

Relatório
Trabalho Prático 2
Fevereiro 2018

Universidade do Minho
Mestrado Integrado em Engenharia de Telecomunicações e
Informática
Gestão de Redes



Raúl Filipe Cruz Antunes A75577

Índice

Introdução	2
Monitorização de Partições do Disco.....	2
Ferramentas Utilizadas.....	2
Objetos Utilizados	3
Questão 1.....	3
Tempo de Poolling	3
Exemplos de Utilização.....	3
Conclusão	4

Introdução

O objetivo do segundo trabalho prático desta unidade curricular é o desenvolvimento de uma aplicação em tempo real que monitorize a ocupação das partições dum sistema operativo dum qualquer host na rede local. Disponibilizando assim na interface gráfica uma tabela com a descrição da partição, o seu tamanho (GB) e o espaço ainda disponível.

Neste relatório é ainda explicada a forma como a aplicação foi desenhada, bem como as ferramentas utilizadas, alguns detalhes dos objetos SNMP estatísticos e a escolha do tempo de poolling.

Monitorização de Partições do Disco

Para este trabalho prático foi desenvolvida uma aplicação que utiliza alguns objetos SNMP para apresentar uma tabela do tamanho das devidas partições e o seu espaço utilizado, desde que ativado o programa até à sua paragem.

Ferramentas Utilizadas

Esta aplicação foi construída em java, usando o NetBeans IDE. Foi usada uma API para comunicar através de SNMP com os objetos da MIB que guardam a informação necessária para gerar a tabela(SNMP4J). Uma classe desta aplicação foi retirada dos exemplos da API SNMP4J (SNMPManager.java).

Objetos Utilizados

Os objetos da MIB utilizados foram:

- “physicalsize” (com o OID 1.3.6.1.2.1.25.2.3.1.5.1);
- “physicalused” (com o OID 1.3.6.1.2.1.25.2.3.1.6.1);
- “virtualsize” (com o OID 1.3.6.1.2.1.25.2.3.1.5.3);
- “virtualused” (com o OID 1.3.6.1.2.1.25.2.3.1.6.3);
- “homesize” (com o OID 1.3.6.1.2.1.25.2.3.1.5.45);
- “homeused” (com o OID 1.3.6.1.2.1.25.2.3.1.6.45);
- “usersize” (com o OID 1.3.6.1.2.1.25.2.3.1.5.43);
- “userused” (com o OID 1.3.6.1.2.1.25.2.3.1.6.43);
- “shmsize” (com o OID 1.3.6.1.2.1.25.2.3.1.5.42);
- “shmused” (com o OID 1.3.6.1.2.1.25.2.3.1.6.42);

Estes guardam o número de bytes da partição respetiva ou o número de bytes utilizados da mesma partição, desde o início do funcionamento do seu host. Para calcular por sua vez o tamanho livre da partição por um respetivo intervalo de tempo simplesmente divide-se por exemplo o “physicalused” pelo “physicalsize” e multiplica-se por 100, obtendo-se a percentagem livre da partição, o que resulta na diferença entre valores ao longo da atualização da tabela.

Questão 1

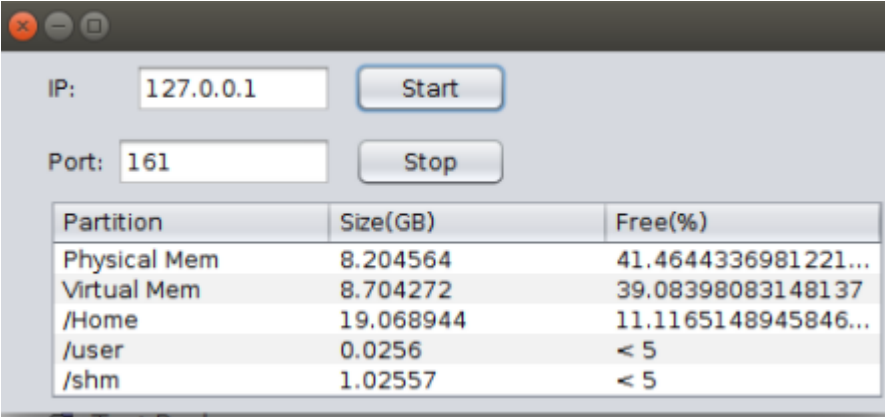
Tempo de Pooling

O tempo de polling escolhido foi de um 30s, visto que em vários testes a fazer polling constante (intervalo de tempo muito curto), verificou-se que a variável não alterava significativamente. Usando esta informação e testes adicionais definiu-se então o tempo de pooling.

Exemplos de Utilização

Na Imagem 1 é apresentado um exemplo de utilização da aplicação, onde podemos verificar uma atualização ao longo do tempo dos valores de espaço livre da partição. De notar que sempre que o espaço livre for menor que 5% do valor inteiro da partição então o resultado é demonstrado como inferior a 5% e não o seu valor real. Foram medidas 5 partições das muitas que

existiam, sendo que destas 5, 3 são partições grandes e 2 pequenas. As restantes partições não foram demonstradas apenas porque são demasiado pequenas e não têm interesse para o projeto.



The screenshot shows a Java application window with a title bar containing standard window controls. The interface includes two input fields: 'IP:' with the value '127.0.0.1' and 'Port:' with the value '161'. Next to each field is a button, 'Start' and 'Stop' respectively. Below these controls is a table with three columns: 'Partition', 'Size(GB)', and 'Free(%)'. The table contains five rows of data representing system resources.

Partition	Size(GB)	Free(%)
Physical Mem	8.204564	41.4644336981221...
Virtual Mem	8.704272	39.08398083148137
/Home	19.068944	11.1165148945846...
/user	0.0256	< 5
/shm	1.02557	< 5

Conclusão

O objetivo deste trabalho foi atingido, visto que a aplicação apresenta os dados estatísticos corretos e de forma intuitiva e agradável ao olhar, com uma interface de fácil utilização.

Os maiores problemas no desenvolvimento desta aplicação foram a obtenção dos valores dos OID e a passagem deles para a própria aplicação, sendo esta depois resolvida devido à utilização do `SNMPManager.java`.

É importante mencionar que a aplicação funciona em qualquer plataforma, visto que é feita em java, desde que o SNMP esteja funcional e que a aplicação tenha acesso garantido aos objetos SNMP em causa.