UNIVERSIDADE DE LISBOA
Faculdade de Ciências
Departamento de Informática

INVESTIGAÇÃO E DESENVOLVIMENTO DE MECANISMOS AUTOMÁTICOS E DISTRIBUÍDOS DE DETECÇÃO, ANÁLISE E MENSURAÇÃO DE EVENTOS RELEVANTES PARA A AVALIAÇÃO DA QUALIDADE DE PROTECÇÃO DE *CALL CENTERS* DE GRANDE DIMENSÃO

Hugo Rosa da Silva

PROJECTO

MESTRADO EM ENGENHARIA INFORMÁTICA
Especialização em Arquitectura, Sistemas e Redes de Computadores

2012

UNIVERSIDADE DE LISBOA

Faculdade de Ciências
Departamento de Informática

INVESTIGAÇÃO E DESENVOLVIMENTO DE MECANISMOS AUTOMÁTICOS E DISTRIBUÍDOS DE DETECÇÃO, ANÁLISE E MENSURAÇÃO DE EVENTOS RELEVANTES PARA A AVALIAÇÃO DA QUALIDADE DE PROTECÇÃO DE *CALL CENTERS* DE GRANDE DIMENSÃO

Projecto realizado na

Portugal Telecom

por

Hugo Rosa da Silva

Projecto orientado pelo Prof. Doutor António Casimiro Ferreira da Costa

e co-orientado pelo Eng. José António dos Santos Alegria

MESTRADO EM ENGENHARIA INFORMÁTICA
Especialização em Arquitectura, Sistemas e Redes de Computadores

2012

Agradecimentos

Desde muito cedo que o mundo das novas tecnologias me fascinou provocando um crescendo de curiosidade e desejo de encetar novas descobertas, por esse motivo os meus pais sempre me apoiaram nas decisões de aprendizagem, estimulando-as e acompanhando de perto, especialmente nos momentos mais complexos da minha vida académica, nomeadamente, nos vários projectos desenvolvidos já na faculdade.

A minha vida académica foi abraçada com empenho e trabalho, mas também por alguns momentos de lazer, tendo proporcionado o surgimento de novas amizades e a consolidação de outras, que muito me apoiaram, a quem agradeço. São de referir os meus amigos do Departamento de Informática da Faculdade de Ciências da Universidade de Lisboa, que foram os verdadeiros companheiros de aventuras.

Agradeço à minha mãe Maria João todo o conhecimento e valores que me transmitiu nas mais diversas áreas, pelo carinho e compreensão que só uma mãe especial sabe. Agradeço ao meu pai João pelo apoio e conselhos dados, de alguém que muito sabe da vida. Agradeço à minha avó Lourdes que sempre esteve próximo de mim vendo-me crescer. Agradeço à minha esposa Inês pela companhia de todos os dias e pelo sorriso matinal dando-me ânimo para mais um dia de trabalho. Todos vós sempre me destes as maiores alegrias que um homem pode ter.

Por fim e não menos importante, agradeço: ao meu professor e orientador António Casimiro pelos relevantes conselhos e observações dados ao longo da elaboração deste relatório, motivando-me em cada capítulo desenvolvido; ao Eng. José Alegria pela confiança depositada em mim na realização de um projecto desta envergadura; aos Engºs Pedro Inácio e Pedro Simões o importante acompanhamento no decurso do estágio.

Às minhas famílias.

Resumo

Este projecto foca-se na segurança de redes empresariais de grande dimensão, envolvendo informação sensível e onde não é possível garantir segurança absoluta.

Para ter uma concepção do nível de segurança de uma rede empresarial é vital conhecer todos os atributos e acções tomadas pelos dispositivos que a constituem. Ainda que uma porção dos activos seja bem conhecida, existem outros que possuem uma actividade intermitente e desconhecida, passíveis de infringir determinadas políticas de segurança da empresa.

O projecto desenvolvido tem como objectivo, conhecer a estrutura de uma rede empresarial em tempo-real. Para atingir este propósito é concretizada uma solução de *software* e *hardware* eficiente e simples, permitindo a monitorização da rede em tempo-real, quer por meios intrusivos ou não intrusivos. Esta solução possui características automáticas e distribuídas. As automáticas não necessitam da intervenção humana na monitorização, no entanto, nas distribuídas a monitorização é efectuada através de uma federação de sondas colocadas em pontos geograficamente estratégicos.

A etapa final é desenvolvida com vista à concretização de uma solução, reflectindo uma avaliação da sua escalabilidade, bem como, uma avaliação crítica dos factores que influenciam a qualidade e a quantidade de dados recolhidos através da monitorização da rede.

Palavras-Chave: Segurança, Monitorização Activa, Monitorização Passiva, Qualidade de Protecção, Tráfego de Rede

Abstract

This project focuses on the safety of large enterprise networks, involving sensitive information and where it is not possible to guarantee absolute safety.

To acknowledge the level of security of an enterprise network is vital to know all attributes and actions taken by the constituent devices. Although a portion of the assets is well known, there are others who have an unknown and intermittent activity that may violate certain company security policies.

This project is developed with the aim to know the structure of a corporate network in real time. To achieve this aim is developed a solution of simple but efficient software, which allows monitoring the network in real time. This solution has automatic features as needs no human intervention in monitoring, and has distributed characteristics, because the monitoring is done through a federation of probes, placed in strategic geographical points.

The final step, given the realization of the solution reflects an evaluation of its scalability, and also a critical evaluation of the factors influencing the quality and quantity of data collected by monitoring a network.

Keywords: Security, Active Monitor, Passive Monitor, Quality of Protection, Network Traffic.

Conteúdo

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

1.1 Motivação

"Mantêm os teus amigos perto, e os teus inimigos ainda mais perto."

O facto de um dispositivo desconhecido se ligar é rede interna de uma empresa pode não ter um impacto prejudicial, no entanto, se este problema envolver a ligação de mil dispositivos desconhecidos no período de um dia, é então imprescindível uma observação do estado da rede para decidir as acções a adoptar.

A monitorização em tempo-real de grandes redes empresariais não só aumenta o conhecimento do número de dispositivos e suas características a cada instante, alertando também para anomalias passíveis de serem determinadas através de valores padrão definidos inicialmente.

É de vital importância toda a informação que possa ser recolhida através da monitorização da rede, pois existem milhares de dispositivos dinâmicos e também estáticos que possuem atribuições essenciais na estrutura da rede. É tão importante saber com exactidão se certo dispositivo desempenha um papel de servidor, como ter o conhecimento das características de um dispositivo que efectua comunicações de duas em duas horas com esse mesmo servidor. Toda a informação é proeminente quando se necessita de um mapeamento total de uma rede empresarial complexa.

O factor *"tempo-real"* é o ponto de viragem na segurança interna de qualquer rede, pois não se considera segura uma rede que, embora possua um conhecimento total de todos os seus dispositivos, a sua última actualização tenha sido efectuada há cinco meses.

Deste modo, é necessária uma solução que possibilite uma monitorização minimamente intrusiva, passível de responder às necessidades de uma empresa, no que

concerne ao seu nível de segurança interna, não possuindo no entanto, capacidades de actuação directa no problema, mas apenas especificidades no contexto de captura e análise de informação.

Será uma solução que visa a renovação e adição de novas funcionalidades de monitorização automática da plataforma Pulso/*Discovery*.

1.2 Objectivos

Neste projecto foram definidos quatro objectivos fundamentais que concorrem para uma concretização eficiente de uma solução de monitorização que preencha as lacunas no projecto Pulso/*Discovery*, e adicionando novas funcionalidades. Investigação dos requisitos da solução, desenho e arquitectura coerentes com os requisitos, desenvolvimento da solução no ambiente pretendido e por fim uma avaliação crítica das suas potencialidades.

Foi assim concretizada uma solução de *software*, permitindo monitorizar em tempo-real os dispositivos da rede empresarial, através de uma integração de ferramentas especializadas de monitorização já existentes no mercado *open-source.*

De modo a cumprir os objectivos propostos, foi necessário um planeamento concreto relativamente às fases de desenvolvimento da solução.

- Estudo da arquitectura da federação de sondas de rede existentes e sistemas associados às mesmas, estudo de ferramentas e componentes necessários ao projecto, nomeadamente o *nmap* [18], *p0f* [19], *IPAudit* [21], *pads* [20], bem como, a investigação de projectos relacionados com a temática da monitorização.

- Investigação exploratória inicial, relativa à integração das diversas componentes atrás referidas, num modelo de informação coerente, tendo em vista a melhoria significativa do processo de construção de uma base de informação válida.

- Aplicação da investigação anterior à federação de sondas, de modo a sobrevir uma arquitectura susceptível de ser escalável e distribuída, despendendo de um número mínimo de recursos.

- Desenho e implementação de um protótipo eficiente na federação de sondas, distribuindo a monitorização de acordo com a quantidade de tráfego de cada sonda,

utilizando monitorização activa e monitorização passiva, tendo em conta o nível de intrusão da primeira e os protocolos de conectividade com um servidor central.

- Desenvolvimento das componentes de *software* que controlam todo o processo de captura e análise de tráfego, e gerem os canais de comunicação com o servidor central.

- Recolha de dados típicos, permitindo construir métricas de segurança, para melhor compreender a Qualidade de Protecção (QoP) da rede, em determinados períodos de tempo.

- Teste de um protótipo da solução, num segmento de rede real, avaliando a sua escalabilidade e eficácia perante possíveis cenários encontrados na rede empresarial.

- Extensão do protótipo da solução possibilitando o funcionamento em federação colaborativa com todas as sondas da rede.

- Integração e automação da implementação do modelo desenvolvido.

1.3 Equipa de Trabalho

A empresa que engloba este projecto é a Portugal Telecom, com origens em 1994, e possui a maior cobertura de telecomunicações do país.

A Portugal Telecom é composta por diversas empresas, sendo uma das quais a PT Comunicações desempenhando um papel fulcral nas comunicações tanto nacionais como internacionais.

Dada a iminente e constante transformação das tecnologias empresariais, foi inevitável a criação de uma equipa que assegure eficiência, disponibilidade e segurança. Assim, foi formado o departamento de Eficiência, Disponibilidade e Segurança (PT-EDS), estando-lhe adstrito o desenvolvimento do sistema Pulso.

1.4 Organização do Documento

No primeiro capítulo deste documento, focam-se os pontos mais importantes no que concerne à problemática da segurança informática no interior de uma rede empresarial de grande dimensão.

O capítulo segundo apresenta a plataforma/sistema Pulso, encontrando-se em fase produção.

O capítulo terceiro apresenta uma panorâmica teórica do projecto, explicando-se sucintamente as técnicas utilizadas na monitorização de grandes redes.

Seguidamente, são introduzidas no capítulo 4 as características das ferramentas utilizadas para efectuar uma optimização da monitorização da rede, revelando o seu tipo de funcionamento e configurações ideais para o ambiente empresarial. São também evidenciadas várias ferramentas que poderão ser integradas futuramente.

O capítulo cinco reflecte a arquitectura da solução final, bem como a sua implementação, desde a construção dos módulos mais básicos, até aos protocolos de comunicação mais complexos.

De modo a efectuar uma validação da solução, foram efectuados diversos testes experimentais em diferentes cenários do ambiente empresarial. Assim, o capítulo seis demonstra a robustez e eficácia da solução através de dados estatísticos.

Os dados estatísticos apresentados no capitulo seis, bem como, alguns problemas encontrados durante o desenvolvimento da solução são discutidos no capítulo sétimo.

Finalizando, no capítulo oitavo encara-se o projecto como uma base bem estruturada, no sentido de serem integradas novas ferramentas, enriquecendo a informação e permitindo um maior nível de profundidade.

Capítulo 2

Sistema Pulso

2.1 Apresentação

O sistema Pulso [1][2][3][15] foi desenvolvido pela equipa responsável da gestão de segurança, risco técnico e qualidade técnica dos sistemas e tecnologias de informação do grupo PT Comunicações.

É uma plataforma composta pelo programa de projectos, pela estrutura técnica que a suporta e por um interface externo, permitindo visualizar todos os dados e análises de Qualidade de Serviço (QoS).

O Pulso é desenvolvido focando-se na detecção, colecção, análise e comunicação de eventos, de modo a suportar a prevenção e a reacção em tempo útil de incidentes que possam ocorrer. Assim, efectua o registo de forma contínua e automática de todos os colaboradores e sistemas envolvidos, tal como o seu grau de risco dentro da empresa, é também possível identificar tendências de evolução a nível da fiabilidade e segurança.

O Portal do Pulso permite ter uma visão centralizada e organizada, podendo encontrar-se todos os dados relevantes para efectuar uma boa gestão de risco técnico e de segurança. É o espaço de comunicação entre os utilizadores dentro da organização.

2.2 Arquitectura

A arquitectura técnica do sistema Pulso é constituída por diversos *frameworks* que permitem o seu desenvolvimento incremental.

Um *framework open-source* que permite o desenvolvimento de Portais, alojando todas as funcionalidades e resultados produzidos concernentes a um ponto de interacção com os utilizadores.

Um *framework* que permite a captura de eventos relevantes através de colectores distribuídos e autónomos, utilizando para esse efeito máquinas já descontinuadas.

Um *framework* que após a captura dos eventos, os processa, correlaciona e analisa.

Um *framework* de produção de alarmes, que após o processamento dos eventos poderá lançar um alerta, notificando todos os utilizadores com responsabilidades na área onde se despoletou o alarme.

Por fim, um *framework* que permite o desenvolvimento de visualizadores gráficos, onde os factos derivados da análise de eventos serão apresentados através de uma metáfora do estado do tempo sobre um mapa de Portugal.

A arquitectura física do sistema Pulso é composta por sistemas distribuídos, que efectuam a medição da qualidade da rede em locais específicos dentro da PT Comunicações. Para este efeito existem três módulos principais que permitem a captura e análise de tráfego de rede entre os pontos pretendidos, designando-se por sondas de rede (*collectors*), agentes de sistema (*targets*) e *loggers.*

Os *collectors* e os *targets* possuem uma ligação lógica entre eles, possibilitando medir parâmetros de rede, e cada agente reporta somente á sonda que o contactou.

Os agentes de sistema ou *targets* são distribuídos por sistemas que possuem um elevado número de utilizadores e aplicações, sendo críticos para a empresa.

Os *loggers* desempenham um papel importante na arquitectura, sendo responsáveis por periodicamente recolher e armazenar na base de dados do Pulso a informação referente às medições efectuadas pelos *collectors* na rede.

Deste modo, as sondas efectuam as medições de acordo com o agente especificado, e armazenam a informação obtida localmente, permitindo que essa informação não se perca, na eventualidade de o *logger* não efectuar a recolha periódica, no entanto logo que, exista uma ligação disponível, entre a sonda e o *logger*, são actualizados e sincronizados todos os parâmetros em falta.

Após todas as medições serem processadas, são gerados relatórios e gráficos, que integram o portal do Pulso, sendo também reportados os incidentes detectados e analisados.

A Figura 1 representa esquematicamente a arquitectura da *framework* de captura e análise do Pulso.

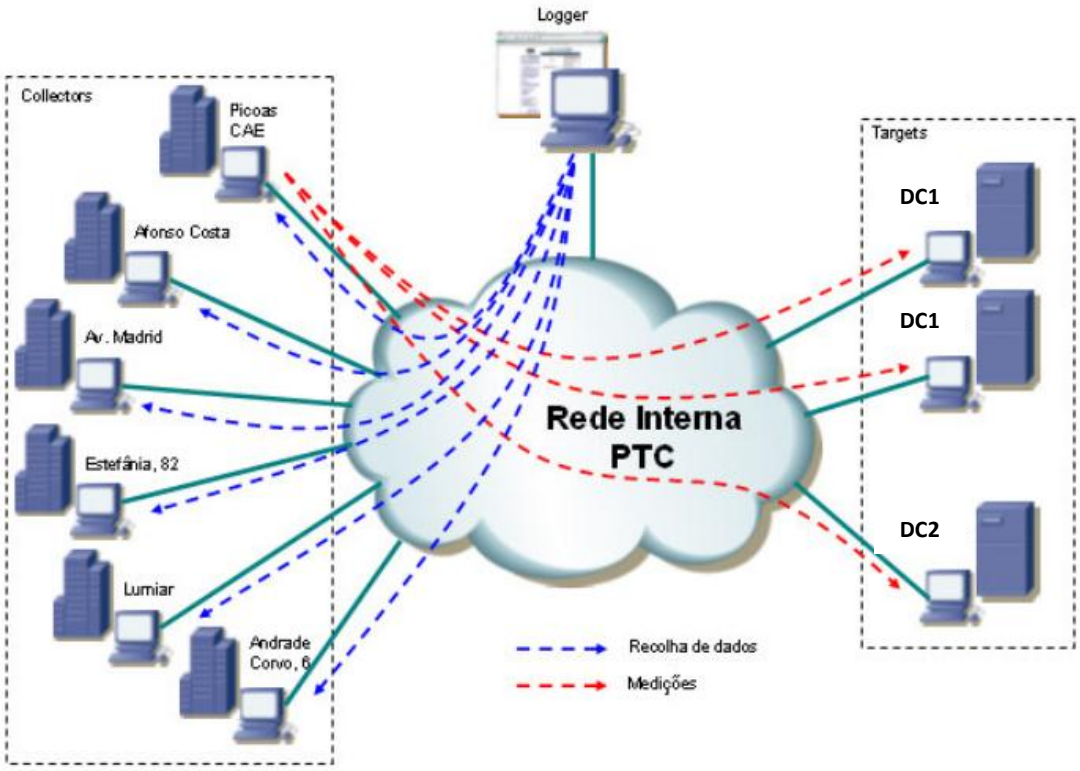

Figura 1 - Arquitectura de *collectors, targets e loggers*

2.3 Resumo

O sistema Pulso é completamente *open-source*, eficaz e de baixo custo.

O desenvolvimento incremental do sistema Pulso permite, o aumento das próprias capacidades, bem como, a introdução de projectos de utilidade imediata por parte de equipas jovens traduzindo-se num sistema robusto, são estes factores que qualquer rede empresarial deseja ver implementado para o progresso do seu negócio.

O projecto descrito neste documento insere-se assim na corrente de desenvolvimento e arquitectura do Pulso, e através dos resultados de uma boa monitorização, sendo ainda, mais fácil detectar zonas na rede propícias a incidentes.

Capítulo 3

Enquadramento Teórico do Projecto

Para se monitorizar uma rede empresarial de grandes dimensões é importante ter em conta a heterogeneidade e a distribuição geográfica da mesma. É impossível ter uma visão completa da rede apenas com dados recolhidos mensalmente, ou até mesmo semanalmente, pois as variações do número de dispositivos ligados são enormes.

A monitorização em tempo-real vem então preencher o espaço deixado pela problemática que é a falta de informação instantânea. Este tipo de monitorização divide-se em duas técnicas principais, a *Monitorização Activa* e a *Monitorização Passiva*, sendo a primeira bastante intrusiva, mas disponibilizando uma boa quantidade de informação relativamente ao dispositivo monitorizado, e a segunda técnica a menos intrusiva possível numa rede de grandes dimensões.

Através da análise cuidadosa das medições efectuadas por estes dois tipos de monitorização, pode-se construir *Métricas de Segurança* que permitem saber por exemplo o nível de segurança de uma empresa.

3.1 Técnicas de Monitorização Activa

As técnicas de Monitorização Activa [5][6][9] têm evoluído a cada dia que passa, acompanhando a tendência dos elevados níveis de segurança aplicados nas empresas. Assim, com a utilização de um modelo de monitorização intrusivo, é possível mapear os dispositivos e os serviços que eles fornecem dentro da rede, descobrindo assim certas vulnerabilidades.

O seu funcionamento passa pelo envio de pacotes *TCP* ou *UDP*, previamente construídos, e enviados directamente ao dispositivo alvo, e aguardar por uma resposta que pode ser composta por, informação relativa ao estado da máquina, os serviços disponíveis em cada porto, e o sistema operativo instalado na respectiva máquina.

Na posse da informação disponibilizada, é possível identificar áreas de rede vulneráveis, quer pelo número e tipo de serviços disponíveis, quer pelo sistema operativo utilizado.

Existindo um nível de intrusão elevado, e passível de introduzir tráfego excessivo na rede, é essencial escolher um conjunto de máquinas alvo que não sejam críticas à empresa, efectuando uma monitorização, assente em três formatos diferentes: *Open Scanning, Half-Open Scanning, e Stealth Scanning*.

A avaliação do tipo de rede deve estar sempre presente na escolha do formato utilizado, pois quer o *Open Scanning* quer o *Half-Open Scanning* são facilmente detectáveis em redes que possuam *firewalls.*

Os pacotes utilizados nesta monitorização são tipicamente *TCP* e/ou *UDP*, com as *flags* de controlo *ACK*, *SYN*, *FIN*, *RST* e *PSH.*

- *ACK* – Receptor de confirmação que é igual ao número de sequência do remetente mais o tamanho ou quantidade de dados na camada de *TCP*.
- *SYN* – É utilizada durante a sessão para concordar com os números de sequência inicial.
- *FIN* – Utilizada durante uma sessão normal, quando o remetente não possui mais dados para enviar.
- *RST* – É uma anulação instantânea da sessão em ambas as direcções.
- *PSH* – Envio forçado de dados sem esperar por *buffers* de preenchimento.

3.1.1 *Open Scanning*

Esta técnica de monitorização envolve uma ligação completa, tipicamente *Three-Way Handshake* com um porto da máquina alvo, que após a troca da primeira mensagem responde com determinadas *flags* activas, caso o porto se encontre aberto ou fechado.

Independentemente do estado do porto a ligação é terminada, e logo de seguida é iniciada uma nova ligação com outro porto de destino.

A resposta enviada pelo alvo à primeira mensagem da sonda, possui as *flags SYN* e *ACK*, isto significa que o porto em questão se encontra aberto. Caso o porto se encontre fechado, a resposta do alvo ao estímulo da sonda possui as *flags RST* e *ACK*, estabelecendo o fim da ligação, tal como demonstrado nas Figuras 2 e 3.

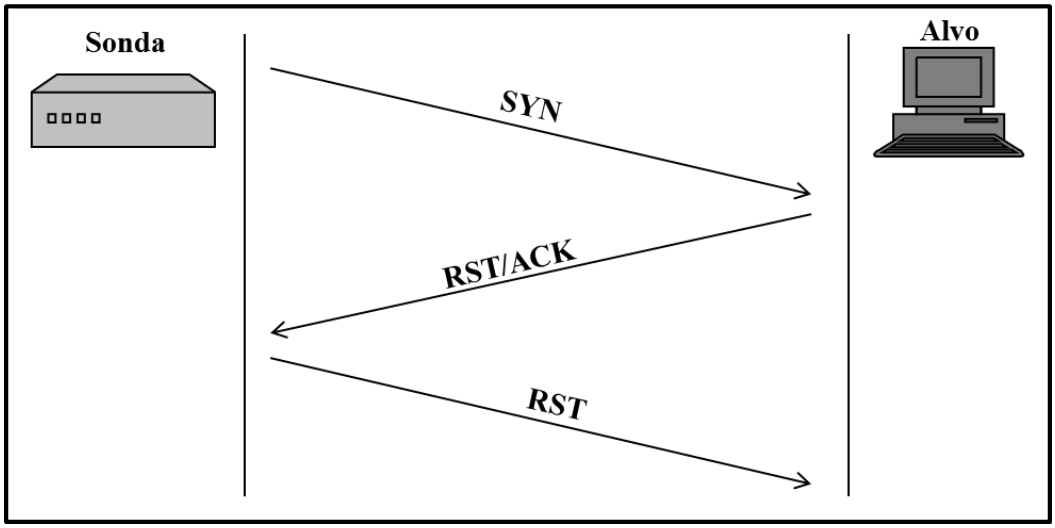

Figura 2 - Comportamento de um porto fechado com *Open Scanning*

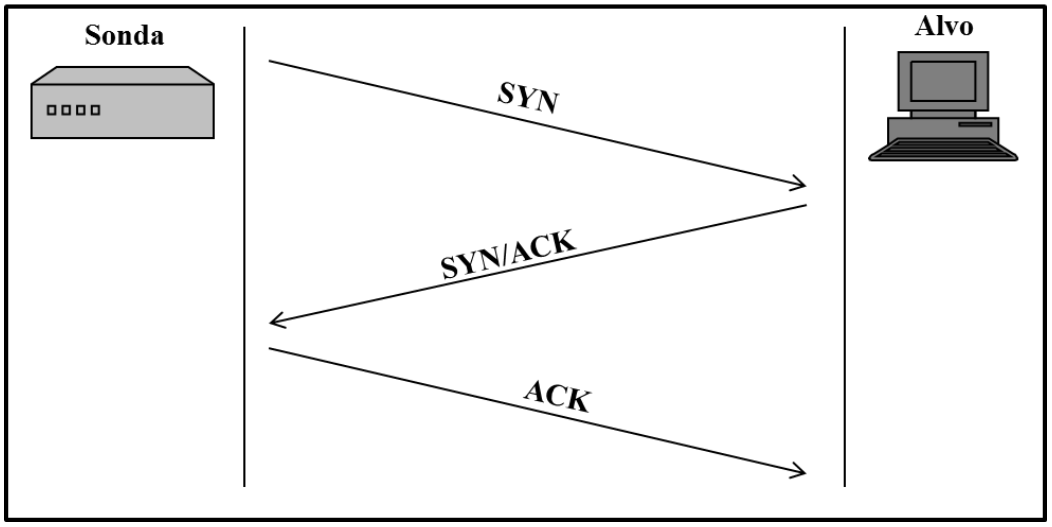

Figura 3 - Comportamento de um porto aberto com *Open Scanning*

Este tipo de técnica é facilmente identificável por sistemas de detecção de instruções, pelo facto de iniciar uma ligação por cada porto de destino, terminando logo após receber uma resposta. No entanto a sua rapidez e precisão revelam que é uma técnica excelente num ambiente sem sistemas de detecção de instruções.

3.1.2 *Half-Open Scanning*

Esta técnica, tal como a anteriormente mencionada utiliza uma ligação *Three-Way Handshake,* no entanto, a sonda não efectua a comunicação por completo terminando a ligação.

Um exemplo da técnica *Half-Open Scanning* é também denominado por *SYN Scanning*, pois a sonda apenas envia uma mensagem do tipo *SYN*, aguarda pela resposta do alvo, e termina de imediato a ligação, conforme apresentado nas Figuras 4 e 5.

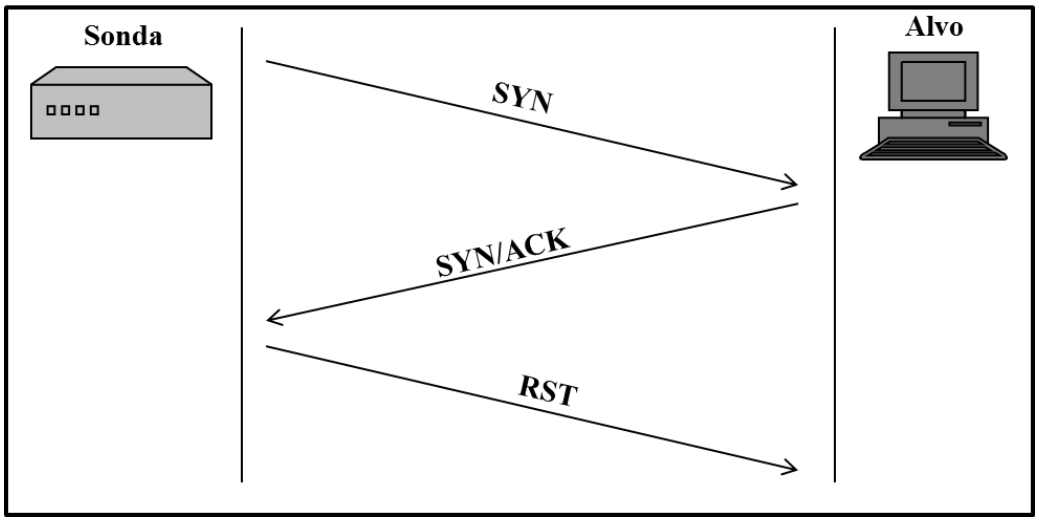

Figura 4 - Comportamento de um porto aberto com *Half-Open Scanning*

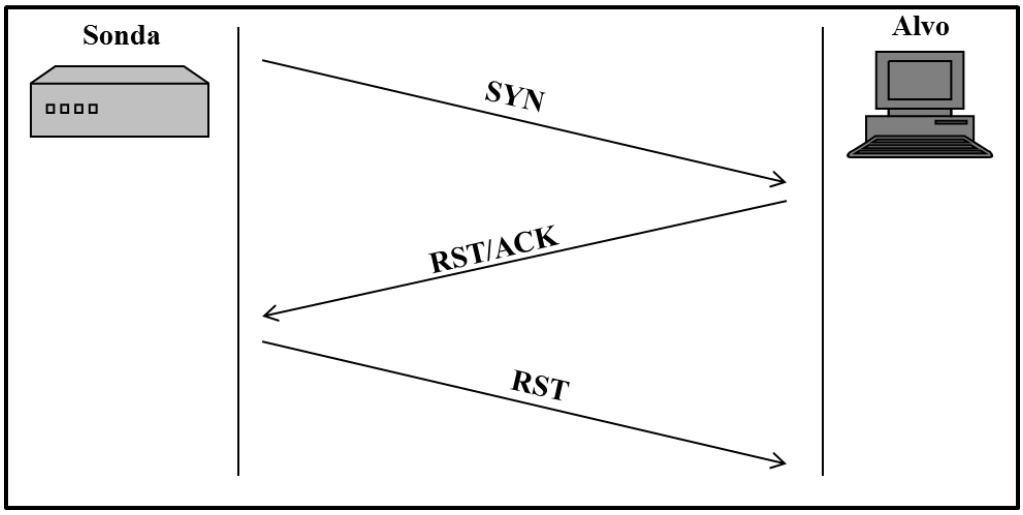

Figura 5 - Comportamento de um porto fechado com *Half-Open Scanning*

Assim, poderemos afirmar que os resultados encontrados relativamente ao estado dos portos são iguais aos resultados obtidos com a aplicação da primeira técnica, no entanto, com a técnica *Half-Open Scanning* é a sonda a responsável do envio da mensagem com a *flag RST* para o alvo, de modo a finalizar a ligação.

Utilizando esta técnica rápida e fiável, é possível evitar sistemas de detecção básicos, pois, o facto de não completar a ligação típica, deixa poucas pistas sobre as suas intenções.

3.1.3 *Stealth Scanning*

Esta técnica foi primariamente definida para evitar a sua detecção e ultrapassar sistemas protegidos por *firewalls*, logo, o seu comportamento aparenta ser apenas tráfego de rede considerado normal, modificando para esse efeito, as *flags* existentes nas suas mensagens, tornando-a numa técnica de mapeamento inverso.

- *SYN|ACK Scanning*

Um bom exemplo de *Stealth Scanning* é o *SYN|ACK Scanning*, que divide a ligação típica *Half-Open Scanning*, enviando a sonda apenas uma mensagem com as *flags SYN* e *ACK*. Caso o alvo responda, a mensagem deve possuir a *flag RST*, significando que o estado do porto é fechado, verificando-se a não existência da mensagem *SYN* inicial e terminando automaticamente a ligação, ou seja, estes factos determinam a interpretação da ocorrência de um erro de transacção, pela existência da recepção de uma mensagem *SYN|ACK* antes de uma mensagem *SYN*, conforme Figura 6.

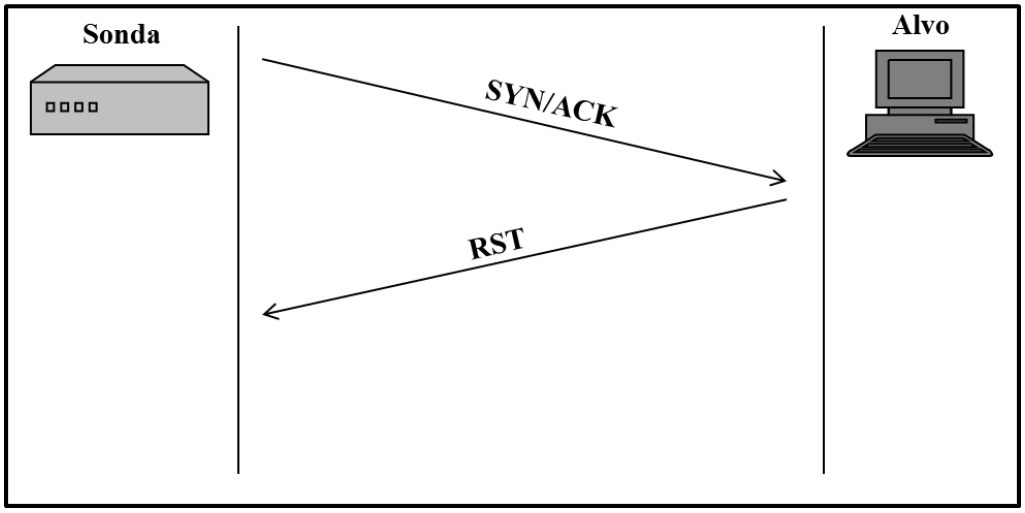

Figura 6 - Porto fechado com *SYN/ACK Scanning*

Por outro lado, um porto que se encontre aberto não iria enviar qualquer tipo de mensagem. O alvo descarta a mensagem *SYN|ACK* por não ter recebido anteriormente uma mensagem *SYN*, e limita-se a não responder, o que leva à produção de falsos positivos, pois o facto de não receber qualquer tipo de resposta pode indicar que o

pacote tenha sido filtrado por um sistema de detecção ou até mesmo excedido o seu tempo de vida na rede, tal como se apresenta na Figura 7.

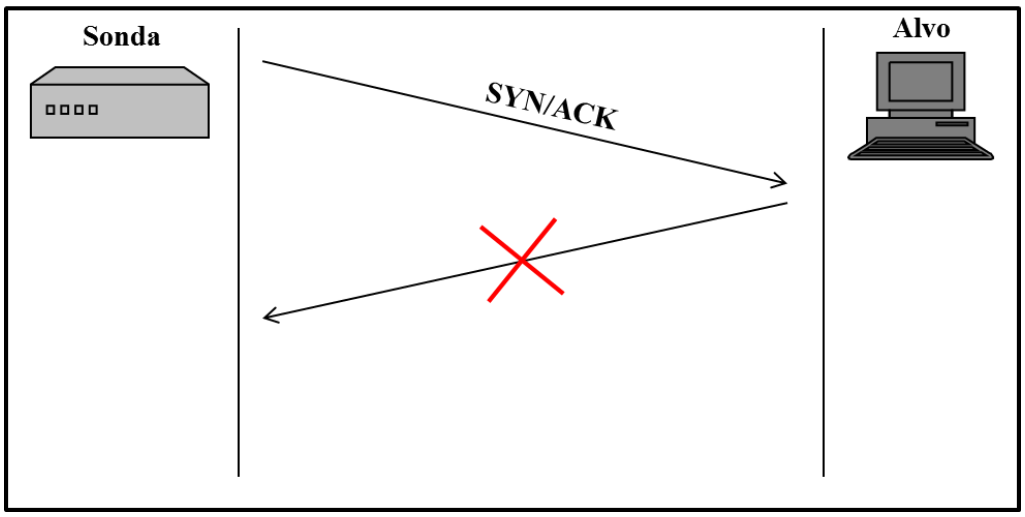

Figura 7 - Porto aberto ou filtrado com *SYN/ACK Scanning*

- *FIN Scanning*

Outra mais-valia na utilização da *Stealth Scanning* é o *FIN Scanning*, empregando também um mapeamento inverso de modo a descobrir os portos que se encontram fechados. Tal como o nome indica, é enviada uma mensagem para o alvo com a *flag FIN* activa, e no caso do porto se encontrar fechado é enviada do alvo para a sonda uma mensagem com a *flag RST*, no entanto, encontrando-se o porto aberto não surgirá qualquer resposta a uma mensagem sem antecedentes, tal como se verifica nas Figuras 8 e 9.

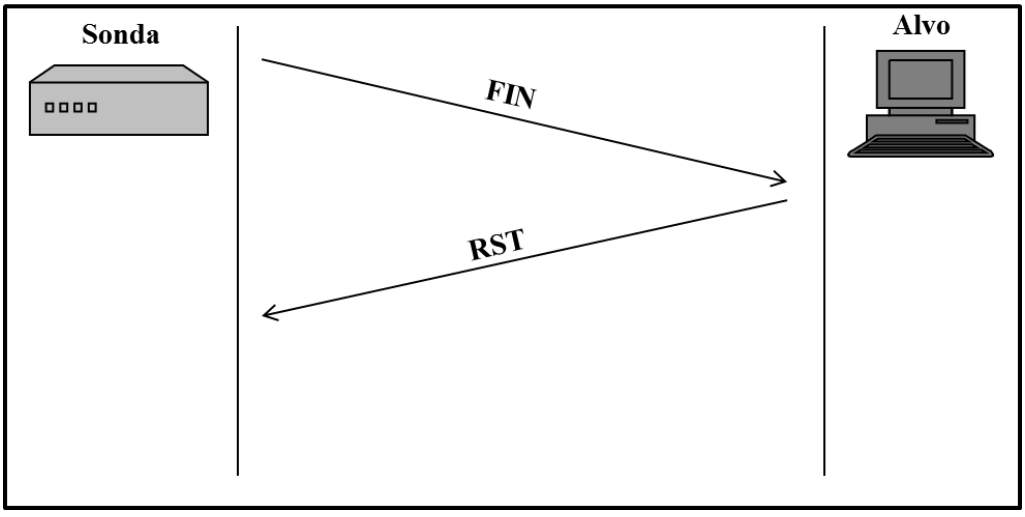

Figura 8 - Porto fechado com *FIN Scanning*

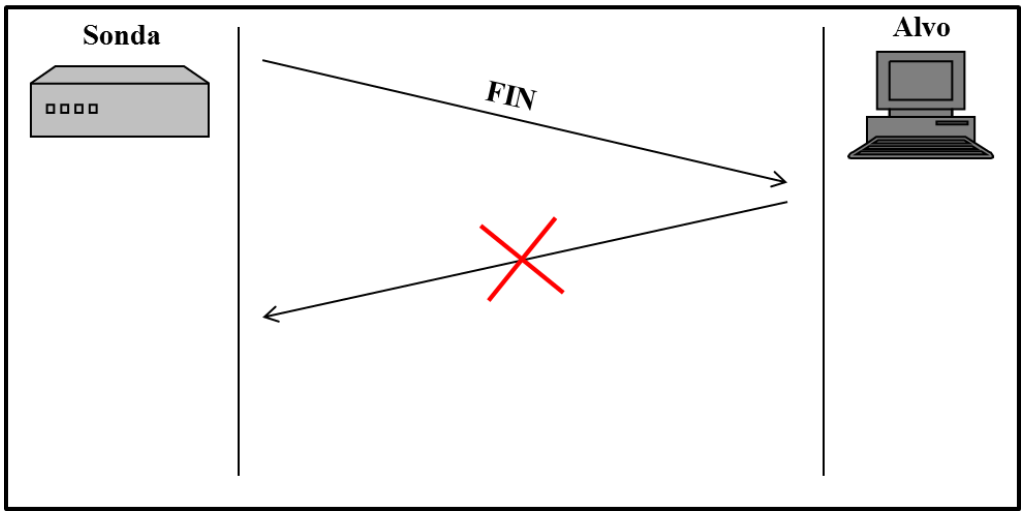

Figura 9 - Porto aberto ou filtrado com FIN Scanning

- *NULL Scanning*

Tal como a técnica de *FIN Scanning*, o *NULL Scanning* utiliza o mapeamento inverso para a descoberta do estado dos portos alvo, e deste modo, todas as *flags* (*ACK*, *FIN*, *RST*, *SYN*, *URG*, *PSH*) de uma mensagem são desactivadas. O resultado alcançado é semelhante ao verificado na técnica anterior, não obtendo qualquer tipo de resposta quando o porto se encontra aberto, no entanto, é recebida uma mensagem *RST* caso o porto se encontre fechado, conforme Figuras 10 e 11.

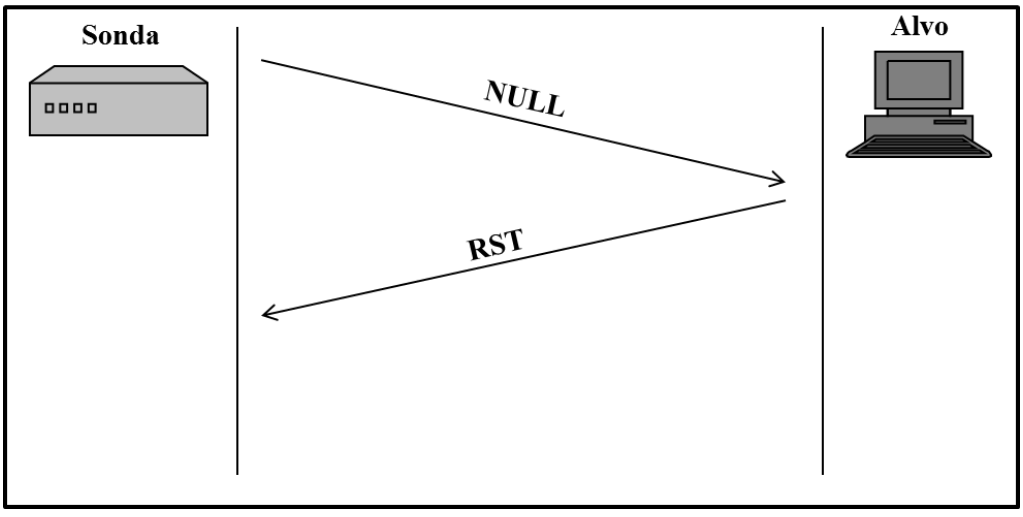

Figura 10 - Porto fechado com *NULL Scanning*

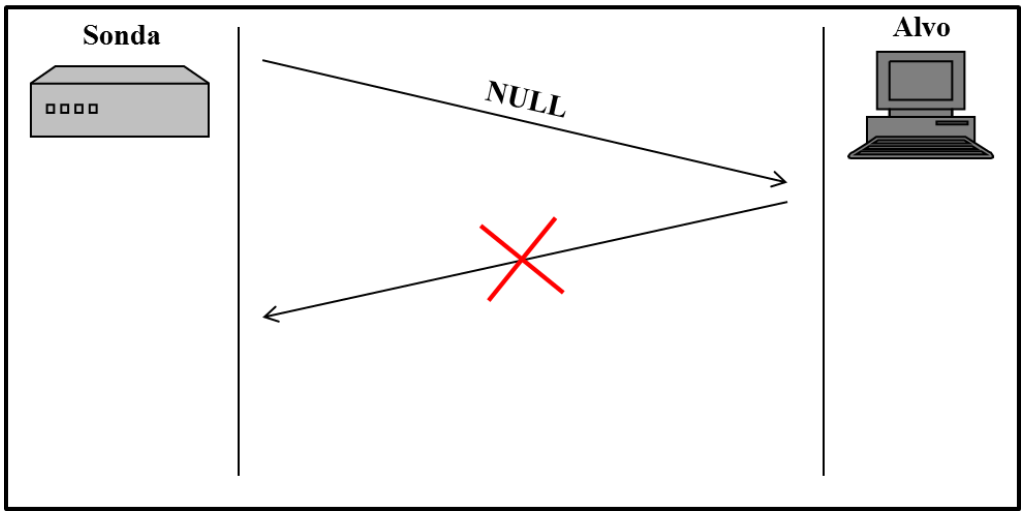

Figura 11 - Porto aberto ou filtrado com *NULL Scanning*

Por não existir um protocolo específico que indique como cada máquina deve responder a este tipo de mensagem, vários sistemas operativos podem responder de forma diferente.

- *XMAS Scanning*

Por fim, é apresentada uma técnica completamente inversa à técnica anterior, pois todas as *flags* se encontram activas na mensagem enviada da sonda para o alvo, mantendo o mesmo tipo de mapeamento inverso. Assim, assume-se que, se não for obtida qualquer resposta do alvo, o porto encontra-se aberto ou filtrado por uma *firewall,* caso contrário, ao receber uma mensagem *RST* deduz-se que o porto se encontra fechado, conforme Figuras 12 e 13.

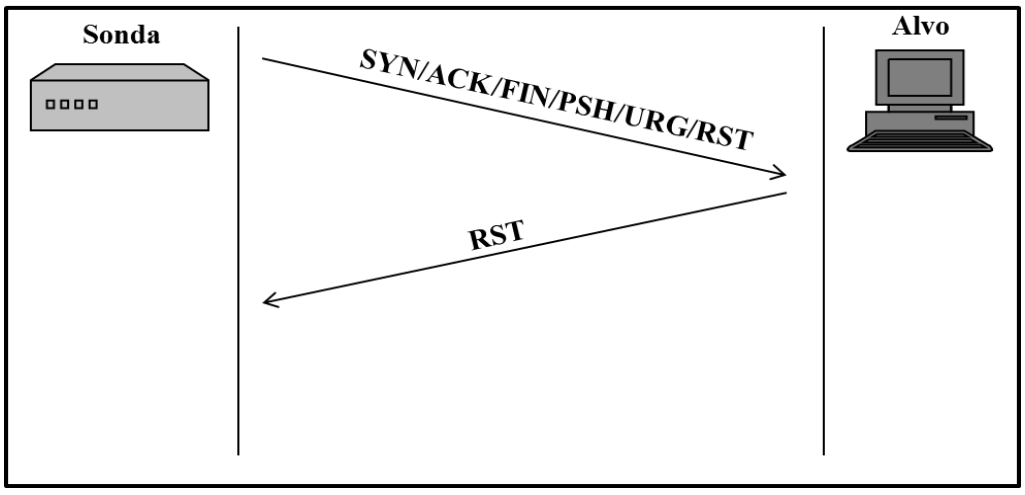

Figura 12 - Porto fechado com *XMAS Scanning*

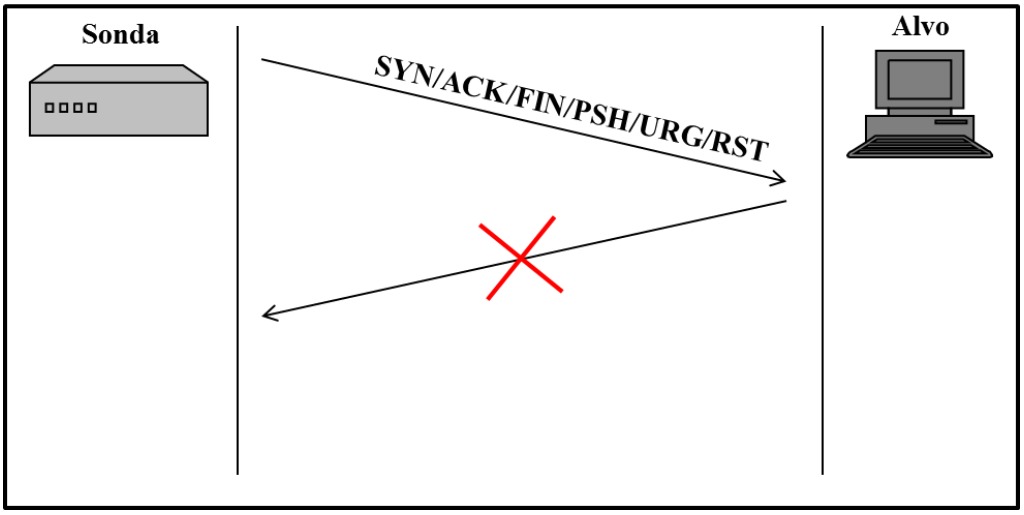

Figura 13 - Porto aberto ou filtrado com *XMAS Scanning*

Em suma, as técnicas de *Stealth Scanning* são rápidas e conseguem ultrapassar muitos dos sistemas de detecção de intrusões, no entanto, a produção de falsos positivos é uma desvantagem, no que diz respeito á descoberta exacta do tipo de serviços fornecido por cada porto.

3.2 Técnicas de Monitorização Passiva

As técnicas de monitorização passiva [7][8][10][12][13][14] são transparentes numa rede, pois não produzem tráfego adicional que possa provocar perdas no desempenho da mesma. É o melhor método para monitorizar a rede em tempo-real, sendo a análise do tráfego efectuada paralelamente à captura do mesmo.

As capacidades desta monitorização focam-se nos fluxos de tráfego, e nas componentes da rede que os produzem, sendo possível a descoberta de:

- Nós activos na rede;
- Sistemas Operativos;
- *Uptime* de uma máquina;
- Papel desempenhado por cada máquina (Servidor, *Switch*, Terminal, etc.);
- Serviços oferecidos (Servidor Web, *DHCP*, etc.);
- Protocolos suportados;
- Configurações da rede *IP*.

A descoberta de nós na rede é uma das capacidades mais importantes na monitorização passiva, embora só seja possível, caso os nós se encontrem activos e a produzir tráfego. Designadamente, o protocolo de resolução de endereços (*ARP*), é uma ajuda valiosa nesta descoberta, pois é um protocolo *broadcast*, emitido sempre que uma máquina necessita de obter o endereço físico de outra máquina, produzindo assim tráfego suficiente para identificar quais os nós se encontram activos, e até mesmo recolher alguma informação sobre o período de actividade dos nós.

A identificação do sistema operativo de uma máquina é essencial para a detecção de vulnerabilidades. A análise dos cabeçalhos de pacotes *TCP* permite essa descoberta, no entanto, pode-se também observar a camada aplicacional desse mesmo pacote, e procurar por expressões específicas que permitam identificar o sistema operativo.

Os protocolos de comunicação que cada máquina utiliza, são também uma capacidade da monitorização passiva. A inspecção do cabeçalho *IP*, nomeadamente, o campo *Protocol*, permite saber que tipo de protocolo está a ser utilizado na comunicação, e assim inferir o papel do dispositivo dentro da rede.

Descobrir os serviços que cada dispositivo disponibiliza, é possível também com monitorização passiva, através da análise dos campos de dados existentes nos fluxos, que possuem expressões regulares fixas para cada tipo de serviço. Assim, com uma base de dados de expressões, é possível inferir em tempo-real qual o serviço em utilização.

A análise da informação recolhida passivamente, pode ser efectuada através de vários mecanismos diferentes, sendo os mais simples e eficazes: *Singleton*, *Sample* e *Stimulus*.

3.2.1 *Singleton*

Este tipo de mecanismo é orientado para um pacote específico, capturado após a aplicação de filtros definidos tendo em vista o objectivo final. Este mecanismo é aplicado, por exemplo, quando um analista necessita de saber se existe alguma máquina que esteja a utilizar protocolos de transferência de ficheiros (*FTP*), para isso, basta capturar apenas um pacote enquadrável com esse objectivo.

3.2.2 *Sample*

Este mecanismo foca-se numa amostra de tráfego, definida por um filtro específico, cuja informação é capturada e guardada durante um certo período de tempo, por fim todos os dados são derivados. A utilidade deste mecanismo consiste em filtrar todo o tráfego de um determinado endereço durante um dia, e posteriormente analisar a

quantidade de dados recebidos e enviados nesse período de tempo através desse endereço. Tendo em conta a enorme quantidade de dados recebidos, pode eventualmente depreender-se que estamos perante um servidor.

3.2.3 *Stimulus-Response*

O mecanismo *Stimulus-Response* necessita de pelo menos dois pacotes para que retorne resultados satisfatórios. Neste caso, um estímulo enviado de uma máquina para outra, aguarda sempre uma resposta, no entanto, existe possibilidades de tal não acontecer, por estes factos é exequível inferir o estado das duas máquinas. Observando o comportamento de uma máquina, que envia uma mensagem (*Stimulus*) para um servidor, e este não responde (*Response*), pode concluir-se que a máquina se encontra activa mas o servidor não. Neste caso o protocolo *ARP* pode assumir o papel de estímulo, aguardando como resposta a resolução do endereço.

3.3 Monitorização Activa Vs. Monitorização Passiva

O facto de existirem máquinas protegidas por sistemas de detecção de intrusões, e máquinas inactivas na rede, implica uma complementaridade entre as duas técnicas de monitorização a activa e a passiva [4], e consequentemente ocorre uma melhor e maior descoberta de dispositivos na rede.

A monitorização activa envolve o envio de pacotes para um conjunto específico de endereços, produzindo tráfego acrescido e tornando-se intrusiva, não permitindo a detecção de máquinas protegidas por uma *firewall*, ou cuja sua actividade seja intermitente.

A monitorização passiva permite descobrir os serviços mais populares de um modo não invasivo, necessitando apenas da existência de tráfego na rede, descobrindo assim máquinas que se encontrem protegidas por *firewalls*.

Em suma, a monitorização activa é a técnica mais rápida para a localização de dispositivos na rede, a monitorização passiva é incapaz de observar máquinas com ausência de tráfego mesmo estando activas. Assim, estes dois tipos de monitorização são afectados fortemente por vários factores exógenos, tais como, a hora do dia em que são executados, a utilização de endereços dinâmicos na rede, e a quantidade de tráfego produzido na rede.

3.4 Métricas de Segurança

A definição de métrica [11][23][24][25] assenta na diferenciação entre métricas e medições. Uma medição é uma visão específica de determinados factores num certo ponto temporal, enquanto uma métrica baseia-se na análise de várias medições durante um certo período de tempo.

A gestão de actividades de segurança só pode ocorrer caso se efectuem medições, mostrando a eficiência dos vários componentes, produtos ou processos de segurança existentes, a análise dessas medições ajuda a identificar o nível de risco que possa surgir caso não sejam tomadas determinadas acções.

Através da observação das métricas de segurança numa empresa, é possível saber, com facilidade, se o nível de segurança da empresa é superior ao nível possuído anteriormente, e até mesmo se se encontra segura.

3.4.1 Técnicas de Produção

Uma boa métrica tem que ser: especifica, mensurável, viável, repetível, e possuir dependência temporal.

Não basta apenas medir o número de ataques de segurança efectuados com sucesso a uma empresa na última semana, não é um indicador suficiente do seu nível de segurança. Assim, é então necessária a análise de vários factores, tais como, o valor da empresa, o nível de ameaça, e as vulnerabilidades existentes. O valor que cada empresa possui deve ser o factor mais fácil de quantificar, enquanto o nível de ameaça não é trivial de analisar, pois na maioria dos casos só factores externos influenciam este valor.

Existem sete passos que guiam o processo de construção de métricas:

1. Definir o objectivo da métrica;
2. Decidir que métrica gerar;
3. Desenvolver estratégias para gerar a métrica;
4. Estabelecer alvos e unidades de avaliação;
5. Determinar como as métricas serão reportadas;
6. Criar um plano de acção;
7. Estabelecer um programa de refinamento cíclico;

3.4.2 Técnicas de análise de métricas

As métricas não são construídas através de um conjunto disperso de dados em bruto. É necessária uma filtragem, ordenação, análise e transformação desses dados de modo a que possuam um formato útil às necessidades empresariais. Assim, um processo iterativo de análise de informação relevante passa por recolher dados em bruto, de seguida transformá-los em informação útil e, por fim, realizar uma análise aprofundada dessa mesma informação.

Para que esta análise seja exacta existem várias técnicas estatísticas comuns, tais como:

- Média
- Mediana
- Desvio Padrão
- Análises de Series Temporais
- Análise de Quartis

Serão descritos os conceitos chaves de cada técnica, indicando as potenciais vantagens quando aplicadas à análise de medições, para produção de métricas.

Média

Em termos estatísticos a Média é uma técnica de agregação *standard*, que devido ao seu tipo de cálculo, torna fácil a sua compreensão. Para obter a média, é necessário um conjunto de elementos, que somados ditam um valor a ser dividido pelo número total de elementos. O valor obtido é o valor médio desse conjunto.

Este tipo de técnica pode induzir um erro comum, pois não é especificada a granularidade que a informação pode ter. Temos como exemplo o conjunto {3,3,3,3,3} possuindo o valor médio 3, no entanto o conjunto {1,2,3,4,5} também tem o mesmo valor médio. Logo, não é possível através da média verificar o tipo de variação que os dados possuem, dando uma falsa impressão dos valores típicos.

Mediana

De modo a evidenciar qual o valor típico de um conjunto de informação, é preferível utilizar a Mediana ao invés da Média, pois esta técnica separa o conjunto em

metade após estar ordenado, e indica qual o valor que separa a metade superior da metade inferior.

Por exemplo, um caso onde é necessário especificar quantos segundos são necessários para decifrar uma palavra-chave de um utilizador, é comum obter uma informação do tipo, "A duração média é de 5060 segundos para decifrar 10 palavras-chave", no entanto, este valor não fornece uma informação concreta, pois vários utilizadores podem possuir uma palavra-chave extremamente difícil de decifrar, o que iria aumentar substancialmente o valor médio. Neste caso, a mediana permite definir quanto tempo demora a decifrar a metade mais fraca do conjunto de palavras-chave, obtendo um valor inferior ao valor médio. Este valor é mais próximo do valor típico pois não sofre demasiadas alterações caso existam palavras-chave de complexidade elevada, oferecendo mais vantagens que a média.

Desvio Padrão

O desvio padrão (σ) permite medir a dispersão de um conjunto de dados relativamente à sua média, ou seja, estabelece se um determinado conjunto de dados tende a aglomerar ou a dispersar, obtendo respectivamente um desvio padrão baixo para o primeiro caso e um desvio padrão alto para o segundo caso.

Esta técnica resume-se inicialmente ao cálculo da média de um conjunto de elementos. De seguida é realizado o cálculo do quadrado da diferença entre cada elemento e a média. Após este passo executa-se a soma dos valores anteriormente obtidos e divide-se pelo número de elementos, obtendo a variância (σ^2). No entanto o valor que nos dará o desvio padrão pretendido será o resultado da raiz quadrada da variância (σ).

Elemento	Valor x_i	Quadrado da diferença entre o valor e a média $(x_i - 7)^2$
1	2	25
2	6	1
3	15	64
4	11	16
5	3	16
Variância	30,5	
Desvio Padrão	5,53	

Tabela 1 - Exemplo do cálculo do Desvio Padrão

Através da Tabela 1, denota-se que o desvio padrão é aproximado do valor médio (7), o que indica que é um desvio padrão alto, apresentando os seus elementos uma posição dispersa no conjunto.

Análise de Series Temporais

Esta técnica foca-se na variação que a informação apresenta ao longo de um período de tempo, observando um determinado atributo medido em intervalos específicos. Este tipo de análise deve possuir um intervalo de observação significativo e passível de ser ajustável a cada caso em estudo possibilitando uma observação coerente, por exemplo, se a análise for ao nível de segurança e depender do factor exógeno ambiente, faz sentido efectuar uma medição diária ou semanal.

Após serem definidos os intervalos de observação, é necessário agregar a informação relativa a esses intervalos, e dentro de cada intervalo efectuar a ordenação por data.

Uma estratégia a adoptar para uma representação da informação de ataques a vulnerabilidades de uma organização seria:

- Data de detecção: Agrupar por mês;
- Nome da aplicação: Única;
- Exploração: Média e Desvio Padrão;
- Impacto da exploração: Médio e Desvio Padrão;
- Custo para recuperação: Média e Desvio Padrão.

A criação de uma tabela com este tipo de informação ordenada, permite uma análise minuciosa para uma tomada de decisões coordenada, podendo-se inferir:

- O número de defeitos de segurança identificados reduziu de *X* para *X-4* sugerindo que a segurança da aplicação aumentou ao longo desse período;
- Embora o número de defeitos tenha reduzido, no fim do período de observação foram detectados alguns defeitos com um impacto elevado;
- O custo para a recuperação da aplicação aumentou ao longo do tempo, tal como o seu desvio padrão, bem como a consistência dos problemas mais complexos.

Em suma, esta análise necessita que os dados estejam ordenados, pois as medições possuem um elevado grau de dependência dentro do período de tempo em que ocorrem, sendo uma das técnicas de análise com melhor posicionamento para efectuar previsões futuras.

Análise de Quartis

De todas as técnicas referenciadas anteriormente esta é uma das mais poderosas em termos de produção de métricas. É assim necessária uma boa estratégia na agregação do tipo de informação recolhida, assumindo para este efeito que os valores mais altos são os melhores.

Esta técnica divide a informação em 4 divisórias ou quartis, sendo o primeiro quartil a representação de 25% dos melhores resultados, o segundo quartil 50%, o terceiro quartil 75%, e o último quartil representa os piores 25%. Esta representação possui uma técnica já referida anteriormente, a mediana, que divide os dois primeiros quartis dos dois últimos, e que por sua vez é utilizada respectivamente na divisão entre o primeiro e segundo quartil e o terceiro e o quarto quartil, como representado na Tabela 2.

98 **90** **89** **89**	Primeiro Quartil
87 **86** **84** **80**	Segundo Quartil
48 **46** **38** **33**	Terceiro Quartil
32 **24** **8** **5**	Quarto Quartil

Tabela 2 - Representação de quartis

Efectuando um sumário dos quartis através do uso das suas medianas, é possível a observação de qual o desempenho de cada quartil em relação aos outros. Assim, se for necessário uma abordagem relativamente a decisões críticas, é possível apenas observar a representação do primeiro e do último quartil, tirando conclusões rápidas com a informação disponível.

3.5 Resumo

Este capítulo descreveu os conteúdos teóricos intrinsecamente ligados ao projecto, focando diversas técnicas da monitorização activa e passiva, que foram utilizadas na concretização da solução. Foram também descritas as características necessárias para a utilidade de uma métrica. O tema da monitorização será aprofundado no capítulo seguinte, com a descrição de várias ferramentas *open-source* simples e eficazes, utilizadas na implementação da solução.

Capítulo 4

Ferramentas e Plataformas utilizadas

Durante a investigação exploratória do projecto, várias ferramentas demonstraram potencialidades relativamente aos objectivos estabelecidos, que se centravam na captura e análise de tráfego de rede em tempo-real.

Relativamente à monitorização activa a ferramenta escolhida foi o *nmap,* enquanto o *p0f,* e o *IPAudit,* foram as escolhas na vertente de monitorização passiva.

Outras ferramentas foram investigadas, tais como o *PADS,* o *DART* e o *SNORT* [22] mas algumas não foram inseridas na solução devido a problemas funcionais ou de integração.

4.1 *Nmap*

O *nmap* é uma ferramenta gratuita e *open-source* para descoberta de redes, e auditorias de segurança. Esta ferramenta entre outras características não menos importantes, utiliza pacotes *IP*, de modo a descobrir os nós activos dentro de uma rede, quais os tipos de serviços disponibilizados, e o sistema operativo que possuem. Foi desenvolvido com o intuito de correr em redes de grande dimensão.

É uma ferramenta flexível, pois suporta várias técnicas de mapeamento da rede, sendo utilizada para monitorizar milhares de máquinas em simultâneo. A sua portabilidade é uma vantagem, pois corre na maioria dos sistemas operativos, e qualquer utilizador encontra facilidades na sua utilização.

- *Funcionamento*

O *nmap* possui uma interface gráfica simples de utilizar, e uma interface de linha de comandos [16], sendo nesta última que nos iremos focar.

A opção mais simples de utilização será escolher um ou vários endereços *IP* e executar o comando *nmap <a.b.c.d.e/(0-32)>*. Após a execução do comando, o *nmap*

envia pacotes *IP* em paralelo para vários portos dos respectivos endereços, aguardando a resposta dos mesmos, tal como descrito na Figura 14.

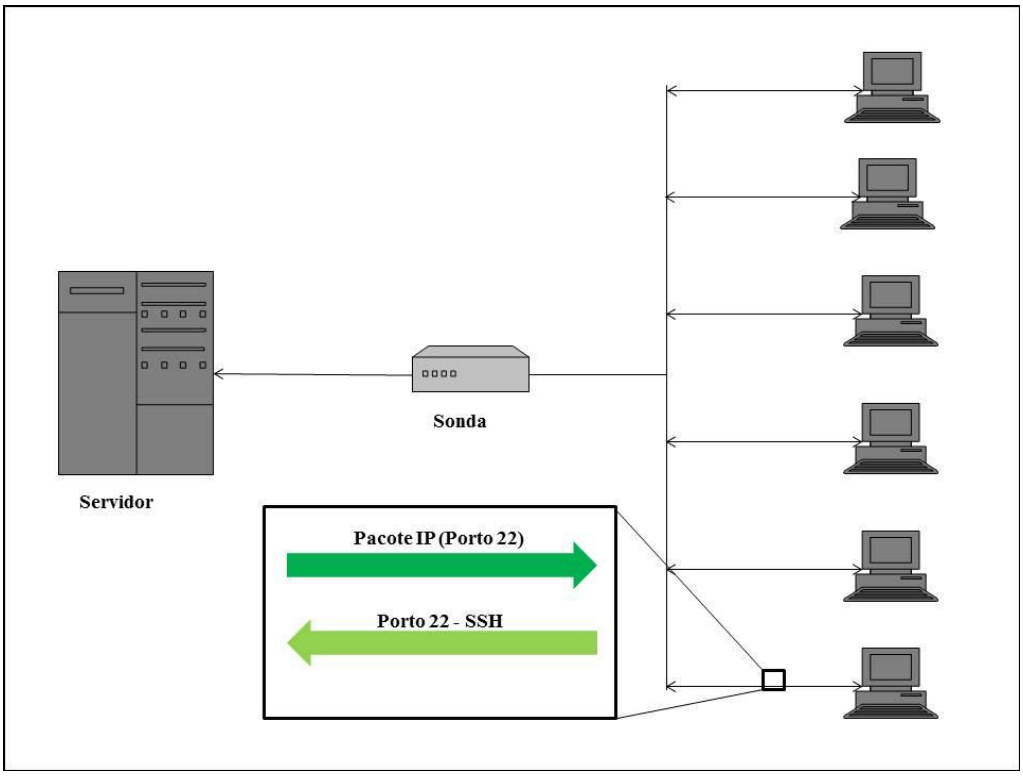

Figura 14 -Funcionamento do *nmap*

Existem outras variantes de execução que permitem um bom desempenho, contudo serão especificadas no capítulo 6.

- *Dados*

A informação recolhida não possui um grande nível de profundidade, no entanto, é já possível saber se a máquina se encontra activa ou não.

Para que o *nmap* forneça informação mais detalhada, é necessário acrescentar um maior número de opções ao comando base, como por exemplo:

- *-O* - Inferir o sistema operativo;
- *-sV* – Descoberta do nome e versão dos serviços disponíveis;
- *-iL* – Introduzir uma lista de vários endereços para monitorizar;
- *-sS (TCP SYN)* – Envio de mensagens *TCP* com a *flag SYN* activa;

- *-sU (UDP SYN)* – Envio de mensagens *UDP* com a *flag SYN* activa;
- *-sN (TCP NULL)* – Envio de mensagens *TCP* nenhuma *flag* activa;
- *-sF (TCP FIN)* – Envio de mensagens *TCP* com a flag *FIN* activa;
- *-sX (TCP XMAS)* – Envio de mensagens *TCP* com todas as *flags* activas;
- *-T<0-5>* - Define a velocidade de envio das mensagens;
- *–PS* – Envia pacotes *ICMP* de modo a obter o estado do endereço.

Com as opções certas, é possível obter com um determinado nível de confiança um conjunto de sistemas operativos que a máquina em questão possa utilizar, bem como, a sua probabilidade. É também exequível adquirir informação sobre todos os portos que se encontram abertos, obtendo os nomes dos serviços disponíveis e as suas versões.

Existem ainda dezenas de opções orientadas ao contorno de *firewalls* e sistemas de detecção, sendo as acima referidas, as mais simples para a produção de dados concretos.

Com toda a quantidade de informação recolhida, o *nmap* tem ainda a hipótese de a guardar estruturalmente, em diversos formatos de ficheiros de texto, sendo o *XML* o formato de eleição.

O grande problema em utilizar o *nmap* é a grande quantidade de tráfego produzido na rede, sendo no entanto possível mitigar esse problema, diminuindo o número de pacotes enviados por segundo, mas perdendo na rapidez de descoberta.

Refere-se que, o *nmap* é considerado a melhor ferramenta de monitorização activa nos dias de hoje, quer a nível funcional, quer a nível da sua base de conhecimento.

4.2 *p0f*

Tal como o *nmap*, o *p0f* é uma ferramenta gratuita e open-source, permitindo analisar as *fingerprints* (campos específicos nos pacotes capturados) dos sistemas operativos de máquinas que se ligam a nós, e também de máquinas às quais nos ligamos. Consegue identificar também, que tipo de ligação as máquinas possuem (*Ethernet*, *DSL*, *etc.*), a sua distância na rede, e até mesmo o seu tempo de actividade.

O *p0f* não gera qualquer tipo de tráfego adicional dentro da rede, tornando-o numa boa ferramenta de monitorização passiva.

- *Funcionamento*

Existem dois tipos de funcionamento do *p0f*. O primeiro armazena a informação num ficheiro de texto simples de analisar. O segundo tipo de funcionamento envolve a criação de um *socket* de sistema que armazena toda a informação capturada em memória.

O *p0f* pode utilizar filtros de tráfego, que são extremamente úteis quando se pretende excluir pacotes específicos, ou até mesmo excluir todo o tráfego de uma certa rede. Estes filtros são essenciais na aplicação de técnicas de monitorização passiva como o *Singleton* e *Sample*. A Figura 15 apresenta esquematicamente o funcionamento do *p0f.*

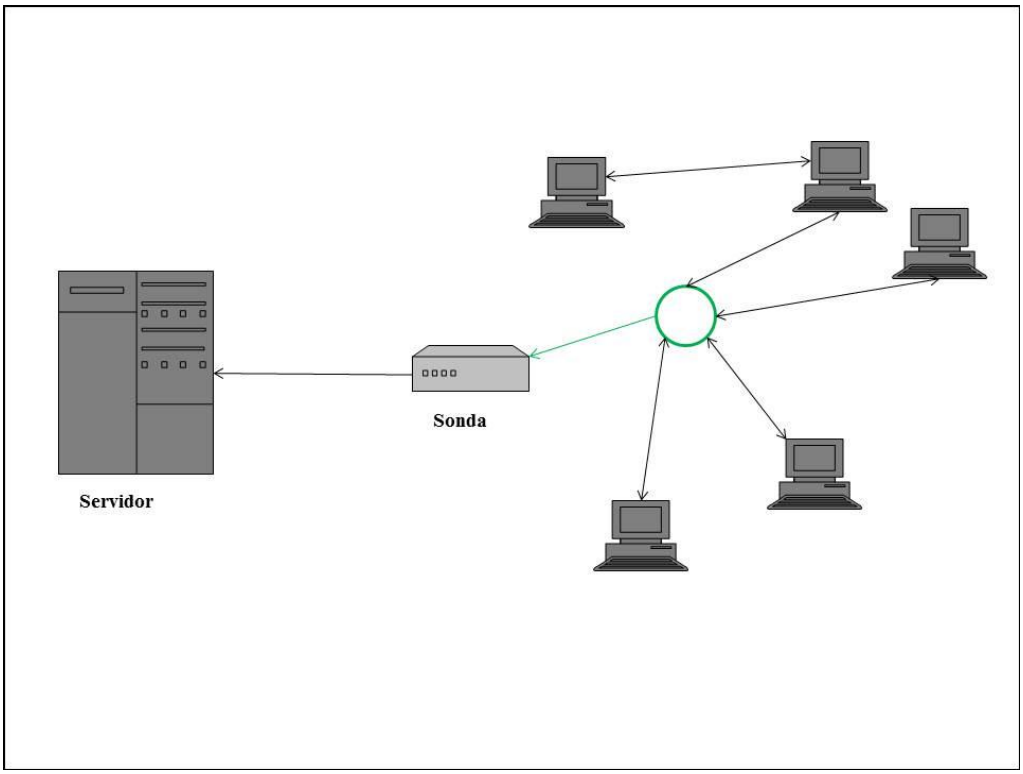

Figura 15 - Funcionamento do *p0f*

- *Dados*

A informação armazenada em memória permite a consulta de vários dados referentes a cada endereço. Os dados possíveis de obter, entre outros, são o número de fluxos efectuados por cada endereço, o sistema operativo utilizado, e o período de tempo que a máquina se encontra ligada.

A informação relativa a cada endereço é mantida em memória, apenas se for capturado passivamente algum fluxo num certo intervalo temporal. Caso contrário, a informação é descartada.

Devido à sua simplicidade e objectividade, o *p0f* é uma ferramenta extremamente útil para uma empresa com necessidade em identificar de modo passivo os sistemas operativos que estão instalados nas suas máquinas.

4.3 *ipaudit*

O *ipaudit* monitoriza a actividade de uma rede, identificando quer as máquinas intervenientes quer os portos e protocolos utilizados por elas, durante os processos de comunicação.

Captura todas as conexões entre dois endereços, monitorizando o início e o fim da ligação, todas as características pertencentes a essa ligação, bem como o porto e o protocolo utilizado.

Esta ferramenta pode ser utilizada para detectar intrusões, ataques de negação de serviço e monitorizar o consumo da largura de banda utilizada.

Encontra-se inserida na classe das ferramentas de monitorização passiva tal como o *p0f*, logo não é passível de produzir tráfego na rede.

- *Funcionamento*

A captura de informação pelo *ipaudit*, depende da existência de duas máquinas que produzam tráfego entre elas. Assim, o seu funcionamento é em tudo semelhante ao *p0f*, excepto no tipo de informação recolhida. A Figura 16 representa de forma esquemática o funcionamento do *ipaudit.*

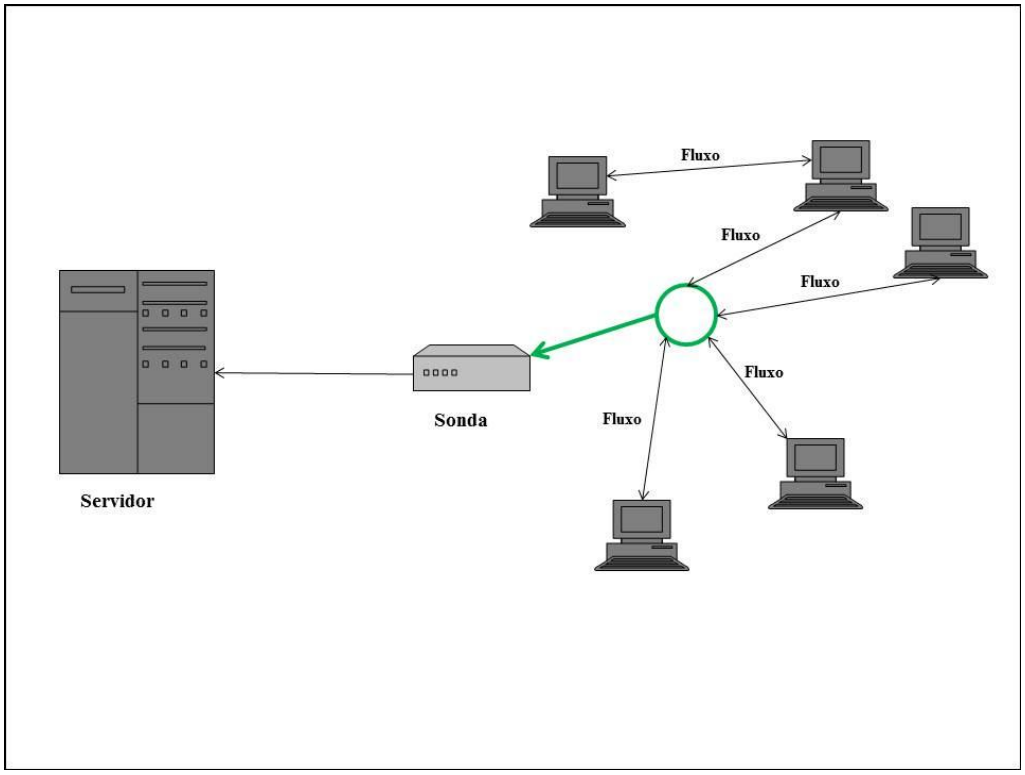

Figura 16 - Funcionamento do *ipaudit*

O *ipaudit* possui um ficheiro de configuração que permite definir as opções de captura ao invés de as adicionar ao comando de execução. É possível inserir opções, tais como, as que limitem o número de pacotes capturados, definir um filtro de tráfego, escolher a interface de captura.

O exemplo seguinte demonstra as possibilidades de configuração:

- *count* 500000 – Número de pacotes capturados para cada ficheiro;
- *ethernet* on – Mostrar o endereço físico;
- *interface* eth4 – Interface em que o *ipaudit* se encontra à escuta;
- *outfile* /path/caminho/ - Directório onde os *logs* serão colocados;
- *shortip* – Representação do endereço *IP* sem zeros à esquerda;
- *writetime* on – Representar o momento exacto da captura do fluxo.

- *Dados*

A informação capturada pelo *ipaudit* reflecte os dados referentes, aos portos e protocolos de comunicação, aos instantes temporais de início e de fim da ligação e de quem a iniciou, a quantidade de *bytes* e pacotes recebidos por cada máquina, e o

endereço físico respectivo. Esta informação é guardada num *log* em formato de texto, correspondendo cada linha a um fluxo de comunicação entre duas máquinas.

4.4 Ferramentas Relacionadas

Existem outras ferramentas direccionadas à monitorização de redes que se englobam na vertente *open-source*, tendo sido algumas desenvolvidas dentro da PT Comunicações.

As ferramentas descritas em seguida não estão presentes na solução final devido a problemas de integração, ou devido às suas funções não se inserirem na presente versão da solução.

- *PADS*

É uma ferramenta passiva baseada em detecção de assinaturas, desenhada para complementar os sistemas de detecção de intrusões.

Desempenhando um papel passivo, não produz qualquer tipo de tráfego na rede, e pode ser facilmente colocada numa máquina remota.

Possui uma grande capacidade em identificar os protocolos (*www, icmp, etc*.), e os nomes das aplicações (*Apache, OpenSSH*, etc.) que as máquinas produtoras de tráfego utilizam.

É possível modificar o ficheiro que contém as assinaturas, de modo a inserir definições de protocolos e aplicações exclusivas da rede empresarial.

Esta ferramenta não é actualizada desde 2005, o que a impede de ter uma boa base de assinaturas, e impossibilita a sua utilização em arquitecturas de 64-*bits*.

- *DART*

Desenvolvida na PT Comunicações, tem como objectivo detectar em tempo-real eventos de *resets* e *timeouts*.

Existe pouca documentação acerca desta ferramenta, mas a sua utilidade não passa despercebida num ambiente onde a eficiência e a disponibilidade são os principais factores de sucesso.

Como a vertente desta ferramenta é a alarmística, talvez seja inserida futuramente.

- *SNORT*

É uma ferramenta de detecção de intrusões na rede, permitindo ainda uma análise de tráfego em tempo-real, análise de protocolos, e detecção de ataques (*buffer overflow*, *stealth port scan, OS fingerprint*, etc.).

Utiliza uma linguagem de regras, de modo a filtrar o tráfego de acordo com as configurações.

Devido à sua eficiência e simplicidade, é uma das ferramentas de detecção de intrusões *open-source* mais utilizada no mundo.

Como o *SNORT* se foca principalmente na detecção de intrusões, recai na temática da alarmística, sendo assim trabalho a desenvolver no futuro.

4.5 Resumo

Este capítulo focou-se na descrição detalhada do funcionamento das ferramentas utilizadas no desenvolvimento e implementação da solução final. Foram assim apresentadas ferramentas pertencentes quer ao grupo da monitorização passiva, quer ao grupo da monitorização activa, identificando-se as funcionalidades e as vantagens de cada uma.

Foi também efectuada uma abordagem de forma descritiva de algumas ferramentas que poderão desempenhar um papel importante numa solução futura, apontando para a temática da alarmística em tempo-real.

No capítulo seguinte, a arquitectura e o funcionamento da solução serão explicados detalhadamente, de acordo com os métodos de integração das ferramentas acima descritas.

Capítulo 5

Solução Desenvolvida

De modo a cumprir os objectivos propostos para este projecto, foi desenvolvida e concretizada uma solução final em *Perl* [17] que tem em conta a integração no sistema Pulso da PT Comunicações.

A solução foi assim concebida para automatizar a captura e análise de tráfego dentro da rede PT, fornecendo informação detalhada e relevante em tempo-real, de modo a inferir o estado dos dispositivos pertencentes á rede.

Numa primeira fase, foram testadas as ferramentas que compõem a solução num pequeno conjunto de máquinas alvo, não críticas e conhecidas. Numa segunda fase foi capturada e analisada uma grande quantidade de tráfego correspondente a um *Data Center* da PT Comunicações.

Neste capítulo serão explicados, os passos do desenvolvimento da solução, tal como todos os seus elementos constituintes e o seu funcionamento.

5.1 Arquitectura

O foco principal da solução incide na captura e análise do tráfego produzido internamente na rede PT, assim, são necessários dispositivos que capturem esse tráfego em pontos estratégicos.

As sondas de rede encontram-se distribuídas, não efectuando qualquer tipo de comunicação entre elas, mas sim com um servidor central que irá armazenar e apresentar graficamente toda a informação previamente analisada.

Estas sondas pertencem à plataforma Pulso, e são máquinas com recursos reduzidos, apenas com o intuito de captura de tráfego.

A solução inicial foi construída com base numa arquitectura orientada a objectos, através de um cliente desenvolvido em *Java*, que analisava toda a informação capturada pelas ferramentas de monitorização, e transformava essa informação em objectos

serializáveis transferindo-os na rede até ao servidor central, também desenvolvido em *Java* que processava os objectos recebidos e utilizava os dados necessários, conforme apresentado na Figura 17.

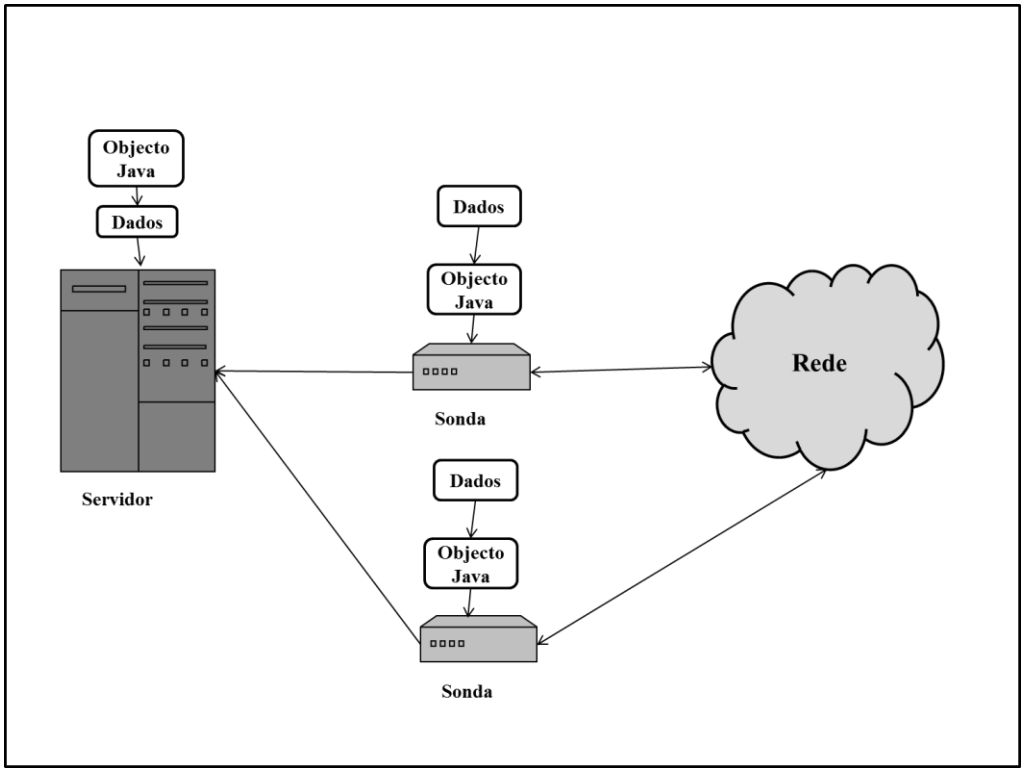

Figura 17 - Arquitectura inicial da solução

No entanto este tipo de implementação não demonstrou resultados positivos no que diz respeito a tempos de processamento, pois a quantidade de informação era substancialmente grande, e a transformação dessa informação em objectos despendia um tempo incomportável para a prática de uma monitorização em tempo-real.

Foi então ponderada uma nova implementação, com uma arquitectura simplificada, que não dependesse de objectos como factor intermediário, mas apenas utilizasse a informação capturada num formato agilizado.

Houve necessidade de uma abordagem menos pesada em termos de programação, descartando a anterior, optando-se por utilizar o *Perl* para a implementação da solução, dado ser prática comum dentro do departamento pois é uma linguagem com bibliotecas simples e integráveis com as ferramentas utilizadas.

Assim, a solução foi dividida em três módulos, sendo os dois principais, o módulo correspondente à monitorização passiva, e o módulo para uma monitorização activa. Estes módulos são independentes no seu funcionamento, pois a monitorização passiva

encontra-se sempre em execução, enquanto a monitorização activa é efectuada em determinados períodos de tempo.

Inseridas no módulo passivo, estão o *p0f* e o *ipaudit*, que possuem uma ligação lógica para melhor entender as características do tráfego produzido na rede.

O módulo activo tem como único constituinte o *nmap*, podendo ser executado automaticamente, ou a pedido, dependendo da quantidade e qualidade de informação pretendida. Existe também um outro módulo, que possui métodos úteis para melhor formatar a informação extraída pelas ferramentas de captura.

Por fim, estes módulos comunicam com um servidor central que permitirá a agregação e visualização de toda a informação recebida.

5.2 Concretização

Como explicado anteriormente no ponto 5.1, a solução encontra-se dividida em três módulos, dois para monitorização (*active.pm* e *passive.pm*), e um para formatação e configuração (*utils.pm*), tal como representado na Figura 18.

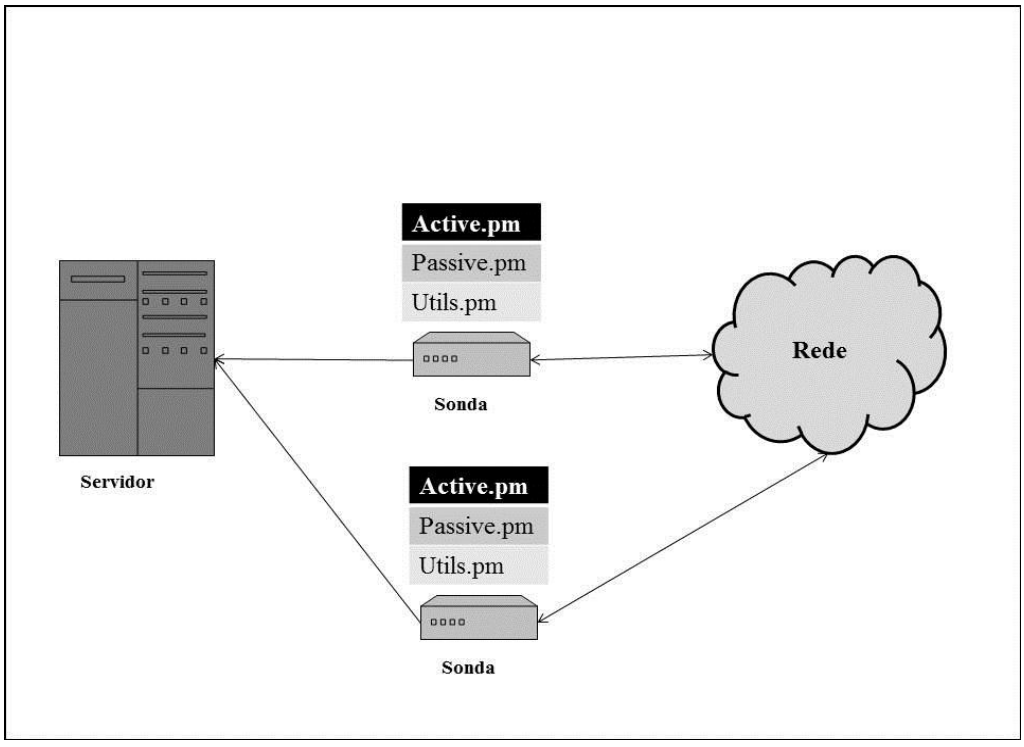

Figura 18 - Módulos *perl* utilizados na solução

5.2.1 *active.pm*

O módulo de monitorização activa da solução é composto exclusivamente pela implementação da ferramenta *nmap*, centrando-se na descoberta de serviços disponíveis nos portos das máquinas alvo e no seu sistema operativo.

- *Funcionamento*

Para que o módulo *active.pm* funcione, é necessário escolher o subconjunto de endereços alvo a partir de um conjunto previamente definido através dos diversos tipos de *range* que existem dentro da rede PT. Deste modo, é possível escolher um conjunto de endereços pertencente a um determinado edifício da PT, seja qual for a sua localização.

Após a escolha dos alvos, o *nmap* inicia o envio de pacotes, e aguarda a resposta aos mesmos, formatando toda a informação num ficheiro do tipo *XML*, conforme consta na Figura 19.

```
<address addr="10.101.3.230" addrtype="ipv4"/>

<address addr="00:1F:29:6E:4E:28" addrtype="mac" vendor="Hewlett-
Packard Company"/>

<port protocol="TCP" portid="25">

<state state="open" reason="syn-ack" reason_ttl="64"/>

<service name="smtp" product="Sendmail" version="8.13.3 rev 1.000"
hostname="ajpixs04.tmn.pt" ostype="HP-UX" method="probed" conf="10">

<osmatch name="HP HP-UX B.11.00 - B.11.31" accuracy="100"
line="23104">

<osclass type="general purpose" vendor="HP" osfamily="HP-UX"
osgen="11.X" accuracy="100">
```

Figura 19 - Informação recolhida pelo *nmap*

Concluída a recolha de dados, o módulo utiliza uma biblioteca já existente (*NMAP::PARSER)* [26] analisando apenas a informação mais relevante, colocando-a num formato simples (*String*), e enviando-a de seguida para o servidor.

Como a quantidade de endereços alvo pode atingir as centenas, é necessário estabelecer um limite numérico, pelo facto do tempo de monitorização e analise poder assumir tempos incomportáveis. Assim, o ideal é cada bloco possuir apenas informação

relativa a 256 máquinas, ou seja, cada ficheiro que o *nmap* produz não deve ultrapassar esse valor de dispositivos. No entanto, verificando-se um aumento da quantidade de máquinas alvo, o desempenho das sondas é afectada.

Todos os ficheiros são guardados na sonda durante um período de tempo, pois é necessário manter um histórico caso existam problemas de conectividade, sendo possível uma sincronização após a resolução do problema.

- *Problemas de integração*

O *nmap* é uma ferramenta cujo funcionamento básico produz quantidades enormes de tráfego, o que pode levar a uma queda de desempenho nas sondas onde é executado. Assim, o desafio foi configurá-lo de modo a diminuir a quantidade de tráfego produzido por segundo de acordo com as capacidades da sonda, sem aumentar demasiado o tempo de monitorização.

Existe um desfasamento relativamente ao conceito de tempo-real, pois é possível efectuar uma monitorização num dispositivo com determinadas características, e logo de seguida essas características mudarem, não existindo informação pontual desse facto.

5.2.2 *passive.pm*

Ao contrário do módulo anterior (*active.pm*) constituído apenas por uma ferramenta, o módulo passivo possui como elementos constituintes o *p0f* e o *ipaudit.*

Estas duas ferramentas estão interligadas, pois a informação capturada pelo *ipaudit* é utilizada para efectuar uma correspondência com os dados existentes na base de informação do *p0f.*

O *ipaudit* captura todos os pares de endereços que compõem uma ligação, permitindo assim identificar um fluxo de dados entre dois pontos.

O *p0f* captura passivamente o mesmo tráfego que o *ipaudit*, no entanto, utiliza um ficheiro com *fingerprints* que lhe permite identificar, por exemplo, o sistema operativo utilizado pela máquina que produziu o tráfego capturado.

A sua execução é permanente, permitindo uma monitorização em tempo-real.

- *Funcionamento*

Após a execução do módulo *passive.pm*, o *ipaudit* adquire um comportamento silencioso, guardando todos os fluxos capturados em *logs* de tamanho fixo (neste caso 500.000 pacotes), analisando a informação relativa a fluxos desde a quantidade de *bytes* recebidos à quantidade de pacotes enviados, remetendo-a posteriormente para o servidor central. O tamanho dos *logs* está estabelecido em 500.000 pacotes, pois, como se verificará no capítulo seguinte, a relação captura/análise tem de encontrar um equilíbrio, efectuando a captura de tráfego em tempo-real, desde que a sua análise não se afaste desse conceito. Estes *logs* são ficheiros de tamanho variável sendo cada linha a representação de um fluxo de dados entre duas máquinas, sendo colocados numa fila de espera para serem analisados, e posteriormente apagados.

```
010.101.001.060 010.101.003.228 6 62809 15000 1043838 862934 4772 4783
2012-08-08-10:33:02.0499 2012-08-08-10:33:09.5177 2 1 001f2932d5b4
001f296eae2a
```

010.101.001.060 –	Endereço envolvido no fluxo
010.101.003.228 –	Endereço envolvido no fluxo
6 –	Protocolo utilizado no fluxo
62809 –	Porto utilizado pelo primeiro endereço
15000 –	Porto utilizado pelo segundo endereço
1043838 –	Número de *bytes* recebidos pelo primeiro endereço
862934 –	Número de *bytes* recebidos pelo segundo endereço
4772 –	Número de pacotes recebidos pelo primeiro endereço
4783 –	Número de pacotes recebidos pelo segundo endereço
2012-08-08-10:33:02.0499 –	Momento exacto de captura do primeiro pacote
2012-08-08-10:33:09.5177 –	Momento exacto de captura do último pacote
2 –	Endereço que iniciou o fluxo
1 –	Endereço que finalizou o fluxo
001f2932d5b4 –	Endereço físico do primeiro interface
001f296eae2a -	Endereço físico do segundo interface

Figura 20 - Captura típica de informação pelo *ipaudit*

Como se pode verificar pelo excerto de informação apresentada na Figura 20, alguns dados não estão formatados por definição, e por isso é utilizado um módulo específico para este tipo de problema sendo descrito posteriormente.

Para uma execução óptima do *p0f*, optou-se por configurar um *socket* de sistema que captura-se o tráfego, e guarda-se em memória a informação relativa a cada endereço. Assim não foram utilizados *logs*, passíveis de ocupar demasiado espaço em

disco, usando-se uma memória de fácil gestão eliminando endereços antigos que não possuam referências a tráfego recente.

A informação guardada pelo *p0f* possui, o sistema operativo da máquina caso exista uma correspondência no ficheiro de *fingerprints*, o número total de fluxos quer de entrada quer de saída para determinado endereço, tal como outro tipo de informações relevantes.

A interligação entre o *ipaudit* e o *p0f* encontra-se nos endereços capturados pela primeira ferramenta, sendo utilizados na análise feita pelo *p0f*. Assume-se que o tráfego capturado pelas duas ferramentas é o mesmo, nesse caso o p0f tem informação relativa aos endereços constituintes dos fluxos capturados pelo *ipaudit*.

A cada análise dos *l*ogs do *ipaudit*, é efectuada uma *query* ao *p0f* através do seu *socket* de sistema, que permite identificar o sistema operativo utilizado pelo respectivo endereço, enviando então para o servidor central a informação agregada destas duas ferramentas.

Este módulo é o mais aproximado de uma monitorização em tempo-real, por isso, é possível que sejam efectuadas *querys* ao *p0f* de endereços iguais em curtos intervalos de tempo, produzindo uma informação coerente e actual. A Figura 21 representa esquematicamente o modo de funcionamento do módulo *passive.pm.*

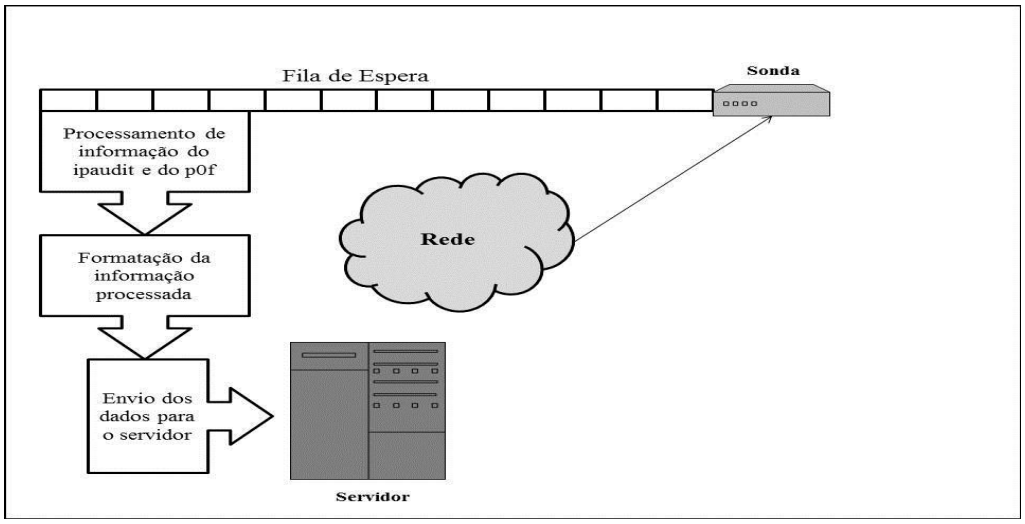

Figura 21 - Processo de análise de dados desde a sonda ao servidor

- *Problemas de integração*

A problemática da quantidade de informação que um sistema pode guardar em disco é parcialmente resolvida através da rotação de *logs*. No entanto, são necessárias diferentes abordagens quando a quantidade de tráfego ultrapassa os milhões de bytes num curto período de tempo, optando-se assim pela utilização de *logs* com o *ipaudit*, e *sockets* de sistema com o *p0f*.

Estas abordagens foram implementadas, pois o *ipaudit* guarda em apenas uma linha de texto a informação referente a um fluxo entre duas máquinas, e o *p0f* guarda em memória dados relativos a cada endereço.

5.2.3 *utils.pm*

O módulo *utils.pm* está estruturado como um elemento auxiliador dos módulos *active.pm* e *passive.pm* na análise e formatação do tráfego capturado.

Nem todo o tráfego capturado pelas ferramentas de monitorização vem formatado para que o possamos utilizar prontamente. Logo são necessários métodos que "construam" a informação baseando-se num formato pré-definido.

As funções do módulo *utils.pm* passam pela formatação correcta de um endereço *IP* ou um endereço físico mal construído, até à definição e configuração dos *sockets* de comunicação entre os módulos de monitorização e o servidor central. Neste caso existem dois *sockets*, um para o módulo passivo e outro para o módulo activo, que funcionam independentemente.

É também neste módulo que são processados os ficheiros de configuração da solução, definindo-se o endereço do servidor central, o porto de destino para a comunicação com o servidor, e as directorias dos *logs* dos respectivos módulos.

5.3 Resumo

A solução final é composta por módulos constituídos por ferramentas de monitorização activa e passiva, que são executados nas sondas pertencentes a uma federação, capturando e analisando tráfego de rede e enviando um conjunto de informação relevante em tempo-real para um servidor central, permitindo assim a sua visualização. Esta visualização é efectuada através duma solução dum projecto paralelo, que recebe em conjunto toda a análise dos dados enviados pelos módulos *passive.pm* e *active.pm*.

A solução passou por várias iterações até atingir a versão final, pois os resultados de vários testes de desempenho não foram satisfatórios, e os recursos disponíveis não

foram constantes ao longo do seu desenvolvimento. Por fim, esta concretização foi implementada numa sonda de grande capacidade, que captura grandes quantidades de tráfego, e o seu comportamento é o mais satisfatório possível, logo é uma solução que pode ser colocada em produção em qualquer rede devido á sua escalabilidade e distribuição.

Capítulo 6

Testes experimentais

Após a apresentação da solução final no capítulo anterior, serão demonstrados neste capítulo os vários testes que foram realizados para verificar os processos aplicados na solução, através da utilização de uma sonda de rede de alto desempenho com uma grande quantidade de tráfego capturado.

Estes testes tiveram em consideração o facto da sonda de rede utilizada possuir um desempenho superior comparativamente às sondas de rede distribuídas pela rede PT. A quantidade de tráfego capturado pela sonda de testes é superior à das sondas de rede simples na ordem das centenas, pois esta sonda foi instalada especificamente junto do *Data Center* da TMN, cujo volume de dados é elevado.

6.1 Tipo de testes

Ao instalar uma sonda na rede de trabalho da PT-EDS não foi possível capturar tráfego suficiente nem útil para uma boa avaliação da solução. Assim foi colocada uma sonda com oito núcleos em *Hyper-Threading* e 32*GB* de memória *RAM*, a recolher uma cópia de todo o tráfego dirigido para o *Data Center* da TMN. Este tipo de sonda não se assemelha às sondas comuns da rede PT, que apenas possuem um núcleo e no máximo 1*GB* de memória *RAM*. No entanto, pode ser efectuada uma relação tráfego/capacidade, verificando-se que quanto maior o tráfego, maior a capacidade necessária da sonda e quanto menor o tráfego, menor a capacidade, ou seja, caso exista um ambiente com uma produção de tráfego diminuta, não é necessário investir em sondas de alto desempenho.

Para encontrar um equilíbrio em todos os factores referidos ao longo deste relatório é necessário analisar a quantidade de tráfego que a solução é passível de capturar, e a sua capacidade em analisar esse mesmo tráfego.

Assim os testes propostos são:

- Análise da quantidade de tráfego produzido durante 24 horas;
- Quantidade média de dados e pacotes capturados por segundo durante 24 horas;
- Análise do tempo de processamento em relação ao tamanho dos *logs* na componente passiva;
- Análise do tempo de execução da componente activa para vários conjuntos de endereços;
- Análise do desempenho da sonda quando a componente activa está a produzir tráfego adicional para a rede.

De modo a analisar o tráfego produzido no *Data Center* da TMN durante 24 horas, foi utilizada uma ferramenta de estatísticas de rede externa à plataforma, o *iptraf*, que permitiu medir vários factores tais como a quantidade de dados produzidos por hora, e a média da velocidade de captura de dados e pacotes a cada segundo. Estes valores permitem antever a quantidade de informação que a solução terá de processar.

Para obter uma relação do tempo de processamento dos dados na componente passiva, foi considerado o período de tempo desde o início da análise de um ficheiro, com os dados referentes ao *ipaudit* e consulta à informação do *p0f*, até ao momento final de envio dos dados para o servidor central.

Os testes efectuados ao desempenho da componente activa, passaram pela execução do *nmap* com alvos bastante distintos quer em quantidade quer em distância.

Durante a execução da componente activa, foi utilizada novamente a ferramenta de estatística de rede *iptraf*, para melhor evidenciar a pequena desvantagem de uma monitorização activa, devido á sua produção de tráfego para a rede.

6.2 Cenários

Nesta secção serão descritos os cenários para cada tipo de teste proposto, apresentando estatísticas que verifiquem os testes efectuados.

Todos estes cenários envolvem a sonda de alto desempenho referida no ponto anterior, e a execução da solução final proposta.

Para um primeiro cenário de análise à quantidade de tráfego, foi configurado o *iptraf* de modo a efectuar medições horárias junto do interface de captura da sonda durante 24 horas, como se apresenta na Figura 22.

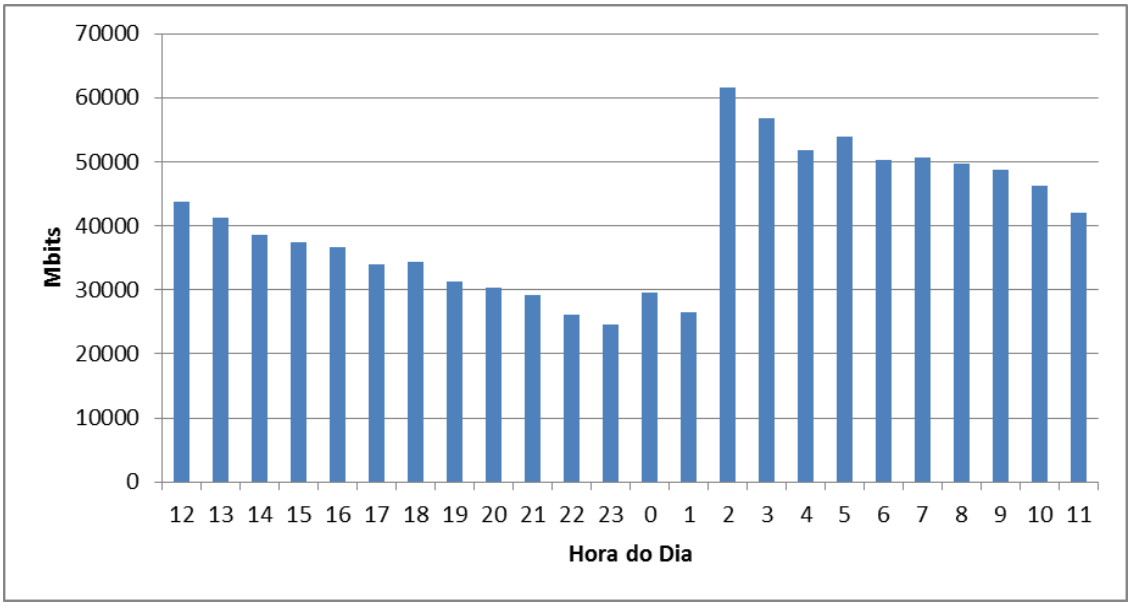

Figura 22 - Quantidade de dados capturados durante 24 horas

Os dados recolhidos foram conclusivos, e após uma análise evidenciou-se que durante certos períodos nocturnos a quantidade de tráfego é superior a qualquer periodo diurno, indicando um possivel processo de *backup* dentro da rede, gerando uma enorme produção de dados.

Durante o período de captura efectuado pelo *iptraf* foi conjugada a velocidade média de transacção de dados por segundo e o número médio de pacotes trocados na mesma unidade de tempo, conforme representado na Figura 23. Estes valores permitem demonstrar o que é expectável de uma análise em tempo-real, e da velocidade necessária de processamento para um desempenho óptimo.

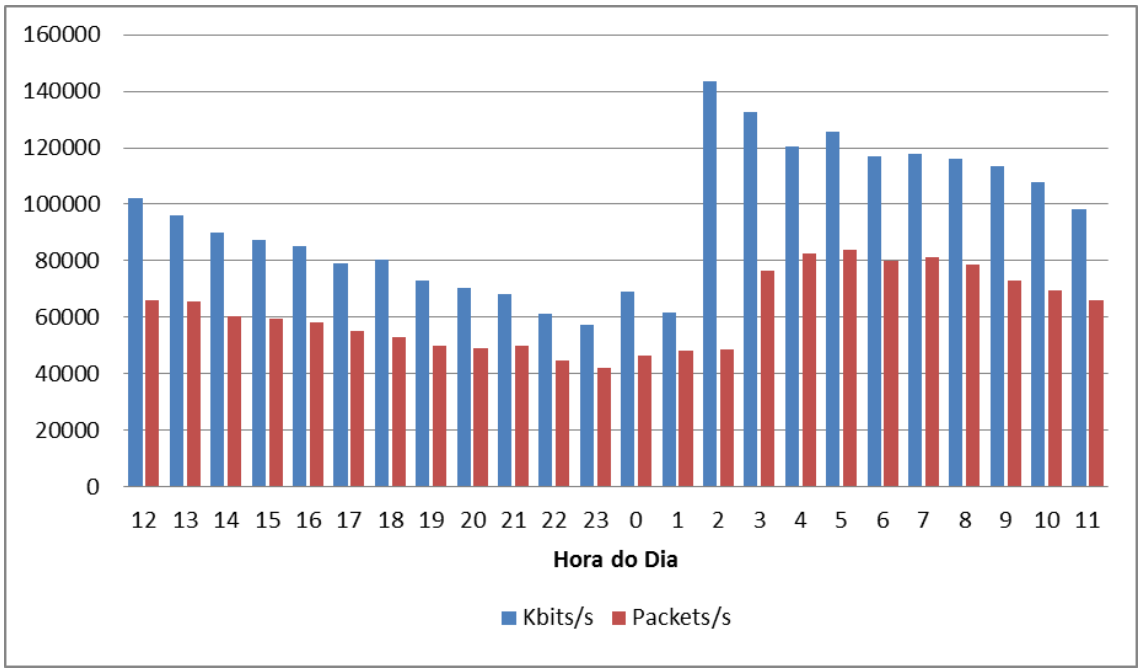

Figura 23 - Numero médio de dados e pacotes processados por segundo durante 24 horas

Estes valores permitem uma melhor configuração das componentes da solução, como por exemplo estabelecer um limite superior no número de pacotes capturados pelo *ipaudit* até efectuar a rotação do *log.*

O segundo cenário passa pela análise de comportamento da componente passiva quando está em execução. Esta componente engloba o *ipaudit* e o *p0f,* executando a análise da informação inserida nos ficheiros de *log* e a simultânea consulta da informação guardada pelo *p0f* relativamente a cada endereço. Assim, as medições iniciam-se com a análise do ficheiro de *log*, seguindo-se a consulta através do *p0f*, finalizando com o envio dos dados formatados para o servidor central.

Foram então efectuadas medições para cada período de análise numa amostra de 9249 ficheiros em tempo-real, como demonstra a Figura 24.

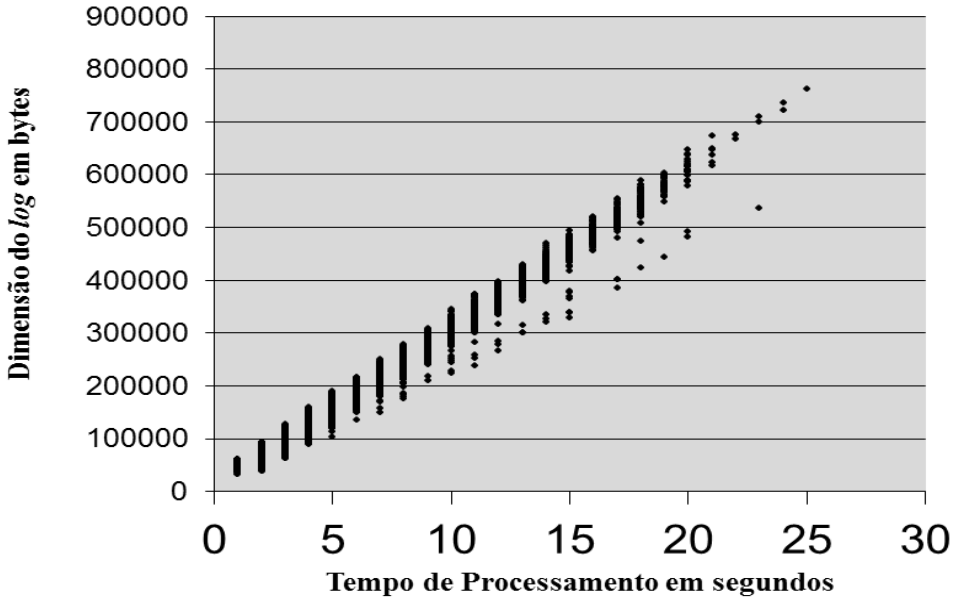

Figura 24 - Relação entre tempo de processamento e dimensão do *log*

Esta valores foram recolhidos apenas durante a execução da componente passiva, não permitindo a existencia de factores externos que pudessem afectar negativamente as medições. Como se denota na Figura 24, quanto maior for a dimensão do ficheiro, maior tende a ser o tempo de processamento. É de notar que cada ficheiro possui 500.000 pacotes, tendo cada pacote um numero variável de *bytes*.

A utilização do *CPU* da sonda durante a execução da componente passiva foi medida através do comando *ps*, que disponibiliza uma média da utilização total do *CPU* ao longo do periodo de execução, cujo valor se situa nos 26% de utilização média para a ferramenta *ipaudit*, e os 13% de utilização para a ferramenta *p0f*.

O seguinte cenário de avaliação envolve a utilização da componente activa para a medição do tempo dispendido na monitorização de diversos endereços dentro da rede PT. Assim, foi utilizada a ferramenta *nmap* recorrendo a vários tipos de comandos, que permitem aumentar a velocidade de envio de pacotes, e o tipo de dados que recolhe ocorrendo uma diminuição do tempo de monitorização.

A Tabela 3 demonstra uma monitorização activa com diferentes comandos sobre um conjunto de endereços definido por *10.101.67.0/24*, *10.101.68.0/24*, *10.101.69.0/24*, cujas máquinas se situam no mesmo edifício que a sonda de captura.

Comando	Tempo	Classificação	Total de endereços – Endereços activos
T5 -sS -A -PS	463s	Bom	Total 768 - Up 59
T5 -sS -A -PS	988s	Bom	Total 768 - Up 86
T5 -sS -PS -sV -O	458s	Muito Bom	Total 768 - Up 87
T5 -sV -sU -A -PP	1813s	Mau	Total 768 - Up 87

Tabela 3 - Medições de monitorização activa

Os comandos utilizados foram escolhidos com o objectivo de aumentar a rapidez de monitorização não descurando a quantidade de informação pretendida.

Após escolher o comando com melhores resultados ***nmap T5 -sS -PS -sV –O***, foram realizadas novas medições a conjuntos de endereços bastante diferentes, quer em quantidade de máquinas quer em distância relativa à sonda.

Localização	Tempo	Total de endereços – Endereços activos	Distância
Beja	**1438s**	**Total 1982 - Up 260**	**178 Km**
Funchal	**1345s**	**Total 2258 - Up 308**	**974 Km**
Cabo Verde	**486s**	**Total 545 - Up 69**	**2915 Km**
Call Center Castelo Branco	**703s**	**Total 1338 - Up 133**	**188 Km**
Faro	**1576s**	**Total 2962 - Up 390**	**278 Km**
TMN Álvaro Pais (Lisboa)	**1670s**	**Total 13465 - Up 399**	**0 Km**
Afonso Costa (Lisboa)	**1613s**	**Total 12910 - Up 304**	**2 Km**
Call Center Braga	**244s**	**Total 296 - Up 23**	**322 Km**

Tabela 4 - Medições de monitorização activa em alguns pontos da rede PT

Durante as medições efetuadas e observadas na Tabela 4, a memória de *CPU* utilizada pela ferramenta *nmap* obteve o valor de 19%, sendo este resultado bastante positivo para uma ferramenta que produz tráfego na rede. No entanto, este valor pode não reflectir o verdadeiro comportamento da ferramenta, pois existem períodos de tempo cuja actividade é nula, e outros passiveis de utilizar 80% da memória.

A Figura 25 reflecte o comportamento real de uma monitorização activa.

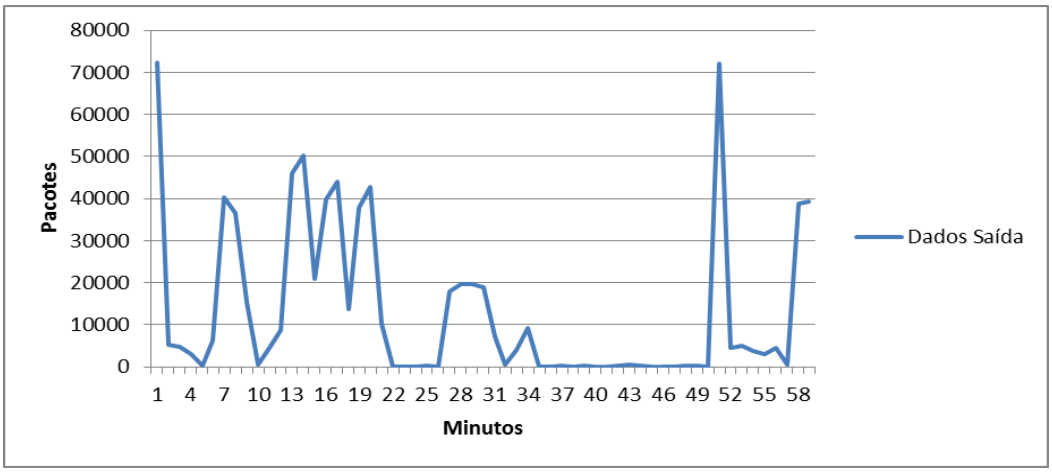

Figura 25 - Tráfego produzido durante a execução do *nmap*

O grafico apresentado na Figura 25 possui traços tipicos de uma monitorização activa, pois destacam-se picos de produção de tráfego, e períodos de estagnação derivados das definições base do *nmap* distribuindo a carga por blocos de endereços, e prosseguindo para um novo bloco quando todos os dados esperados forem recepcionados.

6.3 Resultados e análise

Após a avaliação experimental de diversos cenários, obteve-se os níveis de desempenho da solução através do comando *ps*, situando-se nos 38% de utilização de memória de *CPU*, tendo em conta a execução de todos os módulos e ferramentas em simultâneo. É então possível considerar, que o objectivo de aproximar a solução o mais possível de uma monitorização em tempo-real foi atingido, pois os tempos de captura e análise medidos são aproximados.

Os resultados obtidos não podem ser generalizados para qualquer ambiente de execução, pois cada rede possui uma característica diferente que influência a monitorização. Dentro da rede PT é possível que cada sonda capture informação completamente diferente, demonstrando a sua heterogeneidade. Por conseguinte, cada sonda deve corresponder às necessidades de monitorização utilizando a solução final, que é escalável e por isso útil em qualquer rede de grandes dimensões.

Os valores obtidos na monitorização activa poderão ser melhorados caso se utilize um método distribuído entre as sondas de rede, partilhando os conjuntos de endereços alvo entre elas, reduzindo a carga em cada sonda, sendo mais difícil para um *IDS* detectar este tipo de monitorização.

6.4 Resumo

Neste capítulo foram apresentados os procedimentos e os resultados da avaliação experimental da solução final, utilizando para esse efeito tráfego com características específicas da rede PT.

Os valores e dados estatísticos recolhidos através dos testes realizados foram positivos, quer numa execução conjunta dos dois tipos de monitorização, passiva e activa, quer individualmente.

Após a avaliação experimental pode concluir-se que a solução é escalável, podendo ser adaptada a qualquer rede empresarial caso necessite de uma monitorização em tempo-real.

Capítulo 7

Discussão

7.1 Trabalho inicial

O desenvolvimento deste projecto focou-se na potencial utilidade a nível empresarial, e nesse sentido foi realizado um profundo estudo de artigos científicos que estivessem directamente relacionados com o tema da segurança e monitorização de redes, quer de pequena, média ou grande dimensão. Todo este processo foi desenvolvido tendo subjacente os objectivos da empresa, bem como, o estudo de vários sistemas e plataformas já utilizados PT Comunicações. Algumas das ferramentas testadas e desenvolvidas dentro da empresa foram integradas na solução final.

As reuniões de projecto foram bastante úteis para uma melhor orientação dos objectivos, tendo sido possível uma maior celeridade no desenvolvimento da solução.

7.2 Escolhas efectuadas

7.2.1 Linguagem de programação

O início do desenvolvimento da solução passou pela escolha da linguagem utilizada, que assentou no *Java*. Sendo o *Java* uma linguagem orientada a objectos e possuindo uma extensa biblioteca de rotinas de rede e segurança, foi possível desenvolver uma solução inicial quer nas sondas de rede, quer no servidor, que permitisse a transição de objectos típicos deste tipo de linguagem, uniformizando a estrutura da informação. No entanto, o processo de encapsulamento e transição de um objecto necessita de um processamento maior do que a simples comunicação de texto em claro, tendo vários testes evidenciado este facto, levando a uma ponderação sobre o tipo de linguagem.

Assim, a solução foi adaptada a uma nova linguagem, o *Perl*, que é estável e versátil no processamento e manipulação de texto, além de ser simples e uma prática

comum na empresa. Após esta escolha e adaptação, foram realizados testes simples relativos ao período de tempo ocupado pela análise e transição de dados, e conclui-se que o *Perl* seria a linguagem utilizada na solução final, embora o servidor esteja implementado em *Java,* demonstrando a versatilidade do *Perl*.

7.2.2 Ferramentas de monitorização

Durante a investigação preliminar foram várias as ferramentas disponíveis para uma possível implementação na solução, sendo necessário "dissecar" os manuais de cada uma em busca da ferramenta ideal para cada caso.

O *Ipaudit 0.95* foi uma mais-valia na monitorização de fluxos dentro da rede. Permitiu criar um mapa virtual das comunicações entre máquinas dentro da rede PT, e os tipos de dados transaccionados. Existem também outras ferramentas que nos fornecem esta informação, tal como o *TCPdump,* no entanto, não são orientadas exclusivamente aos fluxos de dados.

O *p0f 3.05b* é uma ferramenta passiva essencial no projecto, pois foca-se prioritariamente no tipo de sistema operativo correspondente à máquina, cujo endereço foi analisado. É uma versão recente, possuindo *fingerprints* de sistemas operativos actuais, sendo um factor positivo numa rede dinâmica.

A monitorização activa da solução está a cargo do *nmap 6.00* lançado em 21 de Maio de 2012, e foi a única ferramenta activa que assumiu um papel de extrema importância na solução, devido à sua versatilidade e adequação aos objectivos propostos.

A ferramenta passiva *pads* não foi inserida na solução final devido a problemas de compatibilidade, e visto o objectivo ter sido a criação de um sistema distribuído e escalável não foi possível tirar partido das suas potencialidades.

Embora existam ferramentas com potencialidades semelhantes às utilizadas não foram integradas por se ter considerado a ocorrência de redundâncias aplicacionais.

7.3 Recursos e Hardware

As sondas de rede disponíveis dentro da rede PT são máquinas descontinuadas e com baixos recursos computacionais. Portanto é necessário desenvolver soluções com essa limitação em mente. As soluções terão de ser leves a nível de processamento, não esgotando todos os recursos disponíveis. No entanto, durante o desenvolvimento da solução final, foi instalada uma sonda de rede com alto desempenho capaz de capturar

uma quantidade substancial de tráfego, permitindo um verdadeiro teste às capacidades de escalabilidade da solução. É de notar que numa situação normal de monitorização existem dezenas de sondas com recursos computacionais limitados a trabalhar em federação, e que dependem das capacidades distribuídas da solução.

Em suma, os recursos foram suficientes, tendo o projecto sido desenvolvido 80% do tempo em sondas com recursos limitados, e o restante tempo numa sonda de elevado desempenho.

7.4 Contribuição

A solução final veio "reanimar" o projecto *Discovery*, actualizando e adicionando novas versões de ferramentas de captura, tornando este processo mais rápido, fluido e dinâmico, diminuindo o processamento exaustivo de informação realizado de 3 em 3 horas ao invés de contínuo. O desempenho melhorou através do desenvolvimento de um sistema de transição de informação mais rápido.

Após a integração da solução, será possível deixar a cargo de um sistema automático e distribuído toda a captura e análise de informação, com o mínimo de interacção humana, passando esta para um interface destinado tanto ao simples utilizador como ao analista de segurança especializado.

Assim, com a junção da presente solução e o projecto paralelo, que permite a visualização de informação através de grafos, o projecto *Discovery* pode seguir um percurso de extrema utilidade e reconhecimento dentro da rede PT.

7.5 Resultados finais

O desenvolvimento do projecto tentou seguir os passos propostos no capítulo 1, mas devido à incompatibilidade de algumas ferramentas e mudanças na arquitectura ocorreram atrasos pontuais. Pode-se no entanto assumir que os resultados foram positivos, tanto no desenvolvimento, como nos testes efectuados com a solução final.

Um projecto deste género é um projecto aberto, ou seja, está sempre apto a receber a implementação de novas ferramentas, que possibilitem o aumento da quantidade de informação apresentando novas temáticas como a alarmística e a detecção de intrusões.

Capítulo 8

Trabalho futuro e conclusões

Conclui-se que este projecto obteve resultados positivos no que diz respeito à monitorização de uma rede empresarial, tendo sido atingido um patamar de conhecimento elevado acerca das exigências de segurança que uma empresa como a PT Comunicações possui.

A solução final apresentada está apta a ser integrada em qualquer tipo de rede empresarial que necessite de ter uma visão dos seus activos, para melhor entender o nível de protecção existente, estando aberta a novas integrações de modo a aumentar as suas funcionalidades e capacidades, focadas em objectivos específicos.

Todas as ferramentas utilizadas fazem com que a solução seja capaz de responder aos mais diversos ambientes empresariais de elevada produção de tráfego, evidenciando uma monitorização em tempo-real, factor extremamente importante para o bom funcionamento de uma empresa de telecomunicações.

O desenvolvimento do projecto inserido numa equipa de trabalho que preza a eficiência, a disponibilidade e a segurança foi uma mais-valia, pois o ambiente de execução da solução era propício para a aplicação destes três princípios, e o apoio prestado por todos os elementos da equipa foi essencial em todos os aspectos.

Durante a execução da solução foram recolhidos dados extremamente úteis, e quantidades massivas de informação que possibilitaram a observação de uma rede empresarial e os seus desvios, tanto a nível de máquinas activas como ao nível de quantidade de tráfego produzido.

Em suma, os resultados foram positivos desde a exploração inicial até à implementação final, pois a temática da monitorização de redes é complexa mas bastante curiosa e interessante, principalmente quando inserida numa empresa de telecomunicações conceituada como a PT Comunicações.

Trabalho Futuro

Durante e após o desenvolvimento da solução, surgiram ideias que poderiam ser implementadas em trabalho futuro de modo a melhorar e introduzir novos aspectos:

- **Reconhecimento de assinaturas**

  Visto existir incompatibilidade com a ferramenta *pads*, que permite o reconhecimento de assinaturas de serviços, é possível desenvolver um sistema simples com funções semelhantes ao *pads* utilizando o número dos portos nos fluxos de comunicação, ou seja, os protocolos utilizados dentro da rede PT são bem conhecidos, e geralmente o porto que utilizam é fixo, logo a sua observação num fluxo permite o seu reconhecimento.

- **Alarmística**

  Uma ferramenta estudada durante o período exploratório foi o *snort*, que permite detecção de intrusões e de ataques. Seria uma adição positiva à solução tornando-a capaz de melhor reconhecer zonas da rede cuja protecção esteja em causa. Existem também outras ferramentas desenvolvidas dentro da PT Comunicações, nomeadamente o *Spygetit*, que analisa e alerta para possível *spyware* na rede, e o *dart*, que produz alertas sobre *resets* e *timeouts*.

- **Comunicação entre sondas**

  Um sistema distribuído preza pela sincronização entre todos os seus elementos, não desvalorizando uma possível comunicação entre as sondas componentes da plataforma. Esta comunicação serve principalmente para optimizar o desempenho da monitorização activa, distribuindo conjuntos de endereços alvos pelas diversas sondas da federação, diminuindo a sua carga e fazendo com que estejam sempre em actividade, recolhendo informação melhorando o desempenho do sistema distribuído. Isto é possível, caso cada sonda possua uma tabela partilhada gerida pelo próprio servidor.

- **Cooperação entre monitorização activa e passiva**

  A existência de uma interligação entre estes dois tipos de monitorização é importante, pois influencia directamente a consistência dos dados que são enviados para o servidor central, ou seja, caso um endereço capturado pela componente passiva não possua informação suficiente, a componente activa tenta colmatar a lacuna, de modo que a informação enviada para o servidor central seja a mais completa possível.

- **Integração na plataforma Pulso/*Discovery***

  Sendo a plataforma Pulso um portal de monitorização de risco técnico, era interessante a integração da solução, possibilitando a qualquer utilizador autorizado a visualização da informação recolhida, e inclusivamente, requerer uma monitorização activa para um certo conjunto de endereços específico.

Bibliografia

[1] José A. S. Alegria, Tiago F. R. Carvalho, Ricardo G. Ramalho. *Uma experiência open source para "tomar pulso" e "ter pulso" sobre a função de sistemas e tecnologias de informação*. PT Comunicações.

[2] Tiago M. S. Sequeira, *Investigação e Desenvolvimento de um Sistema Automático de Detecção, Monitorização e Análise da Propagação de Worms em Redes Empresariais*. Tese Mestrado, Faculdade de Ciências Universidade de Lisboa, 2011.

[3] José A. S. Alegria, Pedro D. Inácio, Luís M. Monge, Nuno F. Almeida, Pedro R. Simões, Tiago G. S. Mendo. *Captura e Análise de Tráfego Aplicacional Detecção e Analise de Vulnerabilidades de Rede*. PT Comunicações, 2005

[4] Christos Papadopoulos, Genevieve Bartlett, John Heidemann. *Understanding passive and active service discovery*. 2007.

[5] Synergy Networks, *Examining port scan methods - Analyzing Audible Techniques.*

[6] Synergy Networks, *Advanced Host Detection, Techniques To Validate Host-Connectivity.*

[7] Richard Lippmann, David Fried, *Passive Operating System Identification From TCP/IP Packet Headers*.

[8] Bruce Schneier, John Kelsey, *Secure Audit Logs to Support Computer Forensics*.

[9] Matta Security Limited. *Hacking for Techies IP Network Scanning & Reconnaissance.*

[10] SANS Institute, *OS and Application FingerPrinting Techniques*.

[11] Henrik Abrahamsson, *Traffic Measurement and Analysis*. 1999.

[12] Insightix. *The Myth of Passive Network Discovery Systems*.

[13] Antonis Papaogiannakis, Demetres Antoniades, Michalis Polychronakis. *Improving the Performance of Passive Network Monitoring Applications using Locality Buffering*.

[14] Annie De Montigny-Leboeuf, Fréderic Massicotte. *Passive Network Discovery for Real Time Situation Awareness*.

[15] António Manuel de Carvalho Alegria. *Verificação Automática de Modelos de Arquitectura Tecnológica de Sistemas de Informação em Rede*. Tese Mestrado, Instituto Superior Técnico, 2009.

[16] Vandyke Software, *an Overview of the Secure Shell*.

[17] *www.perl.org*. Acedido a 7 de Outubro de 2011.

[18] *www.nmap.org*. Acedido a 8 de Outubro de 2011.

[19] P0f project, *lcamtuf.coredump.cx/p0f.shtml*

[20] Pads project, *passive.sourceforge.net/*

[21] Ipaudit project, *ipaudit.sourceforge.net/*

[22] *www.snort.org*. Acedido a 8 de Outubro de 2011.

[23] Andrew Jaquith. *Security Metrics, Replacing Fear, Uncertainty, and Doubt*.

[24] Wayne Jansen. *Directions in Security Metrics Research*.

[25] Michael Aiello. *Information Security Management Polytechnic University*.

[26] *http://search.cpan.org/~apersaud/Nmap-Parser/Parser.pm*.