Redes de Computadores
Relatório de Desenvolvimento
Trabalho Prático

Abel Surreira (39839) Nuno Correia (43179) Miguel Esteves (43165)

3 de Janeiro de 2008

Resumo

Em síntese este documento descreve o relatório de desenvolvimento do trabalho prático da disciplina de Redes de Computadores. Ao longo deste relatório, são apresentadas as fases de desenvolvimento e implementação. O projecto «T1. Identificação e Ranking de Serviços de Rede» consiste num analisador de tráfego de redes TCP/IP avançadas e foi desenvolvido em ambiente Windows recorrendo à linguagem Java. Para a execução do projecto utilizamos como ferramentas de trabalho o IDE Netbeans para o desenvolvimento da aplicação e respectivas interfaces gráficas, quanto ao relatório, este foi produzido recorrendo ao LaTeX.

Conteúdo

1 Introdução

Nesta secção apresenta-se uma breve descrição do trabalho de desenvolvimento da disciplina de Redes de Computadores da Licenciatura em Engenharia Informática, que consiste na apresentação dos objectivos, no contexto do seu desenvolvimento, no planeamento e tecnologias adoptadas. As fases de realização do trabalho são partes integrantes desta introdução.

1.1 Contexto

Nos dias que correm as redes de computadores estão em todo o lado. A Internet, revolucionou, não só o mundo da computação, como também a vida de milhões de várias formas. Usualmente tomamos como garantido que os computadores devem estar conectados, não nos apercebendo da tecnologia necessária para tal acontecer. As redes de computadores são constituídas por nodos inter conectados entre si, por meio de adaptadores de rede ou outros blocos de Hardware(como Routers, Bridges, Switches, etc.) conectados através de um meio físico limitado. Com a evolução da tecnologia as velocidades de transmissão aumentaram, mas não deixam de estar limitadas. Tendo em atenção o grande aumento da utilização de serviços de rede baseados em tecnologias de rede (como

E-Mail, HTTP, VoIP, etc.) torna-se necessário caracterizar e identificar esses serviços mais solicitados, para desta forma conseguir alocar os recursos no sentido de se conseguir minimizar perdas e permitindo assim uma gestão mais eficiente da rede.

1.2 Descrição do Problema

Para este projecto escolhemos a proposta T1. Identificação e Ranking de Serviços de Rede. Este projecto consiste no desenvolvimento de um aplicação capaz de analisar e interpretar logs de tráfego. Disponibilizando informação sobre os serviços de rede suportados e os seus índices de utilização, no sentido de se obter um maior conhecimento do perfil do tráfego, permitindo assim uma melhor gestão da rede.

2 Enunciado

```
1  T1. Identificação e Ranking de Serviços de Rede
2
3  No sentido de se obter um maior conhecimento do perfil do tráfego de rede na UM,
4  o que permite uma melhor gestão da mesma, pretende-se devolver uma ferramenta
5  versátil e intuitiva de apoio à monitorização off-line de redes TCP/IP avançadas.
6  Essa ferramenta deve ser capaz de interpretar e analisar logs de tráfego
7  provenientes de ambientes reais (Netflow, tcpdump, Ethereal), disponibilizando
8  informação sobre serviços de rede suportados e seus índices de utilização.
9  Preferencialmente, essa informação deverá dser disponibilizada através de uma
10 interface gráfica que facilite a filtragem e visualização do ranking dos serviços
11 e a interacção com o utilizador.
12
13 -filtragem (nível de rede, transporte, aplicação)
14 -índices de utilização por intervalo de tempo (horas, dias, ...)
15         nível de rede - IP, ICMP, IGMP ...
16         transporte - TCP, UDP, ...
17         aplicação - http, mail, ftp, VoIP, ...
18         classe - ToS (DSCP)
19         endereços IP - rede IP origem/destino
```

3 Desenvolvimento

Este programa faz sniffing de 3 tipos de ficheiros, nomeadamente netflow, wireshark e tcpdump. Além disso, com a ajuda de uma biblioteca chamada jpcap para java mais o programa winpcap, é também feita a captura de pacotes dos interfaces de rede. É feita uma estatística tanto dos logs como dos pacotes capturados pelo programa por campo de origem, destino, protocolo de aplicação, nível de rede e transporte. Existe a possibilidade de guardar o resultado obtido em ficheiro. Os ficheiros DataWireshark, DataNetflow e DataTcpdump fazem o *parsing* dos logs Wireshark, Netflow e Tcpdump respectivamente. Repeitante ao campo da aplicação, em logs Wireshark já existe a descrição da mesma bem como da porta, o que não acontece em Netflow ou Tcpdump. No caso do Tcpdump, existe pouca informação dos pacotes, ainda assim, com o número da porta, ligando a uma base de dados em texto (portas.txt), é possível fazer o *matching* usando a classe DataPortas. No caso do Netflow é também utilizado este ecesso pela mesmo razão.

Existe ainda outra classe a trabalhar ao mesmo nível que é a DataJpcap. Nesta classe é utilizada a biblioteca Jpcap para capturar uma a um cada pacote recebido. Neste ambiente, é também usado o *matching* das portas como nas outras classes. Além disso no programa é feita uma aprendizagem da aplicação de uma porta desconhecida da base de dados. Quando uma máquina A envia um pedido através da sua porta YYY a uma Máquina B para a porta 80 desta, se esta for um servidor, não pode responder usando a porta 80 pois assim

haveria superlotação. O servidor vai então responder usando uma porta XXX, que vai ser diferente para cada utilizador, para a porta YYY da máquina A. Neste exemplo, a porta YYY vai corrsponder à aplicação da porta 80 que está presente na base de dados das portas.

Em termos algorítmicos, o programa neste ambiente tem aprendizagem da aplicação atraves de uma base de dados dinâmica onde se encontram todas as portas e a respectiva aplicação de pacotes é agora capturados. Assim, no exemplo anterior o primeiro pacote recebido seria algo do tipo A.XXX -> B.80. Após a captura deste pacote, a base de dados dinâmica com as portas e as respectivas aplicações teria mais um registo que seri qualquer coisa como (XXX , HTTP). O segundo elemento do par é a descrição da porta 80 que ia ser carregada da base de dados das portas. Na próxima captura, o *parsing*, além de procurar a descrição da aplicação na base de dados das portas, também o faria nesta base de dados dinâmica. O segundo pacote seria qualquer coisa como B.YYY -> A.XXX e neste caso, a aplicação seria reconhecida como HTTP... Mais um registo seria acrescentado à base de dados dinâmica (YYY , HTTP). Apesar deste algorítmos, existem ainda alguns pacotes sem descrição de aplicação pois quando se tenta aceder a um servidor, existe o conceito de *portmap* e não uma e só uma porta. No exemplo acima, o que poderia acontecer era que o a máquina do utilizador não iria enviar sempre pela mesma porta mas por várias ao mesmo tempo para melhor desempenho. No exemplo acima, o segundo pacote seria qualquer coisa do tipo B.YYY -> A.ZZZ. ZZZ seria neste caso igual a (XXX + 1).

4 Conclusões (Trabalho Futuro)

Pensamos que conseguimos concretizar com sucesso, dentro do que nos foi proposto, este projecto de desenvolvimento, embora, estejamos cientes que podemos melhora-lo em próximas etapas, visto que ainda é possível adicionar muitas mais funcionalidades e *features* a esta aplicação.

Em função disto julgamos ter respondido aos objectivos propostos e conseguimos assim completar com sucesso o desenvolvimento deste projecto.

A Apêndice

Lista-se a seguir os aspectos mais relevantes para o desenvolvimento deste projecto.

A.1 Netflow Log

A.2 Screenshots