicos necessários, ferramentas e linguagens que permitam ao usuário executar diversas tarefas em busca de informações e conhecimentos úteis. Enquanto os sistemas de informação existentes de recuperação (IRS) focam na pesquisa e funcionalidades de navegação, um IRSS atua sobre as funcionalidades de apoio". No contexto da Internet esses sistemas são chamados de *Web Information Retrieval Support Systems* (WIRSS).

A mudança de recuperação centrada no apoio ao usuário causa o aparecimento dos mecanismos de suporte à busca (*Search Support Engines* - SSE). Enquanto que os motores de busca tradicionais são focados nas funcionalidades de pesquisa, os motores de suporte a busca são construídos com foco no suporte a diversos tipos de funcionalidades, principalmente as que guiam e ajudam os usuários na análise e utilização dos resultados da pesquisa.

Outra recente proposta, os sistemas de suporte a busca de informações (do inglês *Information Seeking Support System* - ISSS) enfatizam a necessidade de mudar o foco do estudo de busca de informação para apoio a busca. Marchionini e White [**29**] afirmam que a busca de informações para a aprendizagem, tomada de decisão, e outras atividades mentais complexas que ocorrem ao longo do tempo requerem ferramentas e serviços de apoio que ajude na gestão, análise e compartilhamento das informações obtidas.

4. Solução Proposta e Implementação

Atualmente, os problemas causados por atividades como mensagens eletrônicas não solicitadas (*spam*); atividades fraudulentas como *phishing* e *pharming*[3]; ataques de negação de serviço (do inglês *Distributed Denial of Service* - DDoS); proliferação de vírus e worms; *backscatter*[4], entre outros, fazem com que esse tipo de tráfego não desejado seja considerado uma pandemia, cujas conseqüências refletem-se no crescimento dos prejuízos financeiros dos usuários da Internet.

Parte desses prejuízos se deve a ineficiência das atuais soluções em identificar, reduzir e interromper esse tipo de tráfego. Tipicamente, a efetividade fornecida pelas soluções existentes só é percebida após a ocorrência de algum dano. Além disso, a alta taxa de alarmes falsos e a falta de cooperação com outras soluções ou mesmo com a infra-estrutura de rede são fatores considerados incentivadores do aumento do tráfego não desejado. Como mencionado em [**30**], as soluções usadas para detectar e reduzir os efeitos de ataques DDoS tais como filtragem, limitação de banda, IP *traceback* e esquemas de marcação de pacotes são difíceis de implementar porque necessitam de mudanças na infra-estrutura da Internet. Ao mesmo tempo, soluções tradicionais como firewall e VPN (*Virtual Private Network*) são ineficazes contra códigos maliciosos e *spam*.

Contudo, é possível afirmar que parte destes prejuízos pode ser evitada através da obtenção de informações atualizadas, especialmente no que diz respeito à propagação de tráfego malicioso. Visando resolver a questão de como e onde obter informações úteis, este trabalho propõe um sistema de busca e recuperação de informações na Web, denominado ARAPONGA. Esta solução realiza buscas automatizadas de conteúdo sobre vulnerabilidades e estatísticas de atividades maliciosas divulgados na Internet, armazenando-as em uma base única e, por fim, permitindo acesso a essas informações de forma direta tanto para usuários quanto outros sistemas.

Este capítulo descreve a arquitetura do ARAPONGA, bem como seu funcionamento e o processo de desenvolvimento. Primeiro, uma visão geral do protótipo é apresentada. Em seguida, cada um dos componentes (módulos) será explicado e, por fim, o processo de funcionamento e integração entre os módulos serão detalhados.

[3] *Pharming* refere-se ao ataque de envenenamento de cache DNS cujo objetivo é preparar terreno para atividades de *phishing*.
[4] *Backscatter* é o tráfego recebido de vítimas que estão respondendo a ataques de negação de serviço.

4.1 Arquitetura do ARAPONGA

ARAPONGA foi projeto com o objetivo de concentrar o máximo de informação sobre vulnerabilidades, ataques, *botnets*, *spam* e outras atividades maliciosas em uma base única chamada Base Refinada, facilitando a aquisição de tais informações e permitindo uma busca rápida, fácil e refinada (focada). Na prática, ARAPONGA representa um software capaz de fornecer consultas gerais ou diferenciadas (estruturadas), retornando um conteúdo focado e útil à área de segurança de redes e sistemas.

A idéia central é utilizar os conceitos do suporte a recuperação de informação na web para extrair a máxima quantidade de informações úteis sobre vulnerabilidades e atividades maliciosas, aumentando a precisão das buscas e diminuindo assim o tempo de procura por este tipo de informação.

Semelhante aos WIRSS, o ARAPONGA é composto por cinco módulos: coleta, indexação, adequação, busca e ordenação e interface. A Figura 4.1 ilustra a estrutura do ARAPONGA.

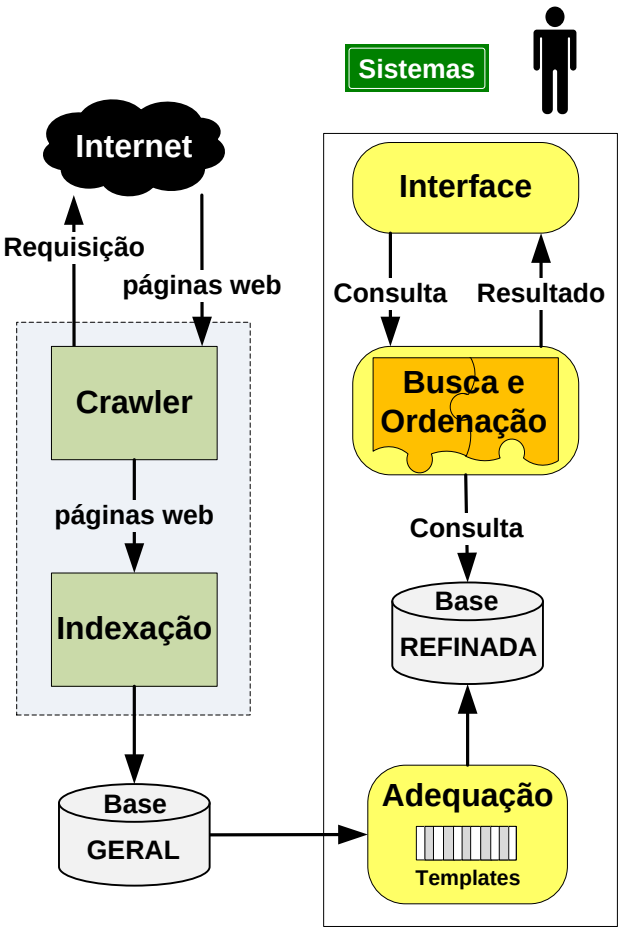

Figura 4.1: Arquitetura do ARAPONGA.

O **módulo de coleta**, também chamado de *crawler*, é responsável pela aquisição de páginas web. Baseado em uma lista contendo as URLs iniciais (focadas em sítios que

divulgam informações de vulnerabilidades e estatísticas sobre ataques e anomalias Internet), o módulo busca em cada página visitada referências para outras páginas. Para resolver os problemas relativos à quantidade e qualidade da informação coletada, o módulo *crawler* utiliza limitadores de profundidade para evitar grandes desvios do ponto de partida inicial (no caso as URLs), limitadores de amplitude para restringir o número de *links* por páginas que podem ser referenciadas, e filtros de URL, consultados todas as vezes que uma nova página está para ser coletada.

O **módulo de indexação** recebe as páginas coletadas pelo módulo *crawler*, cria identificadores de conteúdo do documento e os adiciona à base de dados de conteúdo indexado (base geral). A indexação é feita armazenando-se todo o conteúdo da página com o identificador principal "*content*" e outros identificadores como, por exemplo, a URL no campo "URL", a marca de tempo (*timestamp*) da página no campo "*tstamp*", entre outros.

O **módulo de adequação** é responsável por carregar todos os documentos contidos na base geral e executar um tratamento de conteúdo, visando melhorar a indexação (não indexando apenas pelo conteúdo das páginas). Para tanto, faz uso de modelos (*templates*) para determinar quais partes de uma página devem ser indexadas com *tags* diferentes, possibilitando, assim, buscas diferenciadas. Este módulo também executa a seleção de páginas que não serão indexadas porque não apresentam um conteúdo relevante na solução do problema. Como resultado, este módulo gera uma nova base de dados contendo somente informações uteis e relevantes, denominada base refinada.

O **módulo de interface** é responsável pela comunicação entre usuários e o sistema. Neste módulo são definidas as regras para as consultas e para as respostas. Todas as consultas são enviadas para o módulo de busca e ordenação, que retorna respostas ordenadas baseadas no ranqueamento de cada página.

O **módulo de busca e ordenação** é responsável por receber a consulta e retornar o objeto da consulta de forma ordenada. Este módulo é dividido em dois sub-módulos: tradutor de consultas e ranqueamento. O primeiro recebe consultas em linguagem natural oriundas do módulo de interface, transformando-as em consultas aceitas pelo sistema, buscando as informações na base refinada e repassando as páginas retornadas para o sub-módulo de ranqueamento. O sub-módulo de ranqueamento é responsável em quantificar a relevância dos documentos retornados em relação a consulta e retorná-los para o módulo de interface para exibição.

É importante ressaltar que, na figura 4.1, percebe-se os módulos de coleta e indexação e os módulos de adequação, busca e ordenação e interface estão divididos em grupos distintos. A idéia é representar que os módulos no primeiro grupo não foram implementados neste trabalho e sim utilizados (instalados e configurados), enquanto os outros módulos do segundo grupo foram realmente desenvolvidos integralmente.

4.2 Funcionamento

Para tornar mais claro o processo de funcionamento do ARAPONGA, uma descrição completa de todo processo é exemplificada a seguir.

Em primeiro lugar é preciso entender que os módulos de coleta, indexação e adequação funcionam em conjunto, um após a execução do outro, e de forma *off-line*, ou seja, o processo desde a coleta a preparação da base refinada é realizado isoladamente, acontecendo todos os dias as 03:00 horas. Desta forma, quando um usuário ou sistema efetua uma consulta ao ARAPONGA, a solicitação passa apenas pelos módulos de interface e busca e ordenação.

Tomando como exemplo uma consulta referente a informações sobre ataques, *botnet*, vulnerabilidades, *spam* e boletins envolvendo especificamente o protocolo TCP na porta 80. A consulta efetuada no módulo de interface é a seguinte:

tcp/80 –focus Bulletin,Alert,Spam,Vulnerability,Attack,Botnet

onde o número e tipos dos parâmetros são analisados. Caso estejam de acordo, a consulta é enviada para o módulo de busca e ordenação.

No módulo de busca e ordenação, a consulta passa pelo processo de validação, onde são retiradas as palavras que não tem significância para a consulta (*StopWords*). Então a consulta é enviada ao motor de busca para que efetue a pesquisa nas páginas na base refinada. A idéia é verificar quais páginas contém o valor descrito na consulta (no caso *tcp/80*) de acordo com o parâmetro especificado (*–focus Bulletin,Alert,Spam,Vulnerability,Attack,Botnet*). Páginas encontradas que se encaixam neste perfil (obedecem a essas regras) são ranqueadas e uma lista ordenada de modo decrescente é construída de acordo com o valor de ranqueamento. Por fim, o resultado da busca é enviado ao módulo de interface e então encaminhado ao solicitante.

4.3 Implementação

Esta seção descreve o processo de implementação da solução, focando especificamente os módulos e sua integração. Além disso, alguns aspectos importantes referentes às decisões de projeto e escolha dos dados também são elucidados.

4.3.1 Questões Preliminares

Antes iniciar a explicação do processo de implementação deste trabalho, faz-se necessário esclarecer dois pontos importantes e decisivos no projeto e desenvolvimento do ARAPONGA: a escolha do conteúdo e do *crawler*.

Para definir quais eram as páginas e sítios web mais adequados para a aquisição de informações sobre vulnerabilidades e estatísticas sobre tráfego e anomalias, foi necessário antes realizar a escolha do conteúdo a ser mantido pela Base Refinada, visto

que existem dezenas senão centenas de locais com este tipo de conteúdo na Internet. Após uma avaliação que considerou a relevância e completude das informações, o período de atualização e a facilidade de acesso, foram definidos os seguintes sítios web: Secunia, US-CERT (http://www.us-cert.gov) e US-CERT (http://www.kb.cert.org) para boletins e relatórios de vulnerabilidades; e ATLAS para estatísticas da Internet.

A escolha do *crawler* também foi bem avaliada, uma vez que descobriu-se que alguns domínios, inclusive um dos escolhidos (ATLAS), necessitam de autenticação para que certos conteúdos (informações extras sobre um determinado endereço IP envolvido em ataques DDoS, por exemplo) fossem detalhados. Foram avaliados três *crawlers*: WIRE [**31**], Heritrix [**32**] e Nutch [**33**] (tabela 4.1). Como resultado, o *crawler* Nutch foi escolhido por apresentar características favoráveis quanto a instalação e alteração de seu código, além da capacidade de autenticação.

Tabela 4.1: Comparativo entre os três *web crawlers* testados.

Características	WIRE	Heritrix	Nutch
Instalação (Dificuldade)	Média	Alta	Baixa
Módulo de Autenticação	Não	Sim	Sim
Linguagem de implementação	C/C++	JAVA	JAVA

4.3.2 Módulo de Coleta

O Nutch, projetado e criado pela Apache, é uma mecanismo (*engine*) de busca web que utiliza a biblioteca de busca Lucene [**34**] para armazenar e buscar o conteúdo web baixado. Difundido em escala global, o Nutch é desenvolvido em Java, apresenta simplicidade na modificação do seu código fonte aberto e é bem documentado. Dentre as várias características do Nutch, pode-se citar a capacidade de (i) localizar bilhões de páginas por mês; (ii) manter o índice destas páginas; (iii) pesquisar este índice mais de 1000 vezes por segundo; (iv) prover resultados de alta qualidade; (v) operar com o menor custo possível.

Apesar de todas as vantagens oferecidas, em sua configuração padrão, o Nutch respeita o que está publicado no arquivo robots.txt (presentes na raiz de cada domínio) e as META-TAGs[5] dos HTMLs das páginas, ocasionando deficiências na coleta de páginas.

Em relação ao arquivo robots.txt, tipicamente quando se deseja esconder o conteúdo de um robô, o arquivo tem a configuração apresentada na figura 4.2, onde "*" significa que qualquer agente tem o acesso bloqueado ("*disallow: /*") a todo o diretório.

[5] META-TAGs são palavras reservadas do HTML, "etiquetas", que entre outras coisas descrevem o conteúdo do sítio para os crawlers.

Desta forma, o agente fica impossibilitado de acessar qualquer página dentro do domínio.

```
User-agent: *
Disallow: /
```

Figura 4.2: Exemplo do conteúdo do arquivo robots.txt.

Já em relação às META-TAGs para buscas via *crawlers*, uma configuração típica tem o formato da figura 4.3 onde os valores *índex* e *nofollow* se referem à primeira página do sítio, permitindo a indexação da página inicial, mas não do restante do conteúdo.

```
<meta name="robots" content="index,nofollow">
<meta name="robots" content="noindex,nofollow">
```

Figura 4.3: Exemplo tópico de META-TAGs para evitar o acesso de *crawlers*.

A solução deste problema é a alteração dessas "políticas de bom comportamento" do *crawler*. Por definição, um *crawler* que não respeita essas políticas é conhecido como "*Malware Crawler*". Contudo, uma vez que as informações obtidas neste trabalho serão usadas para fins benignos, o código fonte do Nutch foi alterado para que estas políticas não fossem observadas, o que foi de suma importância para a obtenção de todo o conteúdo necessário para a implementação do sistema.

4.3.3 Módulo de Indexação

A ferramenta escolhida para indexar o conteúdo coletado foi o Lucene [**34**]. Também desenvolvido pela Apache, é uma biblioteca de busca de texto de alto rendimento, com código fonte aberto e recomendada para sistemas que precisam fazer buscas em textos completos.

Em termos de implementação, o Lucene também é escrito em JAVA e disponibiliza uma API que permite acesso a mecanismos de indexação e consulta de documentos.

4.3.4 Módulo de Adequação

O módulo de adequação foi implementado em Java (versão 1.6) utilizando a biblioteca *Jericho HTML parser* [**35**] para extração do HTML dos conteúdos das páginas, além da API do Lucene.

Em linhas gerais, este módulo é responsável por traduzir o conteúdo da base geral (pelos dois módulos anteriores) para a base refinada. Seu funcionamento pode ser dividido em três etapas:

- Uso da API do Lucene para aquisição de todas as páginas baixadas e indexadas pelo *crawler* e o encaminhamento para a segunda etapa;
- Comparação das palavras do título de cada página com um conjunto de identificadores pré-definidos (criados através do reconhecimento de padrões contidos nas páginas relevantes), visando à identificação de modelos (*templates*) que ajudam a referenciar a página. Uma vez que um *template* é encontrado, a página tem seu conteúdo extraído e identificado e, cada bloco de informação tem associado a si uma palavra-chave. Caso a página não tenha um *template* identificado, ela será apenas referenciada pelo seu conteúdo. Após este processamento a página é encaminhada para a terceira etapa. A Figura 4.4 exemplifica a função dos *templates*, onde os círculos representam os campos (*tags*) identificados nos *templates* e os quadrados representam os conteúdos a serem indexados com sua respectiva *tag*;

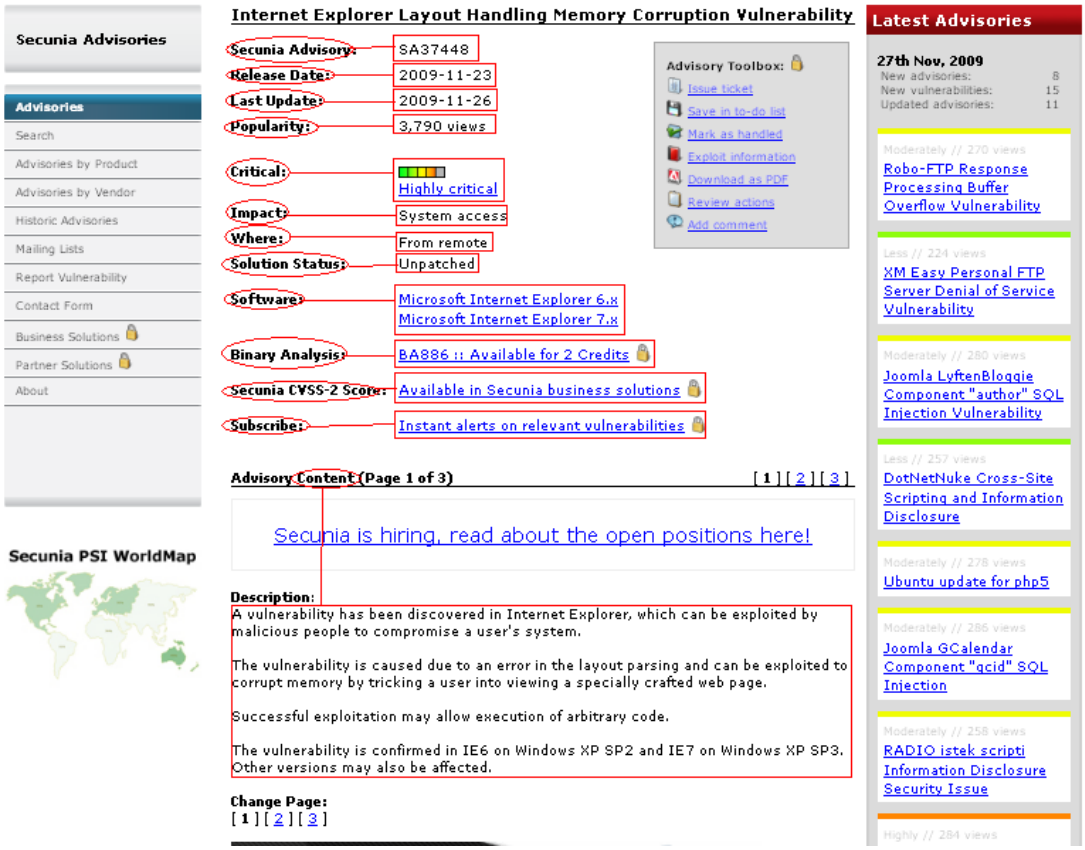

Figura 4.4: *Template* do domínio Secunia.

- Adição dos identificadores de *timestamp*, *title*, *URL*, além dos identificadores de controle interno do sistema de busca/indexação.

Após estas três etapas, a página está pronta para ser indexada.

4.3.5 Módulo de Busca e Ordenação

O módulo de busca e ordenação também foi implementado em Java utilizando a API do Lucene para buscar as páginas que são relevantes à referida consulta. Basicamente, qualquer consulta passa por um processo de eliminação de palavras (*StopWords*) e o resultado deste pré-processamento é enviado ao Lucene. Desta forma, inicia-se o processo de comparação da consulta com os documentos da base. Após efetuadas todas as comparações, os documentos são ranqueados e ordenados em ordem decrescente levando em conta o valor no qual o documento foi valorado.

4.3.6 Módulo de Interface

O módulo de interface fornece dois tipos de saída (ou interfaces). A primeira é visual (GUI) e indicada para consulta pelo operadores humanos do sistema (administradores de segurança e gerentes de TI, por exemplo). A Figura 4.5 ilustra a GUI de consulta.

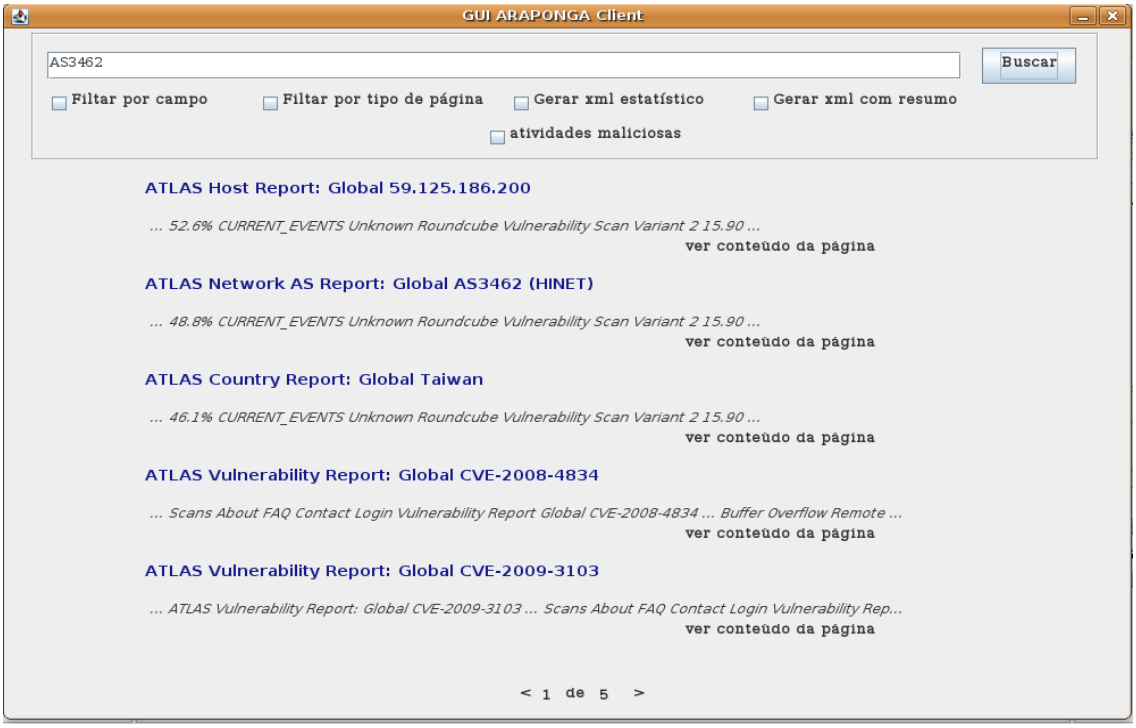

Figura 4.5: Exemplo da GUI de consulta.

A segunda é operada via linha de comando e foi elaborada visando à consulta por outros sistemas como, por exemplo, um sistema de tomada de decisão querendo obter a indicação de que um determinado endereço IP está envolvido em *SPAM* ou DNS fast-flux domain.

Ambas as implementações foram desenvolvidas em Java utilizando a API do Lucene para acessar os documentos indexados. Vale ressaltar que não existe diferença de resultados e nem nos tipos de consulta que podem ser feitas usando o console ou a GUI.

Os tipos de consulta oferecidos pelo módulo de interface são listados a seguir:

- **Geral** – utiliza apenas um parâmetro, a(s) palavra(s) a ser(em) buscada(s), percorrendo o conteúdo de todas as páginas. Um exemplo desta consulta é a busca por páginas que contenham a palavra "botnet";
- **Focada no tipo da página** – utiliza três parâmetros: a(s) palavra(s) a ser(em) buscada(s), o identificador (*tag*) de foco "***-focus***" e o tipo de página em que deve ser pesquisada. Este tipo de consulta percorre somente as páginas com tipo igual ao definido. Um exemplo desta consulta é a seguinte busca: sqlinjection ***-focus*** *Alert,bulletin,vulnerability*, onde somente as páginas do tipo alerta, boletim e vulnerabilidade serão consultadas;
- **Focada no campo da página** – utiliza três parâmetros: a(s) palavra(s) a ser(em) buscada(s), o identificador (*tag*) do campo "***-field***" e o campo a ser considerado na pesquisa. Este tipo de consulta percorre somente as páginas que possuem o campo igual ao definido. Um exemplo desta consulta é a seguinte busca: AS4134 ***–field*** *ASN*, onde somente as páginas que contenham o campo ASN (*Autonomous System Number*) serão consultadas;
- **Sim/Não** – utiliza três parâmetros: a(s) palavra(s) a ser(em) buscada(s), o identificador (*tag*) "***-malicious***" e o tipo *YES/NO*. Este tipo de consulta percorre todas as páginas a procura da(s) palavra(s) buscada(s). O diferencial desta consulta em relação à consulta geral é o resultado: YES caso o que se procura esteja relacionado a qualquer tipo de página que descreve atividade maliciosa, ou NO caso contrário. Este tipo de consulta é bastante útil para averiguar determinadas situações com, por exemplo, se um servidor SMTP está listado em alguma *black list* ou *white list*. Um exemplo desta consulta é a seguinte busca: AS4134 ***–malicious*** *YES/NO*;
- **Resumo de vulnerabilidade** – utiliza três parâmetros: a(s) palavra(s) a ser(em) buscada(s), o identificador (*tag*) "***-summary***" e o endereço canônico do nome do arquivo que será salvo. Este tipo de consulta percorre todas as páginas a procura da(s) palavra(s) buscada(s), retornando um arquivo XML contendo o nível de criticidade ou severidade da vulnerabilidade pesquisada e quantas vezes a vulnerabilidade obteve este nível, as datas de aparição e, por fim, os locais onde essas vulnerabilidades poderiam ser exploradas. Este tipo de consulta é útil porque permite traçar um perfil da vulnerabilidade. Um exemplo desta consulta é a seguinte busca: Microsoft ***–summary*** */home/trodrigues/summary*;
- **Resumo de ataques** – utiliza três parâmetros: a(s) palavra(s) a ser(em) buscada(s), o identificador (*tag*) "***-statistic***" e o endereço canônico do nome do arquivo que será salvo. Este tipo de consulta percorre todas as páginas a procura da(s) palavra(s) buscada(s), retornando um arquivo XML contendo o identificador da vulnerabilidade (padrão CVE), a classificação e a descrição deste ataque. Este tipo de consulta é útil porque permite traçar a abrangência

de um ataque. Um exemplo desta consulta é a seguinte busca: Microsoft **–*statistic*** */home/trodrigues/statistic*;

- **Focada no campo e no tipo da página** – utiliza cinco parâmetros: a(s) palavra(s) a ser(em) buscada(s), o identificador (*tag*) "***-field***", o(s) campo(s) buscado(s), o identificador (*tag*) "***-focus***" e o(s) tipo(s) de pagina(s). Este tipo de consulta realiza uma busca focada nos campos listados e apenas nos tipos de páginas definidas como parâmetro. Um exemplo desta consulta é a seguinte busca: *Microsoft* ***–field*** *high_v* ***-focus*** *Bulletin,Alert*;

Vale ressaltar que foram usadas algumas técnicas de mineração de dados no conteúdo da página com o objetivo de estruturar os conteúdos para consulta, separando cada campo identificado nos *templates* como relevante e indexando-os com os valores da coluna "Campo" demonstrado nas tabelas de *templates*, em apêndice. Também foram aplicadas técnicas de mineração nas consultas (remoção de *StopWords*) para possibilitar o uso de consultas por estrutura, consultas gerais ou consultas com filtragem de domínio.

5. Avaliações e Resultados

Este capítulo mostrará o ambiente em que a solução foi criada e exibirá alguns resultados contrastando-os a fim de avaliar a importância dos mesmos.

5.1 Ambiente de produção/testes

Na construção e testes deste trabalho foi utilizado um computador com processador Intel Core2Duo T5300, 2 Gbytes de memória RAM e HDD de 250 Gbytes. O sistema operacional utilizado foi o Ubuntu 8.04. O ambiente de rede do Grupo de Pesquisa em Redes em Telecomunicações (GPRT) da Universidade Federal de Pernambuco (UFPE) foi utilizado por fornecer um *link* de acesso a Internet de 100Mbps com o PoP-PE (Ponto de Presença da RNP).

5.2 Métricas de Avaliação de Desempenho

Para analisar os resultados do ARAPONGA foram adotadas métricas de avaliação de desempenho da área de recuperação da informação baseadas na noção de relevância, onde um documento é considerado relevante quando possui importância para o tópico considerado.

As métricas de avaliação [**36**] de desempenho utilizadas foram às seguintes:

- **Precisão -** a precisão é definida através da proporção entre o número de documentos relevantes retornados e o número total de documentos recuperados (figura 5.1).

$$\boldsymbol{Precisão} = \frac{\boldsymbol{N_{recuperados} \cap N_{relevantes}}}{\boldsymbol{N_{recuperados}}}$$

Figura 5.1: Fórmula da "Precisão".

- **Abrangência** – Dado-se o conjunto de documentos recuperados, a abrangência é a proporção entre o número de documentos relevantes recuperados e o número total de documentos relevantes na base(figura 5.2).

$$\boldsymbol{Abrangência} = \frac{\boldsymbol{N_{recuperados} \cap N_{relevantes}}}{\boldsymbol{N_{relevantes}}}$$

Figura 5.2: Fórmula da "Abrangência".

- **Média-F** – também conhecida como *F-mean* ou Média Harmônica, é a combinação entre Abrangência e Precisão(figura 5.3). Esta função retorna um valor no intervalo entre zero e um. Quanto mais próximo de zero, menos

relevantes são os documentos e quanto mais próximo de um, mais relevantes são os documentos da base.

$$\boldsymbol{Média - F = \frac{2}{\frac{1}{Abrangência} + \frac{1}{Precisão}}}$$

Figura 5.3: Fórmula da "Média F".

É interessante ressaltar que de acordo com o objetivo de cada processo de descoberta de conhecimento, métricas de avaliação de desempenho diferente das citadas devem ser utilizadas. Por exemplo, uma tarefa de sumarização não será bem avaliada por medidas como abrangência, precisão ou média-F.

5.3 Resultados

5.3.1 Avaliação do Número de Elementos na Base

O teste de avaliação do número de elementos na base foi executado durante três dias (de 24 a 26 de Novembro de 2009) e teve por objetivo analisar o número de páginas coletadas, além de mostrar a diferença entre as bases geral e refinada.

Basicamente, o teste consistiu na execução do *crawler*, operando de forma a capturar no máximo 80 referências (*links*) por página e com profundidade na árvore de busca de até 9 referências. A Figura 5.4 ilustra esse processo de coleta em profundidade, onde os "S" significam os sítios web acessados e "P" as páginas do referido sítio.

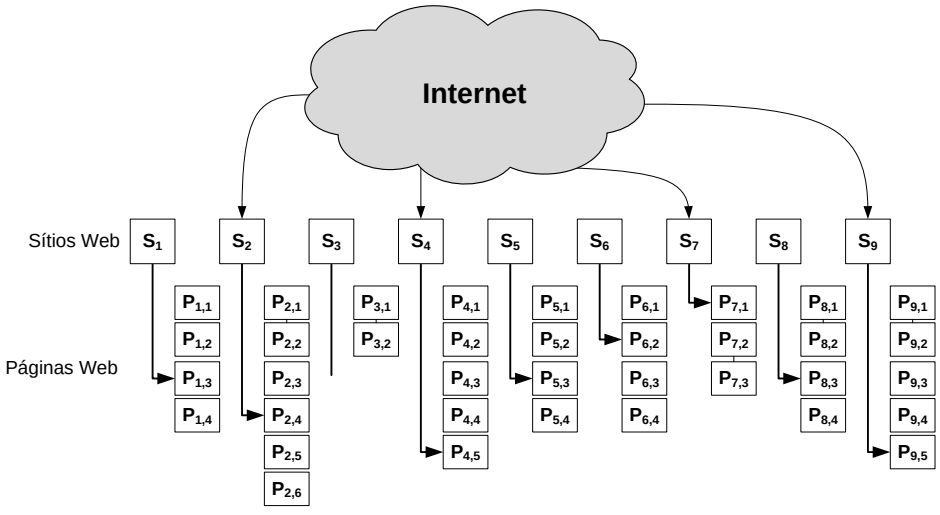

Figura 5.4: Coleta do *crawler* em profundidade.

A escolha por capturar no máximo 80 referências por página com profundidade 9 se deve ao fato de que testes iniciais com 150/10; 120/10; 100/9 e 90/9 (referências e

profundidade, respectivamente) terem resultado praticamente no mesmo número de páginas coletadas.

A Tabela 5.1 mostra o trabalho do módulo de adequação em relação ao número de documentos indexados em cada base, os documentos não indexados e aqueles indexados sem nenhum *template*.

Tabela 5.1: Documentos na base por dia.

	1° dia	2° dia	3° dia
URLs visitadas	5413	5342	5195
Páginas indexadas na base tradicional	388	358	346
Páginas indexadas na base aprimorada	220	213	202
Páginas indexadas sem template	21	18	13
Páginas não indexadas	147	127	131

Nota-se que existe uma diferença notória entre os valores da base geral e refinada. Tal diferença se deve ao esquema de filtragem realizada pelo módulo de adequação, que além de comparar as páginas com os *templates* (criados para extração de informações mais detalhadas de cada HTML e indexação com mais identificadoras) também executa filtragem de páginas por URLs e por conteúdo, de forma que páginas com conteúdos irrelevantes não sejam indexadas ou, case sejam indexadas, seu numero fique o mais próximo possível de zero.

5.3.2 Teste de rendimento

O teste de rendimento entre as duas bases do sistema (geral e refinada) é bastante relevante, pois exprime, em números, o ganho de rendimento de uma busca comum para uma busca diferenciada.

Para realização deste teste foi considerado que todas as páginas passadas pelo processo de filtragem do módulo de adequação e todas as páginas indexadas pelo módulo de indexação são relevantes. É importante ressaltar que dos tipos de buscas possíveis pelo módulo de interface, não foram analisadas aquelas que geram XML como resultado e nem as do tipo Yes/No, pois suas respostas não se enquadram nas métricas empregadas.

Basicamente, este teste consistiu de três consultas: a primeira do tipo focada no campo, a segunda focada no campo e no tipo de páginas e a terceira focada no tipo da página. Para tanto, foi considerada a base do dia 26 de Novembro de 2009, contendo **1092** documentos na base mantida sem adequação (geral) e **687** documentos na base mantida com adequação (refinada).

A **consulta #1** buscou por referências onde a palavra *Microsoft* estive envolvida com vulnerabilidades com alto grau de severidade. Desta forma, a consulta gerada foi à seguinte: *microsoft –field high_v*.

Como resultado, a consulta aplicada na base geral retornou 117 páginas com informações sobre a *microsoft*, onde apenas 20 descreviam vulnerabilidades com alto grau de severidade. Sendo assim, a precisão da consulta nesta base é de 17,09%, ou seja, dos 117 documentos recuperados, apenas 20 eram relevantes de um total de 117 documentos recuperados.

$$\boldsymbol{Precisão = \frac{N_{recuperados} \cap N_{relevantes}}{N_{recuperados}} = \frac{117 \cap 20}{117} = \frac{20}{117} = 0,1709}$$

A abrangência dessa consulta a base geral é de 1,83%, uma vez que dos 117 documentos retornados, apenas 20 eram relevantes de um universo de 1092 documentos.

$$\boldsymbol{Abrangência = \frac{N_{recuperados} \cap N_{relevantes}}{N_{relevantes}} = \frac{117 \cap 20}{1092} = \frac{20}{1092} = 0,0183}$$

Em relação à consulta na base refinada, a precisão foi de 100% uma vez que foram retornados apenas 20 documentos, sendo todos relevantes, e a abrangência foi de 2,91%.

$$\boldsymbol{Precisão = \frac{N_{recuperados} \cap N_{relevantes}}{N_{recuperados}} = \frac{20 \cap 20}{20} = \frac{20}{20} = 1}$$

$$\boldsymbol{Abrangência = \frac{N_{recuperados} \cap N_{relevantes}}{N_{relevantes}} = \frac{20 \cap 20}{687} = \frac{20}{687} = 0,0291}$$

A tabela 5.2 ilustra os valores encontrados na avaliação da Consulta #1.

Tabela 5.2: Resultado das métricas para Consulta #1.

	Abrangência	Precisão
Geral	0.0183	0.1709
Aprimorada	0.0291	1

Na **Consulta #2,** a busca foi por informações do ASN 3462. A intenção é descobrir se este ASN está ou foi listado ou relacionado em atividades como *spam*, *botnet* ou ataques. Desta forma, a consulta gerada foi à seguinte: *AS3462 –field ASN –focus spam,botnet,attack.*

Como resultado, a consulta aplicada na base geral retornou 27 páginas com contendo o AS3462, onde apenas 1 página o relacionava diretamente a atividade

maliciosas. Sendo assim, a precisão da consulta nesta base é de 3,7%, ou seja, dos 27 documentos recuperados, apenas 1 era relevante de um total de 27 documentos recuperados. A abrangência é de 0,09%, uma vez que dos 27 documentos retornados, apenas 1 era relevante de um universo de 1092 documentos.

Em relação à consulta na base refinada, a precisão foi de 100% uma vez que foram retornados apenas 1 documento, sendo relevante, e a abrangência foi de 0,14%, uma vez que apenas 1 era relevante de um universo de 687 documentos

A tabela 5.3 ilustra os valores encontrados na avaliação da Consulta #2.

Tabela 5.3: Resultado das métricas para Consulta #2.

	Abrangência	Precisão
Geral	0.0009	0.0370
Diferenciada	0.0014	1

A **Consulta #3** buscou por referências a palavra *Microsoft* relacionada com atividades maliciosas. Desta forma, a consulta gerada foi à seguinte: *microsoft –focus attack,alert,botnet,spam*

Como resultado, a consulta aplicada na base geral retornou 117 páginas com informações sobre a *microsoft*, onde apenas 42 a relacionavam a atividade maliciosas. Sendo assim, a precisão da consulta nesta base é de 35,89% e a abrangência é de 3,84%. Em relação à consulta na base refinada, a precisão foi de 100% uma vez que foram retornados apenas 42 documentos, sendo todos relevantes, e a abrangência foi de 6,11%.

A tabela 5.4 ilustra os valores encontrados na avaliação da Consulta #3.

Tabela 5.4: Resultado das métricas para Consulta #3.

	Abrangência	Precisão
Geral	0.0384	0.3589
Diferenciada	0.0611	1

De modo geral, observando-se os resultados obtidos é possível notar que o processo de indexar as páginas web utilizando *templates* possibilitou a criação de consultas diferenciadas e aumentou a precisão do sistema para 100%, que era o objetivo a ser alcançado.

5.3.3 Outros resultados

Esta subseção exemplifica as consultas não relatadas nos resultados obtidos anteriormente por não terem comparações pelas métricas estabelecidas.

Resumo de vulnerabilidade

Este tipo de consulta retorna um arquivo de extensão ".xml" contendo um resumo da vulnerabilidade procurada. Basicamente, consiste de uma busca por vulnerabilidades onde as informações de severidade, o impacto da vulnerabilidade e as soluções encontradas são retornadas em um documento semi-estruturado.

A figura 5.5 exemplifica o resultado da consulta *Internet Explorer* ***–summary*** */home/trodrigues/summary"*, onde saída será o arquivo *summary.xml* no diretório */home/trodrigues/*.

Figura 5.5: Sumário da consulta por Internet Explorer.

Resumo de ataque

Este tipo de consulta retorna um arquivo de extensão ".xml" semi-estruturado contendo um resumo de todas as CVEs na qual o ataque consultado é referenciado.

A figura 5.6 exemplifica o resultado da consulta sobre o protocolo TCP na porta 80 (*TCP/80* ***–statistics*** */home/trodrigues/statistic*).

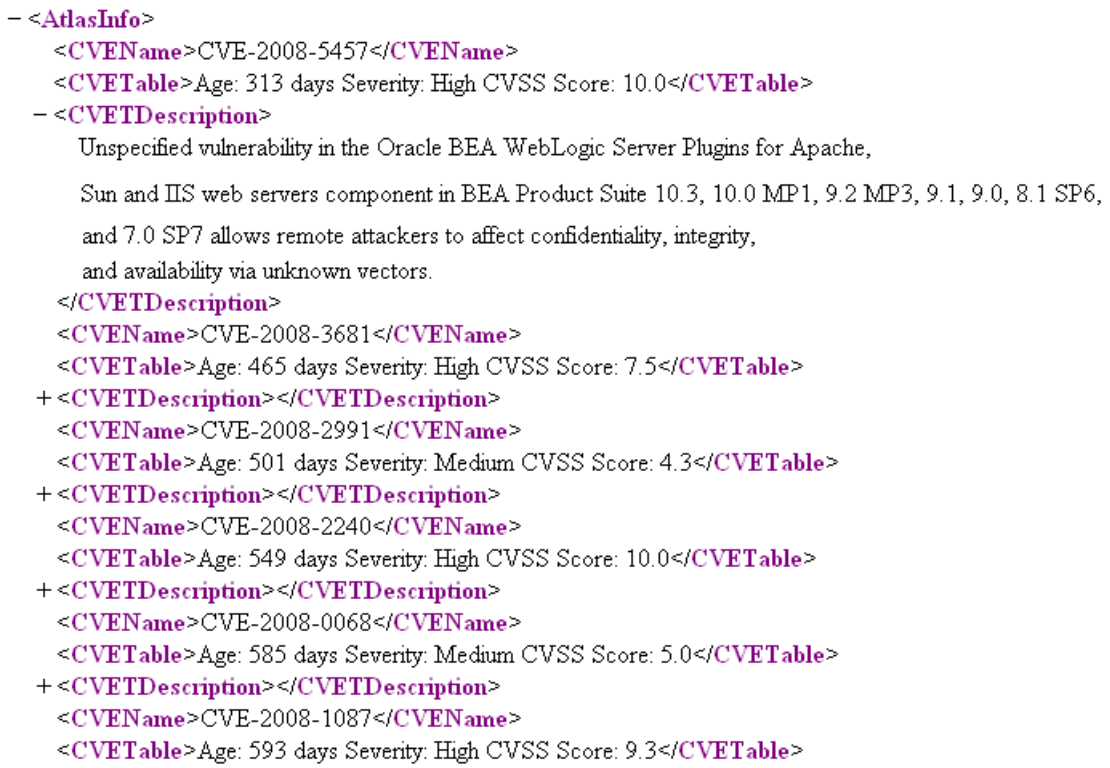

Figura 5.6: Resultado da consulta de resumo de ataque TCP/80.

Resumo de uma consulta Sim/Não (Yes/No)

Neste tipo de consulta, o resultado retornado é uma resposta simples de "*Yes*" quando há a existência da consulta na base de dados e "*No*" caso contrário.

Como exemplo, supõe-se que a consulta deseja receber informações se o AS4134 está ou estava em uma lista de ASNs envolvidos em atividades maliciosas. A consulta gerada é a seguinte: *AS3462 –**field** ASN – **result** YES/NO.*

6. Conclusão

Este trabalho de graduação apresentou uma ferramenta de apoio à recuperação de informações na web. A aquisição das informações na web é feita através de um *crawler* que adquire o conteúdo desejado das páginas HTML. Em seguida, os dados são indexados em uma base de dados para serem posteriormente acessados. O foco deste trabalho foi à busca por informação direcionada e com conteúdo restrito sobre vulnerabilidades e estatísticas do tráfego Internet. Para atingir este objetivo, técnicas de mineração de dados, *templates* e consultas direcionadas foram criadas e acopladas à ferramenta.

Para definir os locais onde as informações seriam coletadas, um estudo detalhado sobre algumas bases de dados e sítios que divulgam informações de atividades maliciosas foi realizado, levando em consideração a completude das informações e do conteúdo divulgados.

As principais contribuições deste trabalho de graduação foram:

- A concentração das informações divulgadas em várias bases de dados de vulnerabilidades e atividades maliciosas em um só lugar;
- A disponibilização de consultas para outros sistemas via console, possibilitando que o processo de tomada de decisão possa ser mais ágil e correto;
- A construção do módulo de Adequação, capaz de extrair somente informações uteis do conteúdo coletado;
- Armazenamento somente das informações relevantes;
- A ferramenta ARAPONGA.

6.1 Dificuldades Encontradas

Muitas dificuldades foram encontradas durante o período de criação deste trabalho de graduação.

A primeira foi em relação às ferramentas de coleta. Durante o período de produção, alguns especialistas na área de recuperação de informação na web foram consultados e os *crawlers* WIRE e Heritrix foram indicados como os mais viáveis para execução deste trabalho. Entretanto, após teste de instalação e execução, verificou-se que o WIRE não implementava esquemas de autenticação em páginas web e o Heritrix tinha como principal dificuldade a instalação, com vários pré-requisitos e algumas vezes apresentando problemas de compatibilidade entre versões do sistema operacional. Por estes motivos o Nutch foi escolhido como ferramenta de coleta.

O segundo problema diz respeito aos conteúdos das páginas web. Apesar de grande parte dos sítios web seguirem um padrão de exibição do conteúdo, o código HTML referente ao conteúdo muitas vezes era disforme criando uma enorme dificuldade na criação dos *templates*. Como exemplo, o código HTML ilustrado na figura 6.1 mostra uma parte do código de uma página do Secunia. Pode-se notar que as palavras-chave estão entre as *tags* HTML e mas, o conteúdo (que está destacado com um quadrado vermelho) não está identificado com nenhuma marcação HTML, dificultando bastante a aquisição destes valores.

```
<tr>
        <td class="AdvisoryTopSection">
                <b>Release Date:</b>
        </td>
        <td class="AdvisoryTopSection">
                2009-11-25
        </td>
</tr>
<tr>
        <td class="AdvisoryTopSection">
                <b>Last Update:</b>
        </td>
        <td class="AdvisoryTopSection">
                2009-11-26
        </td>
</tr>
<tr>
        <td class="AdvisoryTopSection">
                <b>Popularity:</b>
        </td>
        <td class="AdvisoryTopSection">
                1,066 views
        </td>
</tr>
```

Figura 6.1: Exemplo de código disforme em HTML.

Por fim, o Nutch foi concebido para atuar como um *crawler* "respeitável" (robots.txt e META-TAG). Entretanto, devido às necessidades deste trabalho, algumas adaptações e modificações relevantes e dificultosas foram realizadas para permitir que tais políticas fossem ignoradas.

6.2 Trabalhos Futuros

Como trabalhos futuros, podem ser relacionados às seguintes tarefas:

- A criação automática dos *templates* baseado no HTML de cada página coletada;
- A construção de um sistema de recomendação de busca baseado na proximidade das palavras ou sensível ao contexto. Como exemplo deste tipo

de consulta por proximidade pode-se citar uma consulta erroneamente feita pela palavra "*Nicrosoft*", onde o sistema seria capaz de perguntar se a consulta desejada foi pela palavra "*Microsoft*";

- A exibição, de forma visual, da evolução das informações coletadas. Por exemplo, um gráfico que mostra a relação do número de ataques ao BGP durante os 12 meses do ano.

Referências

[1] M. Morin, "The Financial Impact of Attack Traffic on Broadband Networks," *IEC Annual Review of Broadband Communications*, pp. 11-14, 2006.

[2] R. Richardson, "2007 CSI/FBI Computer Crime Survey.," in *12th Annual Computer Crime and Security*, 2007, pp. 1-30.

[3] CERT.br. (2007) Computer Emergency Response Team Brazil. [Online]. http://www.cert.br

[4] CAIS. (2007) RNP's Security Incident Response Team. [Online]. http://www.rnp.br/cais

[5] IBM. (2009) VulDa: A Vulnerability Database. [Online]. http://domino.watson.ibm.com/library/cyberdig.nsf/a3807c5b4823c53f85256561006324be/4cc8fa2ee3af7fc9852567280039a299?OpenDocument

[6] CISCO. (2009) Cisco Security Center. [Online]. http://tools.cisco.com/security/center/home.x

[7] NIST. (2009) National Vulnerability Database (NVD). [Online]. http://nvd.nist.gov

[8] Secunia. (2009) Secunia Advisories. [Online]. http://secunia.com/advisories/

[9] OSVDB. (2009) Open Source Vulnerabilities Database. [Online]. http://www.osvdb.org

[10] Luis. O. C Borba, "Um esquema de divulgação sobre informações de vulnerabilidades," Universidade Federal de Pernambuco, Recife, Trabalho Final de Graduação 2009.

[11] US-CERT. (2009) Technical Alerts. [Online]. http://www.us-cert.gov/cas/techalerts

[12] US-CERT. (2009) Security Bulletins. [Online]. http://www.us-cert.gov/cas/bulletins/

[13] US-CERT. (2009) Alerts. [Online]. http://www.us-cert.gov/cas/alerts/

[14] US-CERT. (2009) KB-CERT. [Online]. https://www.kb.cert.org/vuls

[15] Arbor Networks. (2009) Atlas. [Online]. http://atlas./arbor.net

[16] Insecure.org. (2009) Nmap. [Online]. http://nmap.org/

[17] Jack C. Louis. (2009) UnicornScan. [Online]. http://www.unicornscan.org/

[18] Tenable Network Security. (2009) Nessus. [Online]. http://www.nessus.org/nessus/

[19] GFI. (2009) LanGuard. [Online]. http://www.gfi.com/languard/

[20] Eeye Digital Security. (2009) Retina Network Security Scanner. [Online]. http://www.eeye.com/html/Products/Retina/index.html

[21] Acunetix. (2009) Acunetix Web Vulnerability Scanner. [Online]. http://www.acunetix.com/

[22] N-Stalker. (2009) N-Stalker. [Online]. http://www.nstalker.com/

[23] Ryen W. White and Resa A. Roth, "Exploratory Search: Beyond the Query-Response Paradigm," *Synthesis Lectures on Information Concepts, Retrieval, and Services*, vol. 1, no. 1, pp. 1-98, 2009.

[24] Wikipedia. (2009) Web Search Engine. [Online]. http://en.wikipedia.org/wiki/Web_search_engine

[25] Robotstxt.org. (2009) Robots Exclusion. [Online]. http://www.robotstxt.org/

[26] M. Kobayashi and K. Takeda, "Information retrieval on the web," *ACM Computing Surveys*, vol. 32, no. 2, p. 144–173, Jun 2000.

[27] J. Cho, H. Garcia-Molina, A. Paepcke, and S. Raghavan A. Arasu, "Searching the web," *ACM Transaction on Internet Technology*, vol. 1, no. 1, pp. 2-43, Ago 2001.

[28] Y. Y. Yao, "Information Retrival Support System," in *IEEE World Congress on Computational Intelligence*, 2002, pp. 773-778.

[29] G. and White, R. W. Marchionini, "Information Seeking Support System," *Computer*, vol. 42, no. 3, pp. 30-32, March 2009.

[30] E. L., Aschoff, R., Lins, B., Feitosa, E., Sadok, D. Oliveira, "Avaliação de Proteção contra Ataques de Negação de Serviço Distribuídos (DDoS) utilizando Lista de IPs Confiáveis.," in *VII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, Rio de Janeiro, 2007.

[31] Cwr.cl. (2009) Web Information Retrieval Environment - WIRE. [Online]. http://www.cwr.cl/projects/WIRE/

[32] Heritrix. (2009) Heritrix. [Online]. http://crawler.archive.org/

[33] Apache. (2009) Nutch. [Online]. http://lucene.apache.org/nutch/

[34] Apache. (2009) Lucene. [Online]. http://lucene.apache.org/java/docs/

[35] Jericho HTML Parser. (2009) Jericho HTMLl Parser. [Online]. http://jericho.htmlparser.net/docs/index.html

[36] C. W. Cleverdon, "The Cranfield tests on index langauges devices," *Aslib Proceedings*, vol. 19, pp. 173-192, 1967.

Apêndice - Templates

Este tópico explanará as características comuns nos padrões de cada site de divulgação de vulnerabilidade que foram estudados. Estes padrões que foram descobertos foram chamados de *templates* e nos próximos tópicos serão exibidos com uma breve descrição sobre cada campo.

Secunia Advisores

Campo	Descrição
Secunia	Identificador único da publicação
Release date	Data de lançamento
Popularity	Detalha os produtos afetados
Critical	Mostra quão crítico é a vulnerabilidade
Impact	Como afeta os sistemas
Where	Local do sistema afetado
Solution Status	O que foi feito para resolver o problema, se foi criado um patch ou uma nova versão
Software	Softwares afetados
CVSS Score	Pontuação para o impacto que esta vulnerabilidade ocasiona
Content	Conteúdo da publicação com informações de descrição, solução, referências externas e CVE's referentes ao assunto

KB-CERT

Campo	Descrição
Overview	Resumo da vulnerabilidade
Description	Descrição da vulnerabilidade
Impact	Impacto de um ataque explorando a vulnerabilidade
Solution	Como sanar o problema
Systems Affected	Sistemas afetados pela vulnerabilidade
Referencies	*Links* externos que contêm informações sobre a vulnerabilidade
Credit	Quem descobriu/solucionou a vulnerabilidade
Other Information	Informações das datas de publicação, solução e ultima atualização, ID da CVE e NVD referente à vulnerabilidade

US-CERT

As páginas da US-CERT contêm três padrões diferentes, um para cada tipo de divulgação de informação.

Alerta Técnico [5]

Campo	Descrição
Systems Affected	Lista com os sistemas afetados.
Overview	Resumo da vulnerabilidade.
Description	Descrição da vulnerabilidade
Impact	Impacto de um ataque explorando a vulnerabilidade.
Solution	Como sanar o problema.
Referencies	*Links* externos que contêm informações sobre a vulnerabilidade.

Boletim [6]

Campo	Descrição
Vendor-Product	Faz uma ligação entre o produto e o fabricante.
Description	Descreve brevemente o comportamento da vulnerabilidade.
Published	Data em que a vulnerabilidade foi publicada.
CVSS Score	Divulga uma pontuação para a vulnerabilidade de acordo com a severidade.
Source & Patch Info	*Links* externos para mais informações sobre a vulnerabilidade.

Alerta Simples [7]

Campo	Descrição
Systems Affected	Lista com os sistemas afetados
Overview	Resumo da vulnerabilidade
Solution	Como sanar o problema
Description	Descrição da vulnerabilidade
Referencies	*Links* externos que contêm informações sobre a vulnerabilidade

ATLAS

Campo	Descrição
ID	Identificador do elemento na tabela
Attacks per Subnet	Número médio de ataques que cada sub-rede sofreu
Percentage	Porcentagem no número de ataques em relação a todos os ataques

Além das informações contidas nas tabelas, há uma parte da página com o campo "BACKGROUND" que exibe informações detalhadas e referências externas sobre a ameaça.

As páginas da Arbor exibem mais informações quando o usuário está "logado". Estas informações podem ser visualizadas ao clicar nos Identificadores das Tabelas e contêm

informações detalhadas sobre o elemento selecionado como, por exemplo, ao clicar em um identificador que está na tabela de ASN, são exibidas mais informações sobre aquele ASN que ao usuário não “logado”.