Instituto Universitário de Lisboa

Departamento de Ciências e Tecnologias da Informação

Plataforma para Análise de Fugas de Informação na World Wide Web

Vítor Hugo Silva Sousa

Dissertação submetida como requisito parcial para obtenção do grau de

Mestre em Engenharia Informática

Orientador

Prof. Dr. Carlos Serrão, Professor Auxiliar

ISCTE-IUL

Co-Orientador

Eng. Nuno Teodoro, Information Security Director

Seedrs

Setembro 2016

"Information technology and business are becoming inextricably interwoven. I don't think anybody can talk meaningfully about one without the talking about the other."

Bill Gates

Resumo

A *World Wide Web* e a *Deep Web* são hoje em dia os principais focos de exposição e alojamento de informação corporativa confidencial. Adicionalmente, muitos dos sistemas de informação das organizações estão comprometidos, sendo muitas vezes possível recorrer a serviços online para verificar a existência de vulnerabilidades e exposição de serviços/informação que potencialmente possam colocar a organização em risco. Hoje em dia, indivíduos e empresas enfrentam problemas sérios de perdas de dados e informação que podem ser posteriormente revelados e utilizados para fins maliciosos. Na maioria dos casos, as organizações são reativas e não pro-ativas face a fugas de informação e à análise dos dados da infraestrutura que estão expostos em múltiplos serviços na *WWW*.

Com base neste problema foi desenvolvida uma plataforma que pretende atenuar este problema. Numa primeira abordagem foram feitas pesquisas sobre a técnicas de monitorização e extração da informação presente na *World Wide Web* através de *Web Crawlers* ou *Web Scrappers*. Foram ainda analisadas algumas plataformas de *Data Loss Prevention* comparando-as com a plataforma desenvolvida e as tecnologias de *Big Data* existentes.

Foi estruturada e desenvolvida uma plataforma *Web-based* que permite aos seus utilizadores a procura automática de informação corporativa na Web, que possa estar publicamente disponível, de forma não-autorizada em múltiplos serviços *online*. A utilidade desta plataforma foca-se na procura de informação pessoal ou corporativa através de termos definidos pelo utilizador em duas plataformas distintas, o *Pastebin* (http://pastebin.com) e o *Shodan* (https://www.shodan.io/).

Por fim a plataforma foi disponibilizada *online* e foram convidados especialistas da área da segurança de informação. Das respostas dos utilizadores registados foram extraídos dados sob forma de questionário para validar questões de funcionalidade da plataforma. Foram ainda feitos testes de validação de fugas de informação controlados durante cerca de 6 meses validando que existem de fato imensas fugas de informação presentes na *Word Wide Web.*

Palavras-chave: World Wide Web, Fuga de Informação, Confidencialidade, Big Data.

Abstract

The *World Wide Web* and the *Deep Web* are today the main focus for exposing and hosting confidential corporate information. Adicionally, many of these coporate information systems are compromised, being easy to resort to online services to check the existence of vulnerabilites and exhibition services/information that could potentially put the organization at risk. Nowadays individuals and businesses face serious problems of data loss and information that can later be disclosed and used for malicious purposes. In most cases, organizations are reactive and not proactive against information leakage and analysis of the data infrastructure that are exposed in multiple services on the *World Wide Web*.

Based on this problem we have developed a platform that aims to overcome this problem. In a first approach we researched about monitoring and extraction tecnhiques of the information that was present on the World Wide Web through *Web Crawlers* or *Web Scrappers*. There where also analyzed some *Data Loss Prevention* platforms comparing them with the developed platform and we analysed existing *Big Data* technologies.

It was strutured and developed a Web-base platform that allows users to automatically search for corporate information on the web, that may be publicly available in an non-athorized manner in multiple online services. The utility of this platform is focused on finding personal or corporate information through the terms defined by the user on two different platforms, the *Pastebin* (http://pastebin.com) and *Shodan* (https://www.shodan.io/).

Finally the platform was made available *online* and there have been invited experts of the information security area. From the responses in the questionnaire form of the registered users data was extracted to validate the platform functionality issues. There were also done information leakage validation tests monitored for approximately 6 months validating that there are indeed lots of information leakage present in the *Word Wide Web*.

Keywords: World Wide Web, Data Leakage, Confidentiality, Big Data.

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Códigos

Abreviaturas

WWW	**W**orld **W**ide **W**eb (ver página 1)
HTML	**H**yper **T**ext **M**arkup **L**anguage(ver página 9)
URL	**U**niform **R**esource **L**ocator (ver página 9)
FTP	**F**ile **T**ransfer **P**rotocol(ver página 16)
KPI	**K**ey **P**erformance **I**ndicator(ver página 25)
DLP	**D**ata **L**oss **P**revention(ver página 24)
DNS	**D**omain **N**ame **S**ystem(ver página 27)
IP	**I**nternet **P**rotocol(ver página 26)
CVE	**C**ommon **V**ulnerabilities and **E**xposures(ver página 27)
SaaS	**S**oftware **a**s **a** **S**ervice(ver página 26)
SI	**S**istemas de **I**nformação(ver página 4)
TI	**T**ecnologia de **I**nformação(ver página 4)
API	**A**pplication **P**rogramming **I**nterface(ver página 12)
DOM	**D**ocument **O**bject **M**odel(ver página 12)
CSS	**C**ascading **S**tyle **S**heets(ver página 37)
PHP	**P**HP **H**ypertext **P**reprocessor(ver página 37)
NoSQL	**N**ot **O**nly **SQL**(ver página 19)
OLTP	**O**nline **T**ransaction **P**rocessing(ver página 19)
XAMPP	**A**pache, **M**ySQL, **PHP**, **Perl**(ver página 37)
XML	e**X**tensible **M**arkup **L**anguage(ver página 21)
SQL	**S**tructured **Q**uery **L**anguage(ver página 18)
RDBMS	**R**elational **D**ataBase **M**anagement **S**ystems(ver página 18)
DMS	**D**atabase **M**anagement **S**ystem(ver página 20)
JSON	**J**avascript **O**bject **N**otation(ver página 21)

BSON	**B**inary **J**SON(ver página 21)
GUI	**G**raphical **U**ser **I**nterface (ver página 33)
DoS	**D**enial of **S**ervice(ver página 46)
CAPTCHA	**C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part(ver página 39)
SMTP	**S**imple **M**ail **T**ransfer **P**rotocol(ver página 39)
PDF	**P**ortable **D**ocument **F**ormat(ver página 39)
ASN	**A**utonomous **S**ystem **N**umbers(ver página 39)
XSS	**C**ross-site **S**criptin(ver página 52)
CRSF	**C**ross-site **R**equest **F**orgeryn(ver página 52)
HTPPS	**H**yper **T**ext **T**ransfer **P**rotocol **S**ecuren(ver página 53)
SSL	**S**ecure **S**ockets **L**ayer(ver página 53)
RSS	**R**eally **S**imple **S**yndication(ver página 81)

Capítulo 1

Introdução

1.1 Motivação

A *WWW* é nos dias que correm, uma das melhores fontes de informação e pode ser utilizada para divulgar de forma não autorizada e de forma anónima informação confidencial, bem como expor vulnerabilidade no contexto da infraestrutura organizacional. A maior parte desta informação exposta pode originar grandes perdas por parte da organização, sendo que em alguns casos esta é sensível e vital.

Na maioria dos casos, as organizações são reativas (i.e. só existe reação quando a informação confidencial é exposta) e não pró-ativas face ás fugas de informação e exposição de vulnerabilidades que estão presentes em múltiplas plataformas na *WWW*. O estudo, análise e desenvolvimento de uma plataforma que consiga fornecer uma capacidade pró-ativa e que consiga agregar e correlacionar toda a informação vital dispersa na *WWW* representa uma arma de defesa com um valor inestimável para as organizações.

1.2 Enquadramento

Nos dias que correm Gorden e os autores Shabtai et al. consideram que a fuga de dados é simplesmente a transmissão de dados não autorizada pela organização com um destino exterior e esta pode ser, eletrónica ou física (Gordon, 2007 e Shabtai et al., 2012).

Tendo atenção que normalmente a transmissão não autorizada não quer automaticamente dizer intencional ou maliciosa, a fuga de dados não intencionada também não é autorizada.

Uma vulnerabilidade em termos informáticos é definida como a descoberta de uma irregularidade num sistema, permitindo assim o acesso ao mesmo e dando oportunidade de explorar essa falha. Muitas vulnerabilidades encontradas são corrigidas, pois existe ética e boa conduta por parte do indivíduo que as descobriu mas quando este não é o caso, existe a exposição de vulnerabilidades e ás vezes a fuga de dados que lhe está associada. Assim sendo a exposição de vulnerabilidades tem apenas o intuito de expor as irregularidades do sistema permitindo assim a quem detém esses conhecimentos a oportunidade de comprometer negativamente a organização. Como hoje em dia grande parte das organizações têm sistemas que permitem guardar dados sensíveis informaticamente, e nem tudo está seguro, existem sempre falhas que permitem a pessoas maliciosas comprometer e partilhar esses dados na *WWW*.

1.2.1 Questão de Investigação

A seguinte questão de investigação debruça-se sobre o tema da fuga de informação e exposição de vulnerabilidades e permitem perceber se este tópico é sensível o suficiente para que exista um melhor controlo na abordagem da fuga de informação e exposição de vulnerabilidades.

1. Será que é possível detetar e corrigir fugas de informação na *WWW* usando ferramentas automáticas que pesquisam através de padrões de dados?

1.2.2 Objetivos

O objetivo principal é a criação de uma plataforma que possibilite a agregação de informação proveniente de várias fontes da *WWW*, e que seja modular e costumizável o suficiente para permitir direcionar esforços em relação à fuga de informação corporativa e individual bem como a análise de vulnerabilidades expostas na *WWW*. Para tal, serão utilizados alguns dos métodos de monitorização da *WWW* permitindo assim agregar e correlacionar a informação dispersa. Esta plataforma terá como principais características:

- Possibilidade de selecionar o tipo de dados a visualizar na plataforma, tendo em conta um conjunto de fontes selecionadas como o PasteBin[1], GitHub Gist[2], Shodan[3], Redes Sociais (e.g. Twitter[4], Facebook[5], LinkedIn[6]), entre outras.

- Produção de relatórios e *dashboards* agregados por tipo de fonte e tipo de dados;

- Consumo ou correlação de serviços derivados da *framework "Collective Intelligence Framework"*[7] que permitem combinar várias fontes de informação com ameaças maliciosas conhecidas de forma a prevenir ou mitigar a infraestrutura da organização (p.ex *feeds* de *"Malicious software"*);

- A criação de gráficos e métricas evolutivas dos dados processados , permitindo a geração de relatórios ad-hoc periódicos costumizáveis e gestão de perfis de acesso.

- Uma vez que esta ferramenta lida com informação critica armazenada e gerida pela plataforma, terá obrigatoriamente que ser desenvolvida tendo em conta alguns requisitos adicionais:

[1] http://http://pastebin.com/
[2] https://gist.github.com/
[3] https://www.shodan.io/
[4] https://twitter.com/
[5] https://www.facebook.com/
[6] https://www.linkedin.com/
[7] http://csirtgadgets.org/

- Utilização de técnicas de desenvolvimento seguras que previnam as principais vulnerabilidades aplicacionais (e. g. utilização do *OWASP Top 10*);
- *Hardening* das configurações de segurança dos componentes da infraestrutura que suportarão a plataforma; e
- Documentação técnica do desenvolvimento de toda a plataforma.

1.2.3 Método de Investigação

O método de investigação escolhido para a realização deste trabalho de investigação é o *Design Science Research*, este é fundamentalmente um paradigma para resolver problemas do mundo-real aplicado aos negócios como afirmam Hevner e Chatterjee (Hevner and Chatterjee, 2010). Este paradigma é muito relevante para a investigação dos SI pois este foca-se em dois pontos chave desta disciplina, o papel os artefactos produzidos nas TI aplicados aos métodos de investigação dos SI e a notória falta de relevância profissional na investigação dos SI, afirmam Hevner e Chatterjee baseando-se em vários autores (Hevner and Chatterjee, 2010).

Existem algumas abordagens dentro do método de investigação escolhido. A metodologia escolhida para este caso foi a de *checklist* que permite a resposta a 8 questões e irá orientar todo o processo de investigação abordando aspetos chave do método de investigação. As questões são as seguintes:

Questão 1 - Qual o foco desta investigação?

O foco desta investigação é a agregação de informação proveniente de várias fontes da *WWW*, direcionando esforços em relação à fuga de informação corporativa e individual bem como a análise de vulnerabilidades expostas na *WWW*.

Questão 2 - Qual é o artefacto produzido e como é representado?

O artefacto criado será uma plataforma que permita agregar e correlacionar a informação sobre fuga de informação e exposição de vulnerabilidades dispersa na *WWW*, esta irá permitir a criação de *dashboards* agregados (por tipo de fonte e

tipo de dados), criação de gráficos e métricas evolutivas dos dados processados, permitindo a geração de relatórios ad-hoc periódicos costumizáveis.

Questão 3 - Que processos de desenho vão ser usados para construir o artefacto?

Os processos de desenho serão baseados em algumas técnicas de monitorização da *WWW* (descrito na secção 2) e podem ser alterados dependendo do tipo de fonte de informação.

Questão 4 - Como é que o artefacto e os processos de desenho são fundamentados com a base de conhecimento?

A base de conhecimento será toda a informação corporativa ou individual que o artefacto conseguir reunir de acordo com certos filtros e tópicos mais sensíveis, estes serão processados, analisados e posteriormente será devolvido um relatório costumizável, como foi anteriormente respondido na questão 2.

Questão 5 - Que avaliações são efetuadas durante o ciclo interno de desenho? Que melhorias a nível de desenho são identificadas durante cada ciclo?

Para avaliar a performance das técnicas de monitorização escolhidas, serão verificadas se estas conseguem reunir a informação presente a *WWW*. As melhorias serão notórias quando o artefacto conseguir identificar com sucesso fugas de informação e exposição de vulnerabilidades presentes na base de conhecimento.

Questão 6 - Como é que o artefacto produzido é introduzido no ambiente aplicacional? Que métricas são usadas para demonstrar a utilidade e o melhoramento dos artefactos anteriores?

O artefacto irá primeiramente ser avaliado pelo funcionamento tendo em conta cada ponto definido na secção 1.2.2. Este será ainda avaliado tendo em conta as seguintes métricas:

1. Verificação da usabilidade;

2. Validação dos requisitos de segurança;

3. A percentagem de completude.

Questão 7 - Que conhecimento é adicionado à base de conhecimento e de que forma?

Toda a informação relevante (i.e. fugas de informação) e monitorizada espalhada na *WWW* será guardada numa base de dados permitindo assim aumentar a base de conhecimento.

Questão 8 - A questão que levou à investigação foi satisfatoriamente respondida?

A questão de investigação foi respondida com os testes realizados durante 6 meses à plataforma do *Pastebin* e através dos questionários realizados aos especialistas da área de segurança da informação.

Capítulo 2

Análise do estado da arte

Ao longo deste capítulo serão abordados serviços similares expostos na WWW, técnicas para a recolha de dados, meios de propagação da informação e tecnologias de *Big Data*.

2.1 Existência de serviços similares

Sendo a WWW uma rede enorme, devem existir plataformas ou serviços similares disponíveis para que os utilizadores possam monitorizar e coletar informação que lhes é essencial. Neste capítulo serão analisadas todas as plataformas que se enquadrem na prevenção de fugas de informação e na monitorização da WWW.

2.1.1 Monitorização da WWW

Como podemos comprovar no relatório global de fuga de dados (InfoWatch, 2014) foi registado em 2014 cerca de 1395 casos (Figura 2.1), ou seja, 3.8 por dia ou 116 por mês, sendo maioritariamente o foco destas fugas nos Estados Unidos e na Rússia. A fuga destes dados comprometeu cerca de 767 milhões de dados pessoais sendo que estes são informação crítica essencial pessoal. Na vertente empresarial foram comprometidos cerca de 800 milhões de dados (350 milhões de dados a

partir de ações das pessoas de dentro da empresa e 450 milhões de dados através de ataques externos) e em ambos os casos estas fugas podem provocar perdas catastróficas nas mãos de pessoas maldosas.

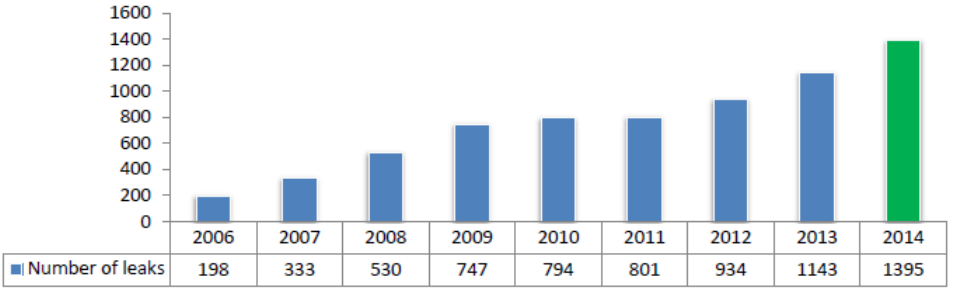

FIGURA 2.1: Número de fugas de dados registados entre 2006 e 2014, Fonte: retirado de InfoWatch, 2014, p. 7

Quando existe exposição de dados não autorizados estes podem variar a sua origem e, de facto, existe a necessidade de serem classificados. Através do tipo de informação, podemos enquadra-los em dois tipos: Corporativos ou Pessoais/Individuais. Dentro da perspetiva corporativa os dados podem ser documentos, isto é, demonstrações financeiras e segredos não revelados (e. g. informação privada ou projetos não revelados) e podem ainda ser informações das bases de dados (e. g. nomes de utilizadores com as respetivas palavras-passe). Na outra vertente os dados podem ser informações pessoalmente identificáveis (e. g. endereços IP, Moradas, Números de Cartões, etc) e Contas pessoais (e. g. contas bancárias online, contas de email, etc).

Dependendo do tipo de dados, os canais de fuga podem variar imenso, estando ainda presente no relatório InfoWatch, 2014, que a WWW teve o peso mais significativo com 35% do total de canais em 2014, seguido de 18% em que não se consegue especificar a fonte de informação sobre o canal de fuga, como se pode observar na Figura 2.2.

De facto existe a necessidade de monitorizar o canal que mais peso tem na propagação da informação hoje em dia, e mitigar com eficiência os danos que esta propagação poderá causar tanto a nível pessoal como corporativo.

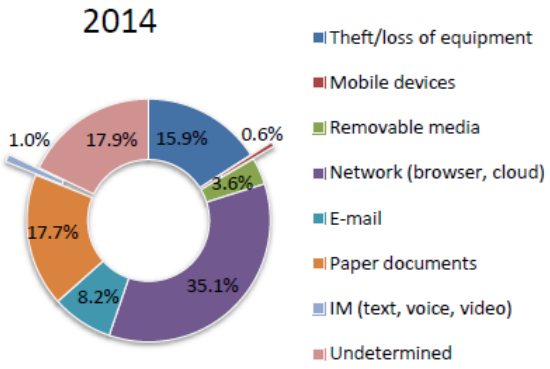

FIGURA 2.2: Distribuição de fugas por canal, Fonte: retirado de (InfoWatch, 2014, p. 13)

Monitorizar a WWW pode ser um processo lento e fastidioso pois esta tem um crescimento exponencial ou seja, não existe um controlo no que toca à criação de páginas web. Para que haja uma forma de procura nesta vasta rede, existem métodos e ferramentas que permitem a interação e extração de conteúdos. *Web Crawling, Web Scraping, Web Minning e Text Minning* são algumas das técnicas que nos providenciam essas características e permitem a correlação e agregação dos dados espalhados na rede.

2.1.1.1 Web Crawling, Web Scraping e as suas técnicas

Um Web Crawler, como Cho indica é um programa que descarrega páginas da web (Cho, 2001). Este percorre a WWW de forma metódica e automatizada e é normalmente utilizado em motores de busca (como o Google, Bing, Yahoo!, etc..) para manter as bases de dados atualizadas.

É criada uma cópia de todas as páginas visitadas para serem pré-processadas e indexadas pelos motores de busca, permitindo assim procuras mais rápidas e eficientes. Estes podem ainda ser utilizados para tarefas de manutenção como verificar *links*, validar código HTML e obter tipos de informação específicos numa pagina web.

Masanés afirma ainda que, dado um número de sementes URL, o *Web Crawler* descarrega todas as páginas endereçadas pelos URL's, extraindo todas as ligações

presentes na página web para o *Crawl Frontier* (ou seja, este serve para guardar uma nova lista de ligações a visitar (Masanés, 2006). Afirma ainda que o *Web Crawler* recursivamente visita estas ligações e descarrega todas as páginas web endereçadas de acordo com as políticas definidas, como demonstrado na Figura 2.3.

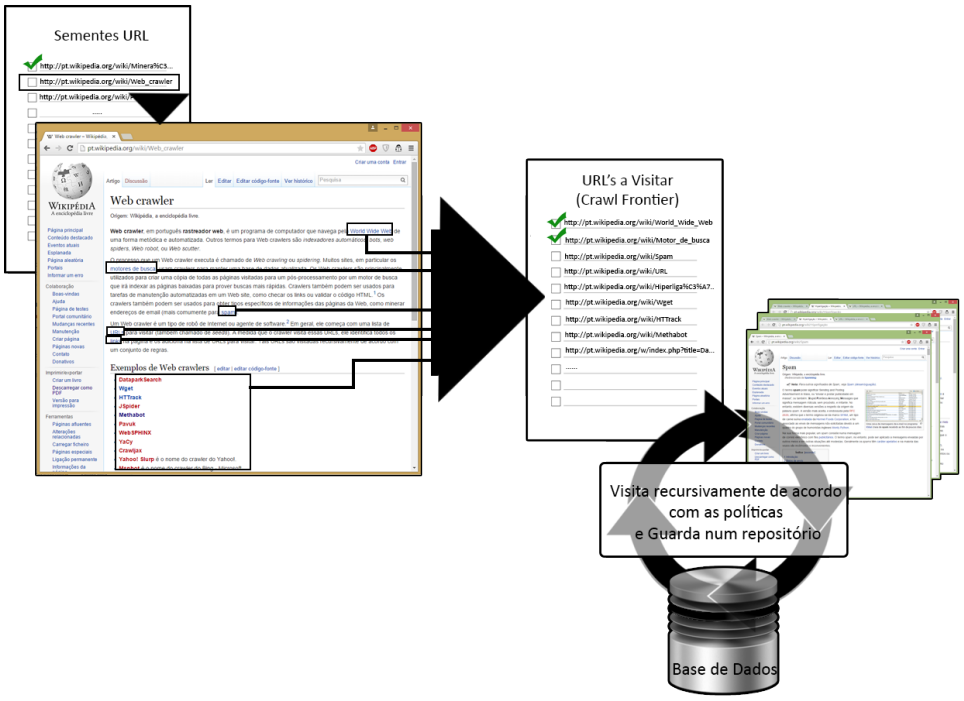

FIGURA 2.3: Como funciona um Web Crawler.

Como Peshave indica existem técnicas que permitem aumentar a eficácia quando se está a procura de determinado conteúdo na WWW (Peshave, 2005). Este afirma ainda que uma das técnicas utilizadas chama-se *Topic/Focused Crawling*, onde é permitido coligir páginas web que satisfazem um determinado tipo de tópicos através da priorização do *Crawl Frontier* e administrando o processo de exploração dos URL's, sendo que assim não existe a necessidade de descarregar todas as páginas reduzindo a quantidade de tráfego na rede e aumenta a produtividade na procura de um determinado tópico.

Para que este tenha sucesso, Peshave afirma que é necessário três componentes principais(Peshave, 2005):

1. O Classificador que julga a relevância das páginas que já visitou e decide sobre a expansão das ligações.
2. O Destilador que determina uma medida de centralidade das páginas visitadas para determinar as prioridades de visita
3. Um Crawler que tem controlos de prioridade dinamicamente reconfiguráveis que são controlados pelo classificador e pelo destilador

Quando estamos a trabalhar com um *Topic/Focused Crawler* temos que medir e monitorizar o rácio de colheita. Este é simplesmente a taxa na qual as páginas relevantes são descarregas e as páginas irrelevantes são filtradas do processo de crawling. Este rácio deve ser alto, caso contrário o *Topic/Focused Crawler* está a perder demasiado tempo a eliminar páginas irrelevantes.

Outra técnica afirma Peshave é a *Distributed Crawling* onde basicamente o processo da atividade de crawling é distribuído via múltiplos processos (mais utilizado por motores de busca na WWW), permitindo assim um aumento na escalabilidade, velocidade de extração das páginas e fiabilidade (Peshave, 2005).

Web Scraping como Cording indica, é o processo de extração de informação estruturada presentes nas páginas web (Cording, 2011). Normalmente quando

queremos extrair estas informações das páginas web, os autores não disponibilizam uma API que permita a extração.

Como os autores Zheng et al. afirmam, a árvore DOM é a representação de uma página HTML (Zheng et al., 2007). Cada nó da árvore DOM representa um *tag* HTML, como se pode observar na Figura 2.4.

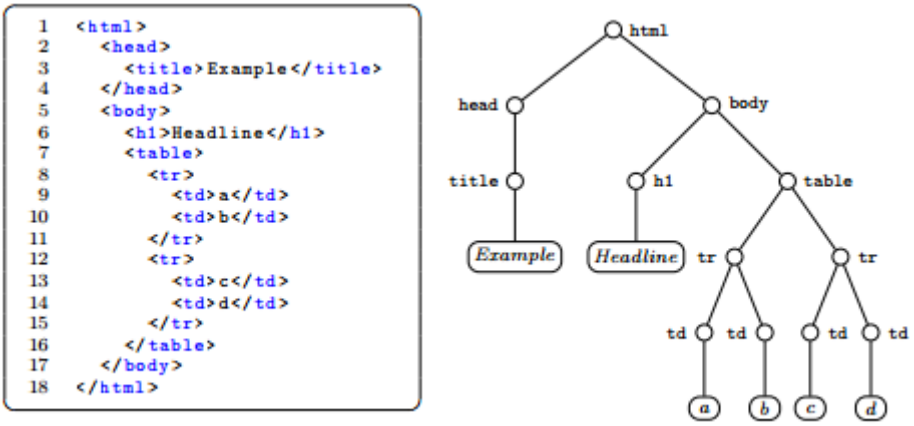

FIGURA 2.4: Representação da árvore DOM (á direita) com o seu respetivo código HTML, Fonte: retirado de Cording, 2011, p .18.

Uma abordagem comum diz Cording, é analisar a página web e representá-la num diagrama em árvore que contem toda a sua estrutura e calcular uma expressão *XPath* (Cording, 2011). O *XPath* mostra o caminho, possivelmente com caracteres universais e quando este é executado numa árvore, o resultado é um conjunto de nós DOM e no final é representando a estrutura total da árvore. Cording afirma ainda que a desvantagem desta técnica é que as páginas web são dinâmicas ou são constantemente atualizadas. Mesmo com os carateres universais do *XPath* estes são vulneráveis a estas alterações porque uma pequena alteração pode ser feita a uma *tag* HTML e esta pode não ser coberta pelos caracteres universais.

2.1.1.2 Data Mining, Text Mining e Web Minning

Os autores Weiss et al. afirmam que o conceito de *Data Mining* é simplesmente a forma de encontrar padrões válidos em dados, e dá resposta às coleções e armazenamento de grandes volumes de dados (Weiss et al., 2005). Afirmam ainda

que os métodos de Data Mining esperam um formato de dados muito estruturado, estes têm que ser tratados de numa forma especial antes que qualquer método de aprendizagem possa ser aplicado. Normalmente os dados de *Data Mining* são representados em forma numérica numa folha de cálculo. Weiss et al. fazem a distinção entre *Data* e *Text Mining*, afirmando que o segundo é apenas representado por texto e documentos, mas sendo diferentes partilham muitos métodos de aprendizagem similares (Weiss et al., 2005).

Os autores Himmel et al. indicam que *Text Mining* é um método para a classificação automática de um grande volume de documentos (Himmel et al., 2009). Afirmam ainda que esta técnica consiste normalmente em passos limitados, como analisar o texto em segmentação de palavras, encontrar termos e reduzi-los (aplicando métodos de truncamento) seguido de procedimentos analíticos como agrupamento e classificação para derivar padrões nos dados estruturados, e finalmente a avaliação e interpretação da saída. Tipicamente tarefas de *Text Mining* incluem a categorização de texto, a extração de conceitos/entidades e sumarização de documentos.

Os autores M.Srividya et al. indicam que *Web Mining* é uma técnica para extrair informação presente na WWW e esta pode ser classificada em três diferentes categorias: *Web Content Mining, Web Structure Mining e Web Usage Mining* (M.Srividya et al., 2013).

Os autores afirmam que *Web Content Mining* é a descoberta de informação útil presente nos conteúdos da web, incluindo texto, imagens, vídeos, etc. O *Web Content Mining* tem como alvo a descoberta de conhecimento em que os principais objetivos são documentos de texto, e mais recentemente documentos de multimédia que estão ligados às páginas Web (M.Srividya et al., 2013). Afirmam ainda que existem ainda dois tipos de abordagens utilizadas nesta categoria. A abordagem baseada em agentes e a abordagem de base de dados. Na primeira abordagem existem três tipos de agentes: os de procura inteligente que automaticamente procuram por informação de acordo com uma *query* específica, os de categorização/filtragem de informação que utilizam um número de técnicas para

filtrar os dados de acordo com instruções pré-definidas e, por fim, os web personalizáveis que aprendem as preferências dos utilizadores e descobrem documentos relacionados com os perfis dos mesmos. A segunda abordagem consiste numa base de dados que contêm esquemas e atributos com domínios definidos.

Ainda os autores M.Srividya et al. afirmam que o *Web Structure Mining* tenta descobrir o modelo que está por debaixo da estrutura de conexão da web (M.Srividya et al., 2013). Este modelo é baseado na topologia dos *hyperlinks* com ou sem descrição das conexões. Este pode ainda, ser utilizado para categorizar as páginas web e é útil para gerar informação como a similaridade ou a relação entre páginas web.

Por fim, *Web Usage Mining* é a aplicação de técnicas de *Data Mining* para descobrir padrões de uso a partir de dados gerados por transações cliente-servidor em um ou mais servidores web (M.Srividya et al., 2013). Esta é a área principal em *Web Mining*, focada em aprender sobre os utilizadores que navegam a web e as suais interações com as páginas web.

Existem duas abordagens que são usadas no *Web Usage Mining*: mapear os dados usados em tabelas relacionais antes de usar técnicas de *Data Mining* e, usar os dados de *logs* diretamente utilizando técnicas de pré-processamento especiais.

2.1.1.3 Meios de Propagação da Informação

Com o crescimento da WWW foram criadas inúmeras formas de propagar a informação (e. g. Wikipédia, Pastebin, WikiLeaks, GitHub entre outras) e é notório que existem fontes de informação mais utilizadas para propagar as fugas de dados e as vulnerabilidades que outras.

O Pastebin é uma plataforma web que permite aos utilizadores a partilha de informação textual (e. g. código de programação, URL, etc) na WWW através da ação de colar (ou "*paste*"), permitindo a criação anónima, escolher uma data de expiração e até opções de partilha (pública, privada ou não listada) como se pode observar na figura 2.5.

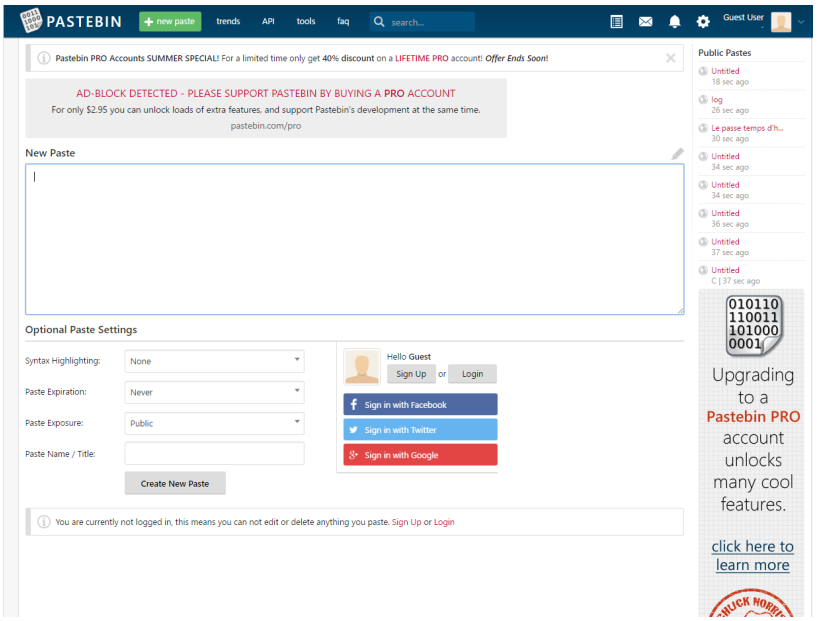

FIGURA 2.5: Página principal do Pastebin onde se podem criar novos *"Pastes"* - imagem retirada de http://http://pastebin.com/

Após a criação é devolvido ao utilizador um URL único que poderá ser partilhado em qualquer lugar na WWW até mesmo em páginas que contêm restrição de caracteres (como é o caso do Twitter). Atualmente não existe qualquer tipo de restrição a nível do tipo de informação partilhada nesta plataforma, até porque esta é utilizada para partilhar informação sensível e roubada, como anunciado:

> *"We monitored pastebin.com from late 2011 to early 2012, periodically downloading public pastes and following links to user-defined posts. We recorded a diverse range of categories of sensitive or malicious information leaked daily: lists of compromised accounts, database dumps, list of compromised hosts (with backdoor accesses), stealer malware dumps, and lists of premium accounts."* - (Matic et al., 2012).

O Pastebin é um dos meios mais utilizados para partilhar informação sensível na WWW, pelo facto de proporcionar anonimato aos seus utilizadores (mas se este fator não existisse era facilmente contornável através de proxychains, VPNs ou mesmo através do browser Tor).

O GitHub Gist é um serviço de partilha de informação textual idêntico ao Pastebin, este permite também o anonimato através da criação de um Gist Público ou Secreto (a diferença é que este não é monitorizado pelos motores de busca e só é visível através do URL privado) como se pode observar na figura 2.6.

FIGURA 2.6: Página principal do GitHub Gist - imagem retirada de https://gist.github.com/

Apesar de serem muito parecidos, a diferença é que o GitHub Gist permite a criação de vários ficheiros de texto (ao invés de um longo bloco que texto do Pastebin) o que possibilita a melhor estruturação da informação a partilhar.

O Shodan apesar de não ser uma plataforma que permita a partilha de informação, permite a encontrar computadores específicos (routers, servidores, etc) através de uma variedade de filtros como se pode observar na figura 2.7.

Este permite encontrar máquinas a correr um determinado tipo de software, quantos FTP anónimos existem, quantas máquinas conseguíamos infetar com uma vulnerabilidade nova. Normalmente os motores de busca tradicionais não permitem uma procura tão específica como esta sendo que esta ferramenta nas mãos erradas pode ser utilizada para proveito malicioso. O canal de televisão norte-americano *Cable News Network* noticiou que :

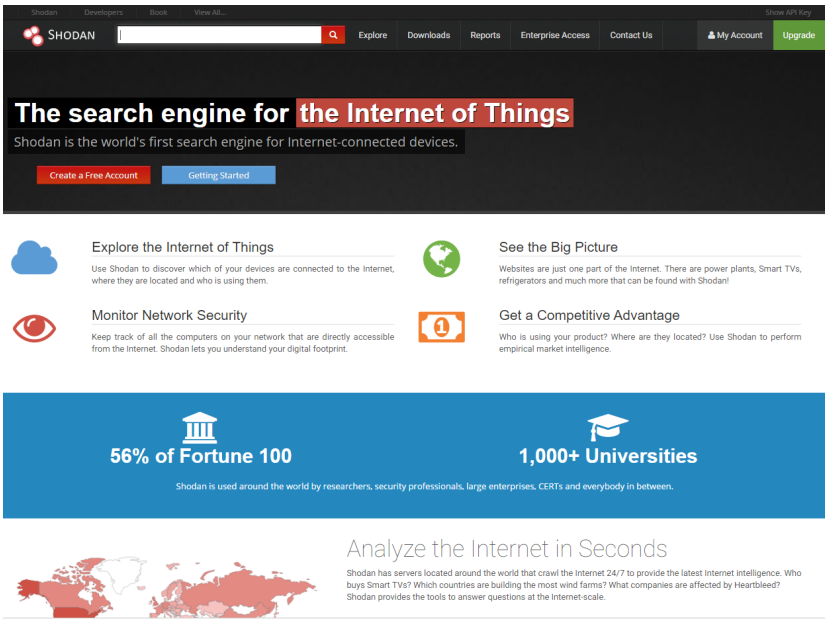

FIGURA 2.7: Página principal do Shodan - imagem retirada de https://www.shodan.io/

> *"The scariest search engine on the WWW"* e que *"Shodan is almost exclusively used for good."* como *"Penetration testers, security professionals, academic researchers and law enforcement agencies are the primary users of Shodan."* - (CNN and Goldman, 2013).

Apesar que os cibercriminosos podem utilizar esta ou outras ferramentas semelhantes para conseguirem o mesmo efeito que o Shodan sem serem detetados.

O Tor[1] é uma das redes anónimas mais populares e procuradas proporcionando privacidade a quem disponibiliza os serviços na WWW (e. g. serviços de publicação web, mensagens instantâneas, etc).Os autores Birykov et al, indicam que os serviços ocultos introduzidos em 2004, possibilitam correr um serviço (e. g. web site, servidor de SSH, etc) permitindo que os seus utilizadores não saibam qual é o endereço IP atual do serviço (Biryukov et al., 2013). Isto é obtido através do *onion routing* proporcionando anonimato em toda a informação entre o cliente e o serviço oculto.

[1]https://www.torproject.org/

Outras plataformas bastante utilizadas para a propagação da informação são as redes sociais. O Facebook e o Twitter servem como meio para a transmissão de informação sensível. No caso do Facebook a criação de páginas, ou a criação de grupos fechados permitem a propagação de informação rapidamente, a limitação de caracteres no caso do Twitter (140 por *tweet*) parece não afetar a propagação, pois os cibercriminosos utilizam outro tipo para diminuir o número de caracteres que podem incluir (e.g. *URL Shortener*).

2.1.2 Tecnologias de Big Data

A plataforma descrita na secção 1.2.2 irá reunir um grande volume de dados, logo é necessário o uso de ferramentas de *Big Data*. Antes de endereçar qualquer tecnologia é importante definir o que é *"Big Data"*.
Esta é considerada por Manyika et al. como uma coleção de grande volume, variedade e velocidade de dados que não consegue ser eficaz nem acessível através das ferramentas de gestão de base de dados convencionais como RDBMS (Manyika et al., 2011).

2.1.2.1 Relational database management system (RDBMS)

Kline afirma que sistemas de base de dados relacionais (i.e *RDBMS*), como *SQL server* e *Oracle*, são os motores principais no que toca a sistemas de informação mundiais, particularmente em aplicações web e sistemas de computação cliente/-servidor distribuídos(Kline, 2001).
Este afirma ainda que um RDBMS é definido como um sistema cujo os utilizadores observam os dados como uma coleção de tabelas relacionadas através de valores comuns. Os dados são armazenados em tabelas e as tabelas são compostas de linhas e colunas. Tabelas de dados independentes podem ser associados (ou relacionados) para outras tabelas se cada uma das colunas de dados (chamadas chaves) representarem o mesmo valor de dados.

2.1.2.2 NoSQL

Stonebraker refere que as bases de dados NoSQL são normalmente consideradas como OLTP(*Online Transaction Processing*) que são baseadas em atualizações e pesquisas intensivas (Stonebraker, 2010). Moniruzzaman e Hossain afirmam que para garantir a integridade dos dados, a maior parte dos sistemas clássicos de base de dados são baseados em transações (Moniruzzaman and Hossain, 2013). Isto garante a consistência dos dados em qualquer situação na gestão. Estas características são conhecidas como *ACID (Atomicity, Consistency, Isolation, and Durability)* mas no entanto, escalar sistemas que são *ACID* já demonstrou ser um problema.
O NoSQL segue o teorema *CAP (Consistency, Availability, Partition tolerance)*:

Consistência (i.e *Consistency*): Todos os clientes vêm a mesma versão dos dados, mesmo com updates aos conjuntos de dados.

Disponibilidade (i.e *Availability*): Todos os clientes podem encontrar sempre pelo menos uma cópia dos dados requisitados, mesmo que algumas máquinas não estejam disponíveis.

Tolerância à partição (i.e *Partition tolerance*): O sistema mantém as suas características mesmo quando está a ser implementado em servidores diferentes, sendo transparente para o cliente.

O teorema como ilustrado na Figura 2.8, exige que apenas dois dos três aspetos possam ser escolhidos e de acordo com os autores Moniruzzaman e Hossain as bases de dados *NoSQL* podem ser classificadas em três tipos (Moniruzzaman and Hossain, 2013):

1. Bases de dados que guardam valores-chave (i.e *key-value*)
2. Bases de dados orientadas às colunas
3. Bases de dados baseadas em documentos.

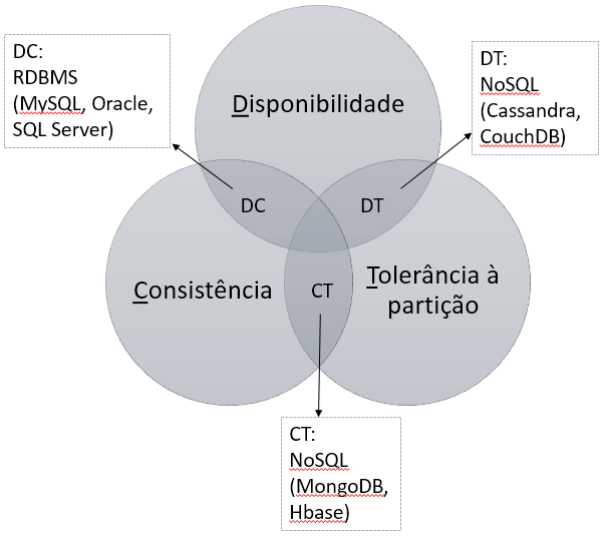

FIGURA 2.8: Teorema CAP exemplificando algumas bases de dados

A primeira, é um sistema de gestão de dados (i.e DMS) que guarda dados como identificadores alfa-numéricos (chaves) e associa valores numa simples tabela (referida como tabelas de *"hash"*). Estes valores podem ser simples strings de texto ou listas complexas. As procuras podem ser feitas apenas às chaves e não aos valores, como exemplificado na Figura 2.9.

FIGURA 2.9: Base de dados NoSQL de valor-chave, Fonte: retirado de https://biauthority.files.wordpress.com/[1].

Os autores afirmam ainda que a simplicidade das base de dados que guardam valores-chave é que são idealmente adequadas para retirar valores de forma ultra-rápida e altamente escalável, estas qualidades são necessárias para aplicações que

[1]https://biauthority.files.wordpress.com/2012/12/image024.jpg

FIGURA 2.10: Base de dados NoSQL de baseada em colunas, Fonte: retirado de https://biauthority.files.wordpress.com/[3].

fazem a gestão de perfis de utilizadores ou para recuperar os nomes dos produtos (e.g. Amazon). Um exemplo de uma destas ferramentas seria o SimpleDB[2] (Moniruzzaman and Hossain, 2013).

As bases de dados baseadas em documentos utilizam uma distribuição orientada às colunas afirmam Moniruzzaman e Hossain. Estas fornecem múltiplos atributos por chave como exemplificado na Figura 2.10(Moniruzzaman and Hossain, 2013). Enquanto algumas das base de dados orientadas às colunas têm um ADN de valor-chave, a maior parte é modelada através da *Google BigTable* (o sistema de armazenamento de dados que a Google desenvolveu).

Este tipo de sistema de gestão de dados afirmam os autores, é ótimo para guardar dados distribuídos, especialmente dados com versões devido ás funções de data e hora (Moniruzzaman and Hossain, 2013). É ótimo ainda para o processamento de dados orientados por grupos como classificação, análise, conversão. A Cassandra[4] e o HBase[5] são dois exemplos de ferramentas deste género.

O terceiro, como o seu nome indica, é desenhado para gerir e guardar documentos indicam Moniruzzaman e Hossain. Estes indicam ainda que os documentos são codificados num formato padrão de dados como XML, JSON ou BSON (Moniruzzaman and Hossain, 2013). Ao contrário da simplicidade das base de dados que

[2]http://aws.amazon.com/pt/simpledb/
[3]https://biauthority.files.wordpress.com/2013/01/image020.jpg
[4]http://planetcassandra.org/
[5]http://hbase.apache.org/

guardam valores-chave, o valor da coluna neste tipo contêm dados especificamente semiestruturados como atributos de pares nome/valor como se pode observar na Figura 2.11.

Uma coluna pode ter centenas de atributos, o número e o tipo dos atributos guardados podem variar de linha para linha. Também, ao contrário das bases de dados que guardam valores-chave, ambas as chaves e valores podem ser procuradas nas base de dados baseadas em documentos.

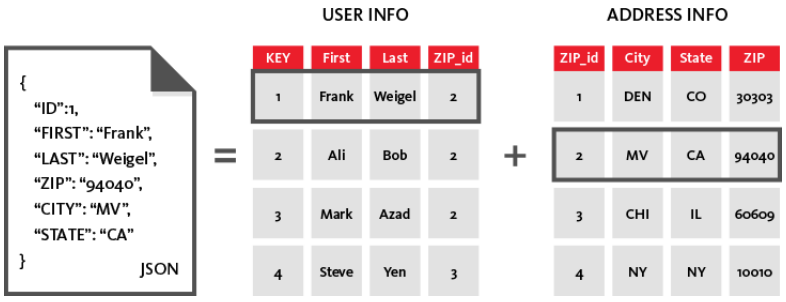

FIGURA 2.11: Base de dados NoSQL de Documentos, Fonte: retirado de http://www.couchbase.com/[6].

Este tipo de base de dados é ótimo para guardar e gerir coleções grandes de documentos de*Big Data*, como documentos de texto, mensagens de email e documentos XML. O CouchDB[7] (JSON) e o MongoDB[8] (BSON) são dois exemplos de ferramentas que se podem utilizar.

Li e Manoharan compararam as tecnologias anteriormente referidas para perceber em termos de performance (ler, escrever e apagar) qual seriam as melhores (Li and Manoharan, 2013). Estes afirmam que as base de dados *NoSQL* estão otimizadas para guardar valores-chave e que as base de dados SQL não estão otimizadas para guardar este tipo de dados.

Nas tabelas de comparação elaboradas por Li e Manoharan podemos observar o panorama geral de cada uma das tecnologias. Assim em termos de performance de leitura (Tabela II , figura 2.12) podemos observar que o *Couchbase, MongoDB, SQL Express* são os mais rápidos, em termos de performance de escrita (Tabela

[6]http://www.couchbase.com/binaries/content/gallery/website/figures/why-nosql-3.png
[7]http://couchdb.apache.org/
[8]https://www.mongodb.org/

III , figura 2.12) podemos observar que o *Couchbase, MongoDB e Cassandra* são os que se destacam, em termos de performance de apagar registos (Tabela IV , figura 2.12) podemos observar que o *Couchbase, MongoDB, SQL Express* são os que são mais rápidos (Li and Manoharan, 2013). A ultima tabela (Tabela V , figura 2.12) demonstra o tempo que cada uma das tecnologias demora a ir buscar todas as chaves. À exceção do *CouchDB* todas as bases de dados são rápidas a ir buscar todas as chaves, o *SQL Express* é o mais rápido e o *Couchbase* é excluído pois não suporta este tipo de funcionalidade.

Database	Number of operations					
	10	50	100	1000	10000	100000
MongoDB	8	14	23	138	1085	10201
RavenDB	140	351	539	4730	47459	426505
CouchDB	23	101	196	1819	19508	176098
Cassandra	115	230	354	2385	19758	228096
Hypertable	60	83	103	420	3427	63036
Couchbase	15	22	23	86	811	7244
MS SQL Express	13	23	46	277	1968	17214

TABLE II
TIME FOR READING (MS)

Database	Number of operations					
	10	50	100	1000	10000	100000
MongoDB	4	15	29	235	2115	18688
RavenDB	90	499	809	8342	87562	799409
CouchDB	71	260	597	5945	67952	705684
Cassandra	33	95	130	1061	9230	83694
Hypertable	19	63	110	1001	10324	130858
Couchbase	6	12	14	81	805	7634
MS SQL Express	11	32	57	360	3571	32741

TABLE IV
TIME FOR DELETING (MS)

Database	Number of operations					
	10	50	100	1000	10000	100000
MongoDB	61	75	84	387	2693	23354
RavenDB	570	898	1213	6939	71343	740450
CouchDB	90	374	616	6211	67216	932038
Cassandra	117	160	212	1200	9801	88197
Hypertable	55	90	184	1035	10938	114872
Couchbase	60	76	63	142	936	8492
MS SQL Express	30	94	129	1790	15588	216479

TABLE III
TIME FOR WRITING (MS)

Database	Number of keys to fetch					
	10	50	100	1000	10000	100000
MongoDB	4	4	5	19	98	702
RavenDB	101	113	115	116	136	591
CouchDB	67	196	19	173	1063	9512
Cassandra	47	50	55	76	237	709
Hypertable	3	3	3	5	25	159
MS SQL Express	4	4	4	4	11	76

TABLE V
TIME FOR FETCHING ALL KEYS (MS)

FIGURA 2.12: Tabelas de comparação retiradas de Li and Manoharan, 2013

Das bases de dados NoSQL que foram testadas pelos autores Li e Monoharan, *RavenDB* e *CouchDB* não têm uma boa performance em termos de operações de leitura, escrita e remoção (Li and Manoharan, 2013). O *Cassandra* é lento nas operações de leitura mas razoavelmente bom em operações como remoção e escrita. O *CouchDB* e o *MongoDB* são os mais rápidos no panorama geral, mas em comparação o *CouchDB* não suporta o retorno de todas as chaves.

2.1.3 Trabalho Relacionado

Nesta secção serão apresentadas e avaliadas algumas plataformas que consigam monitorizar e procurar e agregar informação dispersa na *WWW*.

2.1.3.1 Symantec Data Loss Prevention

Symantec Data Loss Prevention (Symantec, 1995-2015) é um serviço que permite a descoberta, monitorização, proteção e faz a gestão dos dados confidenciais onde quer que estes estejam armazenados e sejam utilizados (e. g. ao longo de terminais, dispositivos móveis, rede e sistemas de armazenamento). Este possui uma plataforma (Symantec Data Loss Prevention Enforce Plataform) que permite gerir as políticas de perdas de dados e *workflows*, rever e corrigir *snapshots* de incidentes, analisar e medir a redução de risco, e incluí ainda um módulo de geração de relatórios que permite a criação fácil de relatórios e *dashboards*. Analisando este serviço, existe a proteção de um dos ativos mais valiosos da empresa a propriedade intelectual. Permite ainda a proteção de ataques maliciosos de dentro da empresa. Como foi referido (InfoWatch, 2014, p. 8), mais de 73% dos ataques são feitos por pessoas de dentro das empresas. No entanto, como esta é uma ferramenta DLP não monitoriza a *WWW* à procura de dados sensíveis. Ao invés monitoriza a rede e analisa todo o tráfego que está a ser enviado (conforme as políticas) e por ventura se existir um pacote suspeito este é descartado.

2.1.3.2 BIGPICTURE 360° Data Analytics

BIGPICTURE 360° Data Analytics (Intelligence, 2015) é uma plataforma que combina de forma rápida e fácil os dados internos com informações externas valiosas (e. g. preços, estatísticas do mercado, economia, notícias, etc..). Em análise esta permite carregar facilmente e automatizar os dados internos das organizações, permitindo a criação de processos paralelos de extração e transformação de dados, com periodicidade desejada. É fornecida uma loja de *feeds* de dados, com preço de produtos, indicadores económicos, estatísticas de mercados específicos,

notícias, indicadores de páginas e *posts* do Facebook, Twitter e Youtube. Esta plataforma utiliza a tecnologia de *text mining* permitindo que não seja necessário sair desta e pode-se classificar automaticamente o texto contido em notícias, *tweets*, licitações, entre outros. Uma vez classificados, estes documentos textuais estão prontos para serem combinados com outras bases quantitativas e analisados como se fossem dados estruturados. Permite a criação de *dashboards* definindo assim o tipo de interatividade que se quer (e. g. filtros comuns, *drills* guiados, etc), sendo assim possível visualizar causas e relações permitindo assim fazer boas previsões. Permite ainda definir *triggers* para que exista o alerta através do email ou num *smartphone* quando um KPI mudar, um cliente anunciar um investimento, um *tweet* negativo ganhar repercussão, uma licitação interessante sair e muito mais. Apesar de monitorizar algumas redes socais, esta plataforma foca-se meramente na forma como as empresas estão a ser vistas e em fornecer ferramentas que ajudem a combater o mercado, e não na procura de fugas de informação e exposição de vulnerabilidades.

2.1.3.3 Cyberfeed Threat-Intelligence

O Cyberfeed Threat-Intelligence (AnubisNetworks™, 2015 & AnubisNetworks™, 2014) é um serviço que permite obter em tempo real *datafeeds* de inteligência acerca de eventos relacionados com ameaças de segurança. Este lida com um volume significativo de dados provenientes de uma variedade de vetores de ataque como e-mail, *WWW* ou *DNS*. A infraestrutura das fontes de dados do *cyberfeed* é um ambiente de *threat intelligence* que oferece uma variedade rica e complementar de diversos *feeds* de segurança, reunir dados que estão próximos da fonte, incluindo informação sobre equipamento infetado e armadilhas que atingem *botnets* ativos e que são processados em tempo real pela plataforma de Cyberfeed Threat-Intelligence. Esta plataforma oferece serviços complementares como o Cyberfeed Live Dashboard, Cyberfeed APIs, Cyberfeed Connectores e Cyberfeed Station que podem ser escolhidos pelas organizações para complementar a sua área de negócio e providenciam um contexto critico ajudando a perceber não apenas quando o ataque acontece, mas onde começou, que IP's foram afetados e a gravidade do

impacto global. Esta plataforma permite uma proteção em tempo real dos ataques que estão a ser feitos e a criação de *dashboards* relacionados com esta vertente mas não permite a procura de fugas de informação nem as vulnerabilidades expostas na *WWW*.

2.1.3.4 Giga Alert

Giga Alert (Indigo, 2015) é uma solução que ajuda a gestão da reputação, monitorização dos componentes e produz ainda ligações críticas para o negócio da organização. Esta plataforma permite a procura por tópicos personalizados, a definição da profundidade de pesquisa e ainda quantas vezes as procuras são executadas e como os resultados são entregues. Estas pesquisas podem ser feitas por domínio, formato de arquivo ou até por parte de uma página web. O *SightPoint* automaticamente aprende quais os resultados das pesquisas mais relevantes para que exista uma personalização das pesquisas. Esta permite ainda a configuração para que se possa receber emails diários em formato HTML, incluindo *links* e texto a negrito destacando os termos da pesquisa. Tal como anteriormente foi abordado, esta plataforma só permite a gestão de reputação permitindo ainda a procura por tópicos específicos presentes na *WWW* mas não permite atingir o objetivo específico da presente dissertação.

2.1.3.5 X-FORCE EXCHANGE

X-FORCE EXCHANCE[2] ou XFE, (IBM, 2015) é uma plataforma do tipo SaaS que permite a partilha de ameaças de segurança recentes e fornece informações detalhadas sobre vulnerabilidades, IP's, URL's e aplicações web. Além disto, esta fornece coleções de dados para que o utilizador procure, armazene e compartilhe essas informações agregadas. As informações das vulnerabilidades estão numa das bases de dados mais antigas sobre vulnerabilidades publicamente disponíveis da *X-Force*, contendo sensivelmente mais de 88.000 vulnerabilidades de segurança.

[2]https://exchange.xforce.ibmcloud.com/

Além das métricas padrão associadas a qualquer vulnerabilidade, o *XFE* fornece informações de cobertura da IBM, bem como referências externas baseadas no número de CVE da vulnerabilidade. Esta fornece ainda uma pontuação de risco (sendo 1 sem risco e 10 o nível de risco mais alto), localização, categorização de informações, conteúdo histórico e informações de DNS para IP's. Em relação às anteriores a *XFE* é a única é parecida ao tema abordado na presente dissertação. Esta permite a procura IP's, vulnerabilidades publicamente disponíveis, *Hashes* específicos e nomes de aplicações mas não procura fugas de informação espalhadas na *World Wide Web*.

2.1.3.6 HIBP (haveibeenpwned)

O haveibeenpwned[3] é uma plataforma que permite ao utilizador a procura sobre um email ou um nome de utilizador. Esta plataforma online procura no *Pastebin* e em outras fontes o argumento passado e verifica se existiu alguma fuga de informação alertando o utilizador se caso for positivo qual a fonte da fuga. Permite ainda a subscrição sem qualquer custo e caso exista uma fuga de informação com o email subscrito este é automaticamente notificado. Apesar desta plataforma permitir a procura de emails e agregar informação espalhada na *World Wide Web*, não é muito flexível no que se pode procurar nem indica qual a fuga de informação que existiu, informa apenas que poderá ser apenas o email, password, URL's, etc.

2.1.3.7 Conclusões e Comparações

Como conclusão desta análise, é notório a existência de algumas plataformas semelhantes mas apenas em relação a monitorização de reputação, prevenção da fuga de informação, ou mesmo a partilha de informações detalhadas sobre vulnerabilidades e reputação de IP's.

Na tabela 2.1 foram comparadas todas as plataformas anteriormente referidas tendo em conta as características descritas na secção 1.2.2 , nomeadamente:

[3]https://haveibeenpwned.com/

- Característica 1 - A plataforma permite a procura de fuga de informação, tendo como objetivo encontrar e monitorizar informação que está dispersa em várias fontes.

- Característica 2 - Permite seleção de tipo de dados a visualizar na plataforma tendo em conta as fontes selecionadas, tendo como o objetivo permitir ao utilizador um controlo na escolha do tipo de dados a visualizar.

- Característica 3 - Produz relatórios e *dashboards*, o objetivo é produzir relatórios e *dashboards* interativo de todos os dados processados na plataforma. Dentro desta existem mais algumas características:

  - Característica 3.1 - Gera gráficos e métricas evolutivas dos dados processados tendo como objetivo dar ao utilizador uma perspetiva de como estão a ser processados os dados para que exista assim uma pró-atividade por parte do utilizador.

  - Característica 3.2 - Permite a geração de relatórios ad-hoc periódicos costumizáveis tendo como objetivo fornecer ao utilizador permissão para conseguir perceber onde é que os dados apareceram durante um determinado período de tempo.

- Característica 4 - Permite a gestão de perfis de acesso, tendo como objetivo permitir a criação de áreas diferenciadas entre cada utilizador.

	C1	**C2**	**C3**	**C3.1**	**C3.2**	**C4**
Symantec DLP	N	N	S	N	N	N/D
BigPicture 360	N	N	S	S	N	N/D
Cyberfeed	N	N	S	N/D	N	N/D
Giga Alert	N	N	N	N	N	N/D
XFE	N	N	S	S	N	N/D
HIBP	S	N	N	N	N	N

Tabela 2.1: Tabela de comparação entre as várias plataformas.

Legenda: S - Sim, N - Não, N/D - Não Disponível.

Através da tabela acima apresentada conseguimos rapidamente perceber que nenhuma das soluções está focada para encontrar fugas de informação presentes na *World Wide Web*.

Até aos dias de hoje não existe uma plataforma que procure fugas de informação e correlacione os dados expostos para que se consiga fornecer uma capacidade pró-ativa em vez de reativa às organizações e utilizadores.

Capítulo 3

Desenho da Solução

Ao longo deste capitulo será abordado o desenho , a análise e implementação de todos os requisitos inerentes à plataforma.

3.1 Análise dos requisitos de alto nível

Após a análise realizada às as plataformas na secção 2.1.3 não foi encontrado uma plataforma que visa agregar e correlacionar fugas de dados que circulam na *WWW*, pelo que foram identificados os seguintes requisitos:

- Permitir o registo de utilizadores, ou seja, qualquer utilizador poderá usufruir da plataforma.
- Permitir a gestão do perfil como a alteração da palavra-chave ou exclusão da conta.
- Permitir a administração da plataforma .
- Permitir a escolha do tipo de dados e fonte de dados, esta possibilita a inserção do tipo de dados a procurar e permitir a escolha das fontes de dados mencionados na secção 2.1.1.3.

- Permitir alertas em tempo real de fugas de informação presentes nas fontes de dados.
- Exibir todos os dados processados num *dashboard* interativo com métricas associadas.
- Assegurar que a plataforma está segura contra as principais vulnerabilidades descritas no OWASP Top 10 como *XSS injection, Command Injection, DoS(Denial of Service), CVE's*, etc..
- Gerar relatórios ad-hoc periódicos e customizáveis.
- Assegurar que a plataforma está sempre disponível 24 horas por dia, 7 dias por semana.

Após a identificação dos requisitos consegue-se perceber quais as funcionalidades base para o desenvolvimento da plataforma.

3.2 Arquitetura da Plataforma

Para a arquitetura desta plataforma optou-se por definir um modelo de separação entre a camada de apresentação (*Front-End*) e a camada de acesso aos dados (*Back-End*) como se pode observar pela Figura 3.1. A razão porque foi escolhido este modelo recai na separação do que é *client-side scripting*, ou seja linguagem de cliente pois simplifica o desenvolvimento e *server-side scripting*, ou seja linguagem de servidor que simplifica a manutenção do código. A separação entre o *Front-End* e o *Back-End* tem haver com a forma de como é feita a interação com o utilizador. A primeira é responsável por todos os componentes manipulados pelo utilizador e é responsável por prepará-los de forma a que o *Back-End* os possa utilizar.

O utilizador após a autenticação na plataforma poderá escolher os termos a procurar na plataforma (fonte de dados) e o tipo de dados no *Front-End*. Através desta especificação o *Back-End* através das metodologias de *Web Scraping* ou

API's monitoriza a *World Wide Web* e sempre que encontra os termos definidos armazena-os numa base de dados para possam ser processados na geração de *dashboards* e relatórios mensais.

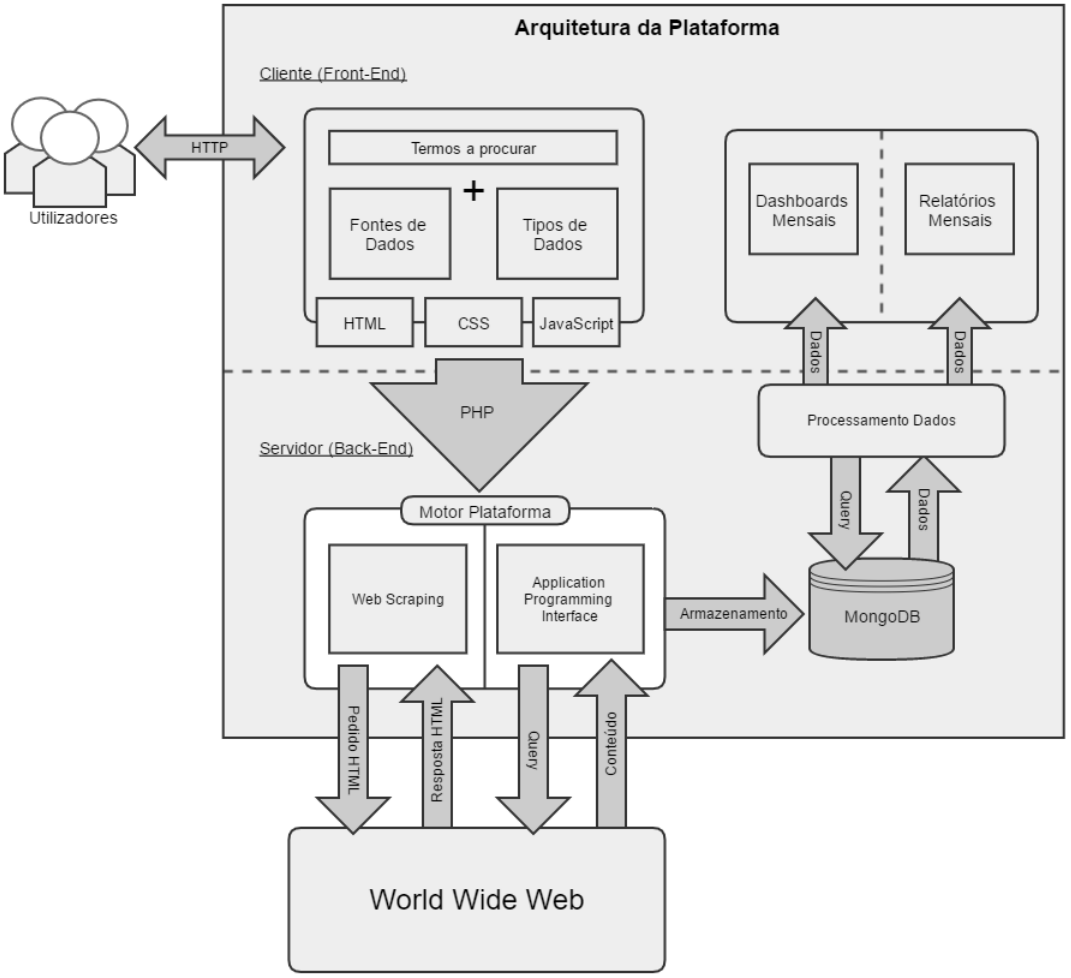

FIGURA 3.1: Arquitetura de alto nível da plataforma

3.2.1 Diagrama de Sequência e *Use Case*

Um diagrama de sequência é usado para demonstrar a interação entre os objetos e a ordem sequencial de como estas interações acontecem. O diagrama 3.2 foi criado para demonstrar a ação de autenticação de um utilizador já registado na plataforma, desta forma a GUI recebe os dados introduzidos pelo utilizador (email e password) e envia-os para o *back-end*.

O gestor responsável pela autenticação recebe os dados e trata de verificar se estes estão válidos, caso afirmativo é criado uma sessão HTML para o utilizador e este é reencaminhado para a página principal. Caso contrário o utilizador será notificado que a autenticação falha. Os restantes diagramas de interação do utilizador com a plataforma são bastante similares. O utilizador interage com a GUI, esta comunica com o seu *Handler* e de seguida este comunica com a base de dados.

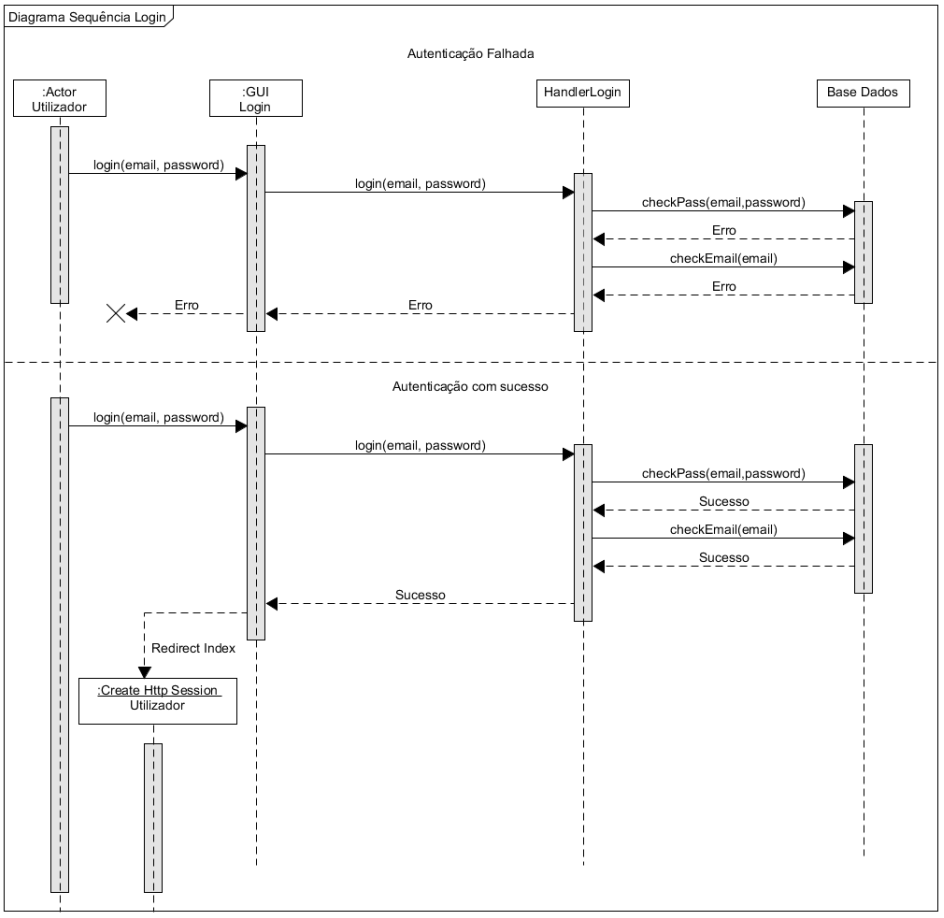

FIGURA 3.2: Diagrama de Sequência da autenticação do utilizador

Tendo em conta o detalhe anterior, a Figura 3.3 também representada em maior detalhe no Anexo A, representa todas as interações com a plataforma bem como os três papéis possíveis que um utilizador pode ter:

1. Utilizador Não Registado
2. Utilizador Registado

3. Administrador

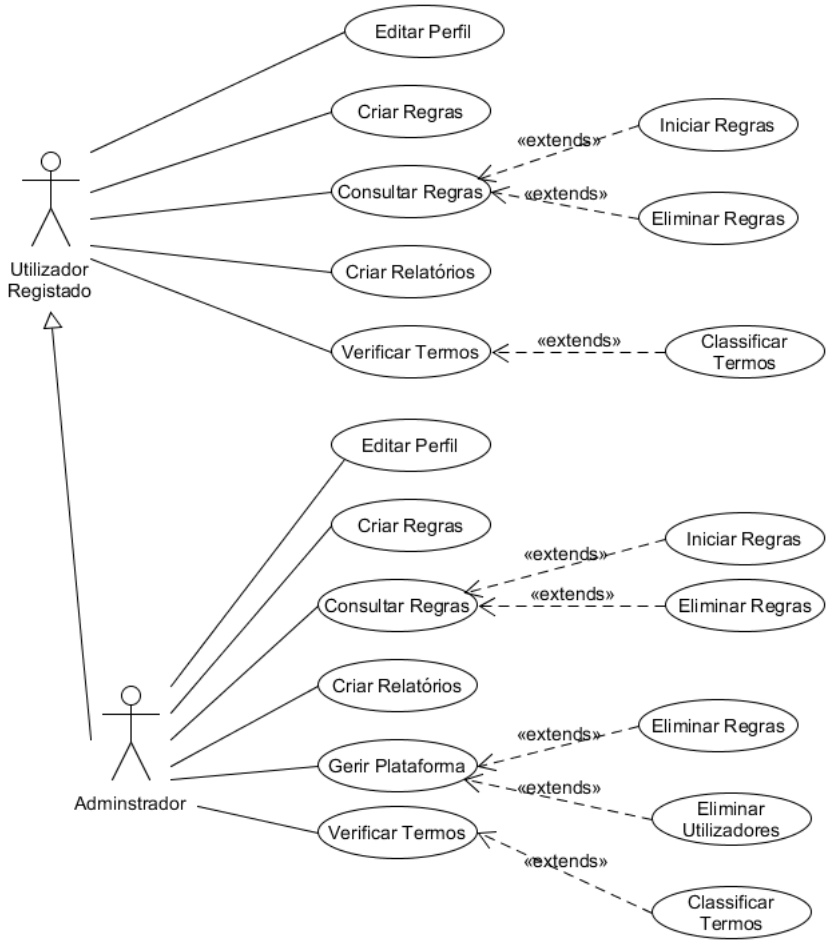

FIGURA 3.3: *Use Case* da Plataforma SDL

Um utilizador não registado (1) não pode aceder a qualquer conteúdo presente na Plataforma e como tal pode-se registar livremente para ter acesso a toda a plataforma.

Quando este se regista e confirma a conta passa a ser um utilizador registado (2) na plataforma podendo logo de seguida criar uma regra para que a plataforma inicie a busca dos termos (figura 3.3, "Criar Regras"). Depois de ter criado uma regra, o utilizador registado pode aceder às regras (figura 3.3, "Consultar Regras") e pode proceder à ativação da regra (figura 3.3, "Iniciar Regras") ou pode eliminar as regras que já não deseja (figura 3.3, "Eliminar Regras").

Um utilizador registado pode editar o seu perfil (figura 3.3, "Editar perfil").

Quando um termo é encontrado o utilizador será notificado e poderá verificar e classificar os termos como entender (figura 3.3, "Verificar Termos"e "Classificar Termos"). Poderá também ter acesso a relatórios periódicos costumizáveis (figura 3.3, "Gerar Relatórios"). O Administrador (3) tem exatamente as mesmas funcionalidades que um utilizador normal mas difere na responsabilidade. Este terá que manter a plataforma a funcionar e poderá eliminar utilizadores registados e regras que possam ser maliciosas (figura 3.3, "Eliminar Utilizadores"e "Eliminar Regras").

3.2.2 Modelo de Dados

De forma a estruturar todo o modelo de dados que serve de base para a plataforma, foi criado um diagrama que presta auxilio à implementação detalhada na secção 3.3. Como o *MongoDB* não é um sistema de gestão de base de dados relacional (RDBMS) mas sim um sistema de coleções a figura 3.4 também presente no Anexo B em maior detalhe, demonstra a estrutura proposta deste modelo de dados que representa a informação que a plataforma deverá gerir.

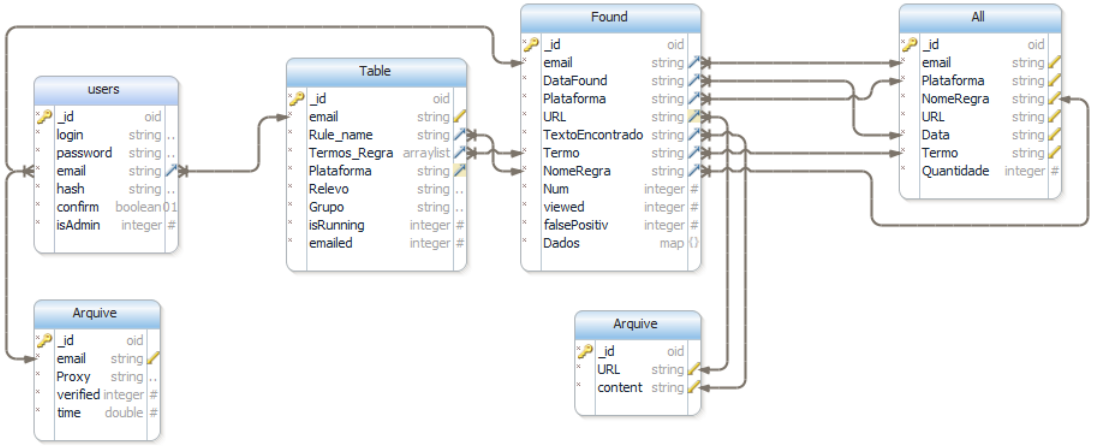

FIGURA 3.4: Modelo de base de dados implementado

Pormenorizando o diagrama acima este é composto por 6 coleções diferentes. O campo email que está presente em quase todas as coleções é um campo único e serve identificar o utilizador bem como garantir que a informação é bem guardada

e propagada. Para garantir maior segurança optou-se por separar cada coleção, assim cada uma faz uso de uma base de dados singular tento como autenticação *usernames* e *passwords* diferentes.
A coleção "*users*" serve de repositório para guardar todas as contas criadas. Cada utilizador registado pode ter várias regras a correr em simultâneo na plataforma, estas informações estão guardadas na coleção "*Table*".
Para efeitos de evitar que o acesso a uma determinada plataforma fosse proibido criou-se a coleção "*Arquives*" para guardar *proxies*, será detalhado a sua utilização na secção 3.3.1.1.
O motor da plataforma está periodicamente a descarregar informação utilizando API's especificas, nesse caso foi necessário criar a coleção "*Arquive*"para guardar essas informações.
Quando um dos termos é encontrado, a coleção "*Found*" é responsável por guardar essa informação para que seja consultada mais tarde. A coleção "*All*" serve de base para a criação dos *dashboards* necessários, contabilizando todos os termos encontrados pelo motor da plataforma.

3.2.3 Ferramentas para a criação da Plataforma

Numa primeira instância foi analisado o *XAMPP*, um ambiente que permite o uso de um servidor *HTTP* (*Apache*) e de uma base de dados *MySQL* convencional de forma a facilitar o desenvolvimento de toda a plataforma.
Inicialmente a escolha da linguagem de programação web como HTML,CSS,Javascript, foi bastante importante pois era precípuo minimizar a forma de aprendizagem. Com conhecimentos adquiridos anteriormente a decisão seria entre o PHP e o *Java*. Apesar de serem bastante parecidas entre performance e robustez, o *PHP* é uma das linguagem mais utilizadas para programação do lado do servidor na *WWW* e na qual os conhecimentos estavam mais apurados.
Após uma curta pesquisa percebeu-se que a fuga de informação é quase sempre disponibilizada em formado de texto, então surgiu a oportunidade de reavaliar qual a solução que permitisse uma melhor gestão de base de dados e introdução

de *Big Data*.

Através de uma comparação rápida de terminologias entre um sistema de gestão de base de dados relacional (i.e *RDBMS*) como é o caso do *MySQL* e um sistema *NoSQL* como é o caso do *MongoDB* ou do *CouchDB*, percebemos na tabela 3.1 que uma solução *NoSQL* rapidamente se torna uma solução fiável para este tipo de contexto pois guarda informação num formato de documentos e é otimizado para o contexto que estamos a desenhar.

RDBMS	**NoSQL**
Tabela	Colecções
Linha	Documentos (JSON,BSON)
Coluna	Campo
ID	ID
Joins	Incorporação de documentos
Chaves Estrangeiras (FK)	Referências de documentos

Tabela 3.1: Terminologia entre *MySQL* e *MongoDB*

Com o detalhe apresentado na secção 2.1.2, percebeu-se que seria uma escolha entre o *CouchDB* e o *MongoDB* pois ambos têm orientação para documentos e tolerância à partição. A tabela 3.2 demonstra a diferença entre cada um destes modelos de *Big Data*.

	CouchDB	**MongoDB**
Teorema CAP	Disponibilidade	Consistência
Configurações	Fácil	Fácil
Aprendizagem	Fácil	Fácil
Rapidez	Milhões\Segundo	Milhões\Segundo
Documentação	Muita	Muita

Tabela 3.2: Tabela de comparação entre *CouchDB* e *MongoDB*.

Analisando a tabela 3.2 o *MongoDB* foi a escolha mais acertada devido à sua inerência à consistência, ou seja, os utilizadores terão sempre acesso à mesma versão dos dados o que trás uma grande vantagem relativamente a fuga de informação.

Foi dado a preferência por um desenvolvimento de raiz, com a aplicação de um *template* web já feito para acelerar todo o processo.

3.2.4 Bibliotecas utilizadas

Para acelerar o processo de criação da plataforma foram utilizadas várias bibliotecas que prestam auxílio a áreas variadas da plataforma.

O *Highcharts*[1] é uma biblioteca *open-source* escrita em *Javascript* que permite a fácil criação de gráficos interativos. Esta biblioteca permite uma escolha variada entre 11 tipologias diferentes de gráficos e permite ainda uma grande customização possibilitando a criação de *dashboards* dinâmicos e simples.

O *Securimage*[2] é uma biblioteca *open-source* que gera imagens complexas e códigos CAPTCHA para proteger a plataforma contra *SPAM bots* e abusos por parte de utilizadores maliciosos. A inclusão desta biblioteca é simples e de fácil compreensão permitindo um aumento de dificuldade e um grande nível de customização.

O *PHPMailer*[3] é uma alternativa fiável á função *mail()* presente por omissão no PHP. Esta função têm que ter um servidor de email local para funcionar e não permite o envio de anexos nem envio de emails baseados em HTML. O *PHPMailer* utiliza uma implementação SMTP e permite o envio de emails sem ser necessário configurar um servidor de email local.

O *RIPE Stat*[4] é uma interface web que serve de base para procurar informação sobre espaços de endereçamentos, ASN's e tudo relacionado com a informação de *hostnames* e países.

Por fim o *tcpdf*[5] é uma biblioteca que permite produzir ficheiros PDF. Este permite uma fácil configuração e uma ampla variedade de codificação de texto como UTF-8 *(8-bit Unicode Transformation Format)*, XHTML, Javascript, Unicode, etc.

[1] http://www.highcharts.com/
[2] https://www.phpcaptcha.org/
[3] https://github.com/PHPMailer/PHPMailer
[4] https://stat.ripe.net
[5] http://www.tcpdf.org/

3.2.5 Front-end

O *Front-end* é responsável pela apresentação visual da plataforma bem como a interação com o utilizador. Como *template* da plataforma e de forma a conseguir re-aproveitar elementos já existentes na Internet foi optado por descarregar um modelo *open source*. As páginas apresentadas são criadas com diversas linguagens de programação, desde *HTML, CSS, JavaScript* entre outras. Após algumas pesquisas foi possível encontrar um *template* que tivesse um *design* apelativo e que pudesse agregar muita informação, tento apenas só uma página de exemplo e alguns elementos pré-configurados. Este tipo de *template* ajuda o programador a iniciar o seu projeto sem ter que criar tudo de raiz.
A informação presente em cada página é preparada para ser entregue ao *Back-end*, a figura 3.5 demonstra quais as páginas implementadas.

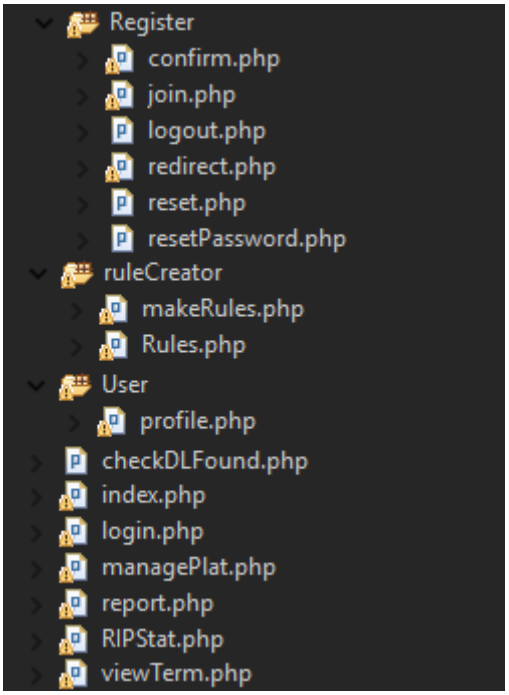

FIGURA 3.5: Páginas de *Front-end* da plataforma

Explicando sucintamente cada uma das páginas:

- Pasta de Registo:
  - **Join** : Página responsável pelo registo de utilizadores que querem procurar informação dispersa na *WWW*;
  - **Confirm** : Página responsável por informar o utilizador o estado do seu registo;
  - **Redirect** : Após o registo, o utilizador é redirecionado para a página de Login;
  - **Logout** : Página de saída da plataforma;
  - **reset** : Página de *reset* da password do utilizador;
  - **resetPassword** : Página para alterar a password do utilizador.
- Pasta de criação de regras:
  - **MakeRules** : Nesta página o utilizador define regras com os termos que está à procura numa determinada plataforma (i.e. Pastebin e Shodan) bem como o tipo de dados associado;
  - **Rules** : Após a criação das regras, nesta página o utilizador pode consultar e eliminar regras criadas bem como começar a busca nas plataformas.
- Pasta de utilizador:
  - **Profile** : Área de perfil do utilizador onde este pode eliminar a sua conta bem como alterar a sua password.
- **Login** : Página inicial da plataforma onde os utilizadores podem autenticar-se bem como registar-se;
- **Index** : Após o **Login** esta página é onde os utilizadores podem consultar os *Dashboards* e navegar através da plataforma;
- **ManagePlatform** : Página de administração da plataforma;

- **CheckDLFound** : Página de verificação dos termos que foram encontrados;
- **ViewTerm** : Página para verificar o conteúdo os termos encontrados;
- **Report** : Página de geração de relatórios periódicos e customizáveis;

3.2.6 Back-end

O *Back-end* é responsável por receber a informação enviada pelo *Front-end*, processa-la e devolve-la de volta para o *Front-end* assim, este é um fluxo normal entre ambos. Na figura 3.6 podemos observar todos os *scripts* que gerem a interação com as diferentes coleções da base de dados.

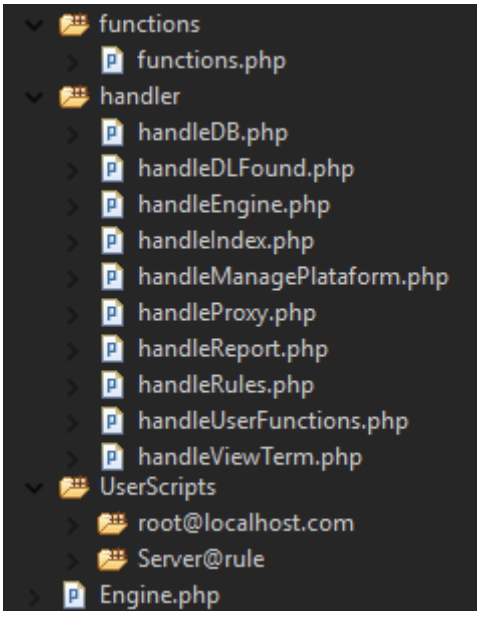

FIGURA 3.6: Páginas de *Back-end* da plataforma

Para uma melhor gestão do conteúdo foram criadas várias pastas e assim podemos ver como a a plataforma está visualmente organizada:

- Pasta de "Funções":
  - **Functions** : Responsável pelas funções mais simples que servem todas as páginas;

- Pasta de "Gestão":
  - **HandleDB** : Módulo responsável pela conexão com todas as coleções da base de dados;
  - **HandleDataLeakageFound** : Responsável por carregar todas as fungas de informações encontradas definidas pelo utilizador;
  - **HandleIndex** : Responsável por carregar toda a informação necessária para a criação dos *dashboards* dinâmicos;
  - **HandleManagePlat** : Responsável por carregar todas as funções de administração da plataforma bem como a promoção de um utilizador a administrador, remoção de contas, gestão de URL's, gestão de *proxies*, etc;
  - **HandleEngine** : Módulo responsável por todas as funções necessárias para o bom funcionamento do motor da plataforma na procura dos termos;
  - **HandleLogin** : Módulo responsável por verificar se as credenciais que o utilizador insere estão válidas;
  - **HandleProxy** : Responsável pelas funções que verificam o estado de um proxy permite a inserção de um *proxy* para ser utilizado;
  - **HandleReport** : Responsável por carregar todas as informações necessárias para a criação dos relatórios customizáveis;
  - **HandleRules** : Este módulo é responsável por carregar todas as regras definidas pelo utilizador;
  - **HandleUserFunctions** : Módulo responsável pela eliminação da conta do utilizador, bem como a redefinição de uma nova password.
  - **HandleViewTerms** : Módulo responsável por carregar o conteúdo dos termos encontrados;
  - Pasta de "Scripts":
    - **Server@Rule** : Pasta que contêm os *scripts* do servidor que correm automaticamente;

– **Engine** : Motor para a procura dos termos definidos pelo utilizador.

3.3 Implementação

Para solucionar o problema da fuga de informação presente na *WWW* e validar esta tese, foi desenvolvida um protótipo de uma plataforma que possibilita a agregação de informação bem como a criação de automatismos na procura de um determinado termo. Após análise na secção 2.1.1.3 denotou-se que o *Pastebin*[6] e o *Shodan*[7] eram as fontes mais fiáveis para fugas de informação e foi analisada a sua estrutura.

3.3.1 Web Scraping

Uma das técnicas utilizadas para retirar informação relevante presente em ambas as plataformas foi a utilização da técnica de *Web Scraping* descrita na secção 2.1.1.1.

Inicialmente e antes de começar desenhar a plataforma foi necessário perceber como é que estas plataformas funcionavam e qual o processo por detrás da indexação de informação.

No caso do *Pastebin*, qualquer informação colocada na plataforma é denominada de *paste* e é automaticamente inserida na página de arquivo[8] com um *URL* único, como tal é necessário termos acesso a todos estes.

Conseguimos analisar o código fonte da página representado na figura 3.7 através da combinação das teclas *CTRL+SHIFT+I* e assim perceber que os *URL*'s são dispostos numa tabela. Em *HTML* as tabelas são dispostas de linhas com sintaxe *"TR (ou seja Table Row)"* e colunas com a sintaxe " *"TD (ou seja Table Data)"*.

[6] http://pastebin.com/
[7] https://www.shodan.io/
[8] http://pastebin.com/archives

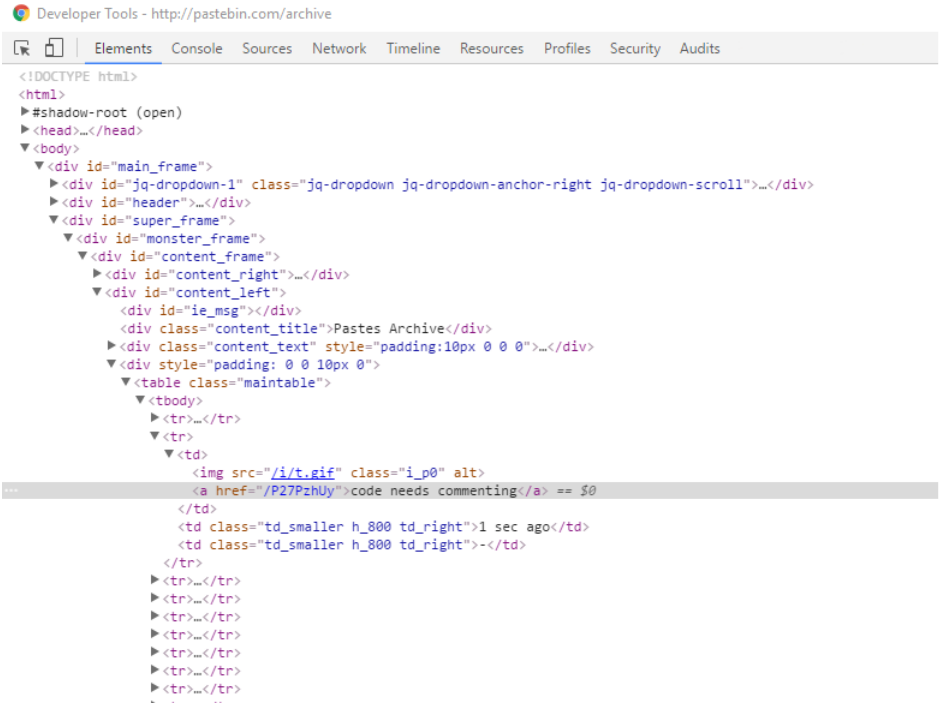

FIGURA 3.7: Código Fonte - *Pastebin.com/arquives*

Por intermédio do código 3.1 conseguimos obter os *hyperlinks* e assim dar início à extração do conteúdo. Nesta última fase o processo é idêntico ao explicado anteriormente com a nuance de que o elemento do código fonte é ligeiramente diferente. Neste caso o conteúdo está presente dentro de uma divisão com um identificador próprio e deste modo a *query* à árvore *DOM* altera-se ligeiramente para *"//div[contains(@id,"selectable")]"*.

```
1  $ch = curl_init();
2  curl_setopt($ch, CURLOPT_URL, 'http://pastebin.com/archive');
3  $page = curl_exec($ch);
4  $xmlPageDom = new DOMDocument ();
5  @$xmlPageDom->loadHTML($page);
6  $xmlPageXPath = new DOMXPath($xmlPageDom);
7  $queryXML = $xmlPageXPath->query('//tr/td/a/@href');
8      for($i = 0; $i < $queryXML->length ; $i++){
9       $URL='http://pastebin.com/'.$queryXML->item($i)->nodeValue;
10     }
```

LISTA DE CÓDIGOS 3.1: Extração de URL's

No caso do *Shodan* o processo é ligeiramente diferente pois este não é uma plataforma de partilha de informação mas sim um motor de busca. Assim, através de uma simples *query*[9] conseguimos procurar a informação que desejamos nesta plataforma.

Após análise o *Shodan* limita a procura apenas à primeira página o que pode ser um problema se existir muita informação.

Outro problema que foi encontrado quando se estava a fazer *web scraping* é o número de pedidos feitos aos servidores destas plataformas. Quando o número de pedidos é muito elevado as plataformas têm um mecanismo de proteção contra ataques de negação de serviço ou *DoS* e recusam os mesmos. Num ataque *DoS* existe a tentativa de interromper o serviço prestado aos utilizadores e como forma de prevenção e mitigação o *Pastebin* e o *Shodan* excluem o acesso durante algum tempo.

3.3.1.1 Utilização de *Web Proxies* e *Web API*'s

Como forma de mitigar o problema acima anunciado foram identificadas duas formas de contornar os bloqueios feitos pelas duas plataformas, a utilização de um *web proxy* ou a utilização de API's específicas (como as API's do Pastebin e do Shodan). Um *proxy* ou servidor *proxy* é utilizado como intermediário para obter recursos de outros servidores a pedido dos clientes.

Um *web proxy* segue o mesmo principio e obtém pedidos de páginas web se for requisitado por um utilizador. A figura 3.8 demonstra como são feitos estes pedidos. Para utilizar o *web proxy* a plataforma conecta-se ao servidor e requisita uma página do *Pastebin* (1), de seguida o servidor *proxy* comunica com o servidor do *Pastebin*(2) e obtêm a página requisitada(3) se a comunicação anterior for feita com sucesso o servidor *proxy* guarda-a em memória e entrega-a à plataforma que requisitou (4).

[9]https://www.shodan.io/search?query=

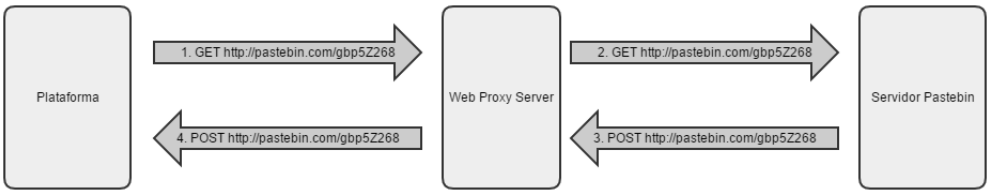

FIGURA 3.8: Pedido ao *web proxy* de uma página do *Pastebin*

Com este método podemos ter uma lista com vários servidores de *web proxies* e ir alternando para que não exista a interrupção da comunicação com o servidor da plataforma e assim recolher toda a informação que necessitamos. Outra análise feita foi escolher qual a origem do servidor de *proxy*, já que um servidor de *web proxy* pode estar em qualquer parte do mundo. Após uma simples análise, optou-se por servidores dos Estados Unidos da América e Canadá já que o *Pastebin* e o *Shodan* estão hospedados em domínios dos Estados Unidos e assim não existe o problema da utilização de um *proxy* bloqueado.

A outra forma de mitigar os problemas descritos é a utilização de *API*'s. Estas são simplesmente funcionalidades que permitem a interação com as interfaces das aplicações/plataformas sem ser necessário perceber o seu código fonte e providenciam a documentação. Normalmente a utilização destes serviços têm associada uma chave *API* que serve de controlo para a forma de como a interface é utilizada e para prevenir o abuso por utilizadores maliciosos.

No caso do *Shodan* e do *Pastebin* o tipo de *web API*'s utilizadas são do lado do servidor. Estes têm interfaces públicas que permitem um sistema de pedidos HTTP que devolvem os resultados em formato *JSON*. Desta forma não existe o bloqueio por parte das plataformas já que a utilização do *API* é feita de forma controlada pelo servidor.

3.3.2 Criação de Automatismos

Anteriormente foi descrito a abordagem para obter os conteúdos presentes nas plataformas. Nesta secção será apresentada a forma como se automatizou a procura (através de *scripts*) e a categorização dos dados (através de expressões regulares).

3.3.2.1 Expressões Regulares

Neste trabalho optou-se pela utilização de uma expressão regular que permitisse a classificação automática de dados encontrados no *Pastebin*. Uma expressão regular (ou *regex* abreviado de *regular expression*) é utilizada para identificar cadeias de caracteres como padrões, palavras ou caracteres em particular. Estas expressões são escritas numa linguagem formal para que um processador as possa interpretar, examinar e devolver se o padrão foi encontrado.

Neste caso foram analisados vários *dumps* de dados e notou-se que existe padrão muito específico no que toca a fuga de informação. Usualmente o padrão têm um email (p.ex user@email.com) com o caractere ':' (dois pontos) seguido da password do utilizador, sendo que no final o padrão tem a forma de *"user@email.com:password"* e através deste foi criado uma expressão regular (ver lista 3.2) para ser utilizada através da função *preg_match* do PHP.

```
/[a-zA-Z0-9_]+([.][a-zA-Z0-9_]+)*[@][a-zA-Z0-9_]+([.][a-zA-Z0
    -9_]+)*[.][a-zA-Z]+[:][^~,]+$/
```

Lista de Códigos 3.2: *Regex* para encontrar fugas de dados

Decompondo a expressão regular em várias partes:

- **[a-zA-Z0-9_]**+ - Combinação de vários padrões:
  - **a-z** - Um caractere único entre a e z (*case sensitive*);
  - **A-Z** - Um caractere único entre A e Z (*case sensitive*) ;
  - **0-9** - Um caractere único entre 0 e 9;

  - **_** - Exatamente o caractere '_';
  - **+** - Entre uma ou várias vezes todos os caracteres que estão para trás.

- **[.]** - Exatamente o caractere '.';
- **[@]** - Exatamente o caractere '@';
- **[:]** - Exatamente o caractere ':';
- ***** - Entre zero ou várias vezes todos os caracteres que estão para trás;
- **[^~,]** - Qualquer caractere existente;
- **$** - Colocar a posição no final da cadeia de caracteres.

A plataforma encontra fugas de dados através de uma comparação simples de texto mas existe sempre a possibilidade de estes serem falsos positivos. Um falso positivo é quando existe a presença do termo mas essa presença não reflete a fuga de dados. Com a utilização das expressão regulares a probabilidade de um falso positivo acontecer torna-se menor mas isto não quer dizer que não se arquive pois o formato pode ser diferente da expressão que se define.

3.3.2.2 Criação de *Batch/Visual Basic Scripts* e tarefas autónomas

Um *script* é um conjunto de linhas de código que permitem a execução de comandos sem que seja necessário a sua compilação. Através destes *scripts* podemos criar automatismos para auxiliar a plataforma e criar tarefas autónomas sem que exista grande esforço por parte do utilizador.

A figura 3.9 demonstra como todo o processo está montado para que seja possível criar as tarefas autónomas.

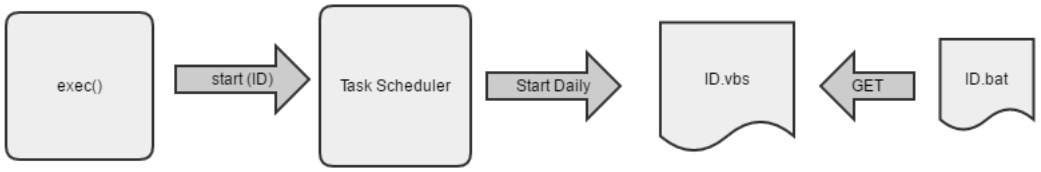

FIGURA 3.9: Diagrama de programação de tarefas

Em primeira instância é criado um ficheiro *batch* como é observado no código 3.3. O nome do ficheiro .bat (descrito como *$filename* no código) é o *ID* que é atribuído automaticamente pelo *MongoDB*, e este identificador é único para cada regra.

De seguida a variável *$cmd* é criada com o intuito de poder criar a instância de PHP para que a regra possa correr. É necessário alterar o caminho de destino para a pasta de *UserScripts* (através do comando *cd*) e a partir daí iniciar o processo *PHP.exe* com o ficheiro que contêm o motor da plataforma (opção -f) e o identificador que lhe foi atribuído (opção -I).

Por fim o conteúdo da variável *$cmd* é guardada dentro do ficheiro *.bat* através do comando de escrita em ficheiros *fwrite*.

```
1 $filename = "..\SDLPlatform\UserScripts\".$_SESSION["email"]."\".
      getIDbyRuleName($Nome_Regra,$_SESSION["email"]).'.bat';
2
3 $cmd = 'cd ..\SDLPlatform\UserScripts\'.$_SESSION["email"].'\
4 "..\php.exe" -f "..\SDLPlatform\Engine.php" -- -I='.
      getIDbyRuleName($Nome_Regra,$_SESSION["email"]).'';
5
6 $fop = fopen($filename, 'a+');
7 fwrite($fop, $cmd."\r\n" );
8 fclose($fop);
```

LISTA DE CÓDIGOS 3.3: Criação do *batch script*

Com este código a plataforma consegue instanciar o motor da plataforma mas para criar regras autónomas necessitamos que estas corram automaticamente.

No caso de sistemas Microsoft conseguimos ter acesso ao programador de tarefas (*task scheduler*) através do comando `%windir%\system32\taskschd.msc /s`.

No caso de sistemas Linux conseguimos ter o mesmo acesso através do *crontab*. Como toda a solução foi desenvolvida em Windows, optou-se pela configuração do programador de tarefas e após análise denotou-se que este permite iniciar programas automaticamente indicados através de ficheiros.

O código 3.4 permite a criação de um *visual basic script* que corre o ficheiro *batch* criado no código acima. Decompondo este código, o *WinScriptHost* corre um programa *batch* que definimos no código 3.3 como um novo processo.

```
1 $vsScript ='Set WinScriptHost = CreateObject("WScript.Shell")
2         WinScriptHost.Run Chr(34) & "'.$filename.'" & Chr(34), 0
3         Set WinScriptHost = Nothing';
4
5 $filenameVBS = "..\SDLPlatform\UserScripts\".$_SESSION["email"]."\
    ".getIDbyRuleName($Nome_Regra,$_SESSION["email"]).'.vbs';
6
7 $fo = fopen($filenameVBS, 'a+');
8 fwrite($fo, $vsScript );
9 fclose($fo);
```

Lista de Códigos 3.4: Criação do *Visual Basic Script*

O PHP tem uma biblioteca nativa chamada *"exec"*, esta permite executar um programa externo. O código 3.5 permite invocar a linha de comandos com o programador de tarefas e criar a automatização necessária para que o motor possa correr dependendo apenas de um único identificador.

```
1 exec("%windi\%\system32\cmd.exe /c Schtasks /create /tn ".$_POST["
    playRule"]." /tr ..\SDLPlatform\UserScripts\\".getUserSpaceByID
    ($_POST["playRule"])."\".$_POST["playRule"].".vbs /sc DAILY /mo
     1 /RI 10 /DU 24:00",null);
```

Lista de Códigos 3.5: Execução do código para o Programador de Tarefas

3.3.3 *Hardening* de Configurações

A nível de segurança da plataforma e de forma a acomodar e garantir que não existem intrusões foram seguidas várias abordagens. A primeira foi garantir que cada coleção do *MongoDB* tinha uma password diferente de autenticação e ativar a autenticação num porto específico através do comando *–auth -port 27017*. Assim, só sabendo as passwords das coleções é que o servidor consegue escrever e ler das bases de dados. Para prevenir ataques por vetor do tipo XSS foi utilizada em toda a plataforma a função de PHP *htmlentities*[10] que transforma todas as *tags* HTML em caracteres.

Por exemplo, a *tag* <B>exemplo<\B> será convertida em <b>exemplo<\b>

Para prevenir ataques por vetor do tipo CRSF foi utilizado a geração de *tokens* quando existe a submissão de dados por parte do utilizador.

Como foi utilizado o *XAMPP* (referido na secção 3.2.3) foi necessário reconfigurar alguns ficheiros. Para esconder a versão do apache quando existe um erro foi necessário reconfigurar o *httpd-default.conf* através do código *ServerTokens ProductOnly* e *ServerSignature Off*.

Para esconder as páginas por defeito que estavam expostas aos utilizadores foi necessário a configuração do ficheiro *httpd.conf* e a inclusão de várias linhas de códigos. Uma das páginas expostos que dá o estado do servidor é a página *server-status* e através do código 3.6 bloqueamos o acesso a todos os utilizadores exceto ao servidor local.

```
1 ExtendedStatus on
2 <Location /server-status>
3  SetHandler server-status
4  Order deny,allow
5  Deny from all
6  Allow from 127.0.0.0/255.0.0.0 ::1/128
7 </Location>
```

Lista de Códigos 3.6: Código para bloquear a página *server-status*

[10]http://php.net/manual/en/function.htmlentities.php

Outra configuração feita foi alterar o encaminhamento de *HTTP* para *HTTPS* que providencia uma camada de segurança sobre o protocolo *HTTP*. Para conseguir este tipo de configurações gerámos um certificado *SSL* para o servidor. Como foi utilizado *HTTPS*, alterámos as *cookies* de sessão para *Secure*. Por fim alterámos as *cookies* de sessão para acomodar *HttpOnly*.

Se as *cookies* de sessão não estiverem como *HttpOnly* um atacante pode aceder às *cookies* e utilizar um ataque XSS para roubar uma sessão ativa do utilizador. Para tal bastou incluir duas linhas de código (*session.cookie_ httponly=True* e *session.cookie_ secure=True.*) no ficheiro de configuração *php.ini*.

3.3.4 Funcionamento da Plataforma

Nesta secção iremos abordar todo o funcionamento bem como o detalhe de cada página presente na plataforma *Seek Data Leakage*. Para uma melhor compreensão criou-se um mapa ilustrativo de toda a plataforma (figura 3.10) que demonstra a divisão feita e qual a hierarquia da mesma.

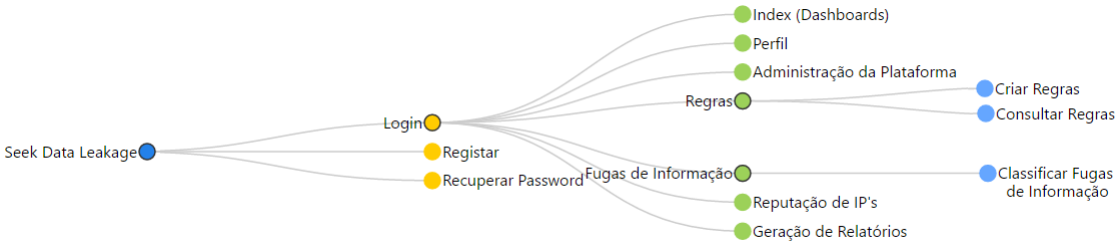

FIGURA 3.10: Mapa da estrutura da plataforma

A página inicial da plataforma será a página de autenticação pelo que se pode observar na figura 3.11.

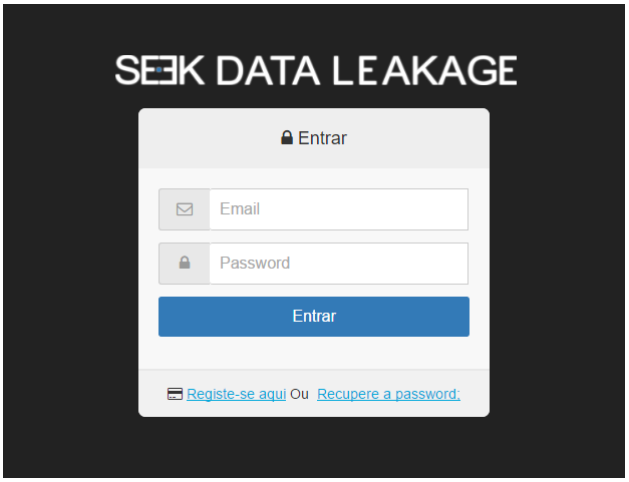

FIGURA 3.11: Página de Autenticação

Antes da autenticação qualquer utilizador que aceda à plataforma têm que se registar. A figura 3.12 exibe todos os campos (Nome, Email, Password) acessíveis para o registo alertando o utilizador da qualidade da password que está a introduzir bem como dois mecanismos de prevenção contra utilizadores maliciosos e *bots*.

O primeiro mecanismo referido na secção 3.2.4, pede ao utilizador para inserir o código *"Captcha"* permitindo ainda caso o utilizador não consiga perceber o código uma maneira de gerar outro. A segunda é um caixa de confirmação para garantir que o utilizador leu os termos e condições presentes na plataforma.

Após o registo, o utilizador irá receber na sua caixa de correio um email de confirmação para que o seu registo fique completo e só após esta confirmação é que o utilizador pode usufruir da plataforma.

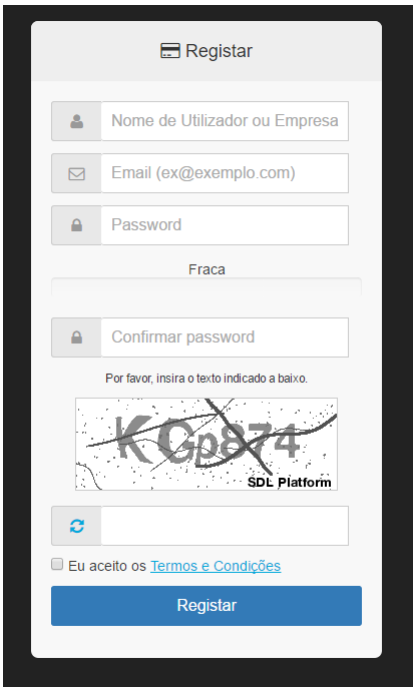

FIGURA 3.12: Página de Registo na Plataforma

Caso o utilizador se esquecer da password que inseriu na plataforma pode através da página representada na figura 3.13 inserir o email de registo. Automaticamente receberá um email na caixa de correio que contêm um URL para a página de *reset* da password (apresentada na figura 3.14.

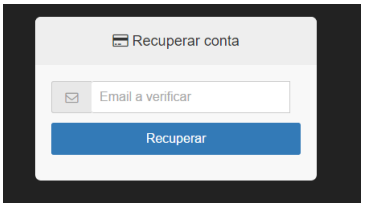

FIGURA 3.13: Página de recuperação da password

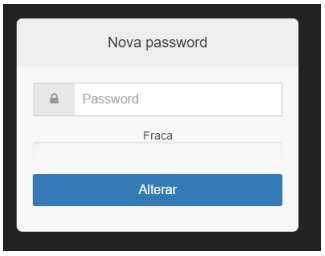

FIGURA 3.14: Página de inserção da nova password

Após a autenticação, o utilizador é redirecionado para a página principal (apresentada na figura 3.15). Nesta primeira área o utilizador pode interagir com o filtro para criar os *dashboards* costumizáveis. Com este filtro o utilizador tem a capacidade de escolher o tipo de plataforma (*Pastebin*, *Shodan*), a regra e o termo (desde que estes tenham sido encontrados pela plataforma).

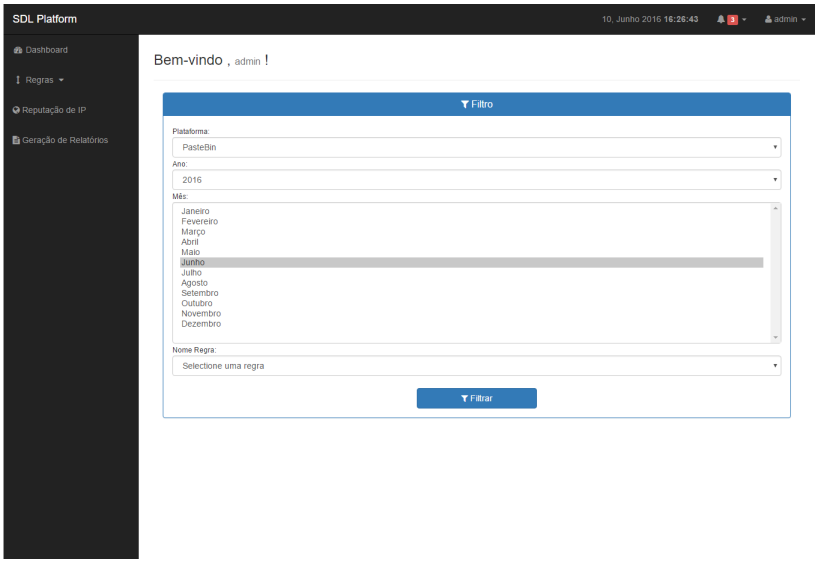

FIGURA 3.15: Página inicial da Plataforma

Quando o utilizador filtra a plataforma devolve 3 tipos de *dashboards* como se pode observar na figura 3.16. O primeiro à esquerda devolve a percentagem das regras com termos encontrados, o segundo à direita representa o *dashboard* mensal e devolve o total de fugas de informação por dia (total, falsos positivos e negativos). O último em baixo representa o *dashboard* anual é dá um panorama global do total de fugas de informação por mês (total, falsos positivos e negativos).

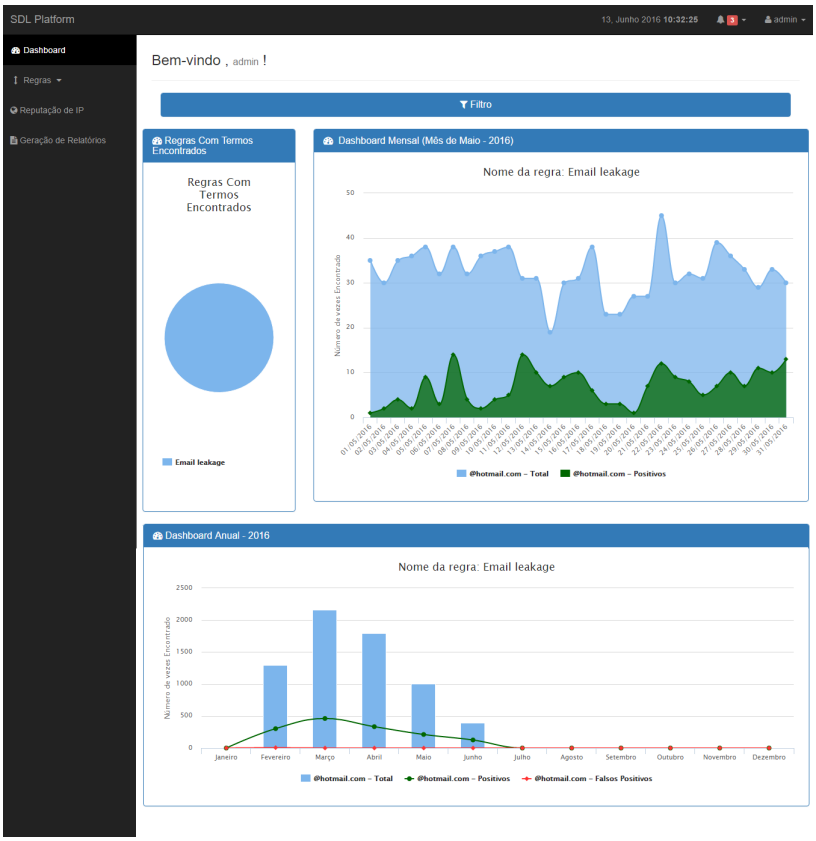

FIGURA 3.16: Página inicial da plataforma com *dashboards*

Uma das áreas a que todos os utilizadores registados da plataforma têm acesso é ao seu perfil (figura 3.17). No perfil é onde o utilizador pode fazer a gestão da password (eventualmente alterar a password se assim desejar) ou eliminar a sua conta.

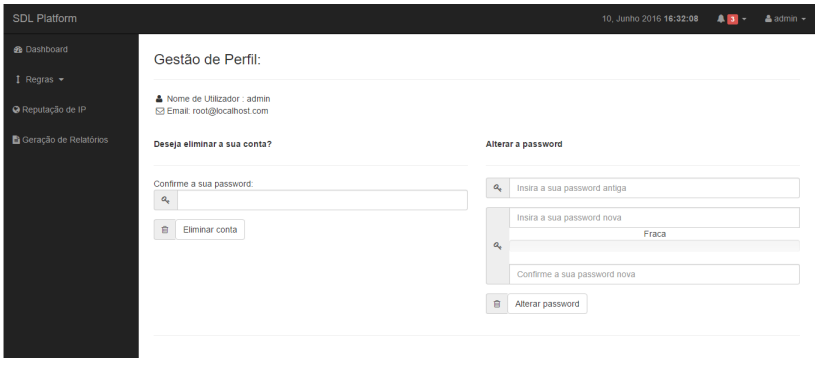

FIGURA 3.17: Página de perfil do utilizador

A figura 3.18 representa a página de administração da plataforma com todas as sub-áreas. Desta forma existe a separação entre cada área e torna-se mais simples verificar cada parte da plataforma permitindo ao administrador alguma flexibilidade de escolha.

Como se pode observar cada uma das sub-áreas são:

- Listagem de Utilizadores - o administrador pode verificar qual o estado de cada utilizador e eliminá-lo da plataforma ou promove-lo a administrador.
- Gestão de Regras - o administrador pode verificar todas as regras criadas de modo a controlar abusos por parte de outros utilizadores, este tem a possibilidade de parar ou eliminar regras existentes.
- Gestão de *URLS* - o administrador pode verificar se a plataforma está a descarregar bem os *URLS* da plataforma *Pastebin*.
- Gestão de *Proxies* - o administrador pode verificar o sinal das *proxies* existentes bem como eliminar ou adicionar *proxies* novas para serem utilizadas pela plataforma.

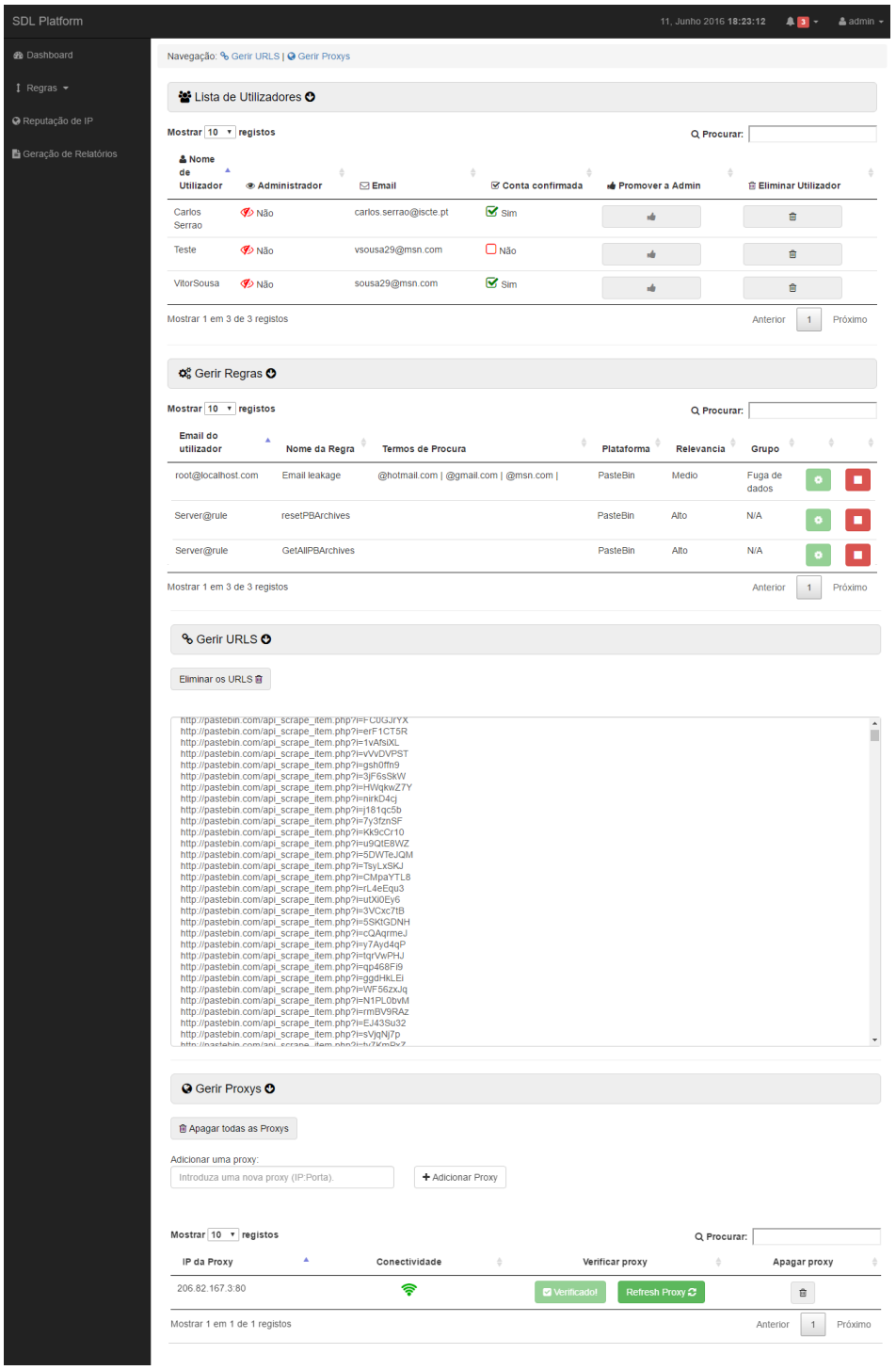

FIGURA 3.18: Página de administração da plataforma

Do lado esquerdo da plataforma temos a parte de navegação onde podemos aceder aos *dashboards*, às regras, à reputação de IP e à geração de relatórios. Para criar regras, qualquer utilizador pode aceder à área presente na figura 3.19. O utilizador tem que definir uma nome para a regra (único) com um ou vários

termos associados e a plataforma que desejam.
Dependo do tipo de fuga de informação, dá-se a opção para definir a periodicidade que o utilizador deseja entre:

- Baixa que procura informação na fonte escolhida de 15 em 15 minutos (notificação é feita na plataforma)
- Média que procura informação na fonte escolhida de 10 em 10 minutos (notificação é feita na plataforma)
- Alta que procura informação na fonte escolhida de 5 em 5 minutos (notificação é feita na plataforma)

Foi incluído ainda uma opção para que o utilizador seja informado na sua caixa de correio mal existam fugas de informação.

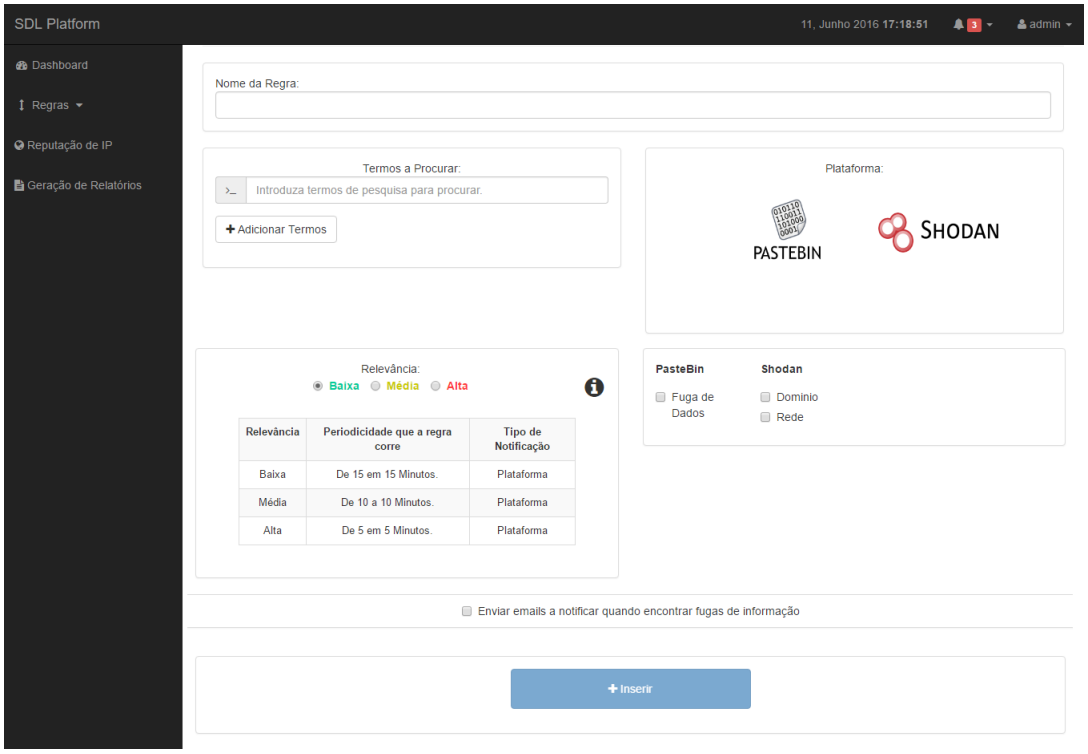

FIGURA 3.19: Página de criação de regras

Quando o utilizador cria uma nova regra é automaticamente redirecionado para a página de consulta de regras (figura 3.20). Nesta página o utilizador pode dar inicio à procura de informação, parar a procura ou eliminá-la.

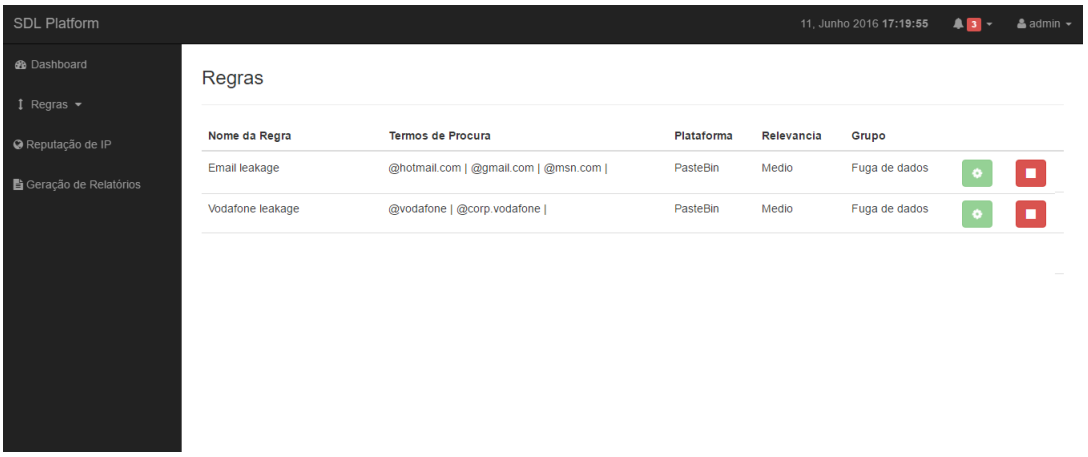

FIGURA 3.20: Página de consulta de regras criadas

Quando algum dos termos é encontrado existe uma notificação presente no topo da plataforma e os utilizadores podem aceder à área de fuga de informação. Nesta área (figura 3.21) o utilizador pode filtrar por plataforma, ano, mês, regra e termo para que seja mais simples obter todas as fugas de informação e assim aceder a cada termo com os dados encontrados.

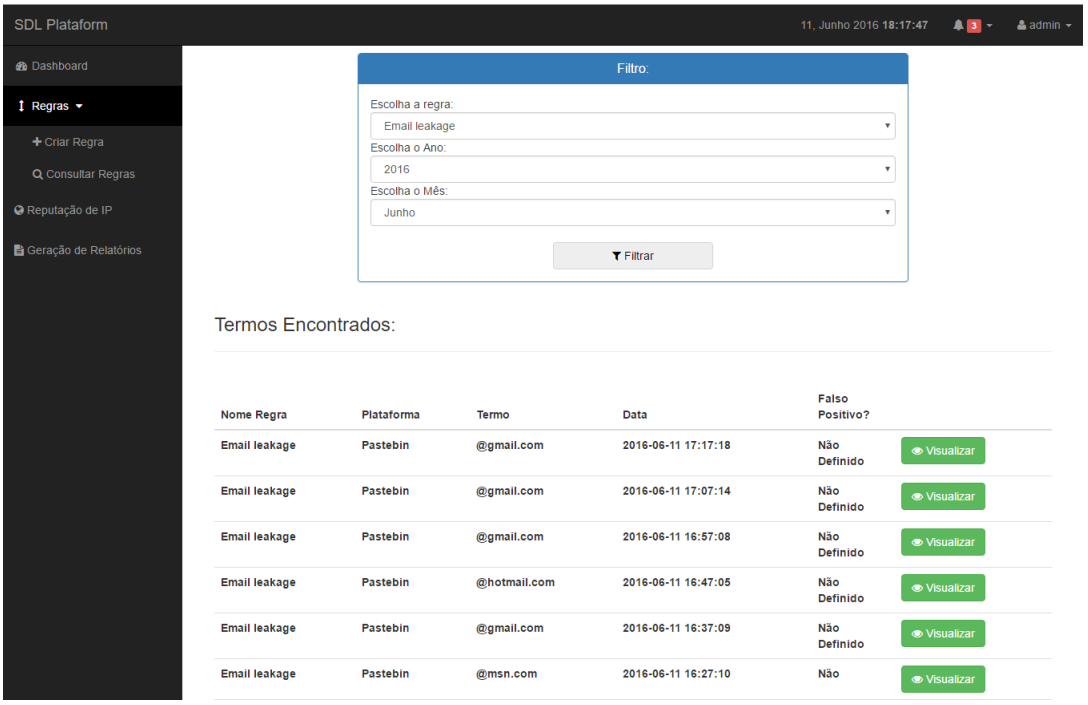

FIGURA 3.21: Página de consulta de fugas de informação

Quando o utilizador carrega em visualizar, será redirecionado para a página referente ao termo que escolheu (figura 3.22). Nesta área o utilizador poderá classificá-lo (sendo falso positivo ou não), verificar qual o texto encontrado na plataforma e qual o *URL* onde esteve presente.

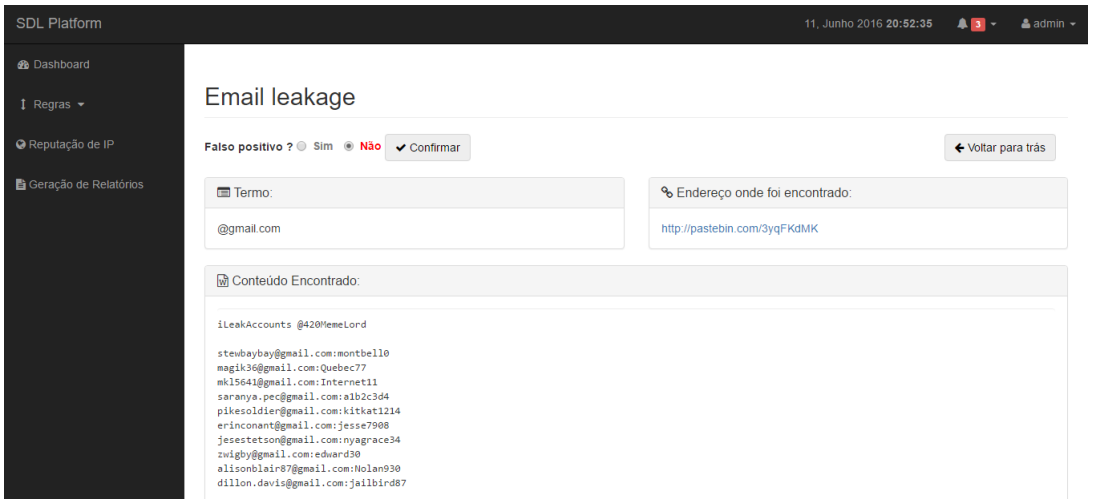

FIGURA 3.22: Exemplo de uma página de informação do termo encontrado

Ainda presente no lado esquerdo temos a reputação de IP (explicada na secção 3.2.4) que serve de auxílio à análise de IP's e ASN's (como se pode observar na figura 3.23). O utilizador pode inserir o seu próprio endereço IP ou um IP/ASN conhecido e esta página irá devolver informação relevante de tudo o que é conhecido.

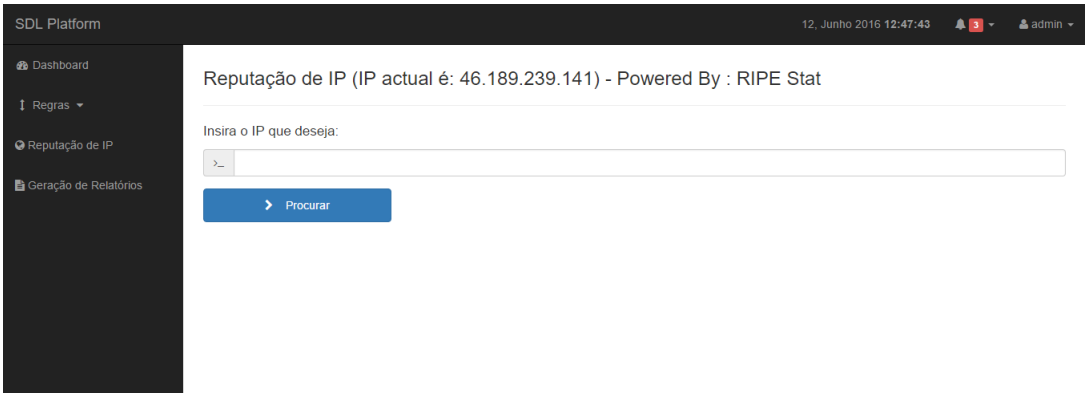

FIGURA 3.23: Exemplo de uma página de informação do termo encontrado

Por fim temos a página de geração de relatórios costumizáveis (representada na figura 3.24). Nesta área o utilizador pode escolher uma imagem para ser introduzida no relatório através de *Upload* para a plataforma, o Ano, o Mês e as regras que o utilizador quer incluir para a pré-visualização do relatório.

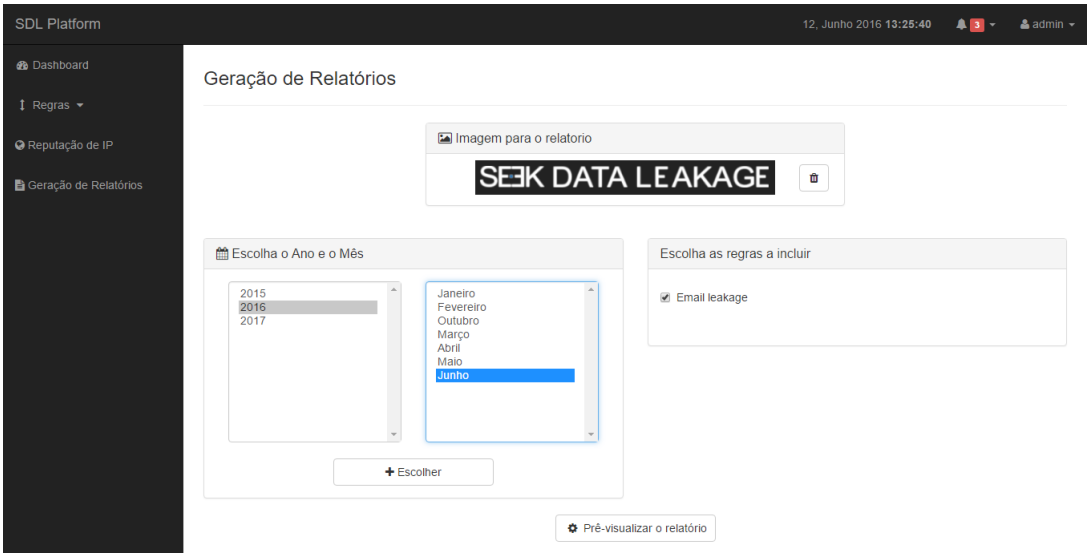

FIGURA 3.24: Página de geração de relatório

Após esta escolha do utilizador o relatório é processado e este é redirecionado para a página de visualização (como se pode observar na figura 3.25). Nesta área o utilizador pode verificar todos os termos com fugas de informação, gráficos evolutivos do mês que escolheu e ainda com a opção descarregar o relatório em formato *PDF* (presente no lado direito).

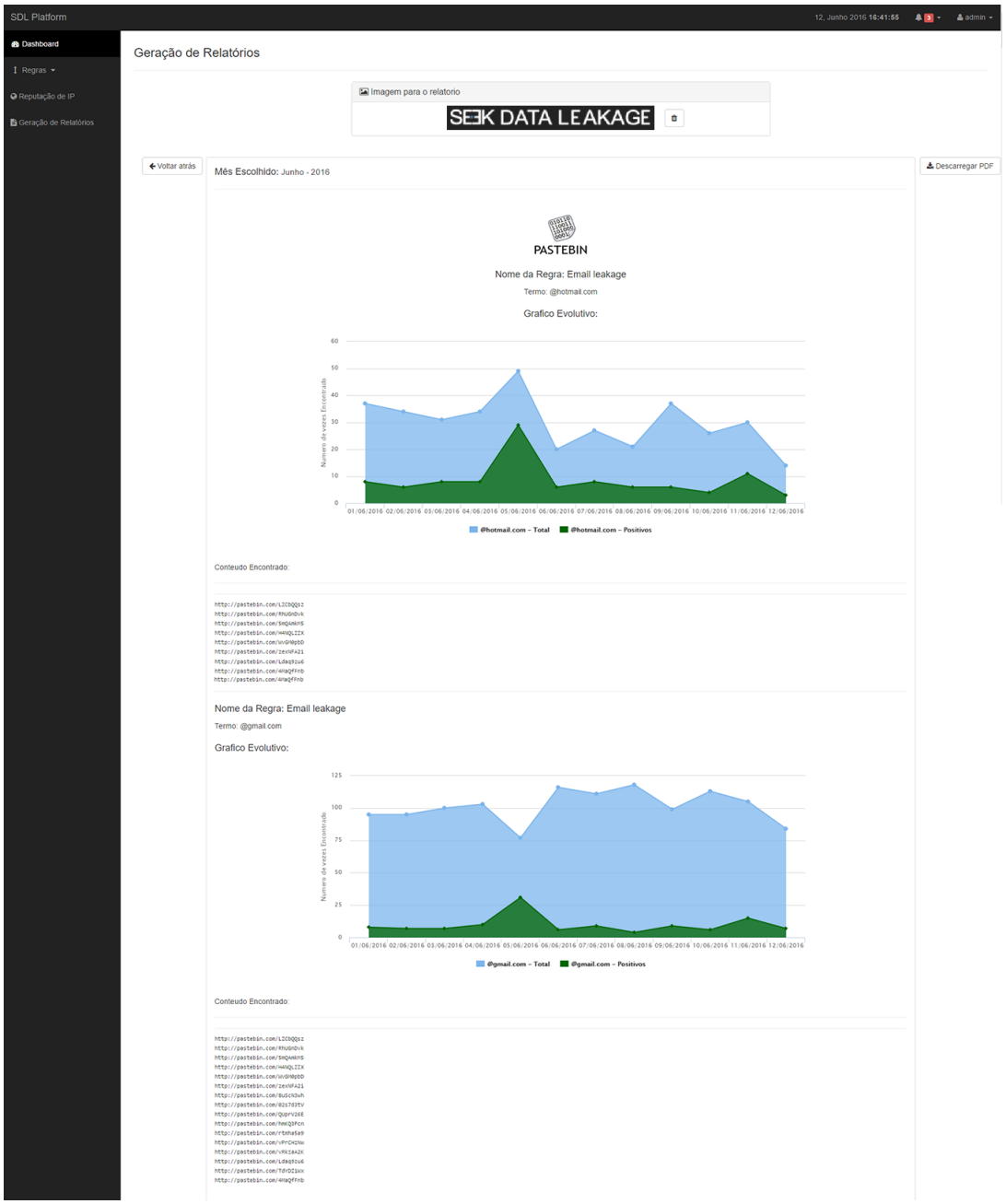

FIGURA 3.25: Página de visualização do relatório

Capítulo 4

Testes e Resultados

Na primeira parte deste capitulo será detalhado o processo bem como os testes de funcionalidade e fugas de informação que foram detetados através da plataforma Seek Data Leakage. Na segunda parte será detalhado todo o processo de recolha de dados dos utilizadores que interagiram com a plataforma, e por fim na terceira parte serão analisados os resultados dos testes detalhadamente para validar todos requisitos da plataforma.

4.1 Teste de funcionalidade e fuga de informação

Como referido na secção 2.1.1.3, o Pastebin é uma das plataformas onde diariamente muitos utilizadores submetem muita informação. Consequentemente, alguma desta informação é considerada como fuga de dados pessoais ou corporativos pois é permitido facilmente por esta plataforma a propagação e disseminação de uma forma não controlada e não autorizada.

Tendo por base esta ideia foi efectuado um estudo entre Fevereiro e Julho de 2016 (sensivelmente 6 meses) e como tal foi utilizada a plataforma Seek Data Leakage para monitorizar todos os conteúdos que eram colocados no Pastebin. O foco deste estudo foram as fugas de contas de emails e para tal, as regras definidas procuravam termos como *"@hotmail.com"*, *"@gmail.com"* e *"@msn.com"*, entre

outros. Foram encontrados neste período um total de 36,372 fugas de dados como se pode observar na figura 4.2 sendo Março o mês onde se verificaram existir um maior número de fugas. Como se seria de esperar e devido à grande abrangência dos termos que foram pesquisados na plataforma existem cerca de 86% (cerca de 31,432 fugas de dados) de falsos positivos e 14% (cerca de 4,940 fugas de dados) de resultados positivos.

As fugas de dados identificadas como positivas foram classificadas automaticamente através da expressão regular apresentada na secção 3.3.2.1 ou manualmente pois a fuga de informação difere do padrão identificado e a plataforma não conseguiu identificá-lo adequadamente. A maior parte dos falsos positivos existem devido à publicação de emails como assinatura ou como contacto e estes poderão ter sido colocados deliberadamente pelo individuo que disponibilizou a informação. No entanto, isto não significa que como existem muitos falsos positivos deveremos ignorar o fato de existirem tantas fugas de informação positivas.

Se analisar-mos o gráfico da figura 4.1 verificamos que onde existe maior disparidade é no termo *"@gmail.com"* que contêm 23,706 fugas de falsos positivos e 2,156 fugas positivas. A seguir na figura 4.2 podemos reparar que o termo *"@hotmail.com"* contêm cerca de 6,380 fugas com falsos positivos e 1,786 fugas positivas. Por fim, o ultimo termo *"@msn.com"* é aquele que está mais equilibrado com 1,346 fugas falsas e 998 fugas positivas.

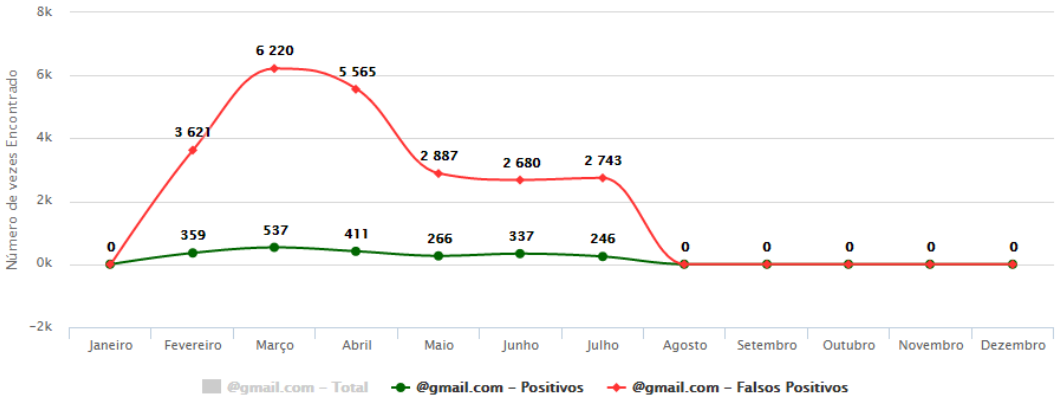

FIGURA 4.1: Gráfico de todas as fugas encontradas por mês para o termo *"@gmail.com"*

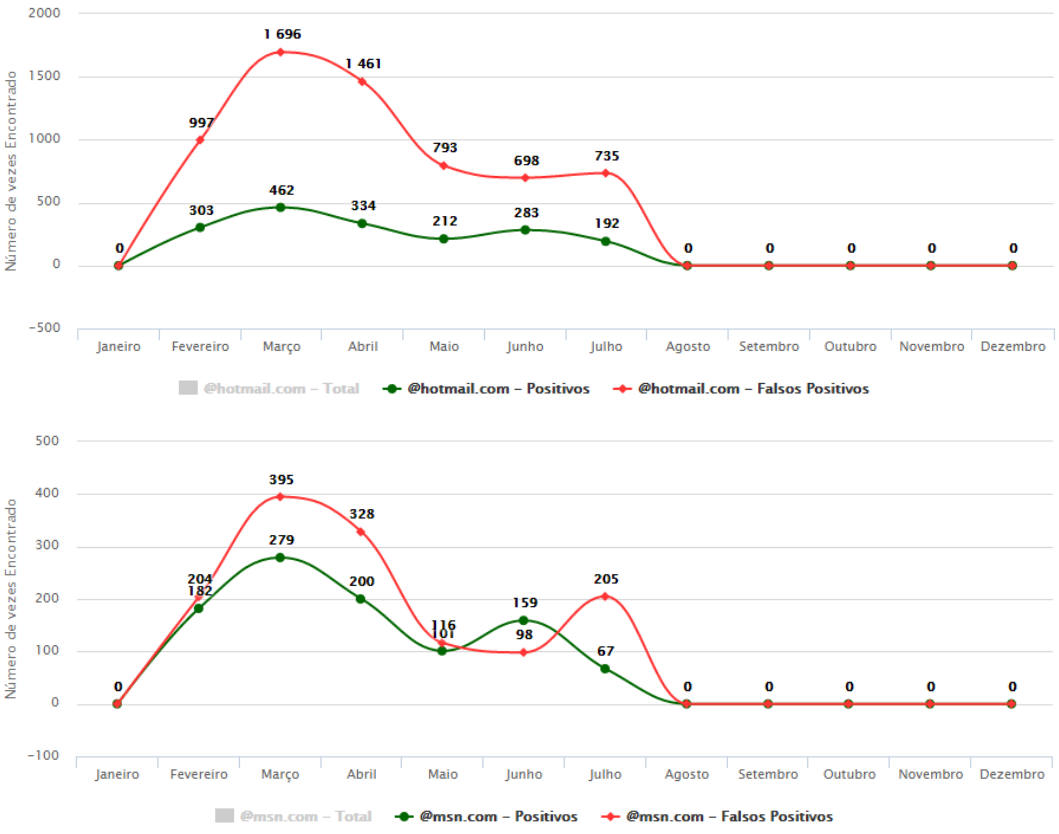

FIGURA 4.2: Gráficos de todas as fugas encontradas por mês para os termos *"@hotmail.com"* e *"@msn.com"*

Um fato a ter em conta é que as fugas de informação das contas de email terem tido origem em bases de dados retiradas de serviços exteriores ao *Outlook*[1] e ao *Gmail*[2], ou seja, o utilizador poderá ter utilizado o seu email para se registar num website e por algum motivo a base de dados foi acedida de forma indevida e a conta do utilizador foi partilhada por outrem.

É recorrente também a publicação de base de dados antigas, em que as passwords já não são utilizadas pelos utilizadores. É notório que existem muitas fugas de informação presentes na *World Wide Web* e uma das formas para mitigar o risco é a alteração periódica de passwords utilizadas em websites exteriores e nunca utilizar a mesma.

[1]http://outlook.live.com
[2]https://mail.google.com/

4.2 Testes de Validação com utilizadores

Entre os meses de Junho e Agosto de 2016 a plataforma esteve disponível na Internet no endereço *https://seekdataleakage.ddns.net/SDLPlatform/login.php*.

Foram convidados utilizadores especialistas da área de segurança da informação alguns a título individual enquanto que outros o fizeram na representação de empresas da área da segurança de informação para testarem a plataforma e emitirem a sua opinião sobre a mesma num questionário que esteve disponível em *https://vhssa.typeform.com/to/wvHJDf*. Nesse mesmo período, registaram-se na plataforma um total de 19 utilizadores. Com este total de utilizadores foram obtidas cerca de 11 respostas válidas ao questionário que era de resposta facultativa. O questionário continha as seguintes perguntas:

1. Indique a sua Idade:
   (a) 18 - 24
   (b) 25 - 30
   (c) 31 - 35
   (d) 36 - 41
   (e) 42-47
   (f) 48+
2. Indique o seu Género:
   (a) Masculino
   (b) Feminino
3. Relativamente à sua Profissão e Empresa:
   (a) Qual a área de negócio de atuação da sua empresa?
   (b) Indique a sua posição na empresa
   (c) Se possível, indique qual a sua empresa

4. Já alguma vez teve que lidar com problemas de fugas de informação critica?

   (a) Sim

   (b) Não

5. Indique os tipos de fugas de informação com as quais já teve que lidar e resolver (indique todos os que se apliquem):

   (a) Dados pessoais

   (b) Dados do meu negócio

   (c) Dados dos negócios dos meus clientes ou parceiros de negócio

   (d) Outro tipo de dados

   (e) Nunca tive de lidar com fugas de informação

6. Independentemente do tipo de dados (pessoais, negócio ou cliente), qual foi o mecanismo que utilizou para encontrar essa fuga de informação? Como é que foi alertado para esta fuga de informação?

7. - As ferramentas apresentas abaixo servem para prevenir a fuga de informação (são denominadas de DLP - Data Loss Prevention) e outras servem para aceder a informação sensível na Internet, selecione da seguinte lista a(as) que conheça.

   (a) Symantec Data Loss Prevention;

   (b) BIGPICTURE 360º Data Analytics;

   (c) Cyberfeed Threat-Intelligence (AnubisNetworks);

   (d) Giga Alert;

   (e) X-FORCE EXCHANGE (IBM);

   (f) Pastebin;

   (g) Shodan;

   (h) Haveibeenpwned;

   (i) Outra;

8. Tem conhecimento de outras ferramentas ou plataformas, para além das visualizadas anteriormente, que permitam prevenir ou alertar para fugas de informação? Qual/Quais?

   Nesta secção serão apenas apresentadas perguntas sobre a plataforma

9. Conseguiu definir regras através da área de criação de regras da plataforma?

   (a) Sim

   (b) Não

10. Através da plataforma, conseguiu encontrar com sucesso fugas de informação com os termos definidos nas regras?

    (a) Sim

    (b) Não

11. Após encontrar informação, os dashboards apresentados na plataforma servem para análise?

    (a) Sim

    (b) Não

12. Se por acaso a plataforma Seek Data Leakage não conseguiu de forma sucinta encontrar e expor a informação que encontrou, consegue explicar porquê?

    (a) Sim

    (b) Não

13. A plataforma não conseguiu de forma sucinta encontrar e expor a informação que encontrou, porque?

14. Chegou a utilizar a reputação de IP? Se sim, com que intuito?

15. Em relação à geração de Relatórios, a informação presente é suficiente e é suficientemente completa para permitir analisar o relatório gerado?

    (a) Sim

    (b) Não

16. Têm alguma opinião geral sobre a plataforma Seek Data Leakage (p.ex aspectos a melhorar)?

17. Quais os pontos que considera mais positivos sobre a plataforma Seek Data Leakage?

18. Quais os pontos que considera mais negativos sobre a plataforma Seek Data Leakage?

19. Tendo em conta a performance da plataforma Seek Data Leakage, que nota lhe atribui?
    Escala: 1 - Muito Má , 5 - Excelente;

20. Tendo em conta as funcionalidades da plataforma Seek Data Leakage, que nota lhe atribui?
    Escala: 1 - Muito Má , 5 - Excelente;

21. Tendo em conta o aspeto visual e de organização da plataforma Seek Data Leakage, que nota lhe atribui?
    Escala: 1 - Muito Má , 5 - Excelente

4.3 Resultados

O género masculino dominou as respostas e testes feitos à plataforma com 100% como se pode observar na figura 4.3. Ainda, a idade mais predominante foi entre 31 e 35 anos com 45% e as restantes idades (de 18 anos a 24 anos e de 25 anos a 30 anos) ficaram empatadas com 27% como se pode observar na figura 4.4.

FIGURA 4.3: Distribuição de género dos participantes no questionário

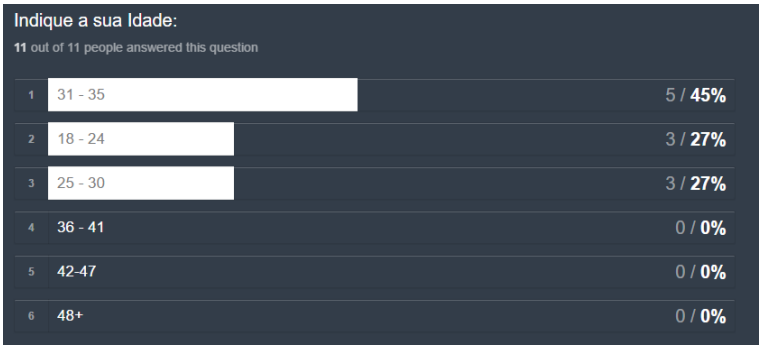

FIGURA 4.4: Distribuição de idade dos participantes no questionário

Relativamente se o inquirido lidou com problemas de fugas de informação crítica, 6 dos inquiridos responderam que sim (55%) e 5 responderam que não (45%) como se pode observar na figura 4.5. Os problemas de fugas críticas continuam a estar presentes no dia a dia com cada vez mais empresas a serem alvo.

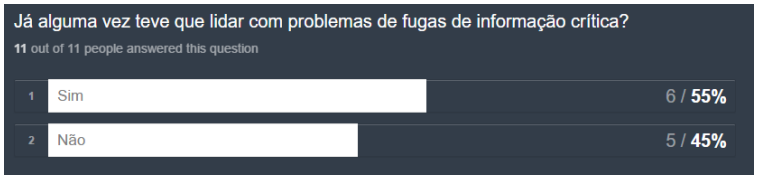

FIGURA 4.5: Distribuição de fugas críticas

Sobre os tipos de fugas de informação com as quais o inquirido já teve de lidar, ambos os dados de negócio bem como os dados pessoais são os que tiveram

mais impacto com cerca de 45%, o que teve menos impacto foi os dados dos negócios dos meus clientes apenas com 9% como se pode observar na figura 4.6. Os mecanismos utilizados pelos inquiridos foram sistemas de monitorização de sites, alertas de outras equipas ou entidades externas (como o Centro Nacional de Cibersegurança[3] ou os próprios intrusos).

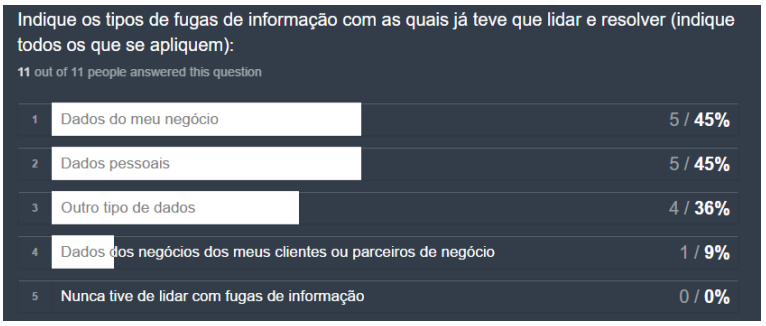

FIGURA 4.6: Distribuição dos tipos de fugas de informação

Quando foram questionados acerca das ferramentas que servem para prevenir a fuga de informação todos os inquiridos conheciam o Pastebin e mais de metade conheciam o Shodan e o Symantec Data Loss Prevention. Como se pode observar na figura 4.7 as ferramentas que têm menos visibilidade são o BIGPICTURE 360° Data Analytics e o X-FORCE EXCHANGE.

FIGURA 4.7: Distribuição das ferramentas conhecidas pelos inquiridos

[3]http://www.cncs.gov.pt/pagina-inicial/index.html

Outras ferramentas conhecidas pelos utilizadores que participaram no inquérito são o BlueCoat DLP[4], TrueDLP[5] e McAfee Total Protection[6]. Todos estes são de prevenção já que se enquadram na categoria de *Data Loss Prevention* e não monitorizam constantemente a *World Wide Web* à procura de informações chave da empresa ou utilizador.

Relativamente aos testes efetuados à plataforma, todos os utilizadores conseguiram definir regras através da área de criação de regras como se pode observar na figura 4.8 apesar de serem sugeridos guias de utilização da plataforma.

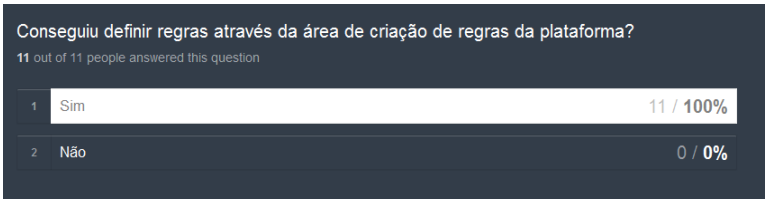

FIGURA 4.8: Gráfico de resposta à pergunta 9, sobre a definição de regras através da área de criação de regras da plataforma

Apenas 3 inquiridos (27%) não conseguiram encontrar fugas de informação com os termos definidos nas regras isto provavelmente pois estes termos eram específicos e não tiveram qualquer fuga na plataforma escolhida, os restantes 8 (73%) afirmaram o contrário e elogiaram a capacidade de resposta da plataforma face às fugas de informação detetadas como se pode observar na figura 4.9.

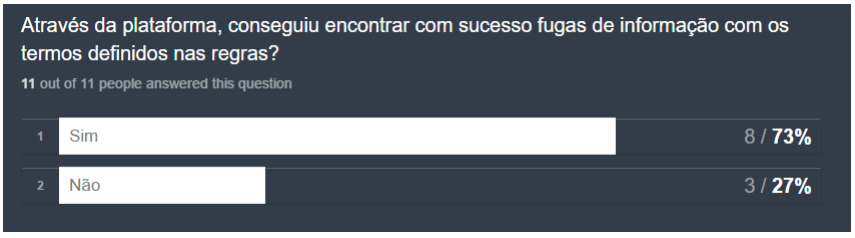

FIGURA 4.9: Gráfico de resposta à pergunta 10, sobre a completude da plataforma em encontrar fugas de informação

[4]https://www.bluecoat.com/products-and-solutions/data-loss-prevention-dlp
[5]https://www.codegreennetworks.com/products/truedlp-data-loss-prevention/
[6]https://www.mcafee.com/consumer/pt-pt/store/m0/catalog/mtp_521/mcafee-total-protection.html

Como se poder observar na figura 4.10 a maior parte dos inquiridos (82%) acharam que os *dashboards* apresentados servem para uma análise composta mensal e anual sobre as fugas de informação. Os utilizadores que responderam negativamente justificaram que não conseguiram visualizar os *dashboards* pois as suas regras não foram encontradas em qualquer plataforma.

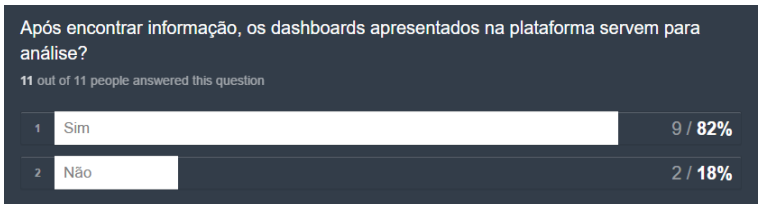

FIGURA 4.10: Gráfico de resposta à pergunta 11, sobre a completude dos *dashboards* para análise

Em relação à plataforma não conseguir de forma sucinta encontrar e expor a informação que encontrou, 7 dos inquiridos conseguem explicar o porque e 4 dos inquiridos não (4.11). Uma das justificações mais dadas foi que provavelmente a regra foi mal configurada pelo utilizador final e daí a plataforma não conseguir expor e encontrar informação. Uma sugestão dada foi a criação de regras padrão para que os utilizadores percebam a lógica ou a criação de um guia de boas vindas para focar o utilizador numa configuração ótima e eficaz.

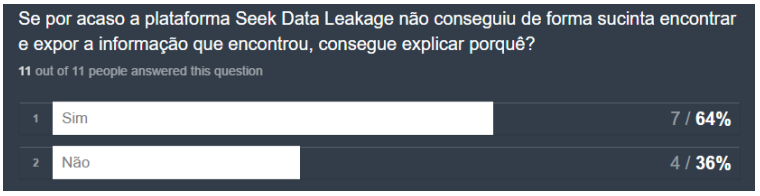

FIGURA 4.11: Distribuição das respostas à pergunta 12 sobre o porque de a plataforma não conseguir encontrar e expor a informação que encontrou

Em relação à reputação de IP's 9 dos inquiridos (82%) responderam que utilizaram para verificar a reputação do próprio IP ou para verificarem a funcionalidade, os restantes 2 (18%) não justificaram o porque de não terem utilizado.

Em relação à geração de relatórios 8 dos inquiridos (73%) afirmaram que a informação é suficiente e é suficientemente completa para permitir a análise do

relatório gerado, os restantes 3 inquiridos responderam negativamente a esta pergunta como se pode observar na figura 4.12.

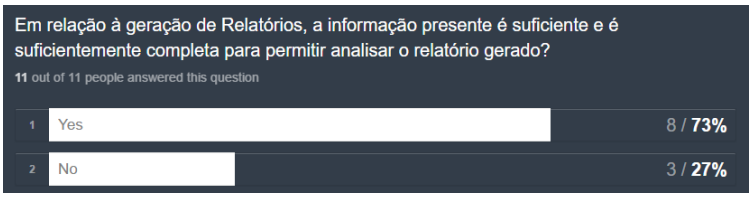

FIGURA 4.12: Distribuição das respostas à pergunta 15 sobre a informação presente é completa o suficiente para analisar o relatório gerado

Sobre a plataforma desenvolvida, os inquiridos classificaram o desempenho ou performance com uma nota de 1 a 5, sendo que 1 significava muito má e 5 excelente. Dos resultados obtidos, a nota que foi mais atribuída foi a nota 4 por cerca de metade dos inquiridos (55%), como demonstra a figura 4.13. A performance da plataforma está relacionada com a rapidez da plataforma e com a disponibilidade da mesma, classificando assim a performance da plataforma com uma nota bastante boa.

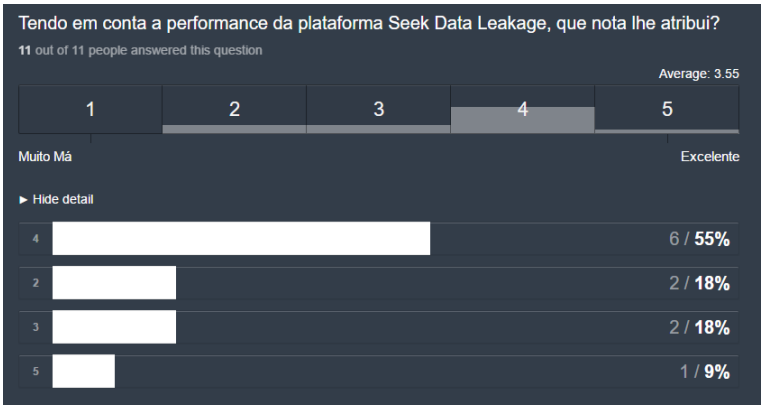

FIGURA 4.13: Distribuição as opiniões em relação à performance da plataforma

Sobre as funcionalidades da plataforma a maior parte dos inquiridos atribuíram a nota 4 (45%), empatando entre a nota 3 (25%) e a nota 5 (25%), como se pode observa na figura 4.14 .

Relativamente ao aspeto da plataforma a nota atribuída foi bastante boa como se pode observar na figura 4.15, sendo que a nota 4 foi a mais predominante com 55% seguida a nota 5 com 27%.

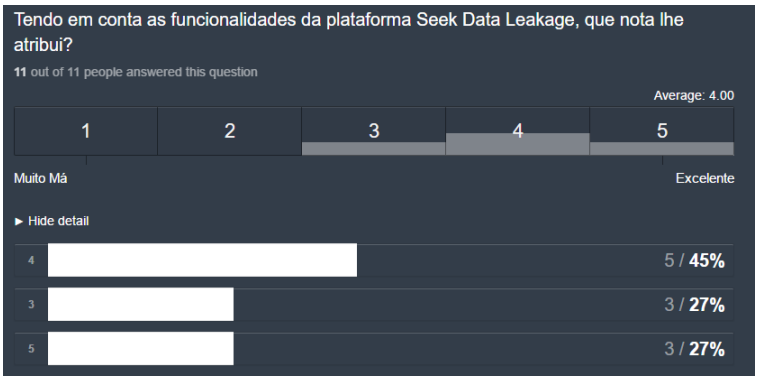

FIGURA 4.14: Distribuição as opiniões em relação as funcionalidades da plataforma

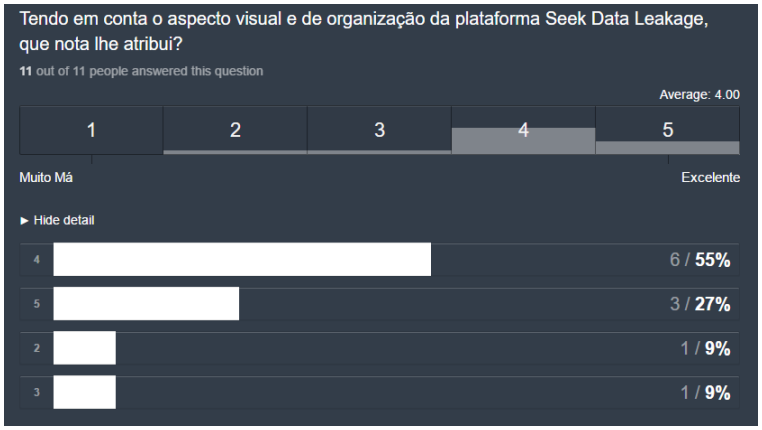

FIGURA 4.15: Distribuição as opiniões em relação ao aspeto da plataforma

De forma a complementar a plataforma foi perguntado se existia algum aspeto a melhorar, identificando-se os seguintes:

- Inclusão de mais plataformas.
- Editar a regra após ser criada.
- Inclusão de uma data de quando a regra foi criada.
- Melhorar o relatório gerado.
- Informação ao utilizador que ainda existiu fugas de informação mas que está recursivamente a procurar.
- Potencial de longevidade e inclusive comercialização através de uma equipa apropriada para o desenvolvimento da mesma.

Em relação aos pontos mais positivos da plataforma, os inquiridos frisaram o fato do aspeto ser simples, leve, apelativo e de fácil utilização. Frisaram ainda a facilidade da criação de regras com atribuição de falsos positivos e relação de fonte (*Pastebin* e *Shodan*), fornecimento dos resultados em tempo real com possibilidade de envio de notificações para o email e a geração de relatórios com os resultados reportados em relação às regras.

Quando inquiridos sobre os aspetos negativos da plataforma foram todos abordados anteriormente com alguns utilizadores a apontarem para a falta de edição de regras, a falta de ajuda para perceberem a forma ótima para configurar as regras e alguns pormenores sobre a forma de como a plataforma geria as regras encontradas.

Capítulo 5

Conclusão e trabalho futuro

Neste capítulo serão abordadas as conclusões bem como todo o trabalho que se poderá realizar futuramente para melhorar a plataforma desenvolvida e garantir uma maior abrangência da *World Wide Web*, na procura e prevenção de fugas de informação crítica, quer em termos de dados pessoais quer corporativos.

5.1 Conclusão

O principal foco desta dissertação é descobrir, ajudar a prevenir e corrigir fugas de informação que se encontram dispersas na Internet, em especial na WWW que, para entidades ou utilizadores é absolutamente vital. Para tal foi desenvolvida uma plataforma online funcional que conseguisse agregar informação proveniente de diferentes fontes online, neste caso foram o *Pastebin* e o *Shodan*. A revisão de literatura permitiu perceber alguns pontos chave de forma a contextualizar o problema identificado e que foi endereçado neste trabalho. Os pontos abordados foram as diferentes técnicas de extração de informação, as tecnologias inerentes à sensibilização e agregação de dados e as diferentes plataformas de *Data Loss Prevention* ou de serviços similares. De seguida foram especificados os requisitos e as funcionalidades da plataforma. Ao longo do desenvolvimento da plataforma foi possível constatar que é cada vez mais notória a falta de visibilidade quando

existem fugas de informação tanto a nível corporativo como pessoal, isto é, existe de fato a falta de uma plataforma que permita a qualquer empresa ou utilizador a monitorização permanente da Internet.

A plataforma esteve disponível online para que um grupo de especialistas na área da segurança de informação convidados pudessem utilizá-la e validá-la. Do grupo de especialistas convidados 19 utilizadores registaram-se e testaram extensivamente a plataforma. A todos os utilizadores registados foi solicitada a resposta a um questionário dos quais foram retirados e estudados os dados como forma de validação da plataforma e da tese. Sobre a questão de investigação proposta inicialmente, "Será que é possível detetar e corrigir fugas de informação na *WWW* usando ferramentas automáticas que pesquisam através de padrões de dados?", esta foi validada na parte da deteção através da criação da plataforma e das expressões regulares aplicadas nas regras como se pode observar na secção 4.1.

A correção de fugas de informação é um tópico mais sensível por parte do utilizador afetado, mas a plataforma criada tem o intuído de fornecer uma capacidade pró-ativa no combate à exposição de informação tornando-se assim uma arma de defesa com um valor inestimável para as organizações e utilizadores. Desta forma a questão de investigação foi respondida de forma positiva sendo que a maior parte dos especialistas afirmaram como ponto positivo que a plataforma consegue detetar fugas de informação de forma automática com tempos de resposta baixíssimos.

Uma das principais limitações é nomeadamente o fato de apenas funcionar em duas fontes de informação (*Pastebin* e *Shodan*) sendo que existem muitas mais fontes de informação que se podem monitorizar conseguindo assim uma abrangência quase total da *World Wide Web*. A solução criada vem contribuir para resolver e mitigar o problema de *information leakage* em geral pelas organizações. A solução desenvolvida foi criada recorrendo a software *open-source* e está igualmente disponibilizada em *open-source* no URL *https://github.com/VSS29/SeekDataLeakage*.

5.2 Trabalho futuro

Existe muito trabalho que pode ser feito na área de segurança da informação e muitas oportunidades que podem ser aproveitadas para além da agregação de informação exposta na *WWW*. Uma das adições que pode ser facilmente implementada é a inclusão de mais algumas plataformas que foram apresentadas na secção 2.1.1.3 como o GitHub Gist e a procura nas Redes sociais. Outro tipo de adição é a introdução de *RSS feeds* que através *Blogs* específicos podem ajudar na monitorização da informação e também a introdução de *Malware Feeds* para complementar a análise de IP's.

Contudo a plataforma ainda que funcional pode ser melhorada e otimizada. A inclusão da edição da regra também poderá ser feita, permitindo assim ao utilizador alterar as regras caso se tenha enganado. Toda a estrutura pode ser revista e o código poderá ser otimizado de forma a conseguir suportar todos os navegadores, resoluções e dispositivos móveis. A geração de relatórios poderia ser mais personalizada, e o *parsing* dos dados poderia ser feito por um motor de dados como o *Kibana*[1], *Splunk*[2] ou *ElasticSearch*[3].

Outro caminho que se poderá tomar é a inclusão e utilização de *Machine-learning* como forma de melhorar a pesquisa e deteção de padrões de informação permitindo assim reduzir os falsos positivos que a plataforma gera e assim ajuda-la a ser bastante mais efetiva nos resultados obtidos.

[1]https://www.elastic.co/products/kibana
[2]https://www.splunk.com/
[3]https://www.elastic.co/

Anexos

Anexo A

Use Case do Utilizador com a plataforma

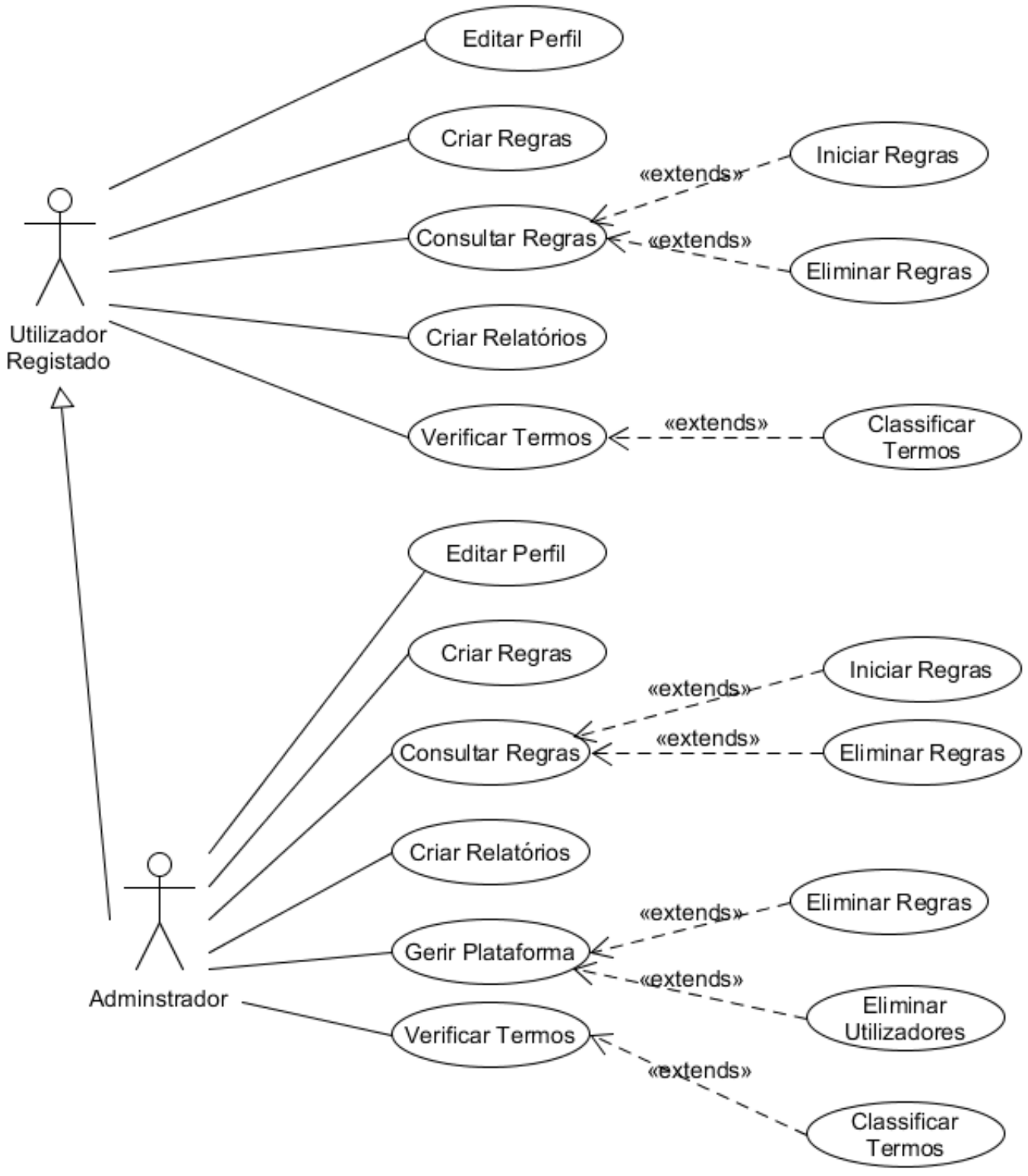

FIGURA A.1: Diagrama de Sequência da autenticação do utilizador

Anexo B

Modelo de base de dados implementado

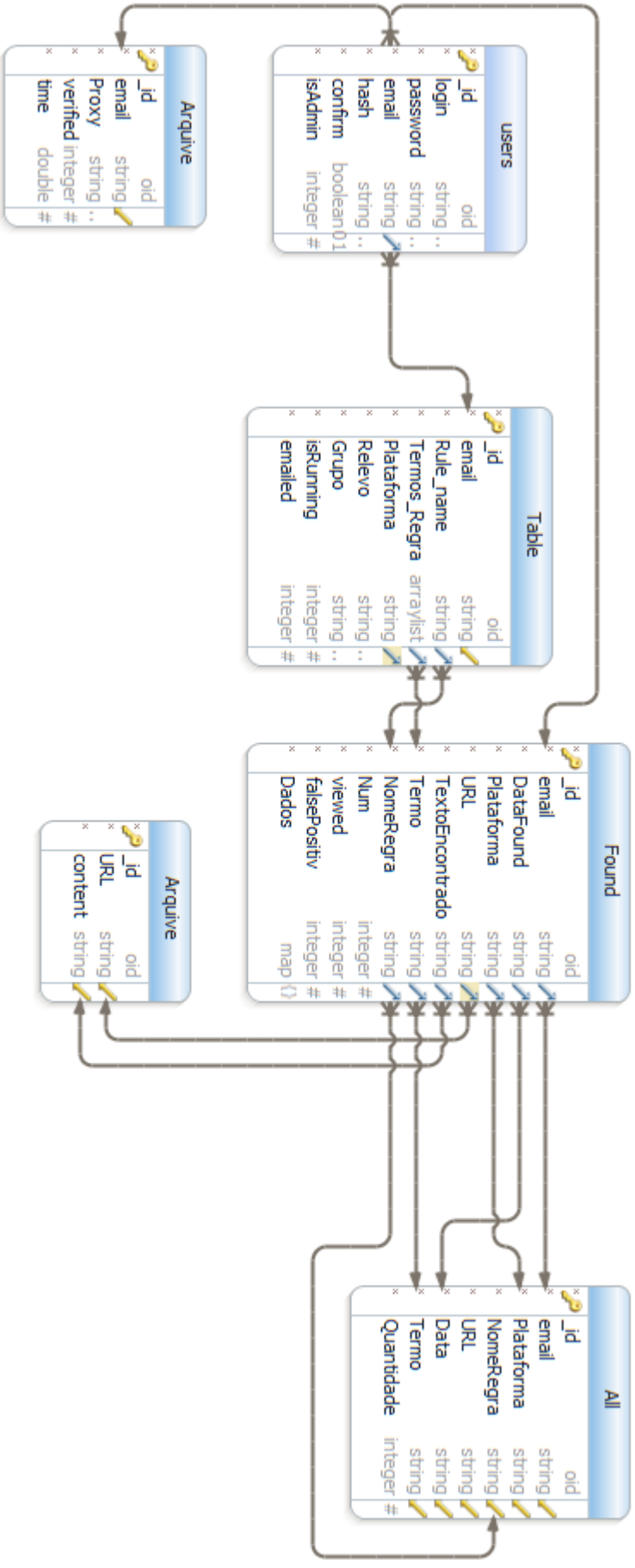

FIGURA B.1: Modelo de base de dados implementado

Bibliografia

AnubisNetworks™. Cyberfeed delivering real security in real time against real threats. Technical report, AnubisNetworks™, 2014.

AnubisNetworks™. Cyberfeed threat intelligence, 2015. URL `https://www.anubisnetworks.com/products/threat-intelligence/cyberfeed/`.

Alex Biryukov, Ivan Pustogarov, and Ralf-Philipp Weinmann. Trawling for tor hidden services: Detection, measurement, deanonymization. *IEEE Symposium on Security and Privacy*, 2013.

Junghoo Cho. *Crawling The Web: Discovery And Maintenance Of Large-scale Web Data*. PhD thesis, Stanford University, Novembro 2001.

CNN and David Goldman. Shodan: The scariest search engine on the internet, Abril 2013. URL `http://money.cnn.com/2013/04/08/technology/security/shodan/`.

Patrick Hagge Cording. Algorithms for web scraping. Master's thesis, Technical University of Denmark, Building 321, DK-2800 Kongens Lyngby, Denmark, 2011.

Peter Gordon. Data leakage – threats and mitigation. *SANS Institute*, 2007.

Alan Hevner and Samir Chatterjee. Design science research in information systems. *Springer*, 2010.

Wolfgang Himmel, Ulrich Reincke, and Hans Wilhelm Michelmann. Text mining and natural language processing approaches for automatic categorization of lay

requests to web-based expert forums. *J Med Internet Res. 2009 Jul-Sep; 11(3): e25.*, 2009.

IBM. Ibm x-force exchange, 2015. URL `https://exchange.xforce.ibmcloud.com/faq`.

Stream Technologies Indigo. Giga alert, 2015. URL `http://www.gigaalert.com/features.php`.

Analytical Center InfoWatch. Global data leakage report. Technical report, InfoWatch Analytical Center, 2014.

Cortex Intelligence. Plataforma bigpicture 360º data analytics, 2015. URL `http://www.cortex-intelligence.com/explore`.

Kevin Kline. *SQL in a Nutshell.* O'Reilly & Associates, Inc, 2001.

Yishan Li and Sathiamoorthy Manoharan. A performance comparison of sql and nosql databases. *Communications, Computers and Signal Processing (PACRIM), IEEE Pacific Rim*, 2013.

James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung Byers. Big data: The next frontier for innovation, competition, and productivity. *McKinsey Global Institute*, 2011.

Julien Masanés. *Web Archiving.* Springer, 2006.

Srdjan Matic, Arstide Fattori, Danilo Bruschi, and Lorenzo Cavallaro. Peering into the muddy waters of pastebin. *ERCIM NEWS*, pages 16–17, 2012.

A B M Moniruzzaman and Syed Akhter Hossain. Nosql database: New era of databases for big data analytics classification, characteristics and comparison. *International Journal of Database Theory and Application*, 6, 2013.

M.Srividya, D.Anandhi, and M.S.Irfan Ahmed. Web mining and its categories – a survey. *International Journal Of Engineering And Computer Science ISSN:2319-7242*, 2, Abril 2013.

Monica Peshave. How search engines work and a web crawler application, 2005.

Asaf Shabtai, Yuval Elovici, and Lior Rokach. *A survey of data leakage detection and Provention Solutions*. Springer, 2012.

Michael Stonebraker. Sql databases v.nosql databases. *Communications of the ACM VOL.53 NO.4 .*, 2010.

Corporation Symantec. Symantec data loss prevention, 1995-2015. URL `http://www.symantec.com/data-leak-prevention/`.

Sholom M. Weiss, Nitin Indurkhya, Tong Zhang, and Fred J. Damerau. *Text Mining - Predictive Methods for Analyzing Unstructured Information*. Springer Science+Business Media, Inc, 2005.

Shuyi Zheng, Di Wu, and Ruihua Song and JiRong Wen. Joint optimization of wrapper generation and template detection. *Research Track Paper*, 2007.