GESTÃO E MONITORIZAÇÃO DE UMA REDE DEPARTAMENTAL

-

UM CASO DE ESTUDO

Hélder Vieira Mendes

Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

2008

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Electrotécnica e de Computadores – Área de Especialização de Automação e Sistemas

Candidato: Hélder Vieira Mendes, Nº 1020359, hvm@isep.ipp.pt
Orientador: Prof. Doutor Jorge Botelho Costa Mamede, jbm@isep.ipp.pt

Departamento de Engenharia Electrotécnica
Instituto Superior de Engenharia do Porto
15 de Dezembro de 2008

Agradecimentos

Os meus agradecimentos vão para todos aqueles que, de alguma forma, contribuíram para o desenvolvimento deste trabalho e para a prossecução de uma experiência enriquecedora, assinalando as referências mais importantes.

Ao Prof. Doutor Jorge Mamede, meu orientador, por todo o apoio científico e pedagógico, e pelas palavras de encorajamento nos momentos mais necessários.

À minha namorada, Sandra Sousa, pela sua permanente compreensão e motivação, em todos os momentos desta jornada.

À minha família agradeço o apoio manifestado e que me fez chegar a este momento da minha vida.

Ao Morten Brekkevold da UNINETT e ao Morten Werner Forsbring da Debian, pelo apoio na resolução de erros do software *Network Administration Visualized* (NAV).

Ao Chuck Syperski, pelo apoio na resolução de erros do software *Free Open Source Ghost* (FOG).

Resumo

De há uns anos a esta parte, no Instituto Superior de Engenharia do Porto (ISEP) tem-se verificado um crescimento da utilização das tecnologias de informação, nomeadamente das redes informáticas. A rede de dados do Departamento de Engenharia Geotécnica (DEG) do ISEP cresceu em complexidade e dimensão, de acordo com as necessidades dos seus utilizadores. Hoje é uma rede que disponibiliza aos seus utilizadores vários serviços informáticos, como o serviço da página de internet, o serviço de contas de utilizadores, o servidor de ficheiros, o serviço de impressão através da rede, entre outros.

A rede informática do DEG não dispõe de um sistema centralizado que permitisse fazer a gestão e monitorização dos equipamentos e serviços informáticos. Para realizar as várias tarefas associadas à gestão e manutenção da rede são utilizadas várias ferramentas em separado, levando muitas vezes a perdas de tempo desnecessárias. A monitorização da rede através de uma aplicação dedicada para o efeito também não é possível. As operações de monitorização estão restringidas às já incluídas nos activos de rede. Uma das operações que gera mais problemas é a de instalação de imagens de software através da rede. Apesar de evitar a formatação dos computadores um a um, a instalação de imagens de software implica que o instalador visite cada um dos computadores, inicie uma aplicação por *Compact Disk* (CD), e depois de algumas configurações, começa o processo de instalação da imagem de software. No final é necessário visitar novamente cada um dos computadores para os desligar. Um outro problema associado à instalação de imagens de software através da rede é o de congestionamento de tráfego nas intersecções de rede. Sempre que se instalam imagens de software nos computadores de uma ou mais salas de informática, existem problemas de congestionamento nos acessos à internet, ao servidor de ficheiros e ao servidor das páginas de internet, entre outros.

O trabalho que conduziu a esta Tese pretende implementar uma solução de software de apoio à gestão e monitorização na rede de dados do DEG. Após uma análise das características da rede de dados do DEG, identificaram-se os principais requisitos de gestão e monitorização. Estes requisitos dividem-se em dois grandes grupos, por um lado a gestão e monitorização da rede, por outro lado a gestão de imagens de software. Uma das tarefas associadas à gestão de imagens de software é a instalação de imagens de software em computadores remotos (através da rede de dados). A instalação de imagens de software

deve ser feita com controlo dinâmico do débito do servidor em função do estado da rede, para evitar o congestionamento de tráfego nas intersecções de rede.

Para dar resposta aos problemas de gestão da rede do DEG desenvolveu-se o *Network Management System with Imaging Support* (NMSIS). O NMSIS resulta da integração entre as ferramentas *Network Administration Visualized* (NAV) e *Free Open Source Ghost* (FOG) e segue as linhas de orientação do modelo funcional de gestão FCAPS (fault, configuration, account, performance, security), definido pela *International Organization of Standardization* (ISO). O modelo FCAPS divide a gestão de redes em 5 grandes áreas: a gestão de falhas, a gestão de configurações, a gestão de contas, a gestão da performance e a gestão da segurança. Desenvolveram-se três módulos para integrar o NAV e o FOG, o *Database Synchronization Module* (DBSM), o *Group Equipment Synchronization Module* (GESM) e o *Dynamic Host Configuration Protocol Synchronizer* (DHCPSYNC).

A instalação de imagens através da rede evita a formatação dos computadores um a um, no entanto, tem o inconveniente de utilizar elevada largura de banda. Com o intuito de minimizar os efeitos negativos da instalação de imagens de software através da rede, desenvolveu-se um software que faz o controlo dinâmico do débito de dados do servidor de imagens, em função da utilização da rede. Deste modo evita-se o conjestionamento de tráfego nas intersecções de rede. O software foi designado de *BandWidth-Traffic Control* (BWTC), sendo parte integrante do NMSIS. O BWTC é constituído por três módulos, o *BandWidth/Traffic Control Address Resolution Protocol* (BWTC_ARP), o *BandWidth/Traffic Control Calculate* (BWTC_CALC) e o *BandWidth/Traffic Control Statistics* (BWTC_STATISTICS). Para verificar a fiabilidade do BWTC foram projectados e executados vários cenários de teste, em ambiente real.

Palavras-Chave

Gestão e monitorização de redes, gestão centralizada, SNMP, ferramentas *Open Source*, gestão de imagens de software, execução PreOS, controlo de tráfego, qualidade de serviço.

Abstract

In the last years, the Engineering School of the Polytechnic Institute of Porto (ISEP/IPP) has observed an increasing use of information technology, including communication networks. The network of the Department of Geotechnical Engineering (DEG/ISEP) grew in complexity and size, according to the needs of its users. Today is a network that offers its users various services, such as the Web site, internet access, user accounts, the file server, the printing over the network, among others.

The network of DEG did not have a centralized system that allows itself the management and monitoring of equipment and services. To perform the multiple tasks associated with managing and maintaining the network were used several tools separately, leading to unnecessary loss of time. Monitoring the network through an application dedicated to this purpose wasn´t possible, till now. The monitoring operations were restricted to those already included in the assets of the network. One of the operations which generated more problems is the installation of software images over the network. Despite avoiding the formatation of the computers one by one, the installation of software images meant visiting each computer, boot from Compact Disk (CD), and only after some settings began the process of installing the software image through network. In the end it was necessary to, again, visit each computer to shut down. Another problem associated with the installation of software images over the network is the traffic congestion in network´s intersections. Every time the installation of images takes place on remote computers, there were problems with access to the Internet, to the file server and the web server, among others.

The work that led to this thesis aims to implement a software solution to support management and monitoring the network of the Department of Geotechnical Engineering (DEG) of ISEP. After an analysis of DEG´s network characteristics, was identified the main requirements for management and monitoring. These requirements are divided into two major groups, on the one hand the management and monitoring of the network, on the other hand the management of software images. One of the many tasks associated with management of software images is the installation of imaging software on remote computers (through the data network). The installation of software images should be made with dynamic control of the flow between server and clients, depending on the state of the network, to avoid the congestion of traffic at intersections points of network.

To deal with the problems of managing the DEG´s network was developed the *Network Management System with Imaging Support* (NMSIS). The NMSIS result of the integration between the tools Network Administration Visualized (NAV) and Free Open Source Ghost (FOG), and follows the guidelines of the management functional model FCAPS (fault, configuration, account, performance, security), defined by the International Organization of Standardization (ISO). The FCAPS model divides the network management in 5 major areas: the fault management, the configuration management, the account management, the performance management and security management. There are three modules to integrate the NAV and FOG, the Database Synchronization Module (DBSM), the Equipment Group Synchronization Module (GESMES) and the Dynamic Host Configuration Protocol Synchronizer (DHCPSYNC).

The installation of images over the network avoids the formatting of the computers one by one, however has the disadvantage of using high bandwidth. In order to minimize the negative effects of this process, it has developed software that makes the dynamic control of the flow of data from the image server, depending on the network usage. The software has been designated Bandwidth-Traffic Control (BWTC), being an integral part of NMSIS. The BWTC consists of three modules, the Bandwidth / Traffic Control Address Resolution Protocol (BWTC_ARP), the Bandwidth / Traffic Control Calculate (BWTC_CALC) and Bandwidth / Traffic Control Statistics (BWTC_STATISTICS). To check the reliability of the BWTC were designed and implemented various test scenarios in real environment.

Keywords

Management and monitoring of networks, centralized management, SNMP, Open Source tools, image management software, PreOS implementation, traffic control, quality of service (QoS).

Índice

Índice de Figuras

Índice de Tabelas

Acrónimos

ACPI	–	Advanced Configuration and Power Interface
AIX	–	Advanced Interactive eXecutive
AJAX	–	JavaScript and XML
API	–	Application Programming Interface
APM	–	Advanced Power Management
AP's	–	Wireless Access Point´s
ARP	–	Address Resolution Protocol
ASCII	–	American Standard Code for Information Interchange
ASN.1	–	Abstract Syntax Notation One
ATM	–	Asynchronous Transfer Mode
BASE	–	Basic Analysis and Security Engine
BIC		Binary Increase Congestion Control
BIOS	–	Basic Input/Output System
BSD	–	Berkeley Software Distribution
BWTC	–	BandWidth / Traffic Control
BWTC_ARP	–	BandWidth / Traffic Control Address Resolution Protocol
BWTC_CALC	–	BandWidth / Traffic Control Calculate
BWTC_STATISTICS	–	BandWidth / Traffic Control Statistics
CBQ	–	Class Based Queuing
CD	–	Compact Disk
CDP	–	Cisco Discovery Protocol
CLI	–	Common Line Interface
CPU	–	Communications Processor Unit
CSF	–	Cliente do Servidor de Ficheiros
CSIMG	–	Cliente do Servidor de Imagens
CSS	–	Cascading Style Sheets
CSV	–	Comma-Separated Values
CSW	–	Cliente do Servidor Web
DBMS	–	DataBase Management System
DBSM	–	DataBase Synchronization Module
DEG	–	Departamento de Engenharia Geotécnica
DHCP	–	Dynamic Host Configuration Protocol
DHCPSYNC	–	Dynamic Host Configuration Protocol Synchronizer

DiffServ	–	Differentiated Services
DNS	–	Domain Name System
DRBL	–	Diskless Remote Boot on Linux
DS	–	Differentiated Services
DSCP	–	Differentiated Services Code Point
EBCDIC	–	Extended Binary Coded Decimal Interchange Code
ESFQ	–	Extended Stochastic Fair Queuing
ESMAE	–	Escola Superior de Música e Artes do Espectáculo
ESTSP	–	Escola Superior de Tecnologia da Saúde do Porto
FAT	–	File Allocation Table
FCAPS	–	Fault, Configuration, Accounting, Performance, Security
FDDI	–	Fiber distributed data interface
FDM	–	Foundry Discovery Protocol
FDP	–	Foundry Discovery Protocol
FIFO	–	First In First Out
FOG	–	Free Open Source Ghost
FTP	–	File Transfer Protocol
FW	–	Firewall
GESM	–	Group Equipment Synchronization Module
GNU	–	General Public License
GRED	–	Generic Random Early Drop
GUID	–	Globally Unique Identifier
HFS	–	Hierarchical FileSystem
HFSC	–	Hierarchical Fair Service Curve
HP	–	Hewlett Packard
HPFS	–	High Performance File System
HTB	–	Hierarchical Token Bucket
HTC	–	High Tech Computer Corporation
HTTP	–	Hypertext Transfer Protocol
HTTPD	–	Hypertext Transfer Protocol Daemon
HTTPS	–	Hypertext Transfer Protocol over Secure Socket Layer
IBM	–	International Business Machines Corporation
ICMP	–	Internet Control Message Protocol
IDE	–	Integrated Drive Electronics
IDM	–	Identity Drives Manter

IETF	–	Internet Engineering Task Force
IntServ	–	Integrated Services
IP	–	Internet Protocol
IPP	–	Instituto Politécnico do Porto
ISCAP	–	Instituto Superior de Contabilidade e Administração do Porto
ISEP	–	Instituto Superior de Engenharia do Porto
ISO	–	International Organization of Standardization
ITU	–	International Telecommunication Union
JFS	–	Journaled File System
JMX	–	Java Management Extensions
LAN	–	Local Area Network
LDAP	–	Lightweight Directory Access Protocol
LLDP	–	Link Layer Discovery Protocol
LRS	–	Linbox Rescue Server
LVM	–	Logical Volume Management
MAC	–	Media Access Control
MAN	–	Metropolitan Area Network
MBR	–	Master Boot Record
MIB	–	Management Information Base
MRTG	–	Multi Router Traffic Grapher
NAT	–	Network Address Translation
NAV	–	Network Administration Visualized
NFS	–	Network File System
NIM	–	Network Immunity Manager
NMS	–	Network Management System
NMSIS	–	Network Management System with Imaging Support
NNM	–	Network Node Manager
NNTP	–	Network News Transfer Protocol
NTFS	–	New Technology File System
NTNU	–	Norwegian University of Science and Technology
OID	–	Object Identifier
OSI	–	Open Systems Interconnection
OSSIM	–	Open Source Security Information Management
PC	–	Personal Computer
PCI	–	Peripheral Componet Interconnect

PCM – ProCurve Manager
PCMCIA – Personal Computer Memory Card International Association
PDA – Personal Data Assistent
PDF – Potable Document Format
Pfifo_fast – Packet First In First Out Fast
PHP – PHP: Hypertext Preprocessor
PID – Proporcional, Integrativo, Derivativo
PME – Power Management Event
PMM – ProCurve Mobility Manager
POP – Post Office Protocol
PPP – Point-to-Point Protocol
PRIO – Priority Scheduler
PXE – Preboot eXecution Environment
QDISC – Queueing Discipline
QoS – Quality of Service
RAID – Redundant Array of Independent Drives
RAM – Random Access Memory
RAW – Data without filesystem
RFC – Request For Comments
RHEL – Red Hat Enterprise Server
RMON – Remote Network Monitoring MIB
RRD – Round Robin Database
SATA – Serial Advanced Technology Attachment
SCSI – Small Computer System Interface
SecureW2 – Secure Wireless 2
SF – Servidor de Ficheiros
SFQ – Stochastic Fair Queuing
SH – Shell
SID – Security identifier
SIMG – Servidor de Imagens / Servidor FOG
SLES – Suse Linux Enterprise
SMB – Server Message Block
SMI – Structure of Management Information
SMIv2 – Structure of Management Information version 2
SMS – Short Message Service

SMTP	–	Simple Mail Transfer Protocol
SNAV	–	Servidor NAV
SNMP	–	Simple Network Management Protocol
SNMPv1	–	Simple Network Management Protocol version 1
SNMPv2	–	Simple Network Management Protocol version 2
SNMPv3	–	Simple Network Management Protocol version 3
SQL	–	Structured Query Language
SSH	–	Secure Shell
SSL	–	Secure Sockets Layer
SW	–	Servidor Web
TBF	–	Token Bucket Filter
TC	–	Linux Traffic Control
TCP	–	Transmission Control Protocol
TCP/IP	–	Transmission Control Protocol / Internet Protocol
TELNET	–	Telecommunication Network
TFTP	–	Trivial File Transfer Protocol
TKIP	–	Temporal Key Integrity Protocol
ToS	–	Type of Services
TRK	–	Trinity Rescue Kit
UDP	–	User Datagram Protocol
UFS	–	Unix FileSystem
UNDI	–	Universal Network Device Interface
USB	–	Universal Serial Bus
UUID	–	Universal Unique Identifier
VLAN	–	Virtual LAN
VNC	–	Virtual Network Computing
WAN	–	Wide Area Network
WfM	–	Wired for Management Framework
WOL	–	Wake-up-On-LAN
WoWLAN	–	Wake-on-Wireless-LAN
WPA	–	Wi-Fi Protected Access
Web	–	World Wide Web
XFS	–	XFS file system - An high-performance journaling filesystem
XML	–	Extensible Markup Language
XMPP	–	Extensible Messaging and Presence Protocol

1. INTRODUÇÃO

A utilização das tecnologias de informação em larga escala levou ao aumento da dimensão e complexidade das redes informáticas. Perante estes factos levantaram-se as questões de gestão, que diversos grupos de normalização procuraram minimizar. Actualmente existem quatro protocolos de gestão estandardizados a nível internacional, são eles o protocolo *Simple Network Management Protocol* (SNMP), disponível no modelo TCP/IP (ARPANET) do *Internet Engineering Task Force* (IETF), o *Common Management Information Protocol* (CMIP) da ISO/OSI, o *Telecommunications Management Network* (TMN) da *International Telecommunication Union* (ITU) e o *Interim Local Management Interface* (ILMI) da ATM Forum [120]. O CMIP foi desenvolvido para o ambiente *Open Systems Interconnection* (OSI), é mais versátil que o SNMP, mas utiliza cerca de cinco vezes mais memória. O TMN é uma *framework* de gestão de redes flexível, escalável e necessita de poucos recursos para funcionar. O ILMI é um protocolo que permite obter das interfaces ATM os parâmetros relacionados com a camada ATM, rotas virtuais, circuitos virtuais, etc.

Actualmente o SNMP é o protocolo de gestão de redes mais utilizado [120]. Na década de oitenta o *Internet Engineering Task Force* (IEFT) apresentou algumas ferramentas de apoio à gestão de redes, o SNMP e a *Management Information Base* (MIB). No final da década de oitenta, o abaixamento do preço dos computadores tornou-os acessíveis a mais utilizadores, massificando da sua utilização. Este facto teve um impacto positivo no vulgarmente denominado *boom* de informação. Esta reviravolta obrigou à criação de novos mecanismos e normas de gestão, mais ricos em funcionalidades e adaptados ao novo

cenário. A elevada dimensão e complexidade das redes de dados leva a outra necessidade, a necessidade de optimizar os custos de criação, manutenção e desenvolvimentos futuros da rede. Para minimizar os custos, são implementados mecanismos de controlo de qualidade de serviço, diferenciação de tráfego (*diffserv* [53]) e o suporte multifacetado de serviços a que as actuais redes de dados têm sido sujeitas. A gestão de sistemas traz vários problemas, que passam pela actualização de software, pela monitorização e controlo remoto dos sistemas, pela gestão de licenças, pela gestão das configurações, pelo inventário de hardware e software, isto é, todo um vasto conjunto de problemas que tem de ser solucionado com as ferramentas e mecanismos de gestão de redes.

Com base nas constatações apresentadas, a gestão de uma infra-estrutura informática é uma tarefa complexa e rigorosa, que está dependente não só dos meios tecnológicos existentes, mas também e sobretudo de técnicos especializados, pois só assim se poderá adequar a infra-estrutura de rede às necessidades dos seus utilizadores.

1.1. CONTEXTUALIZAÇÃO

De há uns anos a esta parte, o Departamento de Engenharia Geotécnica tem verificado um crescente aumento da utilização das tecnologias de informação, nomeadamente das redes de computadores. A rede aumentou em tamanho e complexidade, e surgiu a necessidade de desenvolver uma solução de software que permitisse realizar a gestão e monitorização centralizada da rede informática, de modo a garantir o serviço prestado aos seus utilizadores.

Estando o autor envolvido em enumeras tarefas de gestão da rede do DEG, procurou desenvolver no contexto da Tese um trabalho que pudesse dar resposta às dificuldades associadas ao processo de gestão. Assim, desenvolveu-se um trabalho no âmbito da gestão e monitorização de redes, baseadas na infra-estrutura *Simple Network Management Protocol* (SNMP) [3][19][22][23].

1.2. OBJECTIVOS

O presente trabalho pretende identificar uma série de requisitos de gestão e monitorização de uma *Local Area Network* (LAN), estudar e comparar as soluções do mercado *Open Source* que melhor lhe respondem, proceder à implementação e integração das soluções encontradas e analisar o resultado e aplicabilidade da solução encontrada, a um caso real.

O caso será a rede informática do Departamento de Engenharia Geotécnica do Instituto Superior de Engenharia do Porto.

1.3. ORGANIZAÇÃO DO RELATÓRIO

No capítulo 1 são apresentadas as motivações que deram origem a esta tese, os objectivos pretendidos e a organização do relatório.

Figura 1: Organização da Tese

No capítulo 2 apresentam-se os requisitos de gestão e monitorização da rede do DEG. Na primeira parte deste capítulo são apresentadas as principais arquitecturas e modelos de gestão de redes aplicadas no contexto de uma LAN, bem como os princípios fundamentais da gestão de redes, definidos pela *International Organization of Standardization* (ISO) e mais conhecidos por FCAPS: a gestão de falhas, a gestão de configurações, a gestão de contas, a gestão de performance e a gestão da segurança. Por último faz-se o enquadramento temático do controlo de tráfego, que é parte integrante da gestão de performance da rede. Na segunda fase do capítulo 2 é apresentado a rede do caso de estudo, a rede informática do DEG, sendo também identificados os parâmetros que se pretende controlar na rede.

No capítulo 3 apresenta-se o estado da arte das ferramentas *Open Source* de gestão e monitorização de redes. Das ferramentas apresentadas existem duas que destacam-se por suportarem a maior parte dos requisitos identificados, o OpenNMS e o *Network*

Administration Visualized (NAV). A escolha recaiu sobre o NAV por este possuir um leque mais completo de funcionalidades.

No capítulo 4 apresenta-se o estado da arte no que respeita às ferramentas de gestão de imagens de software. Com base nos requisitos do caso em estudo, faz-se a análise das ferramentas de gestão de imagens de software. A ferramenta de eleição é o *Free Open Source Ghost* (FOG). Apresenta-se o FOG, dando ênfase às funcionalidades de alto nível e aos principais protocolos utilizados.

No capítulo 5 apresenta-se o sistema *Network Management System with Imaging Support* (NMSIS). O NMSIS foi desenvolvido para dar respostas às necessidades de gestão da rede do DEG e resulta da integração entre as ferramentas *Network Administration Visualized* (NAV) e *Free Open Source Ghost* (FOG). Apresentam-se os três módulos que foram desenvolvidos para integrar o NAV e o FOG, são eles o *DataBase Synchronization Module* (DBSM), o *Group Equipment Synchronization Module* (GESM) e o *Dynamic Host Configuration Protocol Synchronizer* (DHCPSYNC). O DBSM permite a sincronização da informação entre as bases de dados NAV e FOG. O GESM organiza em grupos o equipamento de rede (com características semelhante). O DHCPSYNC faz a actualização automática do serviço DHCP, com os novos equipamentos de rede. Todos os programas desenvolvidos estão disponíveis através da interface Web do NMSIS.

No capítulo 6, começa-se por apresentar as principais vantagens e desvantagens das ferramentas de controlo de tráfego em Linux. Apresenta-se a *Framework Iproute2* dos sistemas operativos Linux, com especial detalhe na descrição de um elemento desta *framework*, o *Linux Traffic Control* (TC). São analisados e apresentados os principais elemento do *tc*, servindo de sustentação às opções que foram tomadas para a criação e desenvolvimento do sistema *BandWidth-Traffic Control* (BWTC).

No capítulo 7, apresenta-se o *BandWidth-Traffic Control* (BWTC). O BWTC foi criado com o intuído de atenuar o impacto negativo da instalação de imagens de software através da rede, que resulta no congestionamento de tráfego nas intersecções de rede. O BWTC é composto por três componentes: o *BandWidth-Traffic Control Calculate* (BWTC_CALC), o *BandWidth-Traffic Control Address Resolution Protocol* (BWTC_ARP) e o *BandWidth-Traffic Control Statistics* (BWTC_STATISTICS). Com cada um dos elementos do BWTC pretende-se dar resposta a vários problemas relacionados com a tarefa de modelação do

débito de dados do servidor de imagens em função da carga da rede. Os problemas prendem-se não só com a modelação do débito do servidor de imagens, mas também com a identificação da porta (do activo de rede) em que os computadores clientes estão ligados e qual a taxa de utilização das mesmas. O BWTC funciona ao longo do tempo de uma forma autónoma.

No capítulo 8 é feita a análise do desempenho do software BWTC. Para verificar a fiabilidade do BWTC foram elaborados vários testes à aplicação, em ambiente real. Com estes testes avalia-se: 1) o desempenho do BWTC em situações de descongestionamento e congestionamento de tráfego, quando está a decorrer a instalação de imagens de software em computadores remotos; 2) a capacidade do BWTC optimizar a utilização das portas dos activos de rede para 72% (valor definido pelo gestor) da capacidade máxima. Deste modo garante-se que existe largura de banda disponível para aplicações que entretanto arranquem; 3) qual o impacto nos serviços informáticos do DEG, resultante da instalação imagens de software em computadores remotos (através da rede de dados), sem o BWTC; 4) qual o impacto nos restantes serviços informáticos do DEG, resultante da instalação imagens de software em computadores remotos, usando o BWTC.

No último capítulo são apresentadas as principais conclusões e as propostas de desenvolvimentos futuros.

2. REQUISITOS DE GESTÃO E MONITORIZAÇÃO DA REDE DO DEG

2.1. GESTÃO DE REDES – ENQUADRAMENTO TEMÁTICO

As organizações investem grandes quantidades de tempo e dinheiro na construção de redes de dados complexas. Apesar das empresas/instituições dedicarem um ou mais engenheiros à manutenção da rede de dados, esta operação torna-se mais eficiente se as tarefas rotineiras forem automatizadas, libertando os recursos humanos. Deste modo o engenheiro pode ser alocado a tarefas de maior valor, como por exemplo o desenvolvimento futuro da rede. Desta necessidade nasceu o conceito de gestão de redes. A gestão de redes é o processo de controlar uma rede de dados complexa, maximizando a sua eficiência e produtividade [1].

2.1.1. ARQUITECTURAS DE GESTÃO DE REDES

Uma plataforma de gestão de rede pode usar várias arquitecturas para garantir todas as funcionalidades. As três mais conhecidas são: a arquitectura centralizada, a arquitectura hierárquica e a arquitectura distribuída [1][13].

Arquitectura centralizada

Uma arquitectura centralizada tem um sistema de gestão da rede (NMS – *Network Management System*) localizado num só servidor que é responsável por todas as tarefas de gestão, como se ilustra na Figura 2. Apesar do sistema central ser o elemento de coordenação da gestão da rede, este pode reencaminhar avisos e alarmes para outra consola, através da rede. Uma rede com uma arquitectura de gestão centralizada é utilizada para tratar todos os eventos e alertas da rede, toda a informação de rede, e aceder a todas as aplicações de gestão.

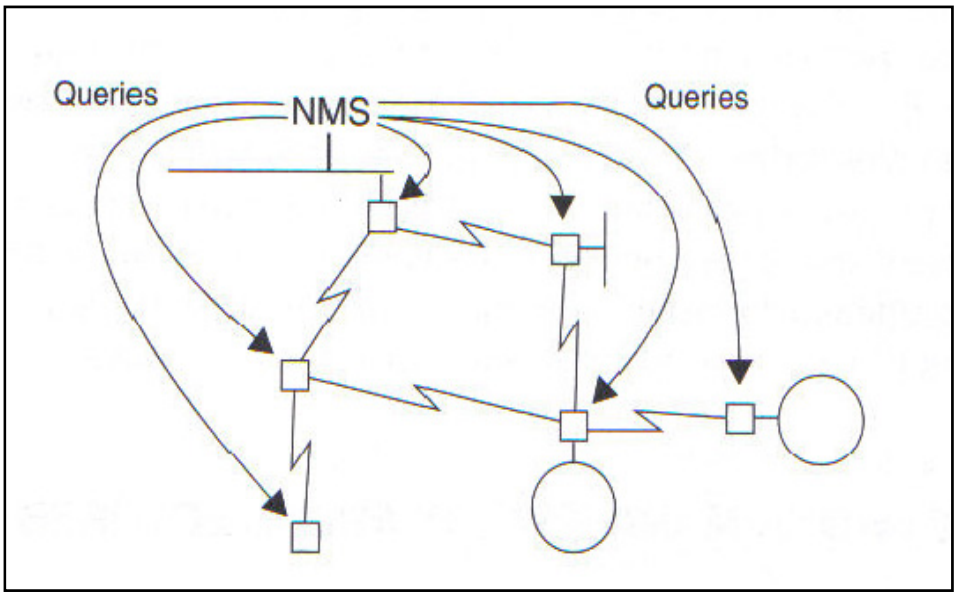

Figura 2: Esquema de uma Arquitectura Centralizada [1]

Ao utilizar uma arquitectura centralizada, o engenheiro de rede dispõe de um lugar único onde pode ver todos os eventos e alertas nela produzidos. Deste modo facilita-se o acesso às bases de informação que auxiliam uma análise de correlação de problemas e sua resolução.

No entanto, o facto de todas as aplicações de gestão dependerem de um só sistema com uma única base de dados, não é aconselhável no que se refere a problemas e falhas do sistema. Assim, as cópias de segurança devem ser regulares, de preferência para outro sistema físico. A eficiência e robustez de uma rede com arquitectura centralizada também estão directamente relacionadas com a dimensão da rede de dados. Quanto maior for a rede de dados maior será a sobrecarga do sistema de gestão e maior é a probabilidade de acorrerem falhas. Ao mesmo tempo, o sistema de gestão tem de abordar todos os dispositivos da rede a partir de uma só localização, aumentando a carga de tráfego nas

ligações de rede. Pelas razões mencionadas, a arquitectura centralizada é viável em pequenas redes de dados, como a rede do Departamento de Engenharia Geotécnica.

A localização da estação de gestão é também um factor crítico neste tipo de arquitectura, uma vez que a quebra de ligação entre esta e a rede de dados, resulta na perda das capacidades de gestão. Assim, a estação de gestão deve ser localizada num ponto central da rede, mesmo que esta não seja a melhor para o engenheiro de rede. Para minimizar este problema, o engenheiro de rede deve ter múltiplos pontos de acesso ao sistema central, nomeadamente consolas *Secure Shell* (SSH), interface Web, ou consolas *Virtual Network Computing* (VNC).

Arquitectura Hierárquica

Numa arquitectura de gestão de rede do tipo hierárquica, ao contrário do que acontece na arquitectura centralizada, são usados múltiplos sistemas de gestão, dos quais se destacam o sistema central e os clientes. Algumas funções da plataforma de gestão estão no servidor e outras nos clientes, o que lhes permite, por exemplo, monitorizar pequenas porções de rede. O esquema da arquitectura hierárquica está ilustrado na Figura 3.

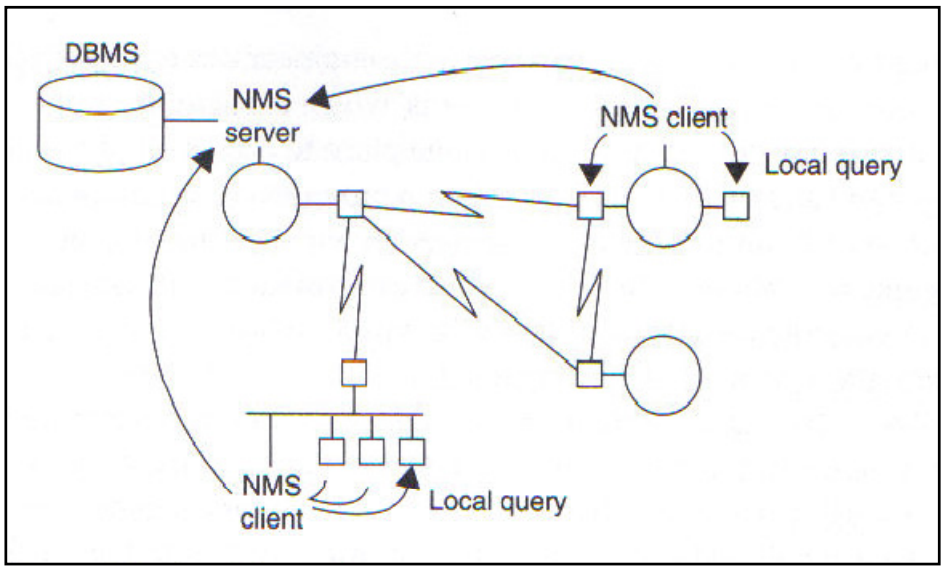

Figura 3: Esquema de uma arquitectura hierárquica [1]

A plataforma de gestão pode usar base de dados do tipo cliente - servidor, deste modo os clientes têm uma base de dados comum disponível no servidor, através da rede.

As principais características deste tipo de arquitectura são: a não dependência de um único sistema de gestão; possibilidade de dispor de tarefas de monitorização e gestão de rede distribuídas; armazenamento centralizado da informação.

A aproximação hierárquica permite aliviar um dos problemas da arquitectura centralizada, distribuindo as tarefas de gestão da rede pelo servidor central e clientes. O administrador da rede pode distribuir a monitorização da rede pelos clientes, poupando largura de banda em pontos de potencial congestionamento. Assim, quanto mais perto o cliente estiver dos dispositivos alvo de gestão, menor será a largura de banda utilizada e menor será a dependência do servidor central. Por outro lado a identificação de quais dos dispositivos são geridos por determinado gestor (servidor ou cliente), evita que dois gestores realizem a gestão do mesmo dispositivo, optimizando-se deste modo todo o processo e recursos associados. Uma desvantagem deste tipo de arquitectura está associada à necessidade da rede ser gerida por múltiplos sistemas, o que pode aumentar as tarefas do administrador da rede.

Arquitectura Distribuída

A arquitectura distribuída combina o melhor das aproximações centralizada e hierárquica. Ao contrário das outras arquitecturas que utilizam plataformas do tipo cliente servidor, a arquitectura distribuída utiliza plataformas multiponto. Neste tipo de plataforma existe um

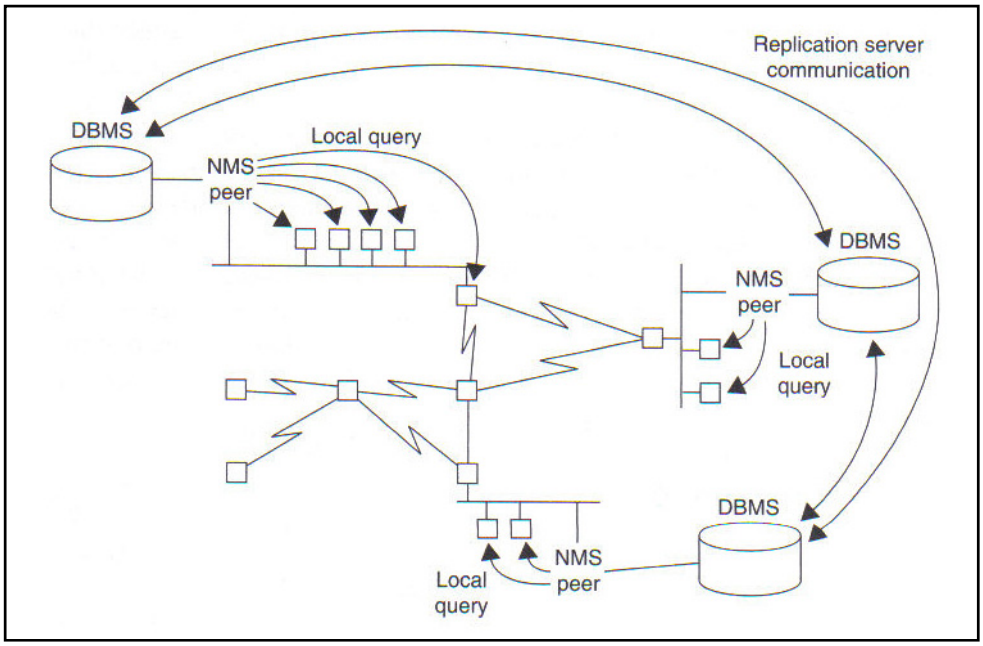

Figura 4: Esquema de uma Arquitectura Centralizada [1]

sistema central que lidera as restantes, no entanto cada plataforma pode ter uma base de dados com informação de rede, o que lhe permite realizar várias tarefas de gestão e monitorização e posteriormente reportar à central.

As vantagens das arquitecturas centralizada e hierárquica estão aqui reunidas, nomeadamente: uma localização para toda a informação de rede, alertas e eventos; a partir de uma só localização pode-se aceder a todas as aplicações de gestão; não depende de um só sistema; as tarefas de gestão estão distribuídas; existe uma distribuição da monitorização de rede, o que permite poupar largura de banda com tráfego de gestão.

Uma das desvantagens inerente à arquitectura distribuída está associada à manutenção e actualização de múltiplas bases de dados. A utilização da tecnologia de replicação da base de dados é extremamente útil neste tipo de plataforma. Nesta tecnologia, através de um servidor de replicação é feita a sincronização dos dados dos diferentes sistemas. Esta tecnologia é muito útil mas também muito complexa, logo é mais propicia a falhas e tem a desvantagem de utilizar maior largura de banda que os sistemas do tipo cliente - servidor.

2.1.2. MODELOS ARQUITECTÓNICOS DE GESTÃO

Modelo OSI

A ISO foi uma das primeiras organizações a definir formalmente uma forma comum de interligar computadores. A sua arquitectura é chamada *Open Systems Interconnection* (OSI) e surgiu na década de 70. Esta arquitectura define um modelo que estrutura os elementos da rede de dados em sete camadas, de forma a obterem-se camadas de abstracção funcional, implementadas por diferentes protocolos [2] [14] [15] [29] [30].

Camada física: Na camada física (camada 1) definem-se as características técnicas dos dispositivos eléctricos (físicos) do sistema; definem-se as características eléctricas e mecânicas do meio de transmissão; movem-se bits ou bytes através de um meio de transmissão; definem-se as taxas de transferência dos dados; definem-se as tensões de alimentação dos equipamentos ou periféricos; etc.

Camada Lógica: Na camada 2, detectam-se e corrigem-se erros que possam ocorrer no nível físico; fornece o suporte para activar, manter e terminar uma ligação sobre a camada física; faz o controlo de fluxo de dados; os sistemas são identificados pelo endereço MAC (Media Access Control); etc.

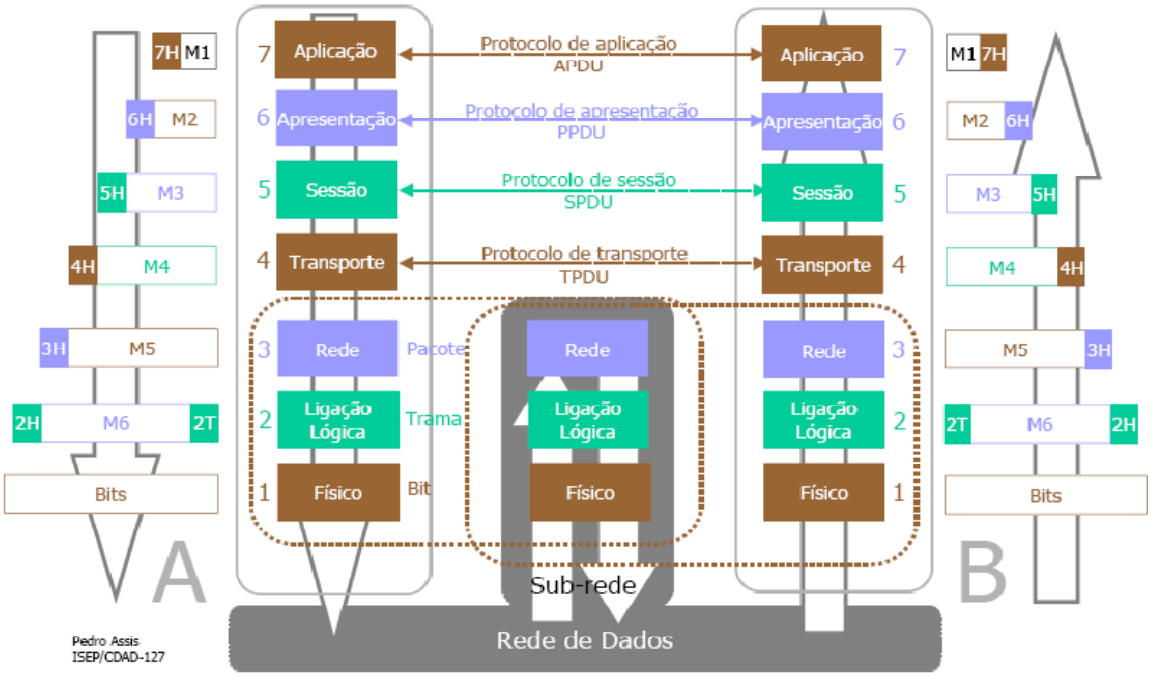

Figura 5: Modelo de referência OSI [78]

Camada de rede: A camada de rede (camada 3) é responsável pelo encaminhamento, fragmentação e endereçamento dos pacotes, convertendo endereços lógicos em endereços físicos (ou IP), de forma que os pacotes consigam chegar correctamente ao destino.

Camada de transporte: No emissor, a camada de transporte (camada 4) é responsável por encaminhar os dados provenientes da camada de Sessão para a camada de Rede, dividindo-os em pacotes. No receptor, acontece o inverso. A ISO define o protocolo de transporte para operar em dois modos: orientado às ligações e não - orientado às ligações.

As quatro camadas inferiores destinam-se a garantir a fiabilidade da comunicação extremo a extremo, as três camadas superiores definem aspectos relacionados com o serviço oferecido ao utilizador final (*user oriented*).

Camada de sessão: A camada de sessão (camada 5) define uma série de regras para estabelecer e terminar trocas de dados entre nós da rede. Atribuições do Nível de Sessão: gestão de testemunhos (*Token Managing*); definição de pontos de sincronização, re-sincronização e conversão de Endereços (*Address Mapping*).

Camada de apresentação: Ao nível da camada de apresentação (camada 6) realizam-se operações de conversão, formatação e sintaxe sobre os dados, de modo a que o aspecto visual dos dados coincida com o do dispositivo receptor. Alguns exemplos de conversões

efectuadas são: *American Standard Code for Information Interchange* (ASCII), *Extended Binary Coded Decimal Interchange Code* (EBCDIC) e ASN.1 (*Abstract Syntax Notation One*).

Camada de aplicação: Na camada de aplicação (camada 7) é estabelecido: o contexto necessário para suportar a comunicação entre aplicações; a descrição da lógica necessária para determinadas aplicações em particular; a interface com o utilizador; a gestão das aplicações; a interface entre o protocolo de comunicação e a aplicação que pediu ou receberá a informação através da rede.

Modelo TCP/IP

O modelo *Transmission Control Protocol / Internet Protocol* (TCP/IP) pode ser visto como um grupo de cinco camadas conceptuais, quatro camadas de *firmware* e software construídas sobre a camada de hardware. Como se ilustra na Figura 6, cada camada assegura um conjunto específico de funções na transmissão de dados, fornecendo um serviço bem definido para os protocolos da camada adjacente [2][19].

Camada de aplicação: Contém os programas que fornecem serviços (por exemplo: o *Telecommunication Network* (TELNET) [RFC318], o *File Transfer Protocol* (FTP) [RFC959] e o *Simple Mail Transfer Protocol* (SMTP) [RFC821]). Nesta camada escolhe-se o tipo de transporte necessário e enviam-se os dados para a camada adjacente. Corresponde às camadas 6 e 7 do modelo OSI.

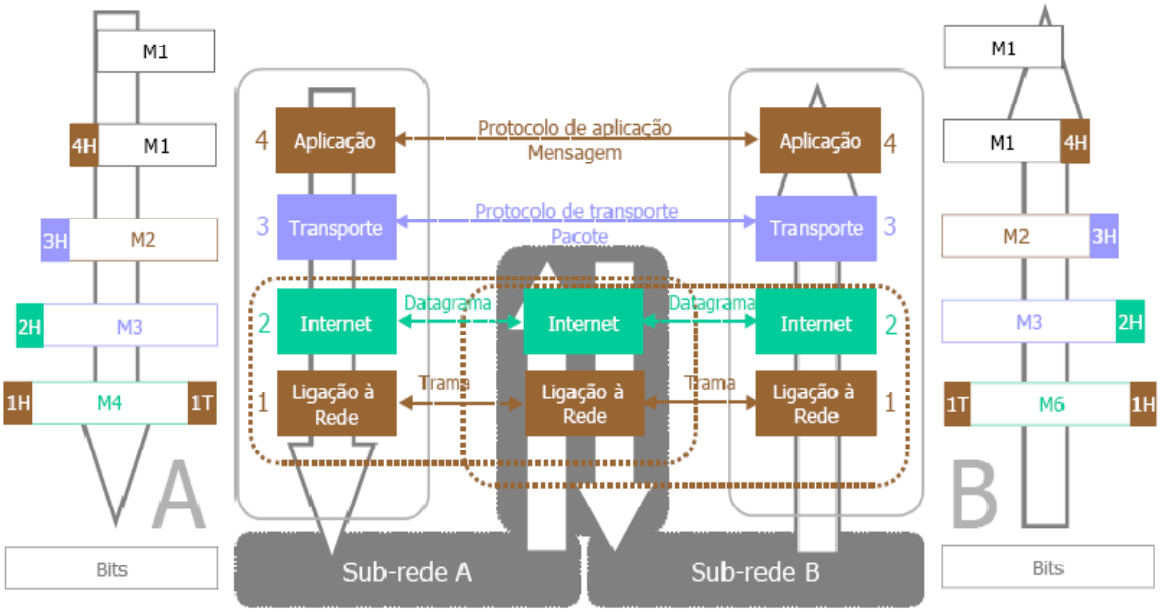

Figura 6: Modelo de referência TCP/IP [78]

Camada de Transporte: Define a forma de comunicação entre duas aplicações (ponto a ponto ou *unicast*); Controlo de erros e fluxo ao nível do TCP; Segmentação e reagrupamento das mensagens. Exemplos de protocolos implementados nesta camada: TCP [RFC793] e *User Datagram Protocol* (UDP) [RFC768]. Corresponde às camadas 4 e 5 do modelo OSI.

Camada Internet (Rede): Fornece a comunicação entre duas máquinas; é responsável pelo encapsulamento e encaminhamento de datagramas IP; verifica a validade dos *datagramas* recebidos; recebe e envia mensagens *Internet Control Message Protocol* (ICMP) de controlo, informação e erros. Corresponde à camada 3 do modelo OSI.

Camada de acesso ao meio: Recebe datagramas IP e envia-os para uma rede física específica; Recebe *datagramas* IP de uma rede física específica. Corresponde às camadas 1 e 2 do modelo OSI.

Um dos protocolos mais utilizados no apoio à gestão e monitorização de redes de dados é o *Simple Network Management Protocol* (SNMP), e que se encontra na camada de aplicação do modelo TCP/IP

2.1.3. SIMPLE NETWORK MANAGEMENT PROTOCOL - SNMP

O *Simple Network Management Protocol* (SNMP) foi introduzido na década de oitenta, no entanto, a sua versão inicial SNMPv1 continua a ser largamente utilizada. O SNMPv1 foi definido na série de standards *Internet Engineering Task Force* (IETF), na qual se definiram, não só, este protocolo, mas também a linguagem que estrutura as *Management Information Base* (MIB). A linguagem que define as MIB's é o *Structure of Management Information* (SMI) [RFC1155] e o seu sucessor SMIv2 [RFC2579].

O protocolo SNMPv1 foi projectado para permitir uma fácil implementação em *routers*, servidores, *switches*, etc., e por isso passou rapidamente a ser utilizado pela maioria dos fabricantes destes dispositivos. A utilização em massa deste protocolo revelou também os seus principais defeitos: falhas na comunicação entre gestores; defeitos no encapsulamento das tramas; falhas na segurança das aplicações; falta de encriptação na troca de dados nem privacidade no acesso às aplicações geridas.

Em 1993 foi proposta a versão SNMPv2. Pretendia-se incluir as funcionalidades de segurança (principalmente ao nível da encriptação dos dados) que faltavam na versão

anterior. Foram propostas várias versões do SNMPv2 (SNMPv2c, SNMPv2u, SNMPv2*), nas quais foram resolvidos parte dos problemas da versão anterior. Mas persistiram alguns dos problemas ao nível da segurança.

Em 1996 foi estandardizada a parte funcional do SNMPv2. Em 1997 começou o desenvolvimento da versão SNMPv3, esta disponibilizava as novas funcionalidades de interoperabilidade e gestão da segurança, colmatando algumas das falhas do SNMPv2, nomeadamente ao nível da segurança. O SNMPv3 inclui desenvolvimentos ao nível dos métodos de autenticação, encriptação, privacidade, autorização e controlo de acesso, e das ferramentas administrativas, por exemplo nomeação de entidades, gestão de utilizadores e chaves, configuração remota via SNMP, interoperabilidade com *proxy servers*.

Todas as versões do SNMP (v1, v2 e v3) partilham a mesma estrutura básica e componentes, e seguem a mesma arquitectura, pelo que as versões mais recentes suportam as versões anteriores. A *Framework* SNMP dá ênfase ao desenvolvimento modular de porções do SNMP, deste modo não é necessário redefinir a estrutura geral da *Framework*.

A MIB é constituída por objectos que representam variáveis e são mantidas e disponibilizadas ao cliente SNMP por um agente SNMP. Os objectos são identificados pelo *Object Identifier* (OID), definido no formato *Abstract Syntax Notation One* (ASN.1), o mesmo formato utilizado no modelo TCP/IP para definir nomes e tipos de variáveis. A MIB (versão inicial MIB-1), uma base genérica para qualquer agente, continha um limite predeterminado de 100 objectos. Com o crescimento tecnológico exponencial verificado na área dos equipamentos de rede surgiu a MIB-2, com conjunto bastante mais amplo de variáveis organizadas em grupos, nomeadamente tcp, udp, ip, system, snmp, egp, transmission, interfaces, entre outros. Após a versão dois, muitas outras especificações foram criadas, merecendo destaque o *Remote Monitoring MIB* (RMON) e o seu sucessor o RMON-2. O RMON é útil na monitorização da rede, uma vez que recolhe informação do estado da rede e não informação dos sistemas [3] [19] [22] [23] [24].

O SNMP foi projectado para funcionar na camada de aplicação do modelo TCP/IP, no entanto opera tipicamente sobre o protocolo *User Datagram Protocol* (UDP).

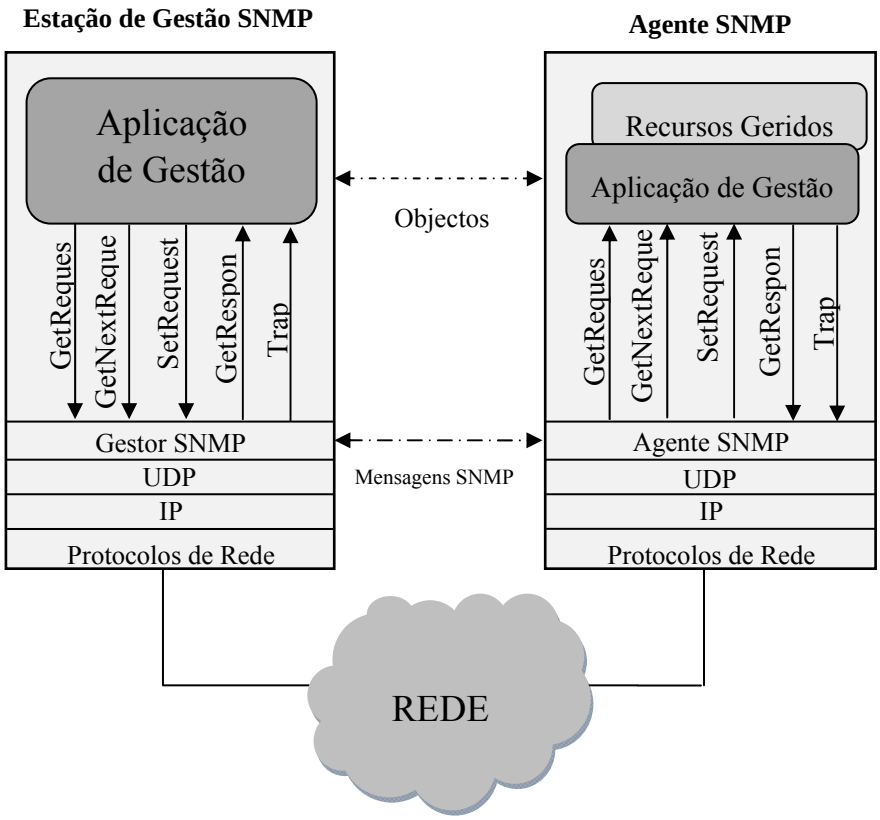

Figura 7: Processo de comunicação SNMP [3]

Cada dispositivo gerido tem um agente que, tal como o sistema de gestão de rede, implementa SNMP, UDP e IP, como se ilustra na Figura 7. A partir de um sistema de gestão são disponibilizadas três tipos de operações: *GetRequest, GetNextRequest e SetRequest*. As três operações são confirmadas pelo agente (no equipamento gerido) na forma de uma notificação *GetResponse* que é devolvida ao Sistema de Gestão. A comunicação entre o sistema de gestão e o agente SNMP decorre por UDP, pelo que não existe ligação permanente entre os dois. O agente do dispositivo a gerir pode ser *Master Agent* ou *SubAgent*, sendo responsável por manter a MIB actualizada, bem como disponibilizar a informação que lhe é solicitada pelo sistema de gestão através dos OID's [3] [19] [22] [23] [24].

2.1.4. MODELO FUNCIONAL DE GESTÃO DA ISO

Para melhor definir a linha de desenvolvimento da gestão de redes, a Organização Internacional de Estandardização (ISO - *International Organization of Standardization*) dividiu a gestão de redes em cinco áreas funcionais: 1) Gestão de falhas (*Fault management*); 2) Gestão de configurações (C*onfiguration management*); 3) Gestão de

contas (*Accounting management*); 4) Gestão do desempenho (*Performance management*); 5) Gestão da segurança (*Security management*). As cinco áreas funcionais são normalmente conhecidas por FCAPS [1][17][27][28].

Gestão de Falhas

A gestão de falhas é o processo de localização de problemas ou avarias numa rede de dados. O uso das técnicas de gestão de falhas permite ao administrador de rede identificar e resolver mais rapidamente os problemas. Alguns exemplos de técnicas utilizadas na gestão de problemas são:

- **Vigilância do estado da rede, gestão básica e complexa dos alarmes de rede.** Entende-se por gestão básica a tarefa de recolher os alarmes e manter o seu registo, bem como visualizar o estado da rede; a gestão complexa dos alarmes engloba conceitos de encaminhamento de alarmes para os técnicos responsáveis (via *pager*, correio electrónico ou SMS, por exemplo); aplicação de filtros aos alarmes e eventos; correlação de eventos e alarmes, para uma resolução inteligente dos problemas;
- **Diagnóstico de falhas, descoberta de problemas, isolamento de problemas e sua resolução;**
- **Gestão de falhas proactiva ou em antecipação:** proceder de modo a evitar que as falhas ocorram.

Gestão de Configurações

A configuração de certos e determinados elementos influencia o funcionamento de uma rede de computadores. A gestão de configurações é o processo de localizar e configurar serviços e elementos críticos, como *bridges, routers, etc.* Alguns exemplos de operações efectuadas na gestão de configurações são:

- **Configuração dos recursos geridos, sejam equipamentos de rede ou serviços da rede**. No centro da gestão de configurações estão as actividades e operações realizadas para configurar o que está a ser gerido, que envolvem o envio de comandos para os equipamentos de rede para alterar a sua configuração. Ao nível dos serviços, a configuração envolve a possibilidade de transformar o serviço, modificando certos parâmetros, ou mesmo desactivando-o. Este último aspecto é muitas vezes esquecido, mas é tão importante como a criação do serviço.

- **Sincronização da informação gerida na rede com a informação das aplicações de gestão.** A informação contida na rede (equipamentos e serviços) muda regularmente, sendo necessário mantê-la actualizada nas bases de dados do sistema de gestão. Existe a necessidade de definir qual das entidades tem a informação mais recente. Deste modo podem considerar-se dois pontos de vista: a rede de dados é considerada o *MASTER* e o sistema de gestão o *SLAVE* e vice-versa. Numa situação de funcionamento normal da rede de dados (sem falhas significativas), os equipamentos de rede e serviços contêm a informação mais recente relativamente à rede. Por outro lado, na eventualidade de ocorrer uma falha num equipamento ou serviço da rede, na sua recuperação é utilizada a informação armazenada na base de dados do sistema de gestão.
- **Inventário, auto-descoberta, descoberta e topologia de rede.** Estas tarefas envolvem consultar a rede para descobrir o que realmente foi configurado; adicionar novos equipamentos e serviços na base de dados de inventário; descobrir novos equipamentos ou quais os equipamentos que estão ligados na rede; realizar o controlo das ligações entre os equipamentos e/ou serviços da rede.
- **Guardar cópias de segurança da configuração da rede e sua reposição caso seja necessário**. Nunca se sabe quando um disco rígido vai ficar danificado ou se um PC contrai um vírus que pode destruir o sistema de arquivos, pelo que é de elevada importância proteger os dados através da realização regular de cópias de segurança.
- **Gerir imagens do software contido no equipamento de rede**: Como acontece com os computadores, os fabricantes de equipamentos de rede emitem, ocasionalmente, revisões de software, pelo que é necessário actualizar os sistemas da rede. Para fazer isso de forma eficaz instalam-se os sistemas a partir de imagens de software, actualizando somente os dispositivos que necessitam, sem interromper o todo o serviço.

Gestão da Segurança

A gestão da segurança é o processo de controlar o acesso à informação numa rede de dados. Alguma informação guardada em computadores ligados à rede poderá ser restrita a alguns utilizadores. Para evitar a utilização indevida de informação confidencial são utilizadas algumas técnicas, tais como: proteger a informação; analisar ameaças e definir indicadores de segurança; rever e analisar regularmente a estrutura de segurança; seleccionar e implementar serviços de gestão da segurança. É necessário distinguir dois

aspectos: segurança da gestão, que assegura que a gestão propriamente dita é segura, e gestão da segurança, que gere a segurança da rede.

Gestão de contas

A gestão de contas envolve: uma classificação e estruturação dos indivíduos em grupos de utilizadores; a definição das permissões de um utilizador no acesso à rede ou serviço, garantindo que cada utilizador tem recursos de rede suficientes; a proibição do acesso à rede a utilizadores não autorizados. Algumas das técnicas utilizadas são a determinação da taxa de utilização dos recursos, o licenciamento de software, a utilização e verificação de contas de utilizadores para acesso aos recursos da rede.

Gestão da Performance

A gestão da performance envolve a medição do desempenho do hardware e software que constitui uma rede de dados. Esta área inclui funções para monitorizar parâmetros de performance como: o número de mensagens erradas num activo de rede; as estatísticas de tráfego em servidores e activos de rede; as estatísticas de utilização de CPU e memória, entre outros. A monitorização e controlo da qualidade de serviço (QoS) são outro exemplo da função gestão da performance. Numa rede de dados a QoS é a capacidade de entregar o tráfego de dados sem erros e com o mínimo atraso, num ambiente em que vários utilizadores partilham a mesma rede [22]. Neste contexto, os valores de *threshold* [73], *jitter* [22][33] , *ceil* [47], *throughput* [22] e *delay* [22] são considerados parâmetros que caracterizam uma rede de dados. Quando o seu valor é excedido, podem ser gerados eventos de notificações para informar o sistema de gestão e o gestor da rede. O *threshold* é um atributo utilizado para definir o patamar aceitável, crítico e não aceitável. Por exemplo, pode ser lançado um alarme quando a taxa de transferência de dados passa o valor de *threshold* predefinido. O *delay* refere-se ao período de tempo que demora a transmitir um bit da origem até ao destino. O *jitter* representa a variação do *delay* ao longo do tempo. O *throughput* é o número efectivo de bits transmitidos, por segundo, através de uma interface de rede ou sistema, tendo em consideração o *delay*. No contexto do controlo de tráfego o *ceil* refere-se à frequência máxima de dados que uma interface de rede ou classe de dados pode atingir.

2.1.5. MECANISMOS DE CONTROLO DE TRÁFEGO EM SISTEMAS LINUX

O que é controlo de tráfego (traffic control)?

O controlo de tráfego é o termo dado a todo a subsistema de organização de pacotes de dados em filas (*queuing*), quer seja num dispositivo de rede ou na rede de dados. O controlo de tráfego é composto de várias operações distintas. [46] [47]

O controlo de tráfego é um tema complexo. Antes de explorar os mecanismos de controlo de tráfego existentes em Linux, vão ser apresentados alguns conceitos chave para uma percepção plena.

- **Queues**

As filas (*queues*) são a base do processo de controlo de tráfego. Uma fila é uma posição (ou *buffer*) que contém um número finito de itens que esperam por uma acção ou serviço. Numa rede de dados, uma fila é o lugar onde os pacotes de dados (itens) esperam para ser transmitidos pelo hardware (o serviço). No modelo mais simples, os pacotes são transmitidos pela ordem de chegada, isto é, o primeiro que entra é o primeiro a sair. Este tipo de fila é conhecido como FIFO (*first-in-first-out*).

Sem qualquer outro mecanismo associado, uma fila não oferece nenhuma funcionalidade de controlo de tráfego. Quando associadas a outros mecanismos, as filas podem ser usadas para atrasar pacotes, reajustar, eliminar e alterar a prioridade de pacotes, em múltiplas filas. Numa fila podem ser consideradas subfilas, muito utilizadas nas operações complexas de *scheduling* (organização e reorganização de pacotes).

Da perspectiva das camadas de software mais elevadas, os pacotes são simplesmente introduzidos nas filas para transmissão, a maneira e a ordem pela qual os pacotes são transmitidos é-lhes irrelevante. Deste modo, para as camadas superiores, todo o sistema de controlo de tráfego pode aparecer como uma única fila. [46] [47]

- **Flows**

Um fluxo (*flow*) é uma ligação distinta ou conversação entre dois *peers*. Qualquer troca de pacotes de dados entre dois anfitriões pode ser considerada como um fluxo. No protocolo TCP o conceito de ligação entre uma origem (IP e porto) e destino (IP e o porto) representa

um fluxo. Um fluxo em UDP representa a troca de dados entre anfitriões, ainda que sem o conceito de ligação entre estes.

Alguns mecanismos de controlo separam o tráfego em classes de fluxos, que podem ser agregados e transmitidos como um fluxo agregado (considerando *DiffServ* [RFC2475] [53]). Outros mecanismos, tal como o *traffic shaping* (modelação de tráfego), dividem a largura de banda de acordo com os fluxos individuais. [46]

- **Tokens - buckets**

Dois conceitos importantes que estão na base para a compreensão da modelação do tráfego (*traffic shaping*) são os conceitos de *Tokens* e *Buckets*. Para controlar o débito de saída de informação das filas, uma implementação pode contar o número de pacotes ou *bytes* para cada item que é retirado da fila, embora isto implique a utilização complexa de temporizadores e medições. Em vez de calcular a frequência de saída dos dados das filas, um método muito usado no controlo de tráfego é introduzir *tokens* (símbolos) no *bucket* à frequência desejada. Assim os pacotes são retirados das filas somente quando um *token* está disponível. A Figura 8 representa o algoritmo *Token Bucket*.

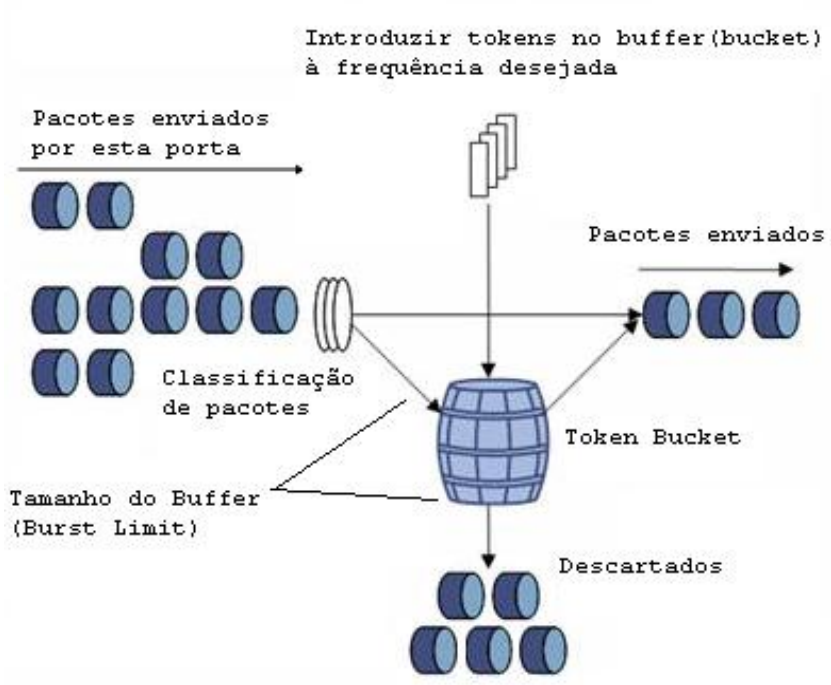

Figura 8: Algoritmo *Token Bucket*

Considere a analogia entre um *Token Bucket* e um parque de divertimento com uma fila de gente à espera para fazer um passeio de barco. Neste parque existem barcos que atravessam um rio e chegam à cabeceira da fila a uma frequência fixa. O parque é análogo a um *bucket,* o barco é análogo a um *token* (símbolo) e a pessoa é análoga a um pacote de dados. Só um determinado número de pessoas pode experimentar o passeio num determinado período de tempo. [46] [47]

Os conceitos de *tokens* e *buckets* estão directamente relacionados com dois dos mecanismos de controlo de tráfego mais relevantes em sistemas Linux: o *Token Bucket Filter* (TBF) e o *Hierarchichal Token Bucket* (HTB).

- **Packets - frames**

A trama (*frame*) é tipicamente usada para descrever uma unidade de dados da camada 2 (camada de ligação lógica do modelo OSI) a serem expedidos ao receptor seguinte. Nas interfaces Ethernet, *Point-to-Point Protocol* (PPP) e T1 as unidades de dados são denominadas *frames*. O controlo de tráfego é aplicado ao nível das *frames*. O termo *packet* (pacote), por outro lado, é o termo utilizado para designar as unidades de dados na camada 3 (camada de rede do modelo OSI). [46] [47]

- **Outros elementos do controlo de tráfego**

A classificação (*classifying*) é um mecanismo pelo qual se identificam as unidades ou pacotes de dados, e se colocam em diferentes fluxos (*flows*), classes (*classes*) ou filas (*queues*). O policiamento (*policing*) é um mecanismo pelo qual se limita o débito de dados que entra ou sai do equipamento de rede. O escalonamento (*scheduling*) é o processo pelo qual os pacotes são organizados ou reorganizados para o processo de transmissão (por exemplo FIFO). A modelação do tráfego (*traffic shaping*) é o processo pelo qual os pacotes são atrasados com um determinado *delay,* e posteriormente transmitidos, de forma a produzir uma determinada taxa de transferência ou limitar a largura de banda. O conceito *dropping* está associado ao processo pelo qual se eliminam pacotes de dados, fluxos ou classificações. A marcação (*marking*) é um mecanismo que permite alterar os pacotes de dados, introduzindo um valor de *Differentiated Services Code Point* (DSCP) [RFC2474] [RFC2475] no campo *Differentiated Services* (DS) da trama de dados. Na arquitectura do *DiffServ* o campo *Differentiated Services* (DS) substitui o campo *Type of Services* (ToS) [RFC791] da trama IPV4 e o campo *Traffic Class* [RFC2460] da trama IPv6 [134].

Posteriormente o valor de DSCP é respeitado nos activos de rede que implementam o standard *DiffServ* (serviços diferenciados) [RFC2475].

2.2. CASO DE ESTUDO – A REDE DO DEG

2.2.1. A INFRA – ESTRUTURA FÍSICA DA REDE

A rede do Departamento de Engenharia Geotécnica (DEG) do ISEP encontra-se ao serviço dos docentes, discentes e alunos do departamento.

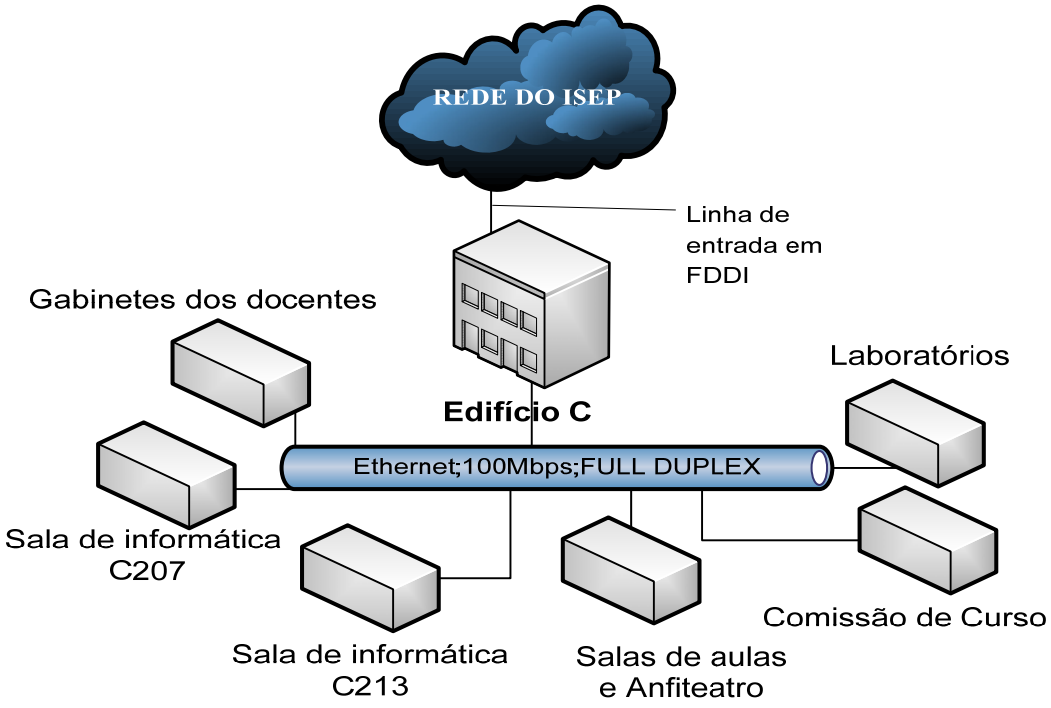

Figura 9: Estrutura lógica da rede de dados do DEG

Na Figura 9 estão representadas as ligações lógicas da infra-estrutura de rede do DEG. Esta rede está distribuída por dois pisos, o rés-do-chão e o 1º andar. A rede do DEG é constituída por oito gabinetes de docentes, duas salas de informática, quatro laboratórios, um anfiteatro e seis salas de aula. Cada uma das estruturas referidas tem uma ou mais ligações de rede do tipo *Ethernet* a 100Mbps, com excepção da ligação à rede geral do ISEP que é feita em fibra óptica (multi-modo 62,5/125µm).

No 1º andar do DEG são utilizadas a maior parte das ligações de rede disponíveis, o seu esquema está representado na Figura 10.

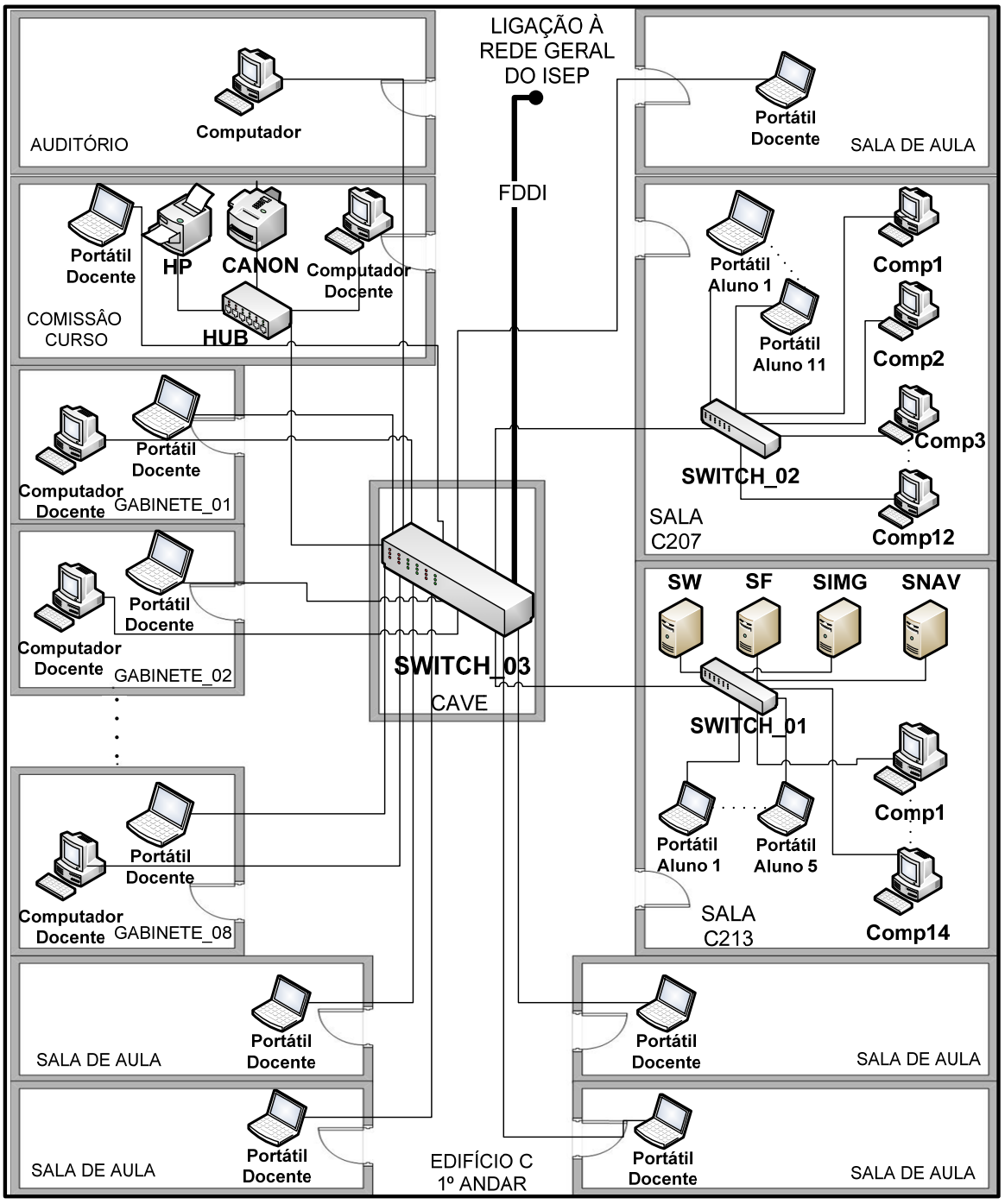

Figura 10: Esquema da rede de dados do 1º andar do DEG

O SWITCH_03, representado na figura anterior, é o ponto central da rede do DEG, e que faz a ligação à rede geral do ISEP. Este activo de rede funciona a 100Mbps, *full-duplex*. As salas de informática C207 e C213 são os locais em que a rede de dados tem maior afluência de utilizadores. No caso particular da sala de informática C213, para além dos computadores de uso geral, a rede de dados também suporta os servidores do DEG. Os servidores do DEG estão

representados na Figura 12, com os acrónimos SIMG, SF, SW e SF. Nas Figuras 11 e 12 estão representadas os equipamentos que estão ligados de forma permanente à rede, e que são objecto de gestão por parte do gestor da rede.

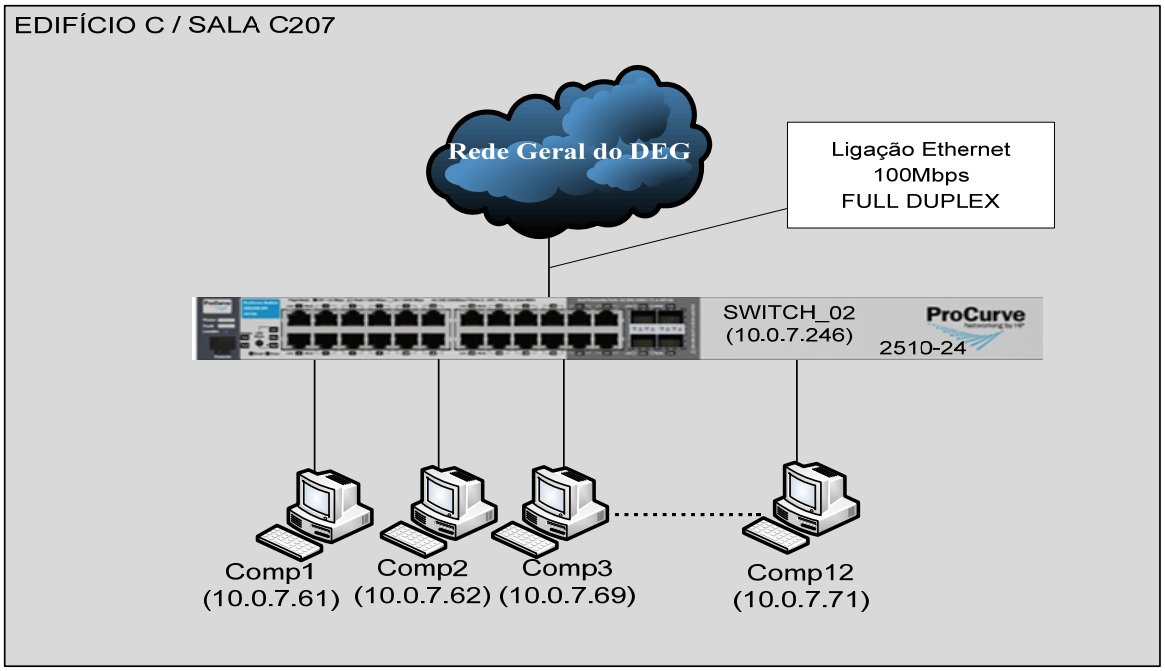

Figura 11: Esquema de rede da Sala de informática C207

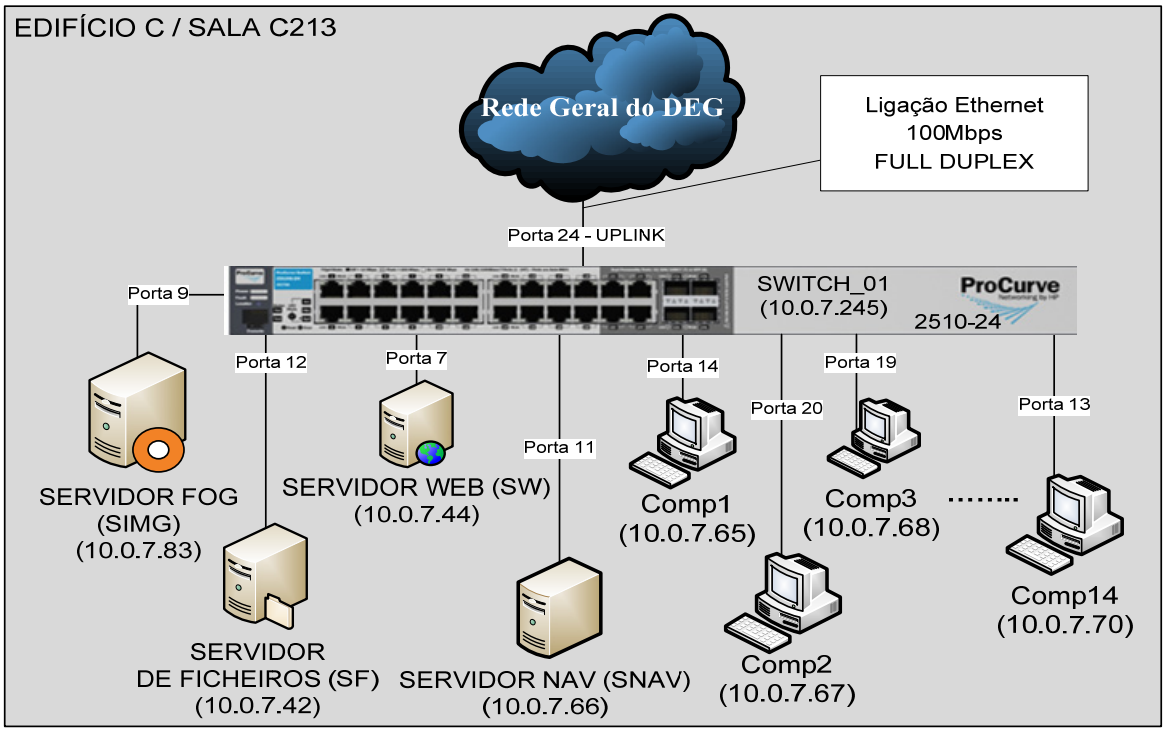

Figura 12: Esquema de rede da Sala de informática C213

A ligação entre as salas de informática C207 e C213 é feita pelo SWICTH_03, representada de forma simples na Figura 10, e com maior pormenor na Figura 13.

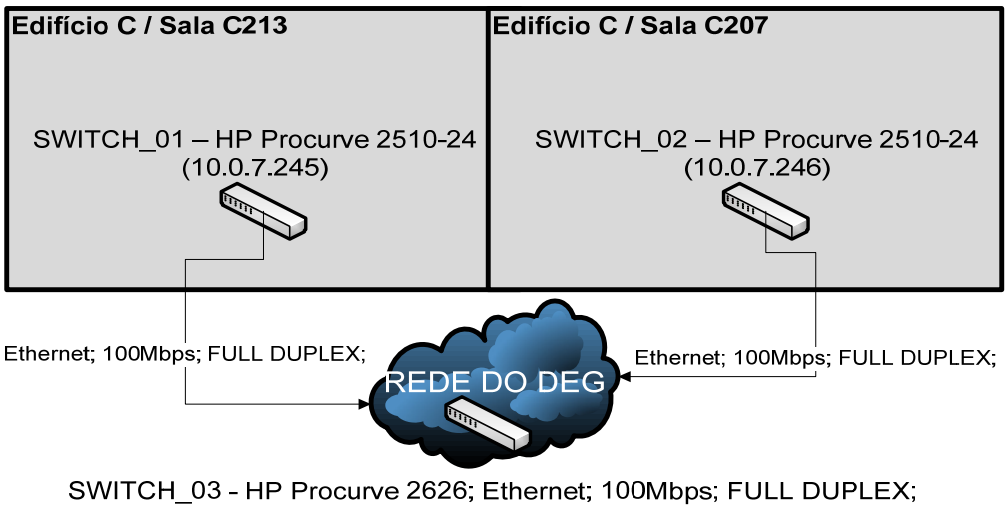

Figura 13: Esquema de ligação entre os SWITCH_01, SWITCH_02 e SWITCH_03

A rede de dados do rés-do-chão do DEG está ilustrada no Anexo B.

2.2.2. A INFRA – ESTRUTURA DE SERVIÇOS

Ao nível dos serviços, esta rede disponibiliza aos seus utilizadores as seguintes valências:

- **Serviço de páginas Web**. A página de internet tem, por um lado, uma vertente informativa, disponibilizando informação departamental aos utilizadores em geral e aos docentes e alunos em particular, e por outro lado, uma vertente pedagógica de auxílio às actividades dos docentes, nomeadamente no que se refere à distribuição de documentação de apoio às aulas.
- **Serviço de internet.** O acesso à internet é facultado pela rede geral do ISEP, a ligação do Edifício C à rede geral do ISEP é realizado por uma linha em fibra óptica multi-modo de 62.5/125 µm. O acesso à internet é exclusivo para utilizadores registados na rede de dados, seja por registo do endereço *Media Access Control* (MAC) no servidor DHCP (no acesso por cabo de rede), seja pelas credenciais de aluno nos acessos *wireless* (autenticação por WPA, TKIP e SecureW2). A maior percentagem dos acessos à internet é realizada pelos alunos nas salas de informática.
- **Servidor de ficheiros.** Na rede do DEG existe um servidor de ficheiros implementado em Windows Server 2003, que server principalmente para apoio às funções realizadas pelo engenheiro gestor da rede e para cópia de segurança de documentos dos docentes.

- **Serviço de contas de utilizadores.** Os alunos dispõem de contas de utilizadores que lhes permite utilizar os recursos dos computadores das salas de informática. Este serviço encontra-se implementado em SAMBA [114] Linux, e está a funcionar no mesmo servidor que suporta o serviço de páginas Web.
- **Serviço de impressão.** O serviço de impressão é exclusivo para os docentes e funcionários do departamento. Neste serviço estão incluídas duas impressoras de rede com suporte para o protocolo SNMP. Este serviço é gerido pela reprografia do ISEP em colaboração com o DEG.

2.3. IDENTIFICAÇÃO DE REQUISITOS DE GESTÃO E MONITORIZAÇÃO

A identificação dos aspectos a controlar e monitorizar na rede será baseada no modelo FCAPS, definido pela ISO. De acordo com as características da rede de dados do departamento, os principais elementos a considerar serão:

- **Gestão de falhas:** vigilância do estado da rede; gestão de alarmes de rede; manutenção de histórico de alarmes; diagnóstico de falhas; detecção, identificação e isolamento de problemas, resolução dos problemas (se possível); controlo e monitorização do estado dos serviços (por exemplo o serviço de impressão e o serviço de páginas Web); identificação do estado dos sistemas da rede, tal como se um servidor ou impressora está alcançável na rede.

- **Gestão de configurações:** inventário do equipamento de rede, necessário para a certificação da instituição pela norma ISO 9001:2000; localização de equipamento na rede; históricos de movimento das máquinas na rede; gestão das imagens de software contidas nos equipamentos de rede, principalmente dos servidores (sistemas operativos Ubuntu, Debian e Windows Server 2003), dos computadores das salas de informática (Windows XP) e dos computadores dos docentes (Windows XP e Vista).

- **Gestão da segurança:** implementar *firewalls* nos servidores para impedir os acessos não autorizados; detectar e eliminar (se possível) vírus informáticos.

- **Gestão da performance:** medição e controlo do desempenho dos equipamentos que constituem elementos críticos da rede de dados, nomeadamente os activos de rede, o servidor Web e o servidor de ficheiros.

- **Gestão de contas:** controlar o acesso a determinada informação de rede, nomeadamente a informação contida no servidor de ficheiros (Windows Server 2003); controlo de acessos à ferramenta de gestão e monitorização da rede, bem como à ferramenta de gestão de imagens de software; serviço de contas de utilizadores em SAMBA – Linux, para limitar a utilização dos computadores das salas de informática aos alunos do DEG.

2.4. IDENTIFICAÇÃO DE REQUISITOS DE GESTÃO DE IMAGENS DE SOFTWARE

Devido à elevada utilização dos computadores das salas de informática, é frequentemente necessário repor imagens de disco (ou imagens de software), não só devido a vírus, falhas no software, cota de disco no limite, etc., mas também pela necessidade de instalar um novo software, *upgrades*, actualizações de segurança, entre outros. Por outro lado, é importante criar regularmente cópias de segurança do servidor da página de internet do DEG.

As razões referidas convergem para a necessidade de desenvolver um módulo de software para o sistema de gestão com as seguintes características: instalação, clonagem, *backup* e recuperação de imagens de software de discos de computadores e servidores; deverá suportar os sistemas de ficheiros *second extended file system* (EXT2), *third extended filesystem* (EXT3), SWAP, *File Allocation Table 16* (FAT16), *File Allocation Table 32* (FAT32), *New Technology File System* (NTFS) e RAW; o módulo de software deverá ter interface Web para gestão, com controlo de acessos, para que se possa utilizar a ferramenta de gestão de imagens a partir de qualquer lugar, e deste modo colmatar uma das desvantagens da arquitectura de gestão centralizada; deve ser construído com ferramentas licenciadas pela *GNU - General Public License*.

A rede de dados do DEG suporta vários serviços informáticos. Um dos problemas intrínseco ao software de instalação de imagens de software é o facto de sobrecarregar a rede de dados, prejudicando significativamente a qualidade dos restantes serviços. Assim, é importante que o mecanismo de instalação de imagens de software actue de forma dinâmica, ao longo do tempo, recolhendo estatísticas de tráfego dos activos de rede e modelando o débito do servidor de imagens de software, de modo a minimizar o impacto negativo da instalação de imagens de software, nos restantes serviços informáticos.

2.5. RESUMO

Neste capítulo identificam-se alguns dos requisitos de gestão da rede de dados do DEG. Numa fase inicial é feito o enquadramento temático sobre a gestão e monitorização de redes; e, posteriormente, uma análise referente ao caso de estudo, a rede informática do DEG; por último, identificam-se os requisitos de gestão da rede do DEG.

Numa primeira fase exploraram-se as principais arquitecturas de gestão de redes, entre as quais, a arquitectura centralizada, a hierárquica e a distribuída. A arquitectura centralizada é a mais utilizada no contexto do caso em estudo. Em seguida foram apresentados os modelos arquitectónicos de gestão de redes mais utilizados, os modelos OSI e TCP/IP, com especial referência ao *Simple Network Management Protocol* (SNMP). O SNMP é o protocolo de gestão que permite a troca de informação de rede entre os equipamentos da rede e o sistema gestor.

Posteriormente apresentaram-se o modelo funcional de gestão definido pela ISO e mais conhecido por FCAPS (*Fault Management, Configuration Management, Accounting Management, Performance Management, Security Management*). As FCAPS são a gestão de falhas, a gestão de configurações, a gestão de contas, a gestão de performance e a gestão da segurança. A gestão de falhas é o processo de localização de problemas ou avarias numa rede de dados. A gestão de configurações é o processo de localizar e configurar serviços e elementos críticos, como *bridges*, *routers*, etc. A gestão da segurança é o processo de controlar o acesso à informação numa rede de dados. A gestão de contas envolve uma classificação e estruturação dos indivíduos em grupos de utilizadores, definindo as permissões de cada utilizar no acesso à rede ou serviço. A gestão da performance envolve a medição do desempenho do hardware e software que constitui uma rede de dados. A monitorização e controlo da qualidade de serviços (QoS) são outro exemplo da função gestão da performance. Depois apresentaram-se alguns conceitos teóricos associados ao controlo de tráfego, que é parte integrante da gestão de performance.

Na segunda fase deste capítulo, foram apresentadas a infra-estrutura física e a estrutura de serviços do DEG. A infra-estrutura física consiste em vários equipamentos de rede, interligados entre si por ligações Ethernet de 100Mbps, full-duplex. A ligação desta rede à rede geral do ISEP é feita em fibra óptica (multi-modo 62,5/125μm). Apresentou-se também a estrutura de serviços informáticos, dos quais se podem salientar o serviço da página de internet, o servidor de ficheiros, e o serviço de acesso à internet. No final do

capítulo são identificados os requisitos de gestão e monitorização da rede de dados do DEG, tendo como referência os princípios fundamentais da gestão (FCAPS), as características físicas da rede do DEG, os serviços informáticos suportados pela rede e as necessidades dos seus utilizadores. No contexto da gestão de falhas destacam-se os requisitos de vigilância do estado da rede, monitorizando o estado de operabilidade dos serviços (por exemplo, o serviço de contas de utilizadores) e servidores (por exemplo, o servidor da página Web e o servidor de ficheiros) da rede do DEG. No âmbito da gestão da configuração dá-se especial ênfase à gestão de imagens de software (dos servidores e computadores de uso geral), com modelação do débito do servidor de imagens em função do estado da rede. Relativamente à gestão da segurança é importante ter *firewalls* nos servidores para impedir os acessos não autorizados. Para a gestão da performance definiram-se os requisitos de medição e controlo do desempenho dos elementos críticos da rede, nomeadamente dos servidores e dos activos de rede (*switches*, *routers*, etc.). Por último, definiram-se os requisitos de gestão de contas, nomeadamente o controlo de acessos ao servidor de ficheiros e o serviço de contas de utilizadores no acesso aos computadores das salas de informática (SAMBA - Linux).

3. FERRAMENTAS DE GESTÃO *OPEN SOURCE*

Actualmente, uma empresa de média dimensão, dispõe de uma rede de dados tão complexa e robusta como uma rede de elevadas dimensões de há dez anos atrás. No entanto, e ao contrário do que seria de esperar, apesar da utilização em massa das redes de dados, o custo das ferramentas de gestão comerciais não as tornou acessíveis à maioria das pequenas e médias empresas. As ferramentas de gestão comerciais como o HP Network Node Manager (NNM) do Hewlett Packard (HP) [75], a Tivoli NetView da IBM [76], a Unicenter TNG da Computer Associates International, a Intel LANDesk Management Suite da Intel [77], requerem licenças dispendiosas, implicando por isso um investimento pouco rentável para as pequenas e médias empresas. De alguns anos a esta parte, mesmo grandes empresas têm adoptado a vertente das ferramentas de gestão e monitorização *Open Source* [20].

3.1. FERRAMENTAS DE GESTÃO DE REDE

Pelo seu carácter de livre utilização, nas próximas secções serão exploradas várias ferramentas de gestão e monitorização de redes, disponíveis para utilização na internet: o HYPERIC HQ, o MRTG, o NAGIOS, o SNORT, o ZABBIX, o NETDISCO, o NAV, o

OPENNMS e o PROCURVE MANAGER (para produtos HP). A caracterização mais ou menos pormenorizada de cada uma das ferramentas é o reflexo da qualidade da respectiva documentação. Devido ao elevado número de elementos que caracterizam cada uma das ferramentas, e sendo muitos deles comuns às diferentes abordagens, nas secções 3.1.1 a 3.1.9 as ferramentas são apresentadas numa perspectiva geral. Na secção 3.1.10 são discriminadas as características de todas as ferramentas na forma de tabela.

3.1.1. HYPERIC HQ

O Hyperic HQ (de "*hi-PEER-ick*") é uma ferramenta muito popular na área de monitorização de redes de dados, e foi criada em 2004 pela equipa que inclui Javier Soltero, Charles Lee e Doug MacEachern [4].

Sistemas operativos suportados e linguagens de programação de implementação:

O Hyperic HQ é uma aplicação distribuída do tipo cliente servidor que opera no topo do software livre JBoss Aplication Server [88]. É escrito em Java e código C portável correndo por isso em Linux, Windows, Solaris, HP-UX e Mac OS X. O portal Hyperic HQ é uma interface escrita em Java e AJAX User Interface.

Princípio de funcionamento e principais funcionalidades:

Esta ferramenta tem a capacidade de identificar todos os recursos de uma rede, bem como as suas características, incluído hardware, sistema operativo, virtualização, base de dados, aplicações e serviços. Permite monitorizar dispositivos e serviços especificados pelo gestor, lançando alertas de acordo com o estado de funcionamento. O Hyperic mantém um histórico de eventos e faz a correlação de eventos para identificar os problemas mais rapidamente. No entanto, o Hyperhic não permite a monitorização de activos de rede, como *routers* ou *switches*. A funcionalidade de auto descoberta está inerente à presença do agente Hyperic HQ nos computadores/servidores a monitorizar, pois é este que verifica os recursos e serviços da máquina em questão, informando posteriormente o servidor Hyperic.

3.1.2. MRTG - MULTI ROUTER TRAFFIC GRAPHER

O Multi Router Traffic Grapher (MRTG), ou simplesmente MRTG, é um software que permite monitorizar e medir a carga de tráfego nas ligações de rede, apresentando-a na forma de gráficos. Foi inicialmente desenvolvido por Tobias Oetiker e Dave Rand para monitorizar o tráfego em *routers*, actualmente permite criar gráficos e estatísticas para quase todos os dispositivos de rede [5].

Sistemas operativos suportados e linguagens de programação de implementação:

O MRTG é escrito em Perl e pode trabalhar em Windows, Linux, Unix, Mac OS e NetWare.

Princípio de funcionamento e principais funcionalidades:

O MRTG utiliza o SNMP para enviar pedidos aos dispositivos, utilizando os identificadores de objectos (OID's). O dispositivo monitorizado, que deve suportar SNMP, dispõe de um agente que identifica na MIB os OID's especificados. Depois de recolher a informação correspondente, o agente envia os dados de volta ao MRTG. O MRTG guarda a referida informação no ficheiro de registo do cliente, juntamente com informação anterior referente ao dispositivo. Posteriormente, o software cria um documento HTML com os ficheiros de registo, contendo a lista de gráficos relativos a cada cliente.

3.1.3. NAGIOS

O Nagios é uma ferramenta de software de aplicação para monitorização de rede, muito popular entre as ferramentas *open source*. O Nagios, originalmente criado com o nome NetSaint, foi escrito e continua a ser mantido por Ethan Galstad, conjuntamente com os grupos de desenvolvimento oficial e não oficial. [6]

Sistemas operativos suportados e linguagens de programação de implementação:

Foi originalmente desenvolvido para operar em Linux, mas pode operar em sistemas da família Unix.

Princípio de funcionamento e principais funcionalidades:

Esta ferramenta realiza operações de monitorização de dispositivos e serviços seleccionados pelo gestor de rede, lançando alertas sempre que o estado de funcionamento piora ou melhora; monitorização de sensores (de temperatura, alarmes, etc.), desde que estes tenham a capacidade de enviar pela rede a informação recolhida; verificação dos serviços de forma paralela; definição hierárquica de equipamentos de rede; detecta, identifica e verifica se um sistema se encontra desligado ou se é inalcançável na rede; monitorização de equipamentos de forma redundante. Apesar da elevada flexibilidade do Nagios, a configuração desta ferramenta é morosa, pois é necessário elaborar *scripts* para quase todas as funções disponíveis, e para cada equipamento a monitorizar.

3.1.4. SNORT

O SNORT é um software de rede para detecção de intrusos ou ataques à rede de dados. Este software foi criado por Martin Roesch, mas actualmente é distribuído e desenvolvido pela Sourcefire [74] em colaboração com a comunidade Snort [7].

Sistemas operativos suportados e linguagens de programação de implementação:

Esta ferramenta funciona em Linux.

Princípio de funcionamento e principais funcionalidades:

Este software permite realizar análise de protocolo e pesquisa de conteúdo, para detectar ataques à rede de dados. O SNORT pode ser usado para detectar vários tipos de ataques à rede de dados, tais como: o *buffer overflows* [89], o *stealth port scans* [90], os ataques de aplicações Web, o *server message block probes* [91], o *fingerprinting attempts* [92], entre outras. Armazena os eventos em ficheiros de registo (*log files*). O SNORT pode também ser usado na prevenção de ataques, por exemplo simulando a ocorrência de ataques e verificando o impacto destes na rede. O SNORT pode ser combinado com outro sistemas, tais como o SnortSnarf [93], o Sguil [94], o *Open Source Security Information Management* (OSSIM) [95], e o BASE (*Basic Analysis and Security Engine*) [96], para disponibilizar a representação visual dos dados de intrusão. Quando associado a software como o ClamAV [97] e ao SPADE [98] permite realizar pesquisa de vírus nos pacotes de dados, bem como detectar anomalias na rede, respectivamente.

3.1.5. ZABBIX

O ZABBIX foi desenvolvido por Alexei Vladishev para monitorizar e identificar o estado de vários serviços de rede, servidores, e outro hardware de rede [8].

Sistemas operativos suportados e linguagens de programação de implementação:

O servidor ZABBIX encontra-se disponível para Linux, Solaris, HP-UX, AIX, Free BSD, Open BSD, MAC OS X, e suporta as bases de dados MySQL [99], PostgreSQL [100], SQLite [103]e Oracle [104]. O agente ZABBIX opera nos mesmos sistemas operativos que o servidor, incluindo o Tru64/OSF1, o Windows NT4.0, o Windows 2000, o Windows XP e o Windows Vista.

Princípio de funcionamento e principais funcionalidades:

Esta ferramenta permite: a monitorização proactiva com reacções complexas a eventos; a realização de testes de performance aos serviços de rede, servidores, etc. Em alternativa à instalação do referido agente, o ZABBIX permite monitorização via SNMP, nas suas versões 1, 2 e 3.

Funcionalidade Auto-Descoberta: Permite a auto-descoberta por gama de endereços IP, por serviços disponíveis e por versão SNMP; permite a monitorização automática de dispositivos descobertos, desde que estes tenham o agente ZABBIX instalado; permite a remoção automática das bases de dados de dispositivos perdidos ou inalcançáveis.

Monitorização Web: Monitorização de aplicações Web, suporta os métodos de obtenção de dados POST e GET e suporta os protocolos HTTP e HTTPS.

3.1.6. NETDISCO

O Netdisco é uma aplicação de gestão orientada para grandes empresas e redes universitárias. O projecto é suportado pela comunidade de software livre, liderado por Eric Miller, Bill Fenner, e Max Baker.

Sistemas operativos suportados e linguagens de programação de implementação:

O Netdisco opera em Free BSD, MAC OS X, LINUX e SOLARIS. Utiliza a base de dados PostgreSQL [100], e a interface Web está escrita em Mason [102].

Princípio de funcionamento e principais funcionalidades:

Nesta ferramenta os dados são armazenados usando um banco de dados SQL, permitindo escalabilidade e velocidade. Quando associada a alguns protocolos com topologia de rede de segunda camada (*Layer 2*) como o *Cisco Discovery Protocol* (CDP) [105] (suportado por alguns *routers* da CISCO) disponibiliza a descoberta automática da topologia da rede. O inventário da rede pode ser feito de dois modos, a partir do modelo do dispositivo ou por sistema operativo. Para cada equipamento de rede que utilize o protocolo TCP/IP é mantido um histórico das portas de *switches* que tem visitado, e os endereços IP que tem utilizado. O Netdisco recebe todos os dados, incluindo informação da topologia de rede, com pesquisas SNMP e consultas a servidores *Domain Name System* (DNS). Esta ferramenta não utiliza *Common Line Interface* (CLI) e não é necessária palavra passe para aceder à sua interface Web de gestão. Tem uma falha grande ao nível da segurança, uma vez que qualquer indivíduo com acesso ao servidor de gestão tem também acesso às funcionalidades de gestão. Apresenta um módulo de localização de pontos de acesso *Wireless* (AP's).

3.1.7. NAV - NETWORK ADMINISTRATION VISUALIZED

O Network Administration Visualized (NAV) [10] é um software avançado para auxílio à gestão e monitorização de grandes redes informáticas.

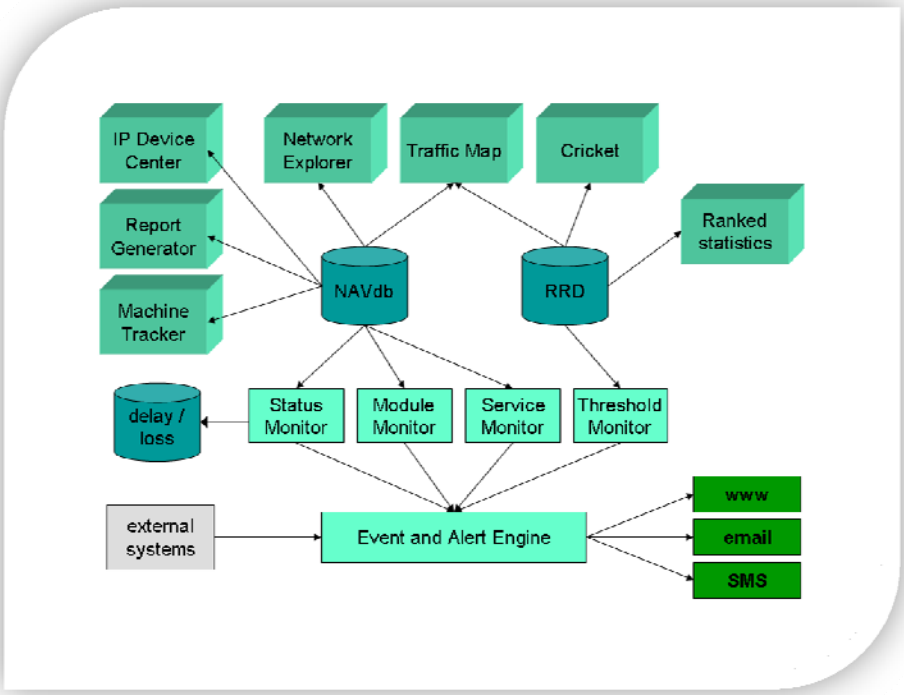

Figura 14: Perspectiva global das funcionalidades do NAV [10]

O NAV foi desenvolvido na Universidade Norueguesa de Ciências e Tecnologia (NTNU – Norwegian University of Science and Technology). O seu desenvolvimento iniciou - se em 1999, tendo origem num conjunto de *scripts* usados internamente pelo NTNU's ITEA Network Group. Em 2001 o UNINETT [86] ficou interessado no seu desenvolvimento, e desde então financia o desenvolvimento do NAV na NTNU, em nome de todas as universidades e colégios na Noruega. Em 2006 o desenvolvimento da NAV foi transferido de NTNU para UNINETT. Actualmente é utilizada por mais de 40 instituições de renome internacional [106]. A Figura 14 fornece uma perspectiva global das funcionalidades do NAV.

Sistemas operativos suportados e linguagens de programação de implementação:

Esta ferramenta opera em LINUX e Free BSD, está escrita predominantemente nas linguagens de programação Python e Java, mas contém Perl, PHP, SH e base de dados PostgreSQL.

Princípio de funcionamento e principais funcionalidades:

O NAV apresenta uma topologia baseada na arquitectura centralizada. A interligação dos elementos da rede, como *switches*, *routers* ou computadores, é de forma auto-detectável. Dispõe de um serviço que faz a monitorização do estado da rede e que detecta quando o tráfego nas intersecções da rede ultrapassa o valor definido pelo gestor (por exemplo 72% de taxa de utilização). O NAV fornece estatísticas de utilização para vários elementos de rede, tais como servidores, *routers* e *switches*, desde que estes suportem o protocolo SNMP; para os vários elementos da rede o NAV apresenta gráficos de utilização de CPU, memória, disco rígido, placas de rede, entre outros; disponibiliza um mapa de tráfego que mostra a topologia da rede ao nível de Rede, e ao nível de Ligação Lógica. O mapa da topologia de rede é construído com base em informação obtida dos *routers*. O tráfego e a carga de CPU dos equipamentos de rede podem ser mostrados no mesmo mapa. Para cada equipamento de rede que utilize o protocolo TCP/IP é mantido um histórico das portas de *switches* às quais esses equipamentos se têm ligado, bem como os endereços IP utilizados.

O NAV disponibilizada uma estrutura gráfica em árvore com a hierarquia da rede, segundo a camada 2 do modelo OSI (ligação lógica). Esta ferramenta permite pesquisar um dispositivo, obtendo-se a árvore de percurso até este.

No NAV é disponibilizado o inventário de todo o equipamento de rede, sejam computadores, impressoras de rede, *routers* e *switches*, etc. Também apresenta a versão de software destes equipamentos, o tipo de equipamento, a sua localização e uma interface gráfica para a utilização de portos de *switches* e *routers* (por exemplo, se as portas estão truncadas, livres, ocupadas, bloqueadas, etc.). A monitorização de carga de CPU, utilização de memória, tráfego nas intersecções de rede, entre outros, pode ser feita de forma permanente. Caso se atinja determinados valores de *threshold* são lançados alarmes (por email ou SMS). Com o NAV é igualmente possível monitorizar o estado de operacionalidade dos servidores, computadores, activos de rede, serviços (HTTP, POP, SMTP, SMB, etc.), impressoras de rede, entre outros. Os alarmes externos à rede também

podem ser recolhidos e processados. Adicionalmente, o NAV disponibiliza uma ferramenta para bloquear portas de *switches* via SNMP - o Arnold, uma outra ferramenta de apoio à manutenção e logística, e um mecanismo de gestão de contas de utilizadores NAV.

3.1.8. PROCURVE MANAGER - PCM

O ProCurve Manager (PCM) [11] é um software de gestão e monitorização para produtos e soluções ProCurve Networking [107], uma divisão da HP. O PCM está disponível em duas versões: uma versão base livre, disponível com todos os produtos ProCurve, e uma versão estendida que incorpora funcionalidades avançadas.

Sistemas operativos suportados e linguagens de programação de implementação:

Esta ferramenta opera em Windows.

Princípio de funcionamento e principais funcionalidades:

O PCM versão base livre utiliza *Link Layer Discovery Protocol* (LLDP), *Cisco Discovery Protocol* (CDP) e *Foundry Discovery Protocol* (FDP) para detectar dispositivos de rede na intranet. A inspecção é feita via SNMP V2c e V3. A gestão de tráfego é feita via *Remote Network Monitoring MIB* (RMON) e detecção de intrusos via SFLOW [121].

O PCM+, a versão estendida do PCM, integra também os módulos: IDM (*Identity Drives Manter*) que implementa mecanismos de segurança no controlo de acessos utilizando 802.1x; o PMM (*ProCurve Mobility Manager*) contém elementos de gestão para ProCurve Access Point (420/520/530) até à versão PCM+ 1.1; o NIM (*Network Immunity Manager*) para PCM+ v2.2 e seguintes.

3.1.9. OPENNMS

O OpenNMS é um projecto *Open Source*, dedicado à criação de uma plataforma de monitorização de rede. O seu desenvolvimento encontra-se a cargo da comunidade de código aberto do OpenNMS, do OpenNMS Group, de Tarus Balog e do The Order of The Green Polo. O objectivo para o OpenNMS é torná-lo verdadeiramente distribuído e escalável, de forma a cobrir todos os aspectos do modelo funcional de gestão FCAPS. [12]

Sistemas operativos suportados e linguagens de programação de implementação:

Esta ferramenta está escrita predominantemente em JAVA e opera em Linux, Solaris, Mac OS X e Windows.

Princípio de funcionamento e principais funcionalidades:

O OpenNMS possibilita a monitorização de elementos da rede usando qualquer browser, ou por ligações telnet, SSL ou SSH. As principais funcionalidades incluem a determinação da disponibilidade e a latência de serviços de rede; a recolha e armazenamento de dados dos equipamentos de rede (nomeadamente valores de performance, latência, tráfego de dados), recolhidos através dos protocolos SNMP, *Java Management Extensions* (JMX), http ou NSClient [108]; monitorização de valores de *threshold,* criando eventos quando esses valores são excedidos; recolha e armazenamento de eventos, quer sejam internos ou externos ao software (incluindo alarmes SNMP); notificação por correio electrónico ou *Extensible Messaging and Presence Protocol* (XMPP) de eventos relacionados com o estado da rede; localização de equipamento numa LAN.

Para a análise comparativa entre as ferramentas apresentadas, a Tabela 1 contém a lista pormenorizada das características de cada ferramenta.

3.1.10. RESUMO DO ESTADO DA ARTE DAS FERRAMENTAS DE GESTÃO E MONITORIZAÇÃO

A tabela seguinte resume as principais características das ferramentas apresentadas nos pontos 3.1.1 a 3.1.9. O significado da numeração está na Tabela 2.

Tabela 1: Tabela de resumo das ferramentas de gestão

	Características																												
Número	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
HYPERIC	■	■	■	■	■	■	■	■	■	■	■	■	■	■			■													■
MRTG	■														■						■									■
NAGIOS	■	■	■	■	■	■		■	■			■	■		■	■	■					■								■
SNORT	■																	■	■											■
ZABBIX	■		■	■		■		■	■					■		■	■			■	■	■			■					■
NETDISCO	■	■				■											■			■			■	■	■		■			■
NAV	■	■	■	■	■						■	■			■	■	■			■	■		■	■	■	■	■	■		■
OpenNMS	■	■	■	■	■						■	■			■	■	■				■				■	■				■
PROCURVE MANAGER	■									■		■			■			■			■		■							

Legenda da tabela: Característica disponível: ■

Lista de características:

Tabela 2: Tabela de características das ferramentas de gestão.

Nº	CARACTERÍSTICA	Nº	CARACTERÍSTICA
1	Suporta o protocolo SNMP.	**16**	Notificações de alertas de rede por E-MAIL, SMS, outras definidas pelo utilizador.
2	Inventário dos elementos da rede	**17**	Interface Web para gestão.
3	Monitorização remota de serviços de rede (SMTP, POP, HTTP, NNTP, ICMP, etc.).	**18**	Detecção de intrusos/ataques ao sistema; prevenção de ataques ao sistema de gestão.
4	Monitorização dos recursos do computador/servidor (CPU, memória, utilização de disco, etc.).	**19**	Análise de tráfego em tempo real.
5	Controlo remoto por túneis SSL ou SSH.	**20**	Base de dados para guardar a informação de rede.
6	Descoberta automática de recursos de rede.	**21**	Boa documentação e rápida curva de aprendizagem.
7	Controlo remoto de recursos, para acções de correcção tais como: iniciar/terminar um serviço;	**22**	Rotação automática de ficheiros de registo (*log files*).
8	*Event handlers* para serviços ou aplicações.	**23**	Tem funcionalidades para bloquear portas de SWITCHES.
9	Resolução de problemas de forma proactiva.	**24**	Utilização das tabelas ARP e tabelas de encaminhamento L2, para permitir a localização da porta do *switch* em que se encontra ligado um equipamento da rede.
10	Correlação de eventos.	**25**	Informação da topologia de rede.
11	Ferramentas de apoio à identificação de problemas e da sua origem.	**26**	Estatísticas e gráficos de tráfego para todas as intersecções na rede, incluindo portas de *switches*.
12	Controlo da segurança e acessos à ferramenta de gestão e monitorização.	**27**	Históricos de movimento das máquinas na rede.
13	*Simple Plug-in design for service checks* (XML, Bash, Perl, PHP, C++, C#, Python, etc.).	**28**	Tem ferramentas de apoio à manutenção e logística da rede. Delegação de tarefas de acordo com o tipo de utilizador.
14	Disponibiliza agente próprio.	**29**	Ferramentas de gestão de imagens de software dos equipamentos de rede; detecção e eliminação de vírus.
15	Monitorização do tráfego nas ligações de rede, com possibilidade de lançar alertas.	**30**	Licenciada segundo a licença GNU - *General Public License.*

3.2. SELECÇÃO DA FERRAMENTA DE GESTÃO E MONITORIZAÇÃO

Nas secções anteriores foi apresentado um leque de ferramentas de gestão e monitorização, que satisfazem a maioria dos princípios fundamentais da gestão. Em seguida serão apresentados os critérios de selecção da ferramenta de gestão, baseados na análise de requisitos de gestão da rede do DEG [2.3].

3.2.1. CRITÉRIOS DE SELECÇÃO DA FERRAMENTA DE GESTÃO

Com base na análise de requisitos de gestão da rede do DEG [2.3] foram definidos 18 critérios de selecção, indicados na Tabela 3.

Tabela 3: Tabela de critérios de selecção da ferramenta de gestão

Nº	CARACTERÍSTICA	Nº	CARACTERÍSTICA
1	Suporta o protocolo SNMP.	**21**	Boa documentação e rápida curva de aprendizagem.
2	Inventário dos elementos da rede	**23**	Tem funcionalidades para bloquear portas de SWITCHES.
3	Monitorização remota de serviços de rede (SMTP, POP, HTTP, NNTP, ICMP, etc.).	**24**	Utilização das tabelas ARP e tabelas de encaminhamento L2, para permitir a localização da porta do *switch* em que se encontra ligado um equipamento da rede.
4	Monitorização dos recursos do computador/servidor (CPU, memória, utilização de disco, etc.).	**25**	Informação da topologia de rede.
10	Correlação de eventos.	**26**	Estatísticas e gráficos de tráfego para todas as intersecções na rede, incluindo portas de *switches*.
12	Controlo da segurança e acessos à ferramenta de gestão e monitorização.	**27**	Históricos de movimento das máquinas na rede.
15	Monitorização do tráfego nas ligações de rede, com possibilidade de lançar alertas.	**28**	Tem ferramentas de apoio à manutenção e logística da rede. Delegação de tarefas de acordo com o tipo de utilizador.
16	Notificações de alertas de rede por E-MAIL, SMS, outras definidas pelo utilizador.	**29**	Ferramentas de gestão de imagens de software dos equipamentos de rede; detecção e eliminação de vírus.
17	Interface Web para gestão.	**30**	Licenciada segundo a licença GNU - *General Public License.*

A análise de requisitos de gestão (secção 2.3) efectuou-se com base no modelo funcional de gestão FCAPS definido pela ISO. Assim, no contexto da gestão das falhas (*Fault management*) definiram-se os critérios 1, 3, 4 e 16; no âmbito da gestão das configurações (*Configuration management*) definiram-se os critérios 1, 2, 21, 23, 24, 25, 27 e 29; para a gestão de contas (*Accounting management*) definiu-se o critério 12; para dar respostas à gestão da performance (*Performance management*) definiram-se os critérios 1, 15 e 26.

Existem dois requisitos de gestão apresentados na análise de requisitos [2.3] e que não foram definidos na Tabela 3, a gestão dos acessos ao servidor de ficheiros (Windows Server 2003) e a implementação de *firewalls* nos servidores para evitar os acessos não autorizados. O servidor de ficheiros está implementado em Windows Server 2003, pelo que já tem ferramentas de gestão integradas que permitem controlar os acessos ao servidor de ficheiros. Em relação às *firewalls*, os sistemas Linux têm a *firewall* IPTABLES, o servidor de ficheiros tem implementado as *firewalls* do Windows e a COMODO [122].

3.2.2. SELECÇÃO DA FERRAMENTA DE GESTÃO E MONITORIZAÇÃO

A selecção da ferramenta de gestão e monitorização passa por identificar as ferramentas que contêm a maior parte dos requisitos definidos na análise de requisitos de gestão da rede do DEG (secção 2.3). Com base na Tabela 4 faz-se a análise das principais vantagens e desvantagens de cada ferramenta em relação aos critérios de selecção definidos.

Tabela 4: Tabela de resumo das ferramentas de gestão *vs* critérios de selecção

	Características																												
Número	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
HYPERIC	●	●	●	●						●		●			●	●					●		●	●	●	●	●	●	●	●
MRTG	●	●	●	●						●		●				●	●						●	●	●	●	●	●	●	●
NAGIOS	●	●		●						●		●			●	●					●		●	●	●	●	●	●	●	●
SNORT	●	●	●	●						●		●			●	●	●				●		●	●	●	●	●	●	●	●
ZABBIX	●	●								●		●			●								●	●		●	●	●	●	●
NETDISCO	●	●	●	●						●		●			●	●	●				●		●	●	●	●		●	●	●
NAV	●	●								●		●			●	●	●				●		●	●	●				●	●
OpenNMS	●	●								●		●			●	●	●				●		●		●		●	●	●	●
PROCURVE MANAGER	●	●	●	●						●		●			●	●	●				●		●	●	●	●	●	●	●	●

Legenda da Tabela:	Característica disponível		
	Critério de selecção	● ● ● ● ● ●	

O HYPERIQ-HQ é uma ferramenta vocacionada para a monitorização de computadores e servidores, permitindo monitorizar carga de CPU, utilização de memória, cota de disco, entre outros, no entanto não permite a gestão e monitorização de activos de rede como *routers* ou *switches*.

O MRTG é uma ferramenta de auxílio à monitorização de parâmetros analíticos dos vários elementos constituintes da rede, como o tráfego nas portas dos *routers*, utilização de CPU e memória, etc. Normalmente é utilizado como módulo nas ferramentas de monitorização mais complexas.

O NAGIOS tem as mesmas funcionalidades que o HYPERIQ-HQ, com a inclusão da monitorização de *routers* e *switches*, e utiliza o MRTG para monitorizar tráfego nas ligações de rede. No entanto não permite a localização de equipamentos de rede numa LAN.

O SNORT é uma ferramenta direccionada para a detecção de intrusos, bem como a protecção e prevenção de ataques à rede de dados, tais como ataques de vírus de rede ou ataques de servidores externos. Como está representado na Tabela 4 o SNORT apresenta apenas duas funcionalidades, das 18 pretendidas.

O ZABBIX está particularmente direccionado para a realização de testes de performance a computadores e serviços de rede, permitindo enviar alertas para o gestor da rede por correio electrónico ou SMS. No entanto não tem, por exemplo, funcionalidades de gestão e controlo de acessos à ferramenta de gestão.

O NETDISCO possui uma arquitectura que se enquadra no contexto das redes universitárias. Esta ferramenta permite, por exemplo, criar gráficos de topologia de rede, localizar equipamento na rede, tem interface Web de gestão, etc. No entanto não faz a monitorização de tráfego nas ligações de rede, a monitorização de CPU, memória ou serviços informáticos. O NETDISCO não tem controlo de acesso à interface Web (por exemplo, *login* e *password*) e não permite o envio de alertas de rede para o gestor, por correio electrónico ou SMS.

O ProCurve Manager (PCM), na versão livre, permite somente a monitorização e configuração de equipamento da HP. O PCM não é *Open Source,* pelo que não se enquadra nos critérios definidos.

Como se pode ver na Tabela 4 existem duas ferramentas que se destacam das outras por preencherem a maior parte dos requisitos estipulados, são elas o NAV e o OpenNMS.

O NAV foi criado numa universidade, para gerir redes em ambientes universitários. Há semelhança das ferramentas HYPERIQ-HQ e MRTG, o NAV permite monitorar e criar gráficos para muitos elementos de análise de uma rede, tais como carga de tráfego em todas as intersecções de rede, carga de CPU nos servidores, entre outros. Tal como o ZABBIX, o NAV permite informar o gestor da rede, por correio electrónico ou SMS, da ocorrência de vários eventos (tais como inacessibilidade de um *switch*, servidor ou serviço na rede). À semelhança do NETDISCO o NAV possui módulos de gestão que permitem, por exemplo, obter a localização física e actual de um qualquer computador, bem como os últimos sítios em que esteve ligado. No entanto o NAV implementa mecanismos de controlo de acessos à ferramenta de gestão e permite enviar alertas por correio electrónico e SMS, ao contrário do NETDISCO.

Como se pode ver na Tabela 4 o OpenNMS contém a maior parte das características do NAV. No entanto o OpenNMS não possui as seguintes funcionalidades: localização de equipamento de rede, com histórico dos movimentos na rede (pontos em que esteve ligado); ferramentas de apoio à manutenção e logística da rede; configuração de portas de

switches; limitações de acesso à ferramenta de gestão, de acordo com o tipo de utilizador (diferentes tipos de acessos de acordo com o tipo de gestor). No NAV as ferramentas que proporcionam as funcionalidades referidas são: o *Machine Tracker* [123], que permite localizar equipamento na rede, bem como os últimos pontos em que esteve ligado; o *Network Explorer* [128], que também permite localizar equipamento na rede, mas fornece a árvore de activos de rede até ao equipamento pesquisado; o *Maintenance* [124], muito útil no apoio às tarefas de manutenção; o *Arnold – the port blocker* [125], que permite bloquear portas de *switches*. O NAV é muito útil nas redes de grandes dimensões, permitindo dividir as tarefas de gestão por vários gestores. O *User Administration* [126] permite associar os utilizadores em grupos, atribuindo a cada grupo a lista de funcionalidades a que tem acesso. O *Alert Profiles* [127] permite definir quais são os tipos de alertas a que cada grupo de utilizadores tem acesso. Por exemplo, o grupo A pode ter acesso a todo o tipo de alertas, pelo contrário o grupo B somente tem acesso aos alertas relacionados com os *routers*, com os serviços de rede e com a tangibilidade das impressoras na rede.

De acordo com os argumentos utilizados, o NAV é a ferramenta de eleição como base de desenvolvimento para a ferramenta de administração da rede do DEG.

3.3. *NETWORK ADMINISTRATION VISUALIZED* - NAV

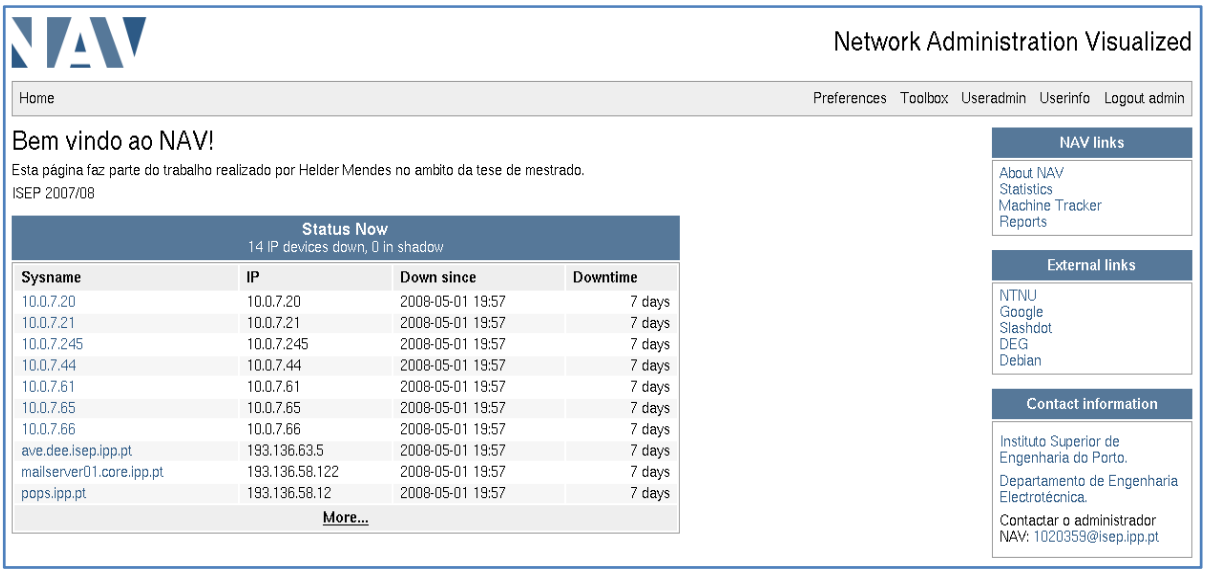

Figura 15: Aspecto da ferramenta NAV

3.3.1. INSTALAÇÃO E CONFIGURAÇÃO DO NAV

Os requisitos de instalação e configuração do NAV são apresentados no Anexo C.

Aceder pela primeira vez à interface Web do NAV

Quando se acede pela primeira vez à interface Web do NAV o servidor Apache disponibiliza a página para utilizadores não autorizados, com acesso restrito. A primeira vez que se faz login na página utiliza-se `username: admin e password: admin.` É extremamente recomendado a alteração imediata da *password* de acesso. O utilizador admin é membro do grupo "NAV Administrators", portanto tem acesso a todas as funções do NAV a partir da interface Web.

Gerir contas de utilizadores e privilégios

Para esta tarefa utiliza-se o painel de gestão de utilizadores *Useradmin*.

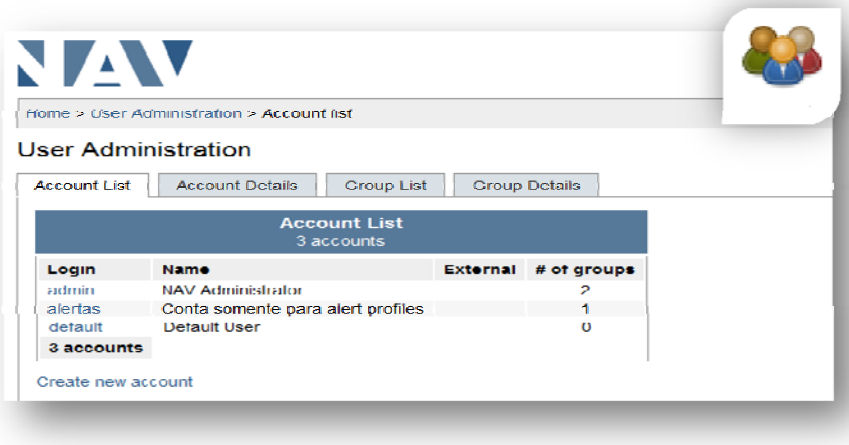

Figura 16: A ferramenta de gestão de utilizadores

Tipicamente esta tarefa permite:

- Criar utilizadores. Se for necessário uma lista grande de utilizadores pode-se utilizar *Lightweight Directory Access Protocol* (LDAP) [135];
- Criar, alterar ou eliminar novos grupos de utilizadores;
- Os utilizadores são organizados em grupos, para cada grupo são definidos os privilégios pretendidos. Por exemplo, é permitido definir quais são as ferramentas do NAV a que o utilizador pode aceder, se o utilizador recebe alertas de sistema por correio electrónico ou SMS, entre outros.

Esta ferramenta em paralelo com a ferramenta `Alert Profiles` dota o NAV de um carácter muito dinâmico. No âmbito da gestão de uma rede departamental é normal existirem vários funcionários responsáveis pelas tarefas de resolução de problemas, com esta ferramenta pode-se delegar funções a diferentes funcionários e assim cada uma recebe os alertas relativamente às funções que lhe estão associadas.

Preparar o equipamento de rede para gestão

O equipamento de rede terá de responder a pedidos de leitura efectuados por SNMP, e se a ferramenta Arnold for utilizada para bloquear portas de *switches* também é necessária permissão de escrita por SNMP. Em qualquer dos casos é importante que somente o servidor NAV (e outros servidores de gestão, se existirem) sejam capazes de aceder ao equipamento com SNMP.

O NAV utiliza o DNS para identificar os nomes do equipamento, se o equipamento não se encontrar registado no DNS é utilizado o IP como nome. É importante, mas não obrigatório, registar o equipamento no DNS antes de ser introduzido na base de dados do NAV.

O NAV organiza o equipamento em divisões/compartimentos (*rooms*). Cada divisão contém o equipamento de rede e os *racks* onde estes de encontram. As divisões são agrupadas geograficamente em áreas denominadas localizações (*locations*). As localizações são agrupadas em organizações (*organizations*) e estas podem ser hierarquizadas. Pode-se fazer uma analogia com o IPP. O IPP (*parent organization*) ramifica no ISEP (*organization*), no ISCAP (*organization*), no ESMAE (*organization*), no ESTSP (*organization*), etc. O ISEP tem diversos departamentos (*locations*), em cada departamento existem diversas divisões (*rooms*), que contêm equipamento de rede e *racks*.

O NAV disponibiliza linhas de orientação [31] para a configuração de *routers*, não são obrigatórias mas facilitam o processo de obtenção de dados dos *routers*. O engenheiro gestor é aconselhado a configurar as portas dos *switches* com nomes, que o NAV recolhe e disponibiliza através de diversas ferramentas.

Registar equipamento na base de dados – Ferramenta Edit Database

O NAV não faz descoberta automática do equipamento de rede, é necessário introduzir informação chave na base de dados recorrendo à ferramenta Edit Database.

Nota histórica: na versão NAV 2 o processo de registar equipamento na base de dados era realizado em ficheiros de texto, na versão actual, NAV 3, todos os registos são realizados na interface Web do NAV utilizando a ferramenta Edit Database que actua directamente na base de dados, `NAVdb`. Esta ferramenta permite fazer *bulk import* através de ficheiro ou directamente na página, permitindo efectuar de uma só vez todos os registos que se pretende, sejam serviços, IP´s, tipos de equipamento, entre outros.

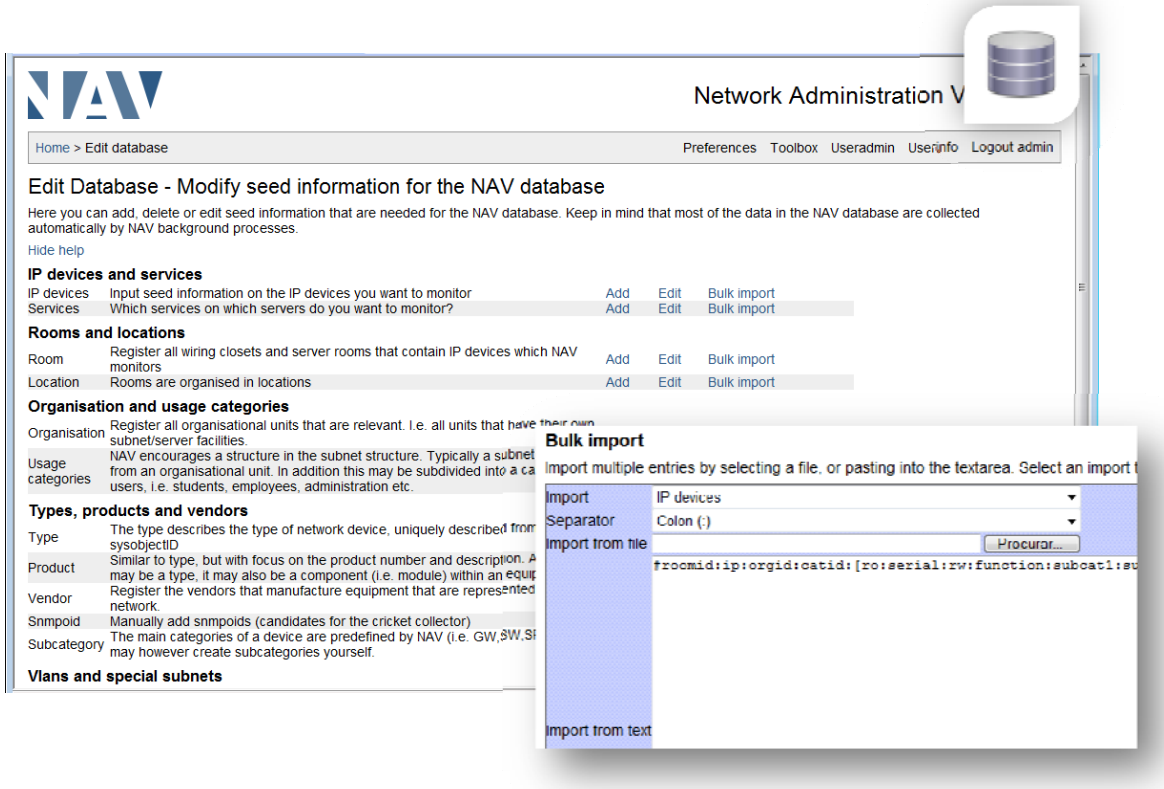

Figura 17: A ferramenta Edit Database

Requisitos mínimos: para registar equipamento na base de dados devem existir no mínimo uma organização (*organization*), uma localização (*location*) e uma divisão (*room*). Para informações pormenorizadas consultar [87].

Verificar se a rede é descoberta – Ferramenta Report

Depois de registado o equipamento na base de dados, os processos do NAV começam a recolher informação via SNMP: o processo `getDiviceData` [87] faz o reconhecimento de novos equipamentos registados na base de dados a cada 5 minutos e classifica-os com base nos OID´s, posteriormente, procede à recolha de informação sobre as portas de *switches*/*routers*. Isto pode ser visualizado na ferramenta reports [87] ilustrada na Figura 18.

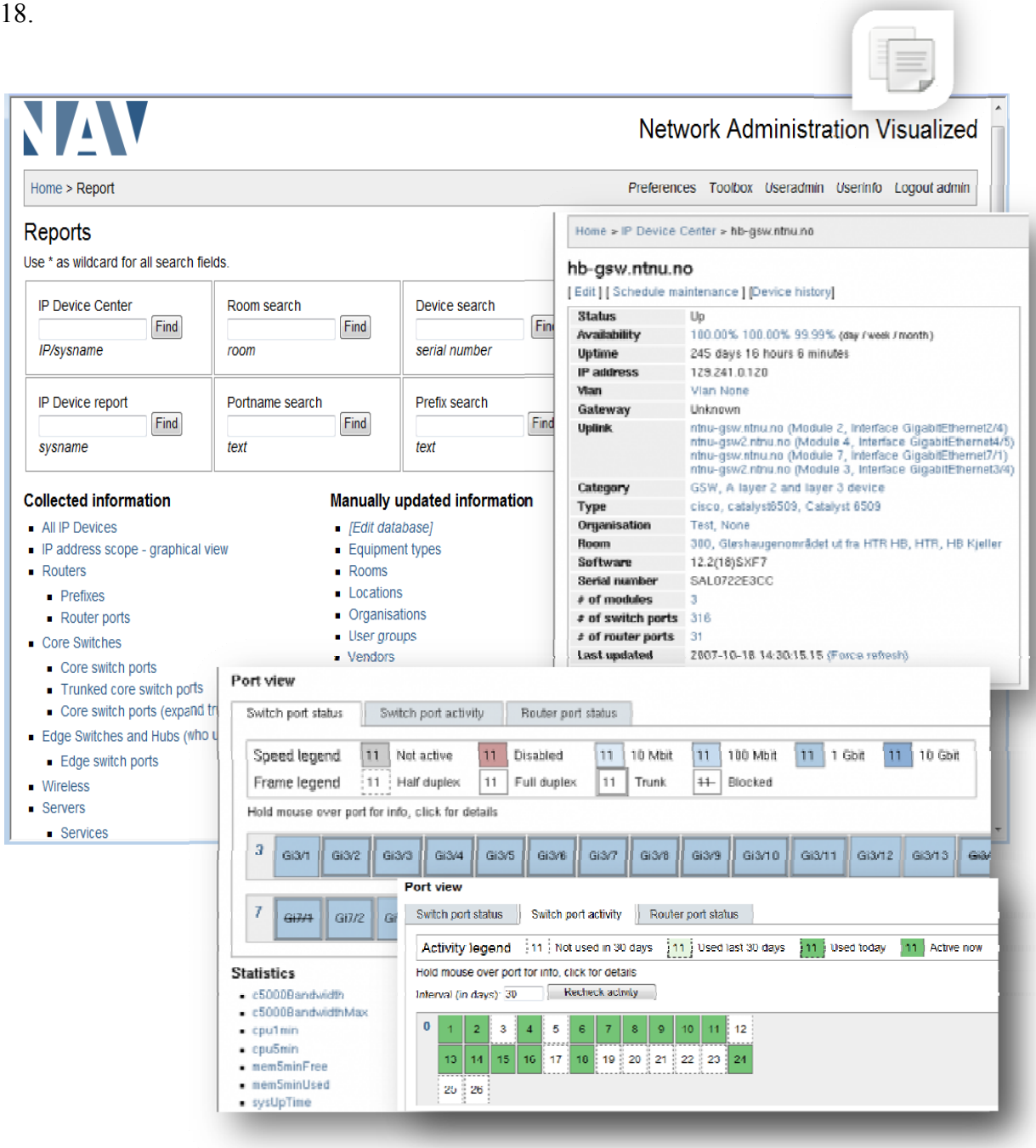

Figura 18: A ferramenta Reports

Depois de ser recolhida a informação para cada componente individual, o processo `mactrace` [87] executa a cada 15 minutos para recolher informação das portas dos activos de rede. Esta informação será utilizada para construir a topologia de rede todas as horas.

3.3.2. CONFIGURAÇÃO DOS ALERTAS – FERRAMENTA ALERT PROFILES

Uma das principais valências que distingue o NAV de todas as ferramentas expostas no estado da arte é a ferramenta Alert Profiles.

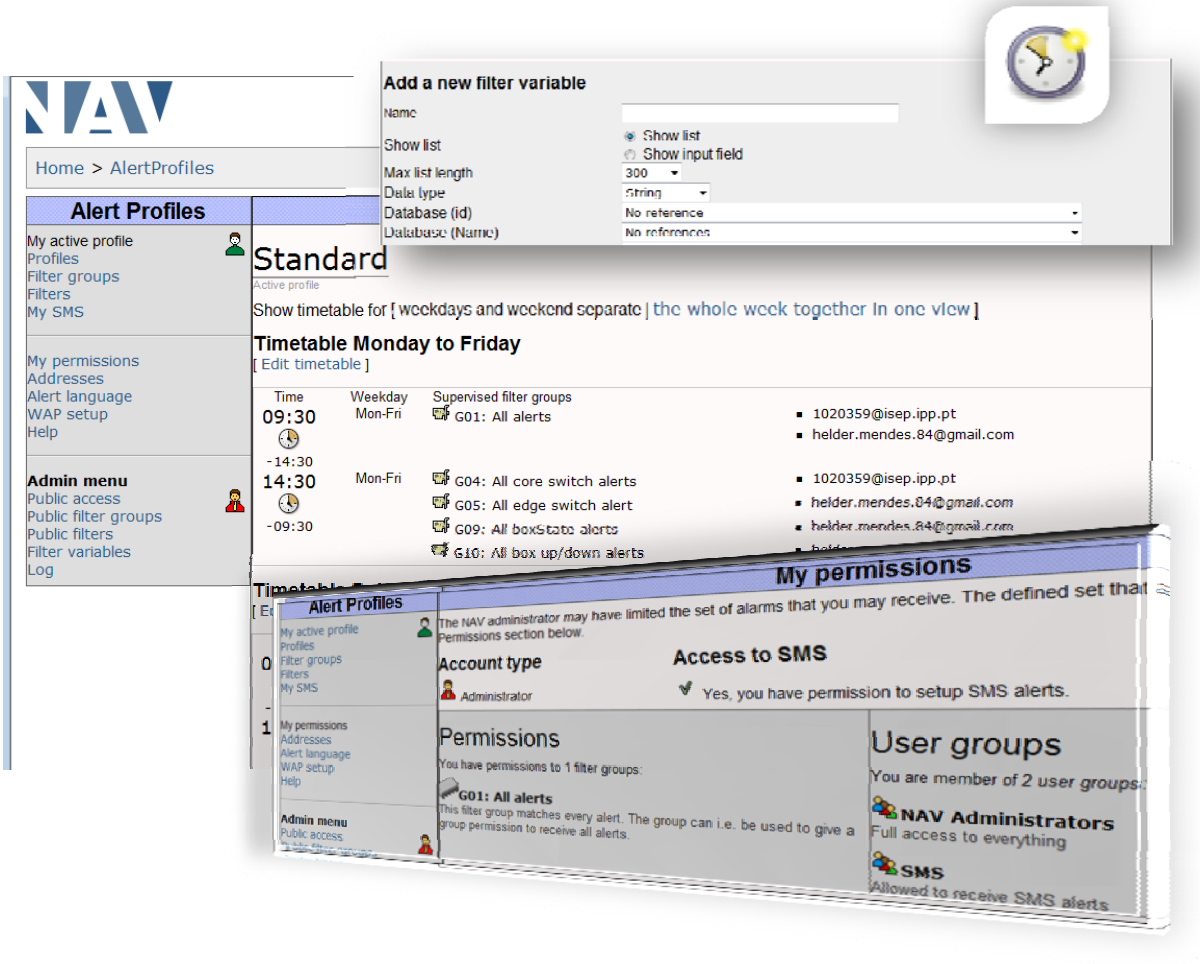

Figura 19: A ferramenta Alert Profiles

O Alert Profiles é uma ferramenta multifacetada que permite aos gestores da rede receber, por correio electrónico ou SMS, vários alertas relacionados com a rede. A Figura 19 fornece uma prespectiva geral das funcionlidades desta ferramenta.

Como principio de funcionamento nesta ferramenta os utilizadores estão associados a um ou mais grupos de utilizadores com funcionalidades de administraçaõ especificas (e têm

um *login* e *password* de acesso). Através do grupo o utilizador dispõe de acesso a determinados conjuntos de alertas relacionados com equipamentos ou serviços. O utilizador pode definir vários perfis de alertas, mas apenas um estará activo de cada vez. Um perfil é análogo a uma base de tempo com diversos intervalos de tempo, para cada intervalo de tempo o utilizador pode definir os alertas que pretende receber. A Tabela 5 contém a lista de filtros predefinidos no NAV.

Tabela 5: Lista de grupos de filtros predefinidos no NAV

G01: All alerts	F01: All alerts
G02: All router alerts	F02: All router alerts
G03: All switch (core and edge) alerts	F03: All switch (core and edge) alerts
G04: All core switch alerts	F04: All core switch alerts
G05: All edge switch alert	F05: All edge switch alerts
G06: All alerts from wireless boxes	F06: All alerts from wireless boxes
G07: All alerts from servers	F07: All alerts from servers
G08: All alerts from OTHER equipment	F08: All alerts from OTHER equipment
G09: All boxState alerts	F09: All boxState alerts
G10: All box up/down alerts	F10: All box up/down alerts
G11: All box sunny/shadow alerts	F11: All box sunny/shadow alerts
G12: All boxState early warnings	F12: All boxState early warnings
G13: All module outage alerts	F13: All module outage alerts
G14: All services alerts	F14: All services alerts
G15: All threshold alerts	F15: All threshold alerts
G16: All device change alerts	F16: All device change alerts
G17: All alerts with severity >= Warning	F17: All alerts with severity >= Warning
G18: All alerts with severity >= Errors	F18: All alerts with severity >= Errors
G19: All alerts with severity >= Critical	F19: All alerts with severity >= Critical
G20: All alerts with severity = Emergency	F20: All alerts with severity = Emergency

3.3.3. ESTATÍSTICAS DE TRÁFEGO – FERRAMENTA STATISTICS

A ferramenta Statistics do NAV mantém estatísticas de tráfego para todas as intersecções na rede, incluindo portas de *routers* e *switches*, octetos I/O, contadores de erros, bem como servidores, incluindo utilização de CPU, disco, memória, tráfego nas placas de rede dos servidores, entre outras. Todas as estatísticas de tráfego são guardadas através de RRDTool [109], o Cricket [110] obtém os dados estatísticos com base na árvore gerada automaticamente pelo NAV. Por omissão a reconfiguração do Cricket com a árvore gerada pelo NAV é feita todos os dias às 00H05, como consequência disto é necessário esperar pelo dia seguinte para ver as estatísticas dos novos equipamentos, ou forçar a reconfiguração do Cricket com a execução do ficheiro `makecricketconfig.pl`. A Figura 20 fornece uma perspectiva geral da ferramenta Statistics.

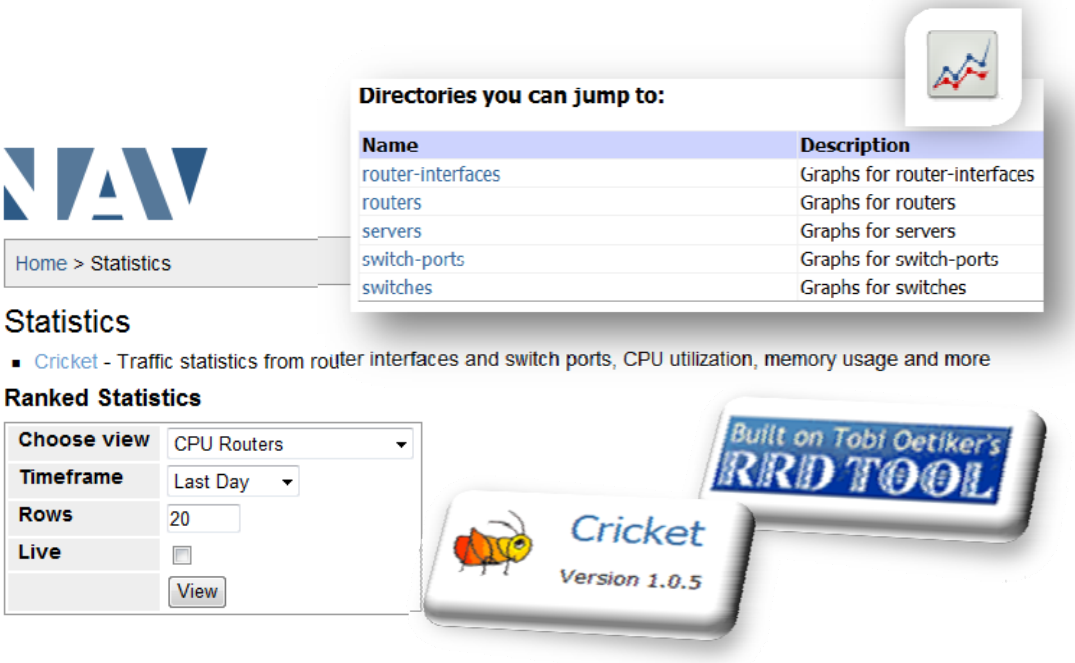

Figura 20: A ferramenta Statistics

3.3.4. ESTADO DA REDE – FERRAMENTA STATUS

A ferramenta Status fornece uma perspectiva global da actividade dos alarmes, incluindo os componentes e serviço da rede que estão ligados e a funcionar.

O NAV dispõe de quatro processo a correr em *background* que monitorizam o estado operacional da rede, estes lançam eventos com base na informação que recolhem e que fica disponível na ferramenta Status. Os processos são: 1) o *status monitor* (pping [87]), que verifica se os equipamentos estão alcançáveis na rede; 2) o *service monitor* (servicemon

[87]) que monitoriza os serviços; 3) o *module monitor* (modulemon [87]) que monitoriza os módulos dentro dos *switches*; 4) o *threshold monitor* (thresholdmon [87]) que monitoriza os dados RRD e envia alertas se os valores limites definidos forem excedidos. Existe ainda o processo snmptrapd que apenas recolhe *traps* enviadas pelos equipamentos de rede, encaminhando-as para a *event queue*. A Figura 20 fornece uma perspectiva geral do funcionamento desta ferramenta.

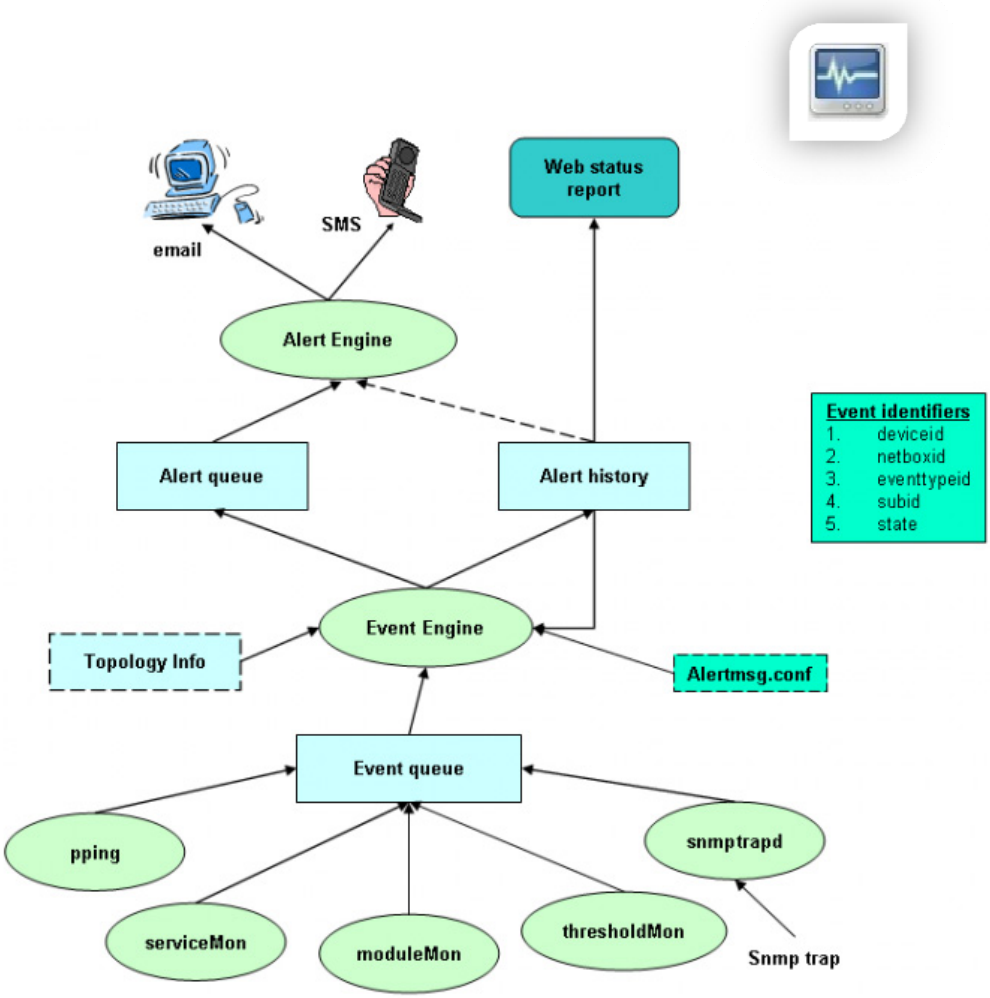

Figura 21: Diagrama da ferramenta Status

3.3.5. EXPLORAR A REDE – FERRAMENTA NETWORK EXPLORER

A ferramenta *Network Explorer* apresenta a rede estruturada em árvore.

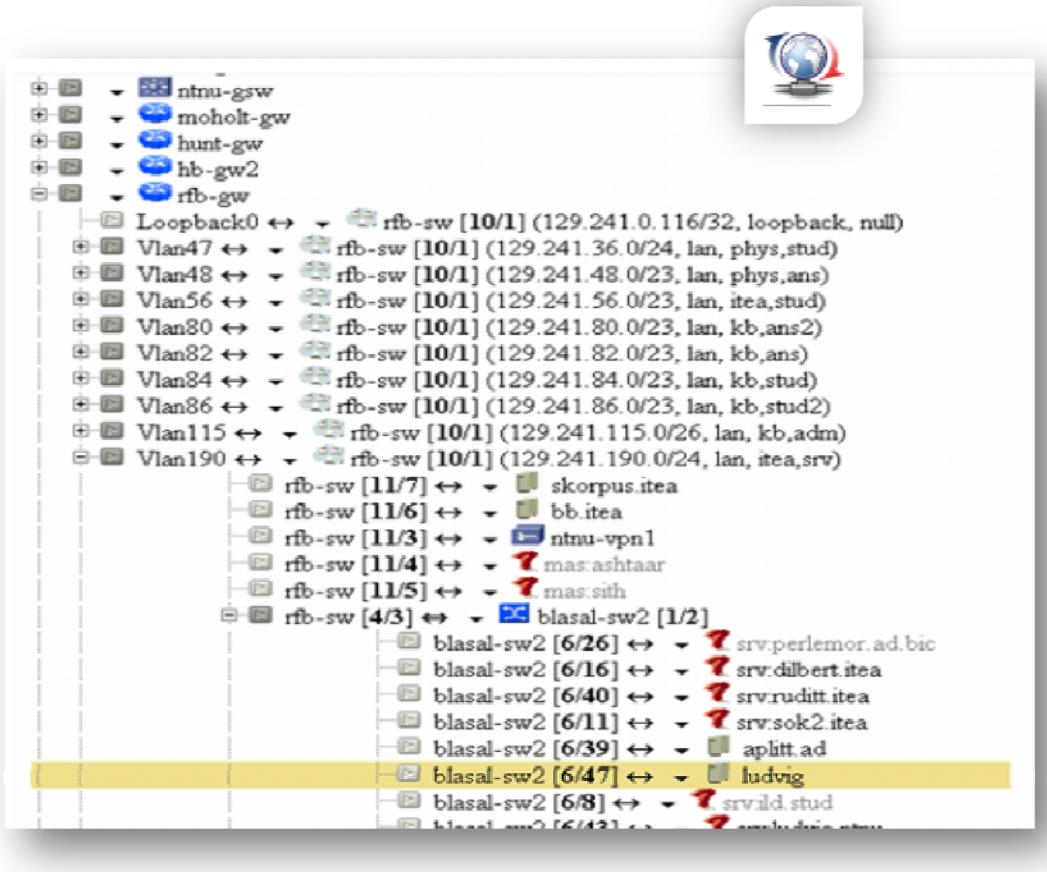

Figura 22: A ferramenta *Network Explorer*

A vista inicial lista todos os *routers* (GW e GSW [87]). Tal como se pode ver na Figura 22 o utilizador pode explorar um router clicando simplesmente o símbolo (+), com isto fica visíveis as informações das VLAN's do respectivo *router*. Clicando novamente em (+) revela-se a informação sobre as VLAN, e por seguinte. Se for conhecida a topologia para um dado interface do *router*, nomeadamente qual o interface do *router* que liga a um *switch*, é possível expandir a árvore de segundo nível. No limite, clicando no interface de um *switch* é fornecido o nome do computador, ou IP e MAC da máquina por de trás da interface.

Esta ferramenta também permite realizar pesquisas por: 1) nome do equipamento; 2) endereço IP; 3) endereço MAC; 4) nome de porta; 5) VLAN; 6) divisão (*room*); 7) Software; 8) número de série; e deste modo disponibiliza automaticamente a árvore de rede até ao item pretendido.

3.3.6. PESQUISA DE MÁQUINAS – FERRAMENTA MACHINE TRACKER

O NAV guarda um histórico de todas as máquinas (endereço IP / endereço MAC) na rede de dados, podendo-se realizar pesquisas por IP, gama de IP´s, MAC ou Switch, como se pode ver na Figura 23.

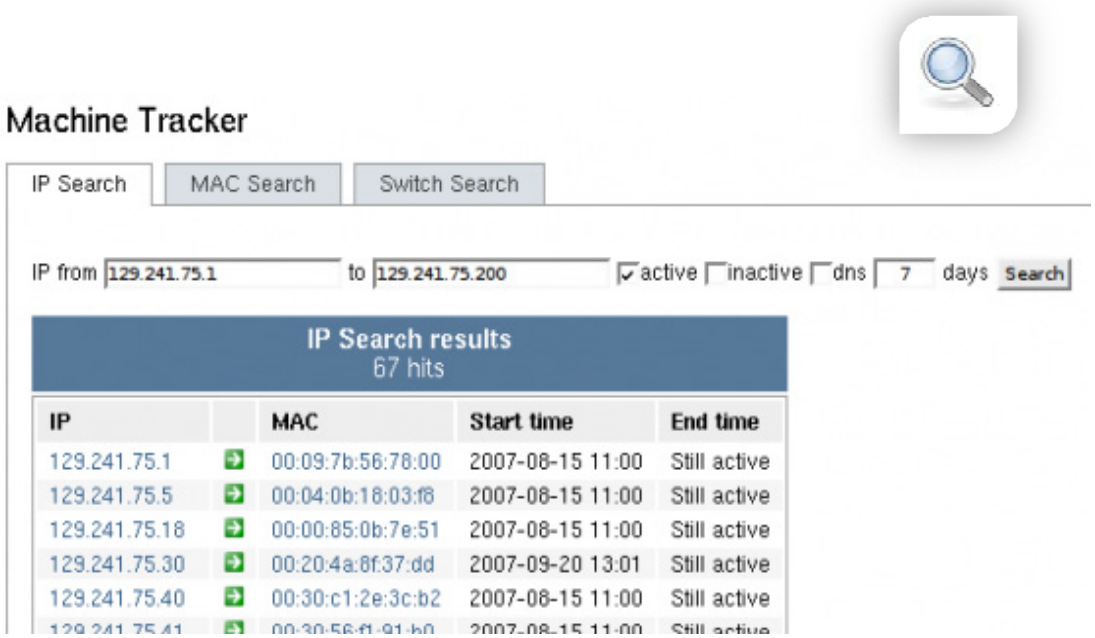

Figura 23: Ferramenta Machine Tracker

3.4. RESUMO

Neste capítulo começou-se por apresentar o estado da arte das ferramentas de gestão e monitorização de redes com código de fonte aberta. No amplo leque de ferramentas estudadas, existem algumas vocacionadas para tarefas peculiares de gestão como: a detecção de intrusos na rede (por exemplo, o SNORT); a avaliação de performance (por exemplo, HYPERIC HQ); ou a monitorização de elementos críticos (por exemplo, o MRTG). Existem também ferramentas mais completas que abrangem diversas áreas funcionais da gestão de redes (FCAPS), como por exemplo o NAV, o OpenNMS e o NETDISCO.

Com base nos requisitos de gestão e monitorização da rede do DEG (definidos na secção 2.3), definiram-se os critérios de selecção e, com base neles, efectuou-se uma comparação entre as referidas ferramentas. Dos critérios definidos podem salientar-se a interface Web de gestão, a monitorização de equipamentos e serviços de rede, com possibilidade de lançar alertas de acordo com o estado de operabilidade destes, a localização de

equipamento na rede com o respectivo histórico de movimentos na rede (pontos da rede onde esteve ligado), etc.

De todas as ferramentas apresentadas existem duas que se destacam por darem resposta à maior parte dos requisitos definidos, o OpenNMS e o NAV. A escolha recaiu sobre o NAV por este possuir em leque mais completo de funcionalidades, nomeadamente na localização física e lógica de equipamento de rede (com histórico de movimentos na rede), nas ferramentas de apoio à manutenção e logística, na interface Web que permite a delegação de tarefas a diferentes gestores, na gestão de alarmes de rede de acordo com o gestor de rede, entre outras.

4. FERRAMENTAS DE GESTÃO DE IMAGENS DE SOFTWARE

Devido à elevada utilização das salas de informática do DEG, é frequentemente necessário repor imagens de disco nos computadores, não só devido a vírus, falhas no software, cota de disco no limite, etc., mas também pela necessidade de instalar um novo software, *upgrades*, actualizações de segurança, entre outros. Por outro lado, é importante criar regularmente cópias de segurança do servidor da página de internet do DEG.

4.1. FERRAMENTAS DE GESTÃO DE IMAGENS DE SOFTWARE

Em seguida serão apresentadas várias ferramentas de manuseamento de imagens de software, disponíveis para utilização na internet. A caracterização mais ou menos pormenorizada de cada uma das ferramentas é o reflexo da qualidade da respectiva documentação.

4.1.1. CLONEZILLA

O Clonezilla [34] foi criado por Steven Shian, K. L. Huang, H. T. Wang, Ceasar Sun, Jazz Wang, e Thomas Tsai. É uma ferramenta de clonagem de discos de software, desenvolvido com base em diversas ferramentas: o *Diskless Remote Boot on Linux* (DRBL) [79], o Partition Image [39], o partclone [80], o dd [81], o ntfsclone [82] e o udpcast [83]. O Clonezilla permite fazer cópia de segurança e recuperação de discos rígidos em máquinas com os sistemas operativos Windows e GNU/Linux, e está disponível em duas versões: Clonezilla Live e Clonezilla Server.

Sistemas operativos e linguagens de programação

A versão servidor do Clonezilla opera em Linux.

Princípios de funcionamento e principais funcionalidades:

A versão Live, disponível para arranque a partir de CD, PXE ou dispositivo de armazenamento USB, permite fazer a cópia de segurança e recuperação de uma só máquina, enquanto a versão *Server* permite a instalação de imagens em várias máquinas simultaneamente. O Clonezilla suporta os sistemas de ficheiros: ext2, ext3, ReiserFS [116], XFS [115] e *Journaled File System* (JFS), usualmente utilizados em GNU/Linux. Nos sistemas Windows os sistemas de ficheiros suportados são FAT e NTFS. Para os sistemas de ficheiros suportados, somente os blocos utilizados das partições são guardados, aumentando a eficiência do processo. Para os sistemas de ficheiros que não são suportados, é realizada uma cópia sector a sector, através da ferramenta "dd". O Clonezilla suportada o *Logical Volume Management* 2 (LVM2) [112] em GNU/Linux (a versão 1 do LVM não é suportada). A instalação em massa de imagens de software, por *unicast* [73] ou *multicast* [73], é suportado na versão servidor do Clonezilla, com auxílio aos protocolos PXE e *Wake-on-LAN*. Utilizando outra ferramenta livre, o drbl-winroll [85], é possível alterar automaticamente o nome do computador cliente, o grupo de trabalho e o identificador de segurança (SID) de máquinas Microsoft Windows clonadas. A interface para o utilizador é do tipo *wizard* com acesso local. O Clonezilla está bem documentado e disponível pela licença GNU – *General Public License*.

4.1.2. GHOST FOR UNIX – G4U

O Ghost for Unix (G4U) [36] é uma ferramenta para clonagem de discos rígidos de computadores, baseado em NetBSD e pode ser carregado a partir de disquete ou CD-ROM.

Sistemas operativos e linguagens de programação

Este utilitário opera em Unix.

Princípios de funcionamento e principais funcionalidades

O G4U realiza a clonagem a partir de informação RAW dos discos ou de suas partições, ou seja, copia sector a sector todo o conteúdo do disco, podendo estas ser comprimidas. O G4U funciona de dois modos distintos, por um lado permite fazer o upload de imagens comprimidas do disco local para um servidor FTP, e o inverso, isto é restaurar a imagem via FTP, descomprimi-la e escreve-la no disco local. Suporta o protocolo DHCP.

4.1.3. LINBOX RESCUE SERVER

O Linbox Rescue Server (LRS) [37] é um *asset management software* [41] que inclui cinco módulos: recuperação de ficheiros, inventário, delegação de software e controlo remoto. Está disponível em duas versões, uma proprietária e outra licenciada pela GPL.

Sistemas operativos e linguagens de programação:

O LRS opera em Linux.

Princípios de funcionamento e principais funcionalidades

A recuperação de ficheiros é baseada no software BackupPC; o inventário de rede é baseado nos agentes ocs-inventory [129], e correm no kernel de arranque por PXE. O agente ocs-inventory faz o inventário ao nível do equipamento de rede, recolhe a informação de hardware e software e retorna-a ao sistema de gestão. O Linbox permite o controlo remoto de equipamento utilizando o software TightVNC [113].

As versões LRS GPL e a proprietária do Linbox disponibilizam interface Web, suporte para discos com interface *Serial Advanced Technology Attachment* **(**SATA), interface

Integrated Drive Electronics (IDE) ou *Small Computer System Interface* **(**SCSI). O Linbox permitem escolher as partições do disco das quais vai ser feita uma cópia de segurança, e controlar as opções de arranque do computador cliente através da interface Web.

A versão GPL do LRS suporta cópias de segurança e restituição de partições em sistemas de ficheiros ext2, ext3, XFS [115], LVM1 e LVM2, enquanto a versão proprietária suporta também os sistemas de ficheiros FAT16, FAT32 e NTFS. Ambas as versões do LRS suportam a instalação de imagens de software em vários computadores em simultâneo, no entanto a versão proprietária suporta o envio das imagens por *unicats* e *multicast*, ao passo que a versão GPL apenas suporta *unicast*. A versão LRS comercial permite fazer configurações pós-instalação e cópia de segurança e restauro através de placas de rede do tipo *Personal Computer Memory Card International Association* **(**PCMCIA), ao contrário da versão GPL.

4.1.4. PARTIMAGE

O Partimage [39], criado por Francois Dupoux e Franck Ladurelle, é um utilitário em Linux que permite copiar partições de discos para o formato de imagem, suportando grande parte dos sistemas de ficheiros Linux e Windows. O Partimage é distribuído em duas versões: software cliente - servidor e em Live CD.

Sistemas operativos, arquitecturas e linguagens de programação

Este utilitário opera nas arquitecturas Intel i386 e PowerPC.

Princípios de funcionamento e principais funcionalidades:

A versão cliente do Partimage tem interface por linha de comandos ou interface gráfica do tipo *wizard*, e permite criar cópias de partições de disco local, ou remotamente através da rede. Deste modo, o cliente Partimage pode interagir com o servidor Partimage ou com um sistema de ficheiros de rede, tais como o *Network File System* (NFS) ou SAMBA [114].

A versão Live CD inclui não só o Partimage, mas outras ferramentas de apoio à recuperação e reparação de sistemas danificados: o GNU Parted; o GParted; ferramentas para os sistemas de ficheiros: e2fsprogs, reiserfsprogs, reiser4progs, xfsprogs, jfsutils,

ntfsprogs, dosfstools; o Ntfs3g; o sfdisk; o Test-disk; e algumas ferramentas de rede: Samba, NFS, ping, nslookup;

Os sistemas de ficheiros suportados pelo Partimage são: Ex2, Ext3, Raiser, FAT16, FAT32, HPFS (*High Performance File System*), JFS, XFS, *Unix FileSystem* (UFS) [130], *Hierarchical FileSystem* (HFS) [131] e NTFS. Os ficheiros de imagem podem ser compactados com os programas gzip e bzip2, para poupar estaco em disco, por outro lado as imagens de software podem ser divididas em vários ficheiros e assim possibilitar a gravação em CD ou DVD. As partições também podem ser copiadas através da rede de dados, utilizando SAMBA ou NFS. O software Partimage está bem documentado e licenciado pela licença GNU – *General Public License*.

4.1.5. FREE OPEN SOURCE GHOST

O FOG [35], criado por Jian Zhang e Chuck Syperski, é um software de clonagem de discos baseado em Linux, que reúne as funcionalidades de várias ferramentas *Open Source*: DHCP, NFS, PXE, TFTP, DD, HTTPD, PARTIMAGE e *Wake-up-On-LAN* (WOL). É uma ferramenta de software baseada em PHP e MySQL, com interface Web. O FOG foi projectado particularmente para a gestão de imagens de software Windows 98, 2000, XP e Vista, mas também permite a cópia de segurança e instalação de imagens de software de qualquer sistema operativo que utilize os sistemas de ficheiros suportados. Não são utilizados CD's (*Compact Disk*) tudo é feito via TFTP e PXE.

Sistemas operativos e linguagens de programação:

O FOG está escrito predominantemente em PHP e SQL, com pequenas porções de código JAVA Script, AJAX e CSS. O FOG opera nas distribuições Ubuntu e Fedora.

Princípios de funcionamento e principais funcionalidades

No FOG as funcionalidades estão disponíveis a partir da sua interface Web, a partir da qual se pode registar, editar e eliminar equipamento de rede, e criar tarefas de execução. No FOG utiliza-se o conceito de tarefas (*tasks*) para designar as operações que se pretende executar sobre os elementos de rede (por exemplo instalação de imagens de software). A maior parte das tarefas feitas com o FOG não implica visitar o PC cliente. Por exemplo,

para instalar uma imagem de software em computadores que suportem os protocolos PXE e *Wake-up-On-Lan* (WOL), basta criar a tarefa. Depois de criada a tarefa o WOL liga os computadores remotamente, o servidor DHCP atribui o IP, o mini sistema operativo (constituído por kernel e initrd) é carregado nos computadores clientes. O cliente informa o servidor que o processo está a decorrer. O Partimage (através do NFS) carrega a imagem de software para o computador cliente. Posteriormente, aquando do término da tarefa, o mini sistema operativo informa o servidor FOG de que a tarefa terminou. O computador cliente reinicia ou desliga, de acordo com os parâmetros escolhidos para tarefa. Se o serviço FOG estive instalado, é alterado o nome e grupo de trabalho do computador e está pronto a ser utilizado.

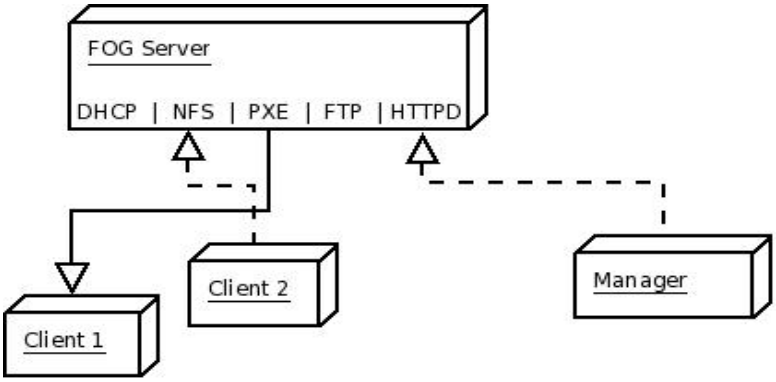

Figura 24: Interacção entre os Clientes e o servidor FOG [35]

Uma das principais vantagens do FOG está na interface Web de gestão, pois permite o controlo de acessos por *login* e *password* e múltiplos acessos de gestão em simultâneo. Pode-se aceder à interface Web a partir de qualquer computador, PDA, iPOD touch, ou iPhone. O FOG utiliza a base de dados MySQL. Os sistemas de ficheiros suportados são**:** NTFS, XFS, JFS, HPFS (*High Performance File System*), FAT16, FAT32, Reiser3, EXT2 e EXT3. Também suporta a cópia de partições no formato RAW. O FOG permite organizar os elementos de rede em grupos, o que facilita a criação de tarefas para várias máquinas em simultâneo.

O FOG não faz somente a gestão, instalação e cópia de segurança de imagens de software, permite também realizar as seguintes tarefas: detecção, eliminação e quarentena de vírus, testes de memória, *disk wiping*, e recuperação de ficheiros. A ferramenta *disk wiping* permite eliminar o MBR do disco rígido, ou mesmo fazer a formatação a baixo nível (colocar o disco rígido todo a zeros ou todo a uns). O FOG pode ser distribuído por vários

servidores, por exemplo: os serviços NFS, apache, PXE, e DHCP podem correr num só servidor ou em diferentes servidores para optimizar a performance. O FOG é modular e permite o desenvolvimento de funcionalidades adicionais.

O FOG está bem documento e licenciado pela licença GNU – *General Public License*.

4.1.6. MONDO RESCUE

O Mondo Rescue [38] é um Live CD que permite realizar cópias de segurança e recuperação de discos de computadores. Foi criado por Hugo Rabson, mas actualmente é desenvolvido por Bruno Cornec e Andree Leidenfrost.

O Mondo Rescue é utilizado por algumas empresas de renome internacional, tais como Lockheed-Martin, Nortel Networks, Siemens, HP, IBM, NASA's JPL, o US Dept. of Agriculture, dezenas de pequenas empresas e centenas de utilizadores.

Sistemas operativos e linguagens de programação suportados:

Este software opera em Linux (i386, x86_64, ia64), RedHat, RHEL, SuSE, SLES, Mandriva, Debian, Gentoo e FreeBSD (i386). O Mondo Rescue está escrito na linguagem de programação C e Unix Shell.

Princípio de funcionamento e principais funcionalidades:

O Mondo Rescue permite fazer cópias de segurança de discos rígidos com o sistema operativo GNU/Linux, para CD-R, CD-RW, DVD-R[W], DVD+R[W], NFS ou para outra partição do mesmo disco rígido, e vice-versa. Suporta LVM 1 e 2, RAID, ext2, ext3, JFS, XFS, ReiserFS, VFAT, UFS, NTFS. Suporta os sistemas *Redundant Array of Independent Drives* (RAID), bem como grande parte dos controladores RAID.

O Mondo Rescue suporta alterações na geometria do disco rígido, incluindo a migração de sistemas não RAID para RAID. O Mondo Rescue é licenciado pela licença GPL v2 (GNU *Public License Version* 2).

4.1.7. TRINITY RESCUE KIT - TRK

O TRK [40], criado por Tom Kerremans, é um utilitário especializado na clonagem e recuperação de discos rígidos com sistemas operativos Windows e Linux. Tendo por base a distribuição Mandriva, se é distribuído em Linux Live CD e construída a partir das ferramentas: partimage, winpass, clonexp, ntfsundeleteall, samba, bridge, setip e setproxy.

Sistemas operativos e linguagens de programação

Esta ferramenta opera em Linux.

Princípios de funcionamento e principais funcionalidades

O TRK é um utilitário de recuperação e clonagem de partições danificadas de discos rígidos. É possível utilizar o TRK de três modos diferentes: arrancando através de CD, de dispositivos de armazenamento *Universal Serial Bus* (USB) ou PXE.

O TRK é operado por linha de comandos excepto em algumas ferramentas que o integram: o qtparted [117], o links, o partition image [118] e o midnight commander [119].

O TRK permite restaurar a palavra passe de computadores com o sistema operativo Windows e Linux e contém módulos que permitem realizar a procura de vírus nos discos.

Com o TRK é possível fazer a clonagem e redimensionamento de partições no formato NTFS, suporta *Logical Volume Management* (LVM) e os sistemas de ficheiros suportados são: Ext2, Ext3, Raiser, FAT16, FAT32, HPFS, JFS, XFS, UFS, HFS e NTFS.

O TRK está disponível pela licença GNU – *General Public License*.

4.1.8. RESUMO DO ESTADO DA ARTE DAS FERRAMENTAS DE GESTÃO DE IMAGENS

Na Tabela 6 é apresentado um resumo das características das ferramentas de gestão de imagens de software apresentadas nas secções anteriores.

Tabela 6: Resumo do estado da arte das ferramentas de gestão de imagens de software

	Característica																					
Nome	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**
Clonezilla Live	■				■		■	■	■	■	■	■	■		■	■				■		■
Clonezilla Server	■				■		■	■	■	■	■	■		■						■		■
G4U		■			■	■	■						■	■						■		■
Linbox Rescue (versão GPL)			■		■			■						■						■		■
Partimage	■							■	■	■	■		■							■		■
FOG			■				■	■	■	■	■				■	■	■	■	■	■	■	■
Mondo Rescue	■						■	■	■	■	■		■							■		■
TRK	■				■		■	■	■	■	■					■				■	■	■

Legenda: Característica disponível ■

Características:

Tabela 7: Caracteristicas das ferramentas de gestão de imagens de software

Nº	CARACTERÍSTICA	Nº	CARACTERÍSTICA
1	Interface de utilização do tipo texto	**12**	Suporta a cópia de discos com *filesystem* HFS+
2	Interface por linha de comandos	**13**	Arquitectura sem servidor
3	Interface de utilização do tipo Web ou gráfica	**14**	Arquitectura do tipo cliente – servidor
4	Utilização por chamadas à linha de comandos, com os parâmetros predefinidos	**15**	Suporta o arranque remoto de computadores via *Wake-up-On-LAN*
5	Disponibiliza Live CD	**16**	Suporta o protocol PXE
6	Permite o arranque a partir de disquete	**17**	Permite a criação de tarefas para vários computadores de uma só vez
7	Suporta a cópia de discos no formato RAW	**18**	Serviço de contas de utilizadores, para controlo de acessos
8	Suporta a cópia de discos com *filesystem* Ext2 e Ext3	**19**	Suporta múltiplos acessos, de vários utilizadores, em simultâneo
9	Suporta a cópia de discos com *filesystem* SWAP	**20**	Suporta a instalação de imagens em vários computadores em simultâneo
10	Suporta a cópia de discos com *filesystem* FAT32	**21**	Suporta a detecção e eliminação de vírus em computadores remotos
11	Suporta a cópia de discos com *filesystem* NTFS	**22**	Licenciada pela *GNU – General Public License*

4.2. SELECÇÃO DA FERRAMENTA DE GESTÃO DE IMAGENS DE SOFTWARE

Nas secções anteriores foi apresentado um leque de ferramentas de gestão de imagens de software, que satisfazem a maior parte dos requisitos de gestão de imagens da rede do DEG. Em seguida serão apresentados os critérios de selecção da ferramenta de gestão, baseados na análise de requisitos de gestão de imagens de software da rede do DEG (secção 2.4), bem como a escolha da ferramenta.

4.2.1. CRITÉRIOS DE SELECÇÃO DA FERRAMENTA

Para seleccionar a ferramentas de gestão de imagens foram definidos 14 critérios de selecção, representados na Tabela 9 (a encarnado), e que reflectem as necessidades de gestão de imagens definidas na secção 2.4.

Critérios de selecção:

Tabela 8: Critérios de selecção das ferramentas de gestão de imagens de software

Nº	CARACTERÍSTICA	Nº	CARACTERÍSTICA
3	Interface de utilização do tipo Web ou gráfica	**16**	Suporta o protocol PXE
7	Suporta a cópia de discos no formato RAW	**17**	Permite a criação de tarefas para vários computadores de uma só vez
8	Suporta a cópia de discos com *filesystem* Ext2 e Ext3	**18**	Serviço de contas de utilizadores, para controlo de acessos
9	Suporta a cópia de discos com *filesystem* SWAP	**19**	Suporta múltiplos acessos, de vários utilizadores, em simultâneo
10	Suporta a cópia de discos com *filesystem* FAT32	**20**	Suporta a instalação de imagens em vários computadores em simultâneo
11	Suporta a cópia de discos com *filesystem* NTFS	**21**	Suporta a detecção e eliminação de vírus em computadores remotos
15	Suporta o arranque remoto de computadores via *Wake-up-On-LAN*	**22**	Licenciada pela *GNU – General Public License*

4.2.2. SELECÇÃO DA FERRAMENTA DE GESTÃO DE IMAGENS

A selecção da ferramenta de gestão de imagens de software passa por identificar as ferramentas que contêm a maior parte dos requisitos definidos na análise de requisitos de gestão de imagens (secção 2.4). Na Tabela 9 apresenta-se um resumo caracterizando as ferramentas de gestão de imagens de software exploradas nos pontos 4.1.1 a 4.1.7, interceptadas com os requisitos de gestão de imagens de software para a rede do DEG.

Tabela 9: Características das ferramentas de gestão de imagens *vs* critérios de selecção

	Característica																				
Nome	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**
Clonezilla Live	■		•		■		•	•	•	•	•	■	■		•	■	•	•	•	•	•	•
Clonezilla Server	■		•		■		•	•	•	•	•	■		■	•	•	•	•	•	•	•	•
G4U		■	•		■	■	•		•		•		■	■	•	•	•	•	•	•	•	•
Linbox Rescue (versão GPL)			■		■		•	■	•	•	•			■	•	•	•	•	•	•	•	•
Partimage	■		•				•	■	•	■	■		■		•	•	•	•	•	•	•	•
FOG			■				■	■	•	■	■				■	■	■	■	■	■	■	■
Mondo Rescue	■		•				•	•	•	•	•		■		•		•	•	•	•	•	•
TRK	■		•		■		•	•	•	•	•				•	■	•	•	•	•	■	■

Legenda:

Critério de selecção	• • • • • •
Característica disponível	■

É essencial que a ferramenta de eleição dê resposta aos requisitos de gestão de imagens de software da rede do DEG (secção 2.4), tal como o suporte para os sistemas de ficheiros Ext2, Ext3, SWAP, FAT32 e NTFS, a capacidade de instalação de imagens de software em vários computadores ao mesmo tempo, possibilitar a criação de tarefas para vários computadores em simultâneo, a interface Web de gestão, a detecção e eliminação de vírus em computadores remotos, etc. Como se pode constatar na Tabela 9 a única ferramenta que dá resposta a todos os requisitos de gestão de imagens é o *Free Open Source Ghost* (FOG) e, como tal, foi a ferramenta de eleição para a gestão de imagens de software.

4.3. *FREE OPEN SOURCE GHOST* - FOG

O FOG está na vanguarda das ferramentas de código aberto utilizadas na gestão de imagens de software de equipamento de rede. Os criadores do FOG tiveram o cuidado de desenvolver uma ferramenta de software virada para o utilizador. É possível executar várias tarefas de manutenção sem que para isso seja necessário estar fisicamente no local, o que permite optimizar o tempo. Em seguida serão apresentadas as características de alto nível do FOG e alguns dos protocolos mais importantes.

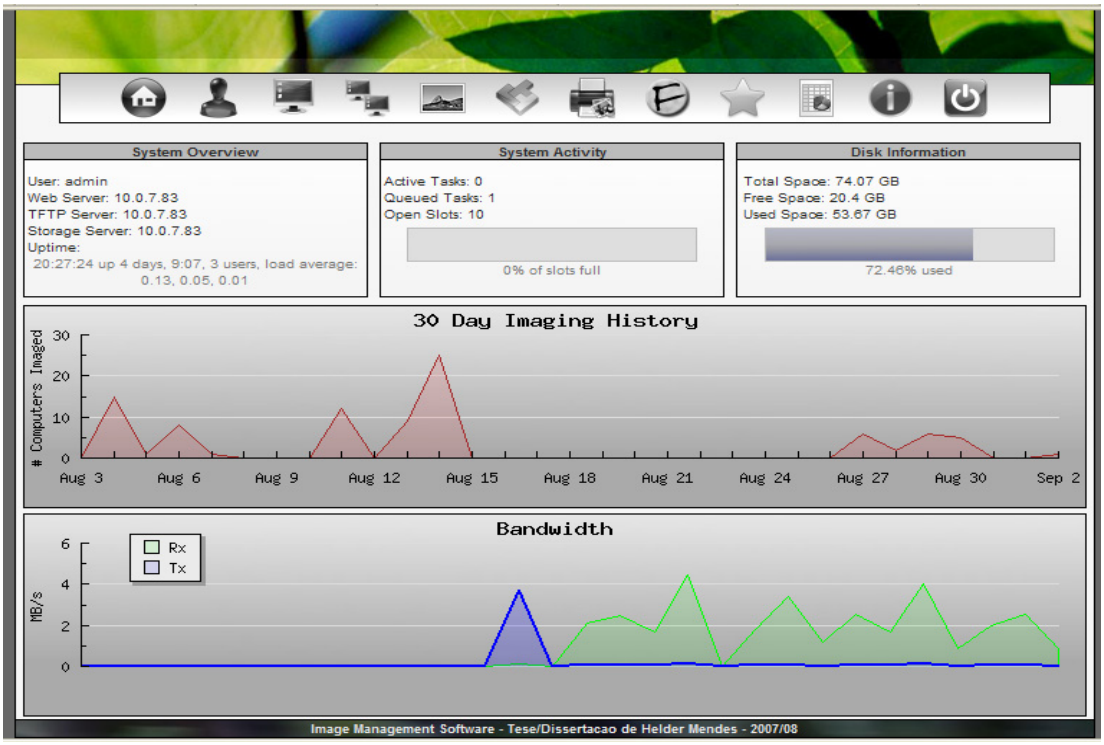

Figura 25: Página de entrada do FOG

4.3.1. PRINCIPAIS FUNCIONALIDADES DO FOG

Os módulos de alto nível do FOG, associados à barra de selecção ilustrada na Figura 26 são os seguintes:

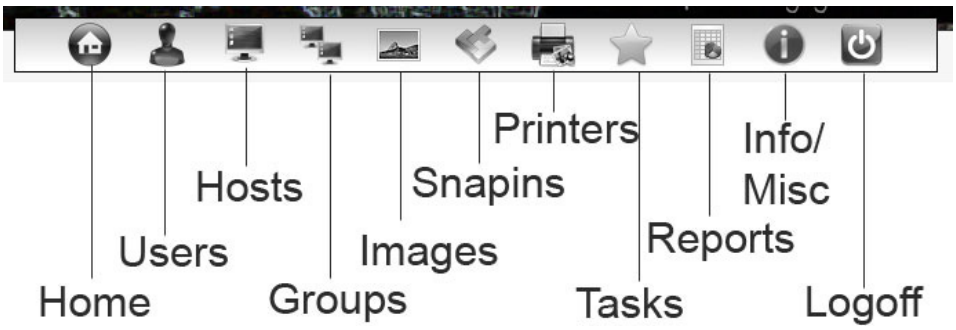

Figura 26: Funcionalidades de alto nível do FOG [35]

***Home*:** é a página de entrada. Como se pode constatar na Figura 25, a página de entrada permite visualizar, em tempo real, várias informações sobre os servidores e as tarefas que estão a decorrer. A título de exemplo, pode-se ver qual é a utilização dos CPU, disco e placa de rede do servidor (físico) que está a correr o FOG;

***Hosts*:** permite não só registar novos equipamentos, mas também listar, alterar e eliminar os já existentes;

***Groups*:** permite criar, listar e eliminar grupos de computadores com características similares;

***Users*:** permite criar, editar ou eliminar os administradores dos recursos do FOG;

***Images*:** é a aplicação que permite criar e eliminar as imagens de software; cada *host* tem associada uma imagem de software;

***Tasks*:** permite criar tarefas a ser executadas pelo FOG, tais como o *download* e *upload* de imagens, testes de memória, recuperação de ficheiros, detecção de vírus, entre outras;

***Snap-ins*:** mecanismo que permite automatizar as operações de pós-instalação de imagens, tais como a instalação de módulos de software adicionais;

***Printers'*:** permite criar impressoras simbolizadas como sendo objectos que posteriormente podem ser atribuídos a computadores ou grupos;

***Reports*:** permite exportar informação da base de dados para o formato pdf, html ou csv;

***Info/Misc*:** é um utilitário que permite actualizar o kernel. Permite actualizar o serviço FOG, que pode ser instalado no sistema operativo MS Windows.

Foram apresentadas as funcionalidades de alto nível do FOG, em seguida serão apresentados alguns dos protocolos que têm um papel especial no funcionamento deste, e que dão um carácter inovador a esta ferramenta, são o PXE, o WOL e o NFS.

4.3.2. PREBOOT EXECUTION ENVIRONMENT – PXE

O *preboot eXecution Environment* (PXE) [42][43] é um protocolo que permite fazer o *boot* de qualquer equipamento de rede que suporte este protocolo, utilizando a sua interface de rede, independentemente da existência de um dispositivo de armazenamento ou sistema operativo instalado. O PXE foi introduzido como parte da *Wired for Management Framework* (WfM) publicada pela Intel e Systemsoft em Setembro de 1999. Este faz uso de diversos protocolos de rede, tais como o IP, UDP, DHCP e TFTP. Introduziram-se os conceitos de *Globally Unique Identifier* (GUID) [45], *Universal Unique Identifier* (UUID) [RFC4122] e *Universal Network Device Interface* (UNDI) e expandiu-se o *firmware* do cliente PXE (equipamento que vai arrancar por PXE) com uma série de APIs predefinidas. Em adição ao suporte do protocolo PXE do lado do cliente, a rede de dados deve ter em um servidor PXE. Existem diversos componentes do PXE preparados para funcionar em cooperação com o servidor *Dynamic Host Configuration Protocol* (DHCP), podendo mesmo estes coexistir no mesmo servidor físico. A Figura 27 demonstra a processo de funcionamento do protocolo PXE.

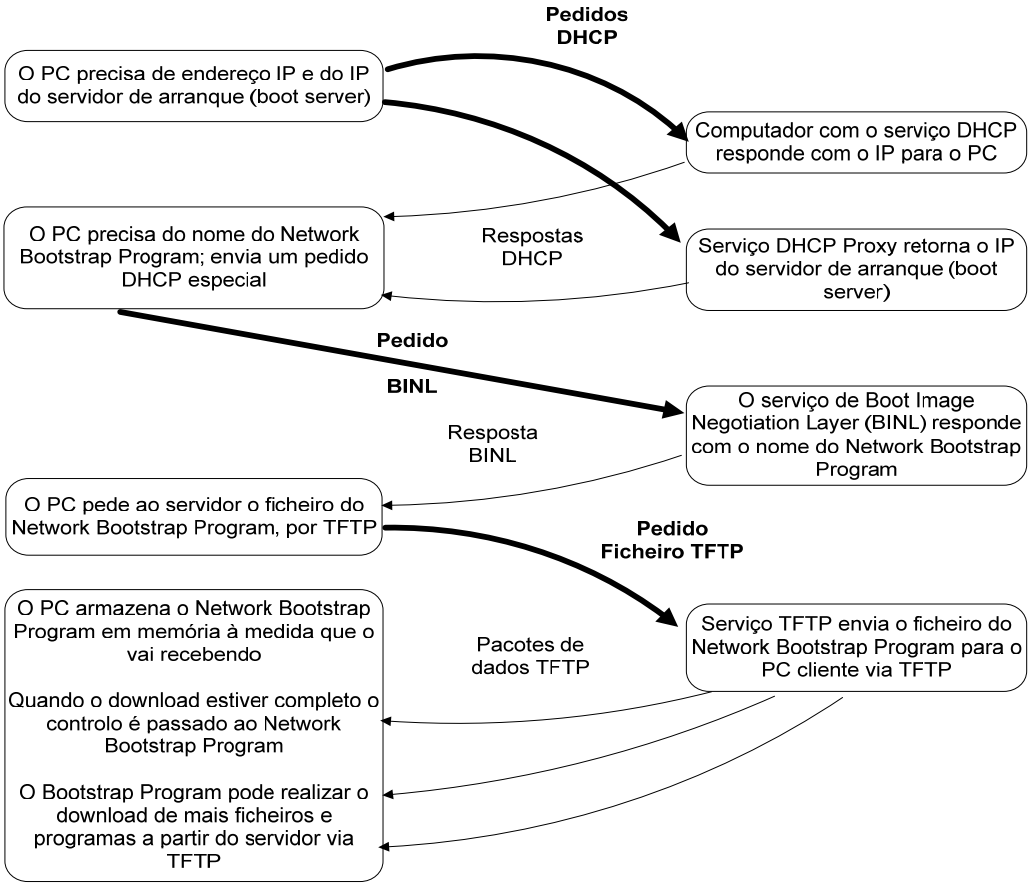

Figura 27: Processo de boot por PXE [43]

4.3.3. *NETWORK FILE SYSTEM* - NFS

Inicialmente desenvolvido pela Sun Microsystems Incorporated, o *Network File System* (NFS) permite o acesso partilhado a ficheiros localizados em servidores, de forma transparente e integrada [19]. A utilização deste protocolo no cliente NFS é quase transparente, uma vez que o utilizador pode executar programas ou utilizar ficheiros localizados no servidor NFS da mesma forma que os utilizaria se estivessem guardados localmente. A Figura 28 ilustra como o NFS é integrado no sistema operativo. Quando uma aplicação é executada, chama o sistema operativo para abrir um ficheiro, obter ou introduzir dados num ficheiro. O mecanismo que controla o acesso aos ficheiros recebe o pedido e passa-o automaticamente para o sistema de ficheiros local ou para o cliente NFS. Depois de receber o pedido, o software cliente utiliza o protocolo NFS para contactar o servidor NFS numa máquina remota e executar o pedido. Quando o servidor NFS remoto responde, o cliente NFS retorna os valores à aplicação local que os solicitou. O cliente acede ao servidor NFS mediante a validação das credencias de acesso. O NFS está especificado no RFC 1094, RFC 1813 e RFC 3530.

O FOG utiliza um sistema operativo embebido que é carregado no computador cliente, e que possibilita por exemplo a instalação das imagens de software. A instalação é feita recorrendo ao Partimage a correr com o NFS, deste modo utiliza as imagens de software guardadas no servidor, de forma transparente e segura.

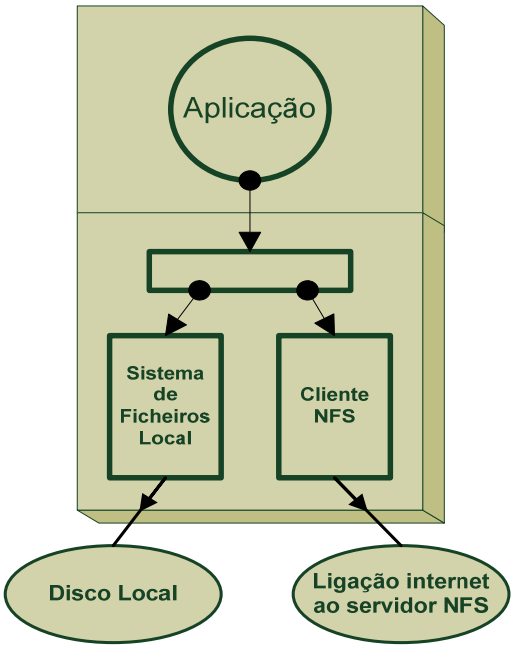

Figura 28: NFS no sistema operativo [19]

4.3.4. WAKE-UP-ON-LAN – WOL

O *Wake-up-On-LAN* (WOL) pode ser implementado em motherboards de computadores. As motherboards mais recentes com placa de rede Ethernet integrada suportam WOL, sem a necessidade de instalar um cabo externo. As motherboards mais antigas com PCI 2.1 e conector WAKEUP-LINK precisavam de um cabo de 3 pinos para ligar à placa de rede Ethernet, de modo a suportam WOL. No entanto, os sistemas que suportam PCI 2.2 e uma placa de rede Ethernet compatível, não precisam do cabo de 3 pinos, uma vez que a informação para ligar o sistema vai pelo barramento PCI. Tipicamente, as motherboards mais antigas vinham equipadas com as funcionalidades *Advanced Power Management* (APM). O WOL deve estar activado na BIOS (secção de gestão de energia da motherboard). [67][68]

Princípios de funcionamento do WOL

O computador, servidor ou portátil pode ser ligado remotamente enviando-lhe um "pacote mágico". Este pacote é enviado em *broadcast*, normalmente através do protocolo UDP (porto 7 ou 9). O pacote mágico contém o endereço MAC da placa de rede, repetido 16 vezes no campo de dados. A placa de rede, que tem de ter energia proveniente do sistema de alimentação *standby*, reage ao pacote de dados que contém o seu endereço MAC enviando um sinal aos circuitos de controlo de energia do computador (através do *Power Management Event* (PME)). Os circuitos de controlo de energia do computador, em resposta, activam a energia do computador de modo a este ligar os vários componentes que o constituem. Isto permite, por exemplo, iniciar o sistema operativo ou arrancar uma aplicação por PXE, para além de permitir ao gestor da rede realizar remotamente as tarefas de manutenção.

Existe também o conceito de *Wake-on-Wireless-LAN* (WoWLAN) [69]. Os novos portáteis com *chipset* Intel 3945 e seguintes (com suporte indicado na BIOS), permitem ligar a máquina remotamente utilizando a rede *wireless* (protocolo 802.11).

Os computadores mais recentes suportam o *Advanced Configuration and Power Interface* (ACPI), que estende o conceito de APM, permitindo ao sistema operativo seleccionar e controlar o modo de energia do computador.

O ACPI suporta vários estados de energia, são exemplos [67]:

- Estado **S0**: o computador está ligado e completamente operacional.
- Estado **S1**: o sistema está em *low power mode* (também conhecido por *sleep mode*). O relógio do CPU está parado, mas a RAM está ligada e a ser refrescada.
- Estado **S2**: Semelhante ao S1, mas a energia é removida do CPU.
- Estado **S3**: Suspenso na RAM (também conhecido por *standby mode*). Basicamente, quase todos os componentes estão desligados excepto a RAM:
- Estado **S4**: Suspenso no disco rígido (também conhecido por *hibernate mode*). Os conteúdos da memória RAM são copiados para o disco rígido e repostos quando o computador é "acordado" do modo de hibernação.
- Estado **S5**: todos os elementos de computador estão desligados, excepto a placa de rede.

No sentido de suportar as placas de rede mais antigas, algumas motherboards recentes continuam a suportar o modo APM, configurável no ACPI. Para que seja possível o arranque remoto, o computador deve estar no modo S1, S3 ou S5, consoante as motherboards. [67]

Efectuou-se um teste de acesso à interface Web do FOG a partir de um PDA HTC TYTN (ligação por *wireless*). A interface Web revelou-se muito flexível, adaptando-se automaticamente à resolução do ecrã do PDA (Figura 29).

Apesar das pequenas dimensões do PDA, a navegação entre os menus do FOG é fácil e intuitiva. Criou-se a seguinte tarefa: instalação de uma imagem de software num grupo de 10 computadores do DEG, com *shutdown* depois da tarefa concluída. Depois de criada a tarefa, o FOG liga automaticamente todos os computadores remotos através do WOL, executa a tarefa e desliga o computador, sem que para isso precisasse de estar no local.

Figura 29: Aceder ao FOG a partir de um PDA

4.4. RESUMO

Neste capítulo foram apresentadas várias ferramentas que permitem a gestão de imagens de software, umas mais simples e outras mais complexas. No leque de ferramentas mais simples consideraram-se o G4U (que funcionam a partir de disquete) e o Clonezilla Live (que funciona a partir de CD-ROM). Quanto às ferramentas mais complexas pode-se destacar o *Free Open Source Ghost* (FOG), que possui uma arquitectura do tipo cliente-servidor, com base de dados para armazenar a informação de rede, *diskless*, com interface Web de gestão, etc.

Com base nos requisitos de gestão de imagens de software da rede do DEG (definidos na secção 2.4) identificaram-se os critérios de selecção da ferramenta de gestão de imagens de software. Dos critérios definidos dá-se destaque ao suporte de vários sistemas de ficheiros (swap, ext2, ext3, fat32 e NTFS), à execução de tarefas remotamente (protocolo PXE), à instalação de imagens em vários computadores em simultâneo e a interface Web de gestão. Em seguida fez-se a análise e selecção da ferramenta de gestão de imagens de software recaindo a escolha no FOG.

O FOG é uma ferramenta com interface Web de gestão, *open source*, flexível, multifuncional e é *user-friendly*. O FOG possibilita a execução remota de um conjunto de tarefas de gestão, isto é, sem a presença física do gestor na rede de dados.

Do conjunto de funcionalidades de alto nível, no FOG destaca-se a interface Web de gestão, a capacidade de armazenar em base de dados a informação de rede, a capacidade de organizar os computadores em grupos, e desta forma executar tarefas de gestão simultâneamente em vários computadores. Das funcionalidades de baixo nível existentes no FOG destacam-se a suporte do protocolo Wake-up-On-LAN (WOL), permitindo ligar remotamente um computador/servidor através da rede, o suporte do protocolo *Preboot Execution Environment* (PXE) que permite a execução de tarefas antes de ser carregado o sistema operativo do computador cliente e o suporte para o protocolo *Network File System* (NFS), utilizado no processo de instalação de imagens de software, e que possibilita a utilização local de ficheiros existentes num servidor remoto.

5. NMSIS - NETWORK MANAGEMENT SYSTEM WITH IMAGING SUPPORT

Muitas vezes as ferramentas de software aberto disponíveis no mercado não possibilitam o tratamento de informações entre aplicações desenvolvidas com diferentes objectivos e muitas vezes por diferentes fabricantes. Neste sentido, pode-se perceber que a capacidade de adaptação, reutilização e potencial de inovação que são inerentes ao software aberto, manipulados com ferramentas de desenvolvimento, possibilitam a geração de novas aplicações passíveis de serem executadas na maioria dos sistemas operativos.

Este modelo de desenvolvimento em que se utilizam ferramentas de código aberto, que podem trabalhar de forma cooperativa (sendo necessário fazer a integração entre estas), pode reduzir os custos de desenvolvimento. Por possibilitarem o controlo desde o código fonte, estas ferramentas possibilitam o auxílio na resolução de possíveis problemas de uma forma simples e directa. A facilidade de integrar novas funcionalidades às ferramentas de código aberto é de grande importância para as equipas de desenvolvimento, fazendo com que a tecnologia em questão tenha a possibilidade de agregar valor ao produto final

desenvolvido. Para dar resposta aos problemas de gestão da rede do DEG foi criado o *Network Management System with Imaging Support* (NMSIS).

5.1. INTEGRAÇÃO NAV-FOG

A solução NMSIS assenta na integração de duas aplicações *Open Source*: o NAV e o FOG. Pode-se dividir a integração entre as ferramentas NAV e FOG em duas fazes distintas: a integração gráfica e a integração funcional. Uma vez que o software NAV está em grande parte escrito na linguagem de programação Python e o software FOG está escrito predominantemente em PHP, a integração gráfica destas ferramentas seria morosa. Assim optou-se por uma integração funcional, com uma estratégia célere que permitisse sincronizar de uma forma rápida a informação mais relevante ao processo de gestão de imagens de software. Neste contexto, optou-se por fazer os registos de equipamento no NAV e desenvolver alguns módulos de software para assegurar a sincronização dos registos entre o NAV e o FOG. Para assegurar a sincronização das bases de dados desenvolveram-se os módulos *DataBase Synchronization Module* (DBSM) e o *Equipment Group Synchronization Module* (GESM), dois programas em código PHP que são acessíveis através da interface Web do FOG e que permitem, de uma forma transparente para o gestor da rede, sincronizar os registos existentes entre as bases de dados referidas.

O NAV apresenta uma documentação muito detalhada, mesmo no que se refere à arquitectura da base de dados, enquanto a aplicação FOG dispõe de uma boa documentação das funcionalidades ao nível de aplicação, mas não apresenta qualquer informação relativamente à arquitectura da base de dados. Para ser possível realizar o processo de sincronização entre as bases de dados do NAV e do FOG foi necessário descobrir como interagiam os vários elementos da base de dados do FOG, e posteriormente definir os algoritmos de integração de ambas as bases de dados.

Depois de sincronizadas as bases de dados, os novos computadores estão registados nas bases de dados do NMSIS mas não no serviço DHCP. A actualização manual do serviço DHCP é morosa, para ultrapassar este problema foi desenvolvido o *Dynamic Host Configuration Protocol Synchronizer* (DHCPSYNC), um programa que permite a actualização automática do serviço DHCP com os novos registos de equipamento efectuados no NMSIS.

5.1.1. DBSM - *DATABASE SYNCHRONIZATION MODULE*

O *DataBase Synchronization Module* (DBSM) sincroniza as bases de dados de informação NAV e FOG. As ferramentas NAV e FOG têm uma semelhança ao nível da arquitectura do software que facilita a sua integração: ambas centram a sua arquitectura em base de dados relacionáveis para o armazenamento da informação. O NAV utiliza PostgreSQL e o FOG utiliza MySQL. Existe no entanto uma diferença ao nível de aplicação, no NAV regista-se o equipamento com base no respectivo IP e não existe na base de dados um campo para introdução do MAC, ao passo que na base de dados do FOG inserem-se os endereços IP e MAC (para os protocolos PXE e DHCP). O esquema da base de dados do NAV está ilustrado na Figura 30.

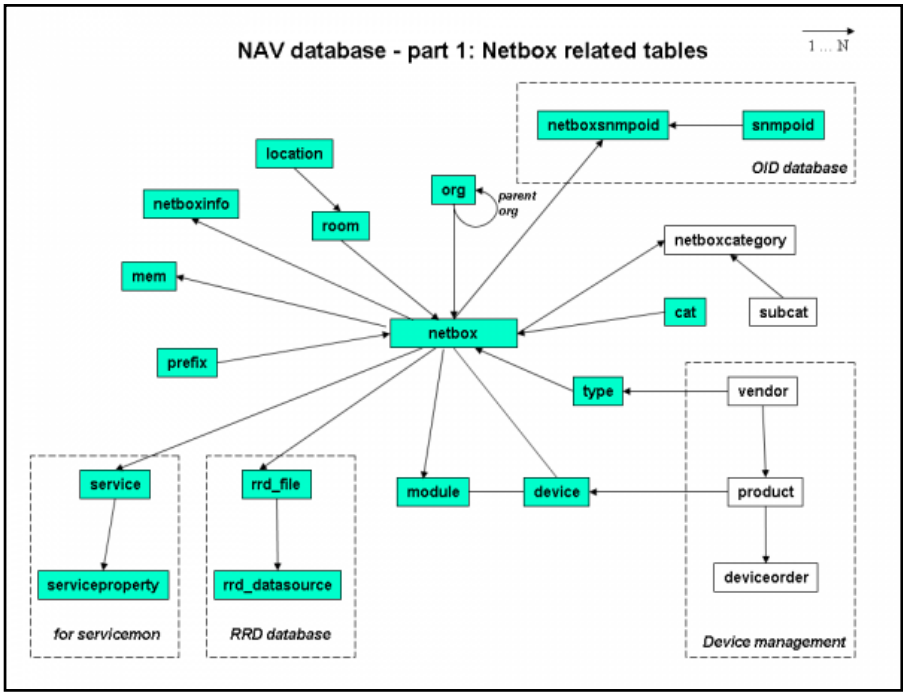

Figura 30: Esquema da base de dados do NAV [87]

Para colmatar esta diferença aproveitou-se o facto do NAV dispor de campos de preenchimento opcional (por exemplo, o campo *Function*) que foram utilizados para introduzir outras informações importantes relativamente ao equipamento em questão (por exemplo o MAC). Neste contexto desenvolveu-se o DBSM, um módulo de software que executa o algoritmo de sincronização das bases de dados (ilustrado na Figura 31), e que satisfazem as seguintes fases: o equipamento a ser gerido em ambos os sistemas será registado no NAV e classificado com a categoria "OTHER" e subcategoria "comp_fog"; no campo de preenchimento opcional "*Function*" (cuja informação pode posteriormente

ser acedida na tabela NETBOXINFO do NAV) introduz-se o endereço MAC, o tipo de sistema operativo, bem como outra informação.

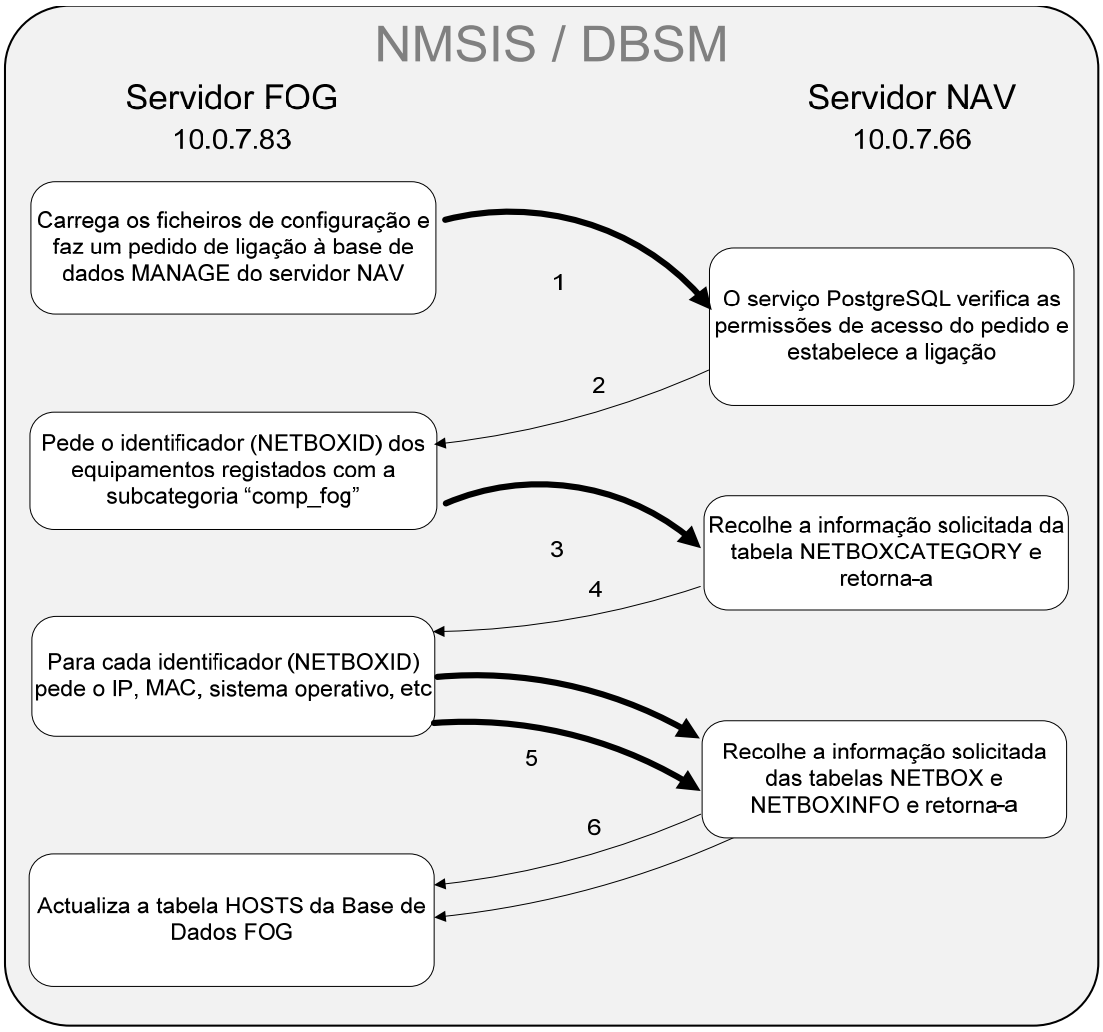

Figura 31: Algoritmo de sincronização das bases de dados MANAGE e FOG

O DBSM foi implementado num *script* em PHP que faz a sincronização da informação entre as bases de dados do NAV e do FOG. A Figura 31 representa o processo de sincronização.

A título de exemplo, o seguinte extracto de código corresponde ao processo de obtenção dos NETBOXID dos equipamentos de tipo "comp_fog" (iterações 3 e 4 da Figura 31):

```
/* LER DA TABELA NETBOXCATEGORY */

$query="SELECT netboxid, category FROM netboxcategory WHERE
category='comp_fog'";

$resultado=pg_query($query);
```

Na Figura 12 da página 25 pode-se visualizar a localização física dos servidores FOG (SIMG) e NAV (SNAV).

Mapeamento entre as bases de dados do NAV e do FOG:

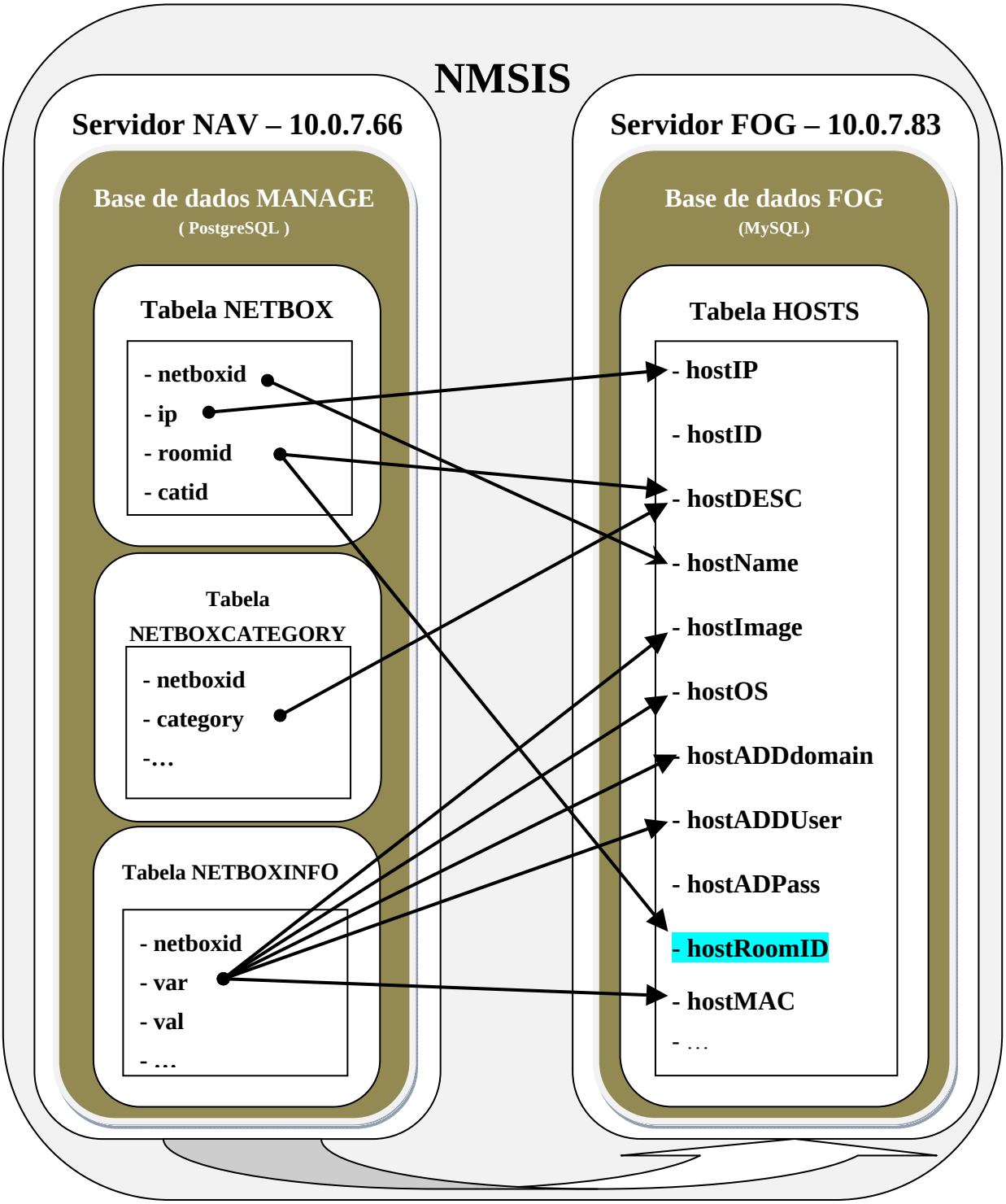

Figura 32: Sincronização do equipamento do tipo “comp_fog” da base de dados MANAGE para a base de dados FOG

Na Figura 32 identificam-se os parâmetros sincronizados. As setas representam fluxo de informação, somente do NAV para o FOG, deste modo a base de dados do NAV contém sempre a informação mais recente. O parâmetro *hostRoomID* representado em relevo não existe na base de dados original do FOG, foi criado para a introdução do compartimento em que se encontra o equipamento. Deste modo reduz-se o número de pedidos à base de dados MANAGE do NAV. Para permitir a sinronização das bases de dados NAV e FOG a partir da internet o *script* de sincronização foi integrado na interface Web de gestão do NMSIS, a partir da qual pode ser activado.

5.1.2. GESM – *GROUP EQUIPMENT SYNCHRONIZATION MODULE*

O *Group Equipment Synchronization Module* (GESM) organiza o equipamento em grupos, de forma automatizada. No NAV organiza-se o equipamento de rede em compartimentos (*rooms*), enquanto no FOG o equipamento de rede pode ser organizado em grupos. Estabeleceu-se então um paralelismo entre estas funcionalidades.

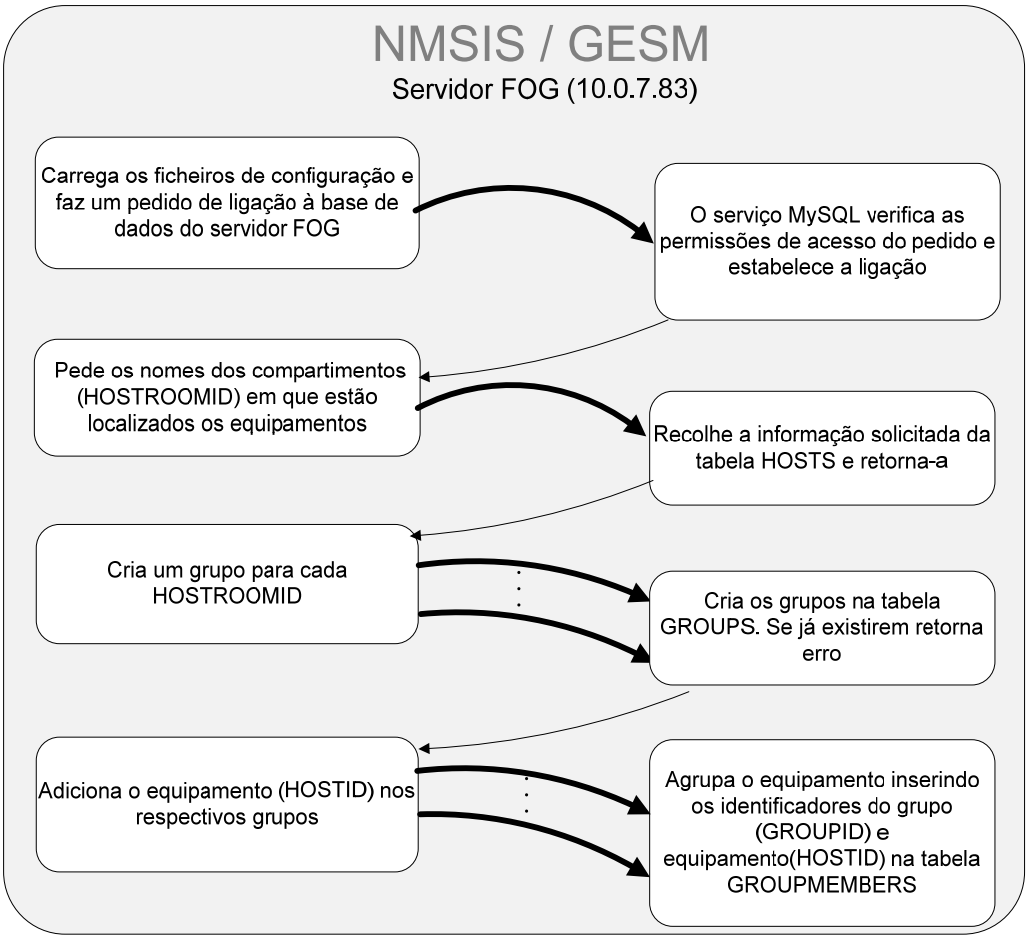

Figura 33: Algoritmo de organização do equipamento em grupos de equipamento

A Figura 33 representa o processo de organização dos equipamentos em grupos com características semelhantes.

O extracto de código que permite obter os HOSTROOMID é:

```
$query = "select hostRoomID from hosts”;
$res = mysql_query( $query, $conn ) or criticalError(
mysql_error(), "FOG/HOSTS - HOSTROOMID :: Erro na base de dados!"
);
```

Quando se procede à sincronização das bases de dados, os nomes dos compartimentos (*hostRoomID*) são utilizados para a criação de grupos de equipamento no FOG.

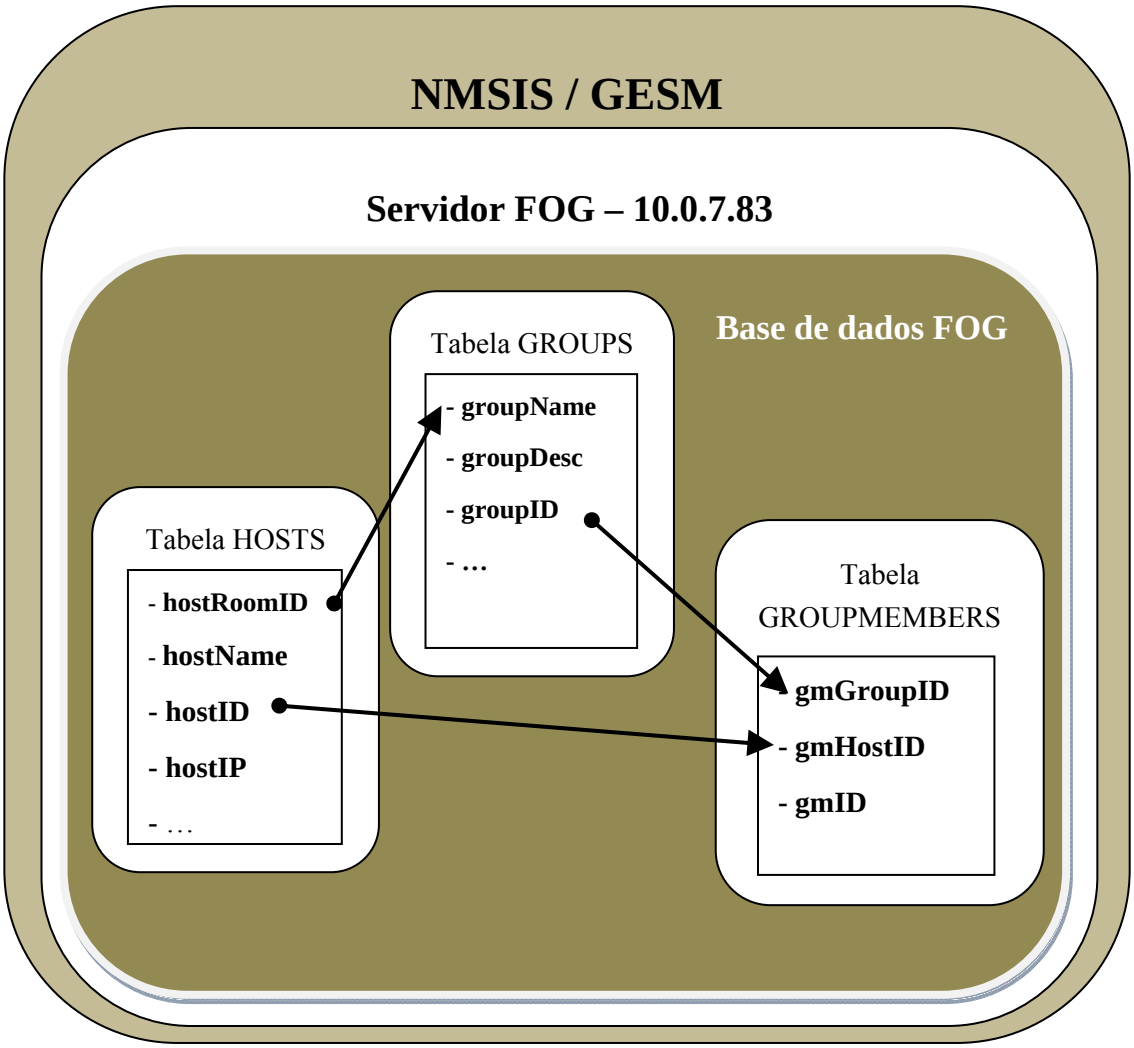

Figura 34: Criação de grupos de equipamento no sistema NMSIS

Depois de criados os grupos, a introdução dos equipamentos nos respectivos grupos têm de seguir a estrutura de software base do FOG. Assim, o GESM permite criar automaticamente os grupos no FOG e agrupar o equipamento de rede em grupos (*hostRoomID*⇔*groupName*), recorrendo à informação da base de dados.

Ao associar os computadores em grupos o FOG permite criar tarefas para vários computadores, em simultâneo. Como está ilustrado na Figura 34, para organizar o equipamento em grupos de forma automática foram seguidas as seguintes directrizes: para associar um equipamento (de nome *hostName* e identificador *hostID*) a um grupo (de nome *groupName* e identificador *groupID*), os respectivos identificadores (*hostID* e *groupID*) têm de ser introduzidos na tabela GROUPMEMBERS. O mapeamento entre as tabelas intervenientes está representado na Figura 34. Para permitir a sincronização dos grupos de equipamento a partir da internet o GESM foi integrado na interface Web de gestão do NMSIS, a partir da qual pode ser activado.

5.2. DHCPSYNC - *DYNAMIC HOST CONFIGURATION PROTOCOL SYNCHRONIZER*

O software *Dynamic Host Configuration Protocol Synchronizer* (DHCPSYNC) permite automatizar o processo de actualização do serviço DHCP.

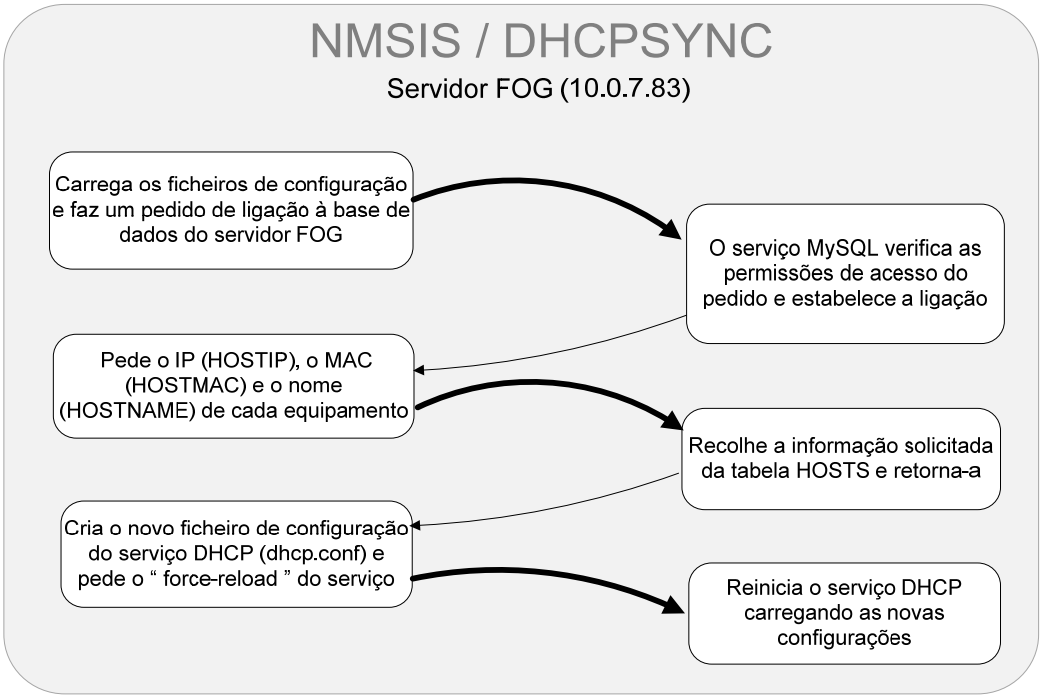

Figura 35: Algoritmo de actualização automática do serviço DHCP

O serviço DHCP é o responsável pela atribuição das configurações de rede aos equipamentos que o solicitam, nomeadamente do endereço IP, máscara de sub-rede, endereço da *gateway*, endereço do servidor DNS, suporte para o PXE, entre outros.

Na Figura 35 está representado o algoritmo implementado pelo DHCPSYNC. O DHCPSYNC é um *script* em PHP que acede à base de dados do servidor FOG, e recolhe a informação mais relevante no contexto da reconfiguração do serviço DHCP. O *script* acede à tabela *hosts* da base de dados e recolhe o nome, IP e MAC de cada equipamento. Em seguida actualiza a configuração do serviço DHCP (em *dhcpd.conf*). Depois reinicia o serviço DHCP para que sejam carregadas as novas configurações. O processo está esquematizado na Figura 36.

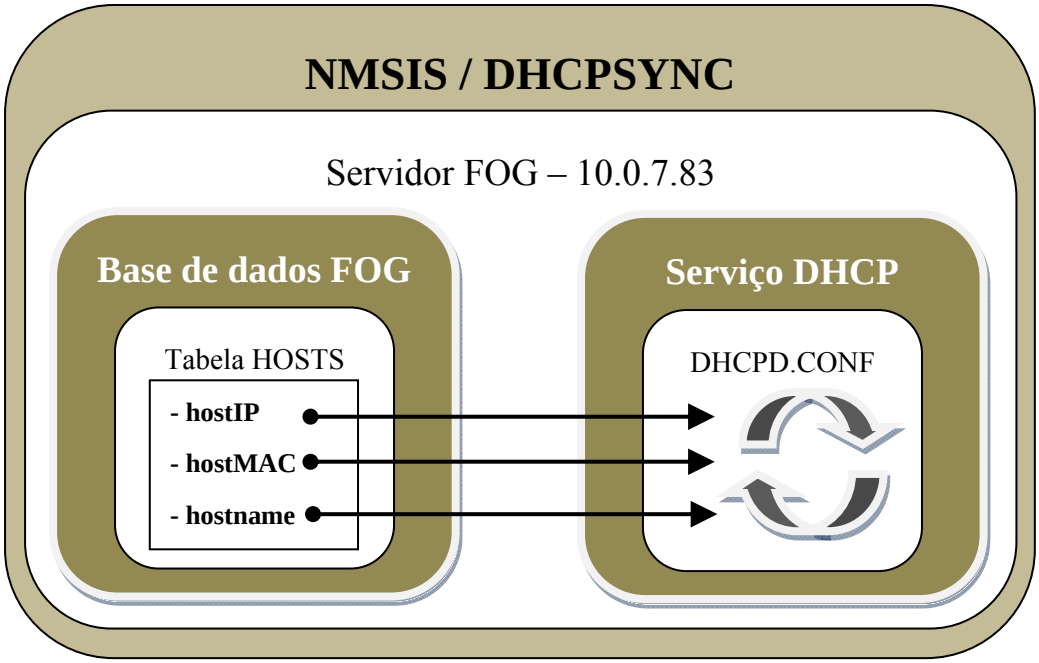

Figura 36: Actualização automática do serviço DHCP

O extracto de código que se segue corresponde ao processo de obtenção do nome (hostname), IP (hostIP) e MAC (hostMAC) para reconfiguração do serviço DHCP:

```
$sql2 = "SELECT hostIP,hostMAC,hostName FROM hosts ORDER BY
hostIP";//Obter tds os IPs do BWTCSameAs

$res2   =   mysql_query(   $sql2,   $conn   )   or   criticalError(
mysql_error(), "DHCP 3 :: Erro na base dados!" );
```

As novas configurações são carregadas no serviço DHCP através da instrução:

```
$output=shell_exec('sudo /etc/init.d/dhcp3-server force-reload');
```

Todo o processo de actualização do serviço DHCP decorre sem intervenção manual. Para permitir a reconfiguração do serviço DHCP a partir da internet o *script* de reconfiguração automática do serviço DHCP foi integrado na interface Web de gestão do NMSIS, na qual pode ser activado.

5.3. RESUMO

Este capítulo descreve a implementação do sistema *Network Management System with Imaging Support* (NMSIS). O NMSIS assenta na integração entre duas aplicações *open source*: o *Network Adminsitration Visualized* (NAV) e o *Free Open Source Ghost* (FOG). Na integração destas ferramentas deu-se pouca relevância à integração gráfica, uma vez que ambas dispõem de interface Web. Assim, grande parte do desenvolvimento assentou na implementação de algoritmos de sincronização de informação de rede, entre as bases de dados dessas ferramentas.

O NMSIS integra duas bases de dados, uma proveniente do NAV (do tipo PostgreSQL) outra do FOG (do tipo MySQL). O equipamento com destino à gestão de imagens de software é registado no NMSIS sob a categoria "OTHER" e subcategoria "comp_fog". O registo inicial fica somente na base de dados NAV mas é sincronizada com a base de dados FOG através do *DataBase Synchronization Module* (DBSM). O DBSM é um *script* desenvolvido em PHP que acede à base de dados do NAV, recolhe a informação (o IP, a localização, o MAC, o sistema operativo, etc.) dos equipamentos da subcategoria "comp_fog", e introduz a informação na base de dados do FOG. Este *script* pode ser executado a partir da interface Web do NMSIS.

O NMSIS permite organizar em grupos, os computadores com características semelhantes, e deste modo executar tarefas (por exemplo, a instalação de imagens de software) para vários computadores ao mesmo tempo, poupando-se tempo. Para automatizar o processo dos equipamentos em grupos implementou-se o *Group Equipment Synchronization Module* (GESM). O GESM é mais um *script* desenvolvido em PHP que cria grupos de computadores, e associa os equipamentos com características semelhantes aos respectivos

grupos, de forma automatizada. Para criar os grupos, o GESM acede à base de dados, recolhe a nome das localizações (por exemplo, sala C207) e cria grupos com o mesmo nome. Para associar os equipamentos aos grupos, o GESM recolhe da base de dados os identificadores dos grupos e dos equipamentos, e introduz esses identificadores na tabela GROUPMEMBERS. Também este *script* pode ser executado a partir da interface Web do NMSIS.

Finalmente, para automatizar o processo de actualização do serviço DHCP com os novos registos de equipamento efectuados no NMSIS desenvolveu-se o *Dynamic Host Configuration Protocol Synchronizer* (DHCPSYNC). O *script* PHP DHCPSYNC foi integrado na interface Web do NMSIS, e executa o algoritmo de actualização do serviço DHCP. O DHCPSYNC acede à base de dados, recolhe o IP, MAC e nome de cada equipamento de rede, actualiza o ficheiro de configuração do serviço DHCP (o *dhcpd.conf*) e reinicia o serviço para que sejam carregadas as novas configurações.

6. CONTROLO DE TRÁFEGO EM LINUX

6.1. CONTROLO DE TRÁFEGO – SOLUÇÃO OU PROBLEMA

As redes de comutação de pacotes (por exemplo, redes baseadas em IP), diferem das redes baseadas em circuito num aspecto muito importante: as primeiras são *stateless* [71]. São *stateless* porque não guardam registos das interacções anteriores e cada pedido de intervenção (por exemplo, a transferências de dados) tem de ser suportado com toda a informação fornecida pelo mesmo. Nas redes baseadas em circuitos (sistemas *statefull*), como a rede telefónica, existe a necessidade de manter a estabilidade da rede e garantir a QoS [61]. Alguns dos parâmetro relacionado com a QoS são o controlo de tráfego, o *delay* na entrega dos pacotes de dados, etc [59]. As redes baseadas no protocolo IP não foram projectadas com ferramentas de controlo de tráfego integradas, com excepção das operações de *traffic routing* [22]. Nas redes ATM o controlo de tráfego é facilitado uma vez que estas foram projectadas com ferramentas de controlo de tráfego integradas.

Em termos simples, o controlo de tráfego permite ao administrador realizar várias operações sobre os pacotes de dados, com base nos seus atributos, e deste modo introduzir o conceito *stateful* [71] nas redes de comutação de pacotes.

Vantagens do controlo de tráfego: Quando é aplicado de forma adequada, o controlo de tráfego torna previsível a percentagem de utilização dos recursos da rede, e menos volátil a alocação dos mesmos. Por omissão, os sistemas Linux utilizam o mecanismo *Packet First*

In First Out Fast (pfifo_fast [144]) para enviar os pacotes de dados através da inerface de rede. O *pfifo_fast*, ao contrário do FIFO, utiliza várias filas de menssagens, em paralelo, para diferenciar o tráfego de acordo com o valor de *Type of Service* (ToS). O tráfego com elevada prioridade (representado pelo valor de ToS) é encaminhado de forma prioritária. Com o controlo de tráfego pode-se mudar este paradigma, o gestor é quem define o modo como são tratados os dados.Por exemplo, em aplicações de baixa prioridade[1], como o serviço de correio electrónico, pode ser alocada largura de banda disponível para este serviço, mesmo que em simultâneo seja processado tráfego de elevada prioridade. O controlo de tráfego permite não só limitar a largura de banda disponível por cliente, mas também realizar operações sobre o tráfego de acordo com as suas características. A título de exemplo, pode-se descartar o tráfego que chega à interface de rede de um servidor e que tem como destino um serviço específico (por exemplo, POP, HTTP, FTP).

Desvantagens do controlo de tráfego: A complexidade é sem dúvida uma das principais desvantagens da utilização de mecanismos de controlo de tráfego. Existem várias maneiras de se familiarizar com o tema controlo de tráfego, com rápidas curvas de aprendizagem. O controlo de tráfego é muito útil quando é aplicado e configurado de forma apropriada, permitindo distribuir com equidade os recursos da rede de dados. No entanto, facilmente se definem configurações erradas, se não houver um conhecimento pleno do tema, levando a alocação inadequada de recursos. Deste modo torna-se um verdadeiro desafio identificar e corrigir problemas relacionados com a configuração errada do software.

Para o uso pessoal, não existem custos associados ao uso do controlo de tráfego, uma vez que existe documentação gratuita e disponível na internet. Para uma empresa existem os custos associados à formação de funcionários. Garantir o conhecimento aprofundado sobre um tema conceptualmente abrangente e complexo como o controlo de tráfego, pode ter elevados custos e a empresa pode simplesmente optar por aumentar a largura de banda, em vez de optimizar a sua utilização.

[1] O parâmetro *Differentiated Services Code Point* (DSCP) define a prioridade dos pacotes de dados

6.2. A *FRAMEWORK IPROUTE2*

O *iproute2* é uma colecção de ferramentas de software disponível para o sistema operativo Linux. Estas ferramentas permitem não só controlar as comunicações feitas segundo o modelo TCP/IP, mas também fazer o controlo de tráfego. Actualmente o iproute2 é mantido por Stephen Hemminger. Alexey Kuznetsov ficou conhecido por introduzir o QoS (controlo de tráfego) no kernel Linux versão 2.2.

Em grande parte dos manuais de configuração de redes faz-se referência ao "ifconfig" e "route" como as principais ferramentas de configuração. No entanto o "ifconfig" é conhecido por ser instável nas redes de dados mais modernas. Apesar disto, a maior parte das distribuições baseadas em Linux continuam a utilizar o "ifconfig". O projecto "/etc/net" [132] procura suportar as tecnologias mais recentes aplicadas às redes de dados, e não inclui o "ifconfig" [61]. O iproute2 é normalmente fornecido num pacote denominado iproute ou iproute2, constituído por um conjunto de ferramentas, das quais se destacam as mais importantes: o *ip* e o *tc*. O *ip* controla a configuração das redes IPv4 e IPv6, ao passo que o *tc* é aplicado no controlo de tráfego [62][63]. O iproute2 pode ser integrado com a *firewall* Linux Iptables [64].

6.3. *LINUX TRAFFIC CONTROL - TC*

Os *kernels* Linux recentes oferecem várias ferramentas para realizar o controlo de tráfego. Uma fatia grande das funcionalidades de controlo de tráfego existentes na versão actual do kernel (versão 2.6) fora introduzida por Alexey Kuznetsov no kernel 2.2 [50]. Esse trabalho foi inspirado nos conceitos apresentados na publicação [51], mas também cobre os mecanismos necessários para suportar a arquitectura desenvolvida pelos grupos "*intserv*" (serviços integrados) [RFC1633] [52] e "*diffserv*" (serviços diferenciados) [RFC2475] [53] do IETF. Neste contexto, em seguida serão apresentados as publicações que serviram de inspiração ao trabalho desenvolvido por Alexey Kuznetsov.

Na publicação [51], *Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism,* é considerado o suporte para aplicações de tempo real em redes ISPN (*Integrated Services Packet Networks*). Em primeiro lugar são analisadas as necessidades das aplicações de tempo real. Desta análise verificam que ao contrário do que é publicamente considerado, que as aplicações de tempo real precisam de um *delay* fixo, algumas aplicações de tempo real são mais flexíveis e podem adaptar-se às

condições actuais da rede de dados. Em seguida propõem uma arquitectura para as redes ISPN e que dá suporte a dois tipos distintos de serviços de tempo real: o *guaranteed service* e o *predicted service*. Posteriormente propõem um mecanismo de escalonamento e organização de pacotes de dados, com suporte para os serviços de tempo real.

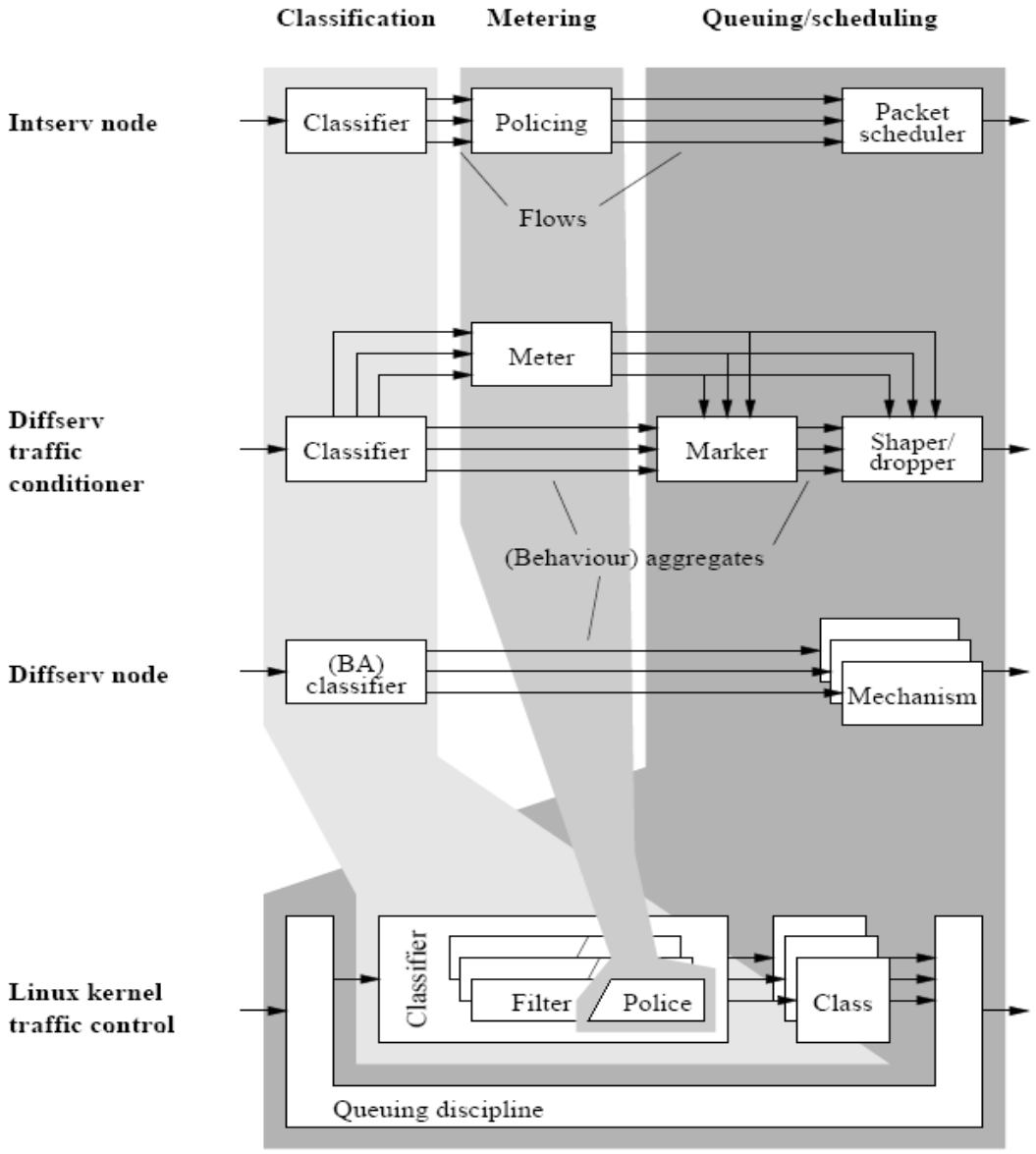

Figura 37: Relação entre os elementos das arquitecturas *intserv* e *diffserv* com o controlo de tráfego do kernel Linux [50];

O *Integrated Services (intserv) working group* foi criado pelo IETF [52]. Com base em experiências este grupo defende que os protocolos de comutação de pacotes são capazes de suportar e garantir a QoS nos Serviços Integrados. Os serviços integrados são: o transporte de áudio, vídeo, serviços de tempo real, e o transporte de dados clássico, numa só infra-

estrutura de rede. As experiencias realizadas por este grupo de trabalho sugerem que expandindo o modelo de serviços da Internet dá-se melhor resposta às necessidades destas aplicações (áudio, vídeo, etc).

Assim, o propósito deste grupo de trabalho é especificar e fortalecer um novo modelo de serviços da Internet, e estandardizar as interfaces e requisitos necessários para o implementar, fazendo da Internet uma infra-estrutura robusta de serviços integrados.

O IETF criou o *Diff-Serv Working Group* [53] para desenvolver um método simples que permitisse diferenciar classes de serviços no tráfego da Internet, garantindo a QoS. O *Diff-Serv* define um novo campo DS (*Differentiated Services*) no cabeçalho das tramas IP, e que substitui o antigo campo ToS (*Type of Service*). O padrão de bits no campo DS especifica o tipo de serviço e o tipo de processamento a ser realizado nos activos de rede que suportam *Diff-Serv*.

Na publicação *Interoperation of RSVP/Int-Serv and Diff-Serv Networks* [54] é apresentada uma proposta de interoperabilidade entre *intserv* e *diffserv*. O *diffserv* e o *intserv* são diferentes aproximações que procuram dar resposta ao mesmo problema, a QoS para a Internet e sistemas. Estas aproximações devem ser capazes de coexistir e operar entre si. Em [54] é apresentado um modelo que visa a interoperabilidade entre os modelos referidos, em que o modelo *diffserv* é utilizado nas *transit networks* [133] no núcleo da Internet, enquanto os *hosts* e redes periféricas (*edge networks*) utilizam o modelo *intserv*. Também discutem alguns modelos alternativos de interoperabilidade entre o *diffserv* e o *intserv*. Na Figura 37 é apresentada a relação destes modelos com o controlo de tráfego do kernel Linux.

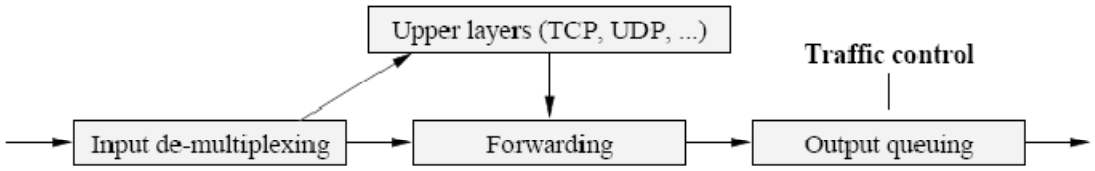

Figura 38: Processamento dos dados com o controlo de tráfego

A Figura 38 mostra, de forma simples, como são processados pelo *kernel* os dados recebidos da rede de dados, e como são gerados novos dados para serem enviados para a rede: os pacotes recebidos são examinados e podem ser directamente encaminhados para a rede (se a máquina estiver a actuar como *bridge* ou *router*), ou encaminhados para as

camadas superiores da pilha protocolar onde posteriormente são processados (por exemplo para um protocolo de transporte como o TCP ou correio electrónico como o POP3). As aplicações e protocolos das camadas superiores podem também gerar dados, encaminhando-os para as camadas inferiores para tarefas como o encapsulamento, o *routing*, ou mesmo a transmissão. O encaminhamento (*forwarding*) inclui a selecção do interface de saída, o encapsulamento, etc. Depois do processo de encaminhamento os dados são colocados nas filas de dados do respectivo interface de saída. Normalmente é a este ponto que ocorre a intervenção do controlo de tráfego (*traffic control*). O controlo de tráfego pode, entre outras coisas, decidir se os dados são colocados nas filas ou descartados (por exemplo, quando a fila atingiu a capacidade máxima, ou o tráfego excedeu a frequência máxima). Pode decidir a ordem de envio dos dados (atribuindo diferentes prioridades a determinados *flows*), pode atrasar o envio dos pacotes de dados (para limitar o débito do tráfego de saída). Assim que o controlo de tráfego liberta o pacote de dados para ser enviado, o *device driver* envia-o para a rede.

6.3.1. COMPONENTES DO *LINUX TRAFFIC CONTROL* (*TC*)

O código do controlo de tráfego existente no kernel Linux tem como base os seguintes conceitos [50]:

- Disciplinas de tratamento de filas (*queueing disciplines* ou *qdisc*)
- Classes (dentro duma *qdisc*)
- Filtros
- Policiamento (*policing*)

O *queueing discipline* (*qdisc*) é o maior bloco no qual é construído todo o controlo de tráfego em Linux. O *qdisc* é um escalonador (*scheduler*) de pacotes de dados. Na terminologia do *tc* são usados muitas vezes os termos *egress qdisc* e *ingress qdisc*, e na realidade estas estruturas não aplicam controlo nas filas de dados, são antes localizações onde se podem anexar as estruturas de controlo do tráfego, de saída e entrada respectivamente. Cada interface de rede contém ambos. O mais utilizado no controlo de tráfego é o *egress qdisc* (também conhecido como *root qdisc*), porque suporta a tratamento dos pacotes de dados em classes de acordo com os diferentes critérios definidos com os filtros. O *ingress qdisc* não suporta a distinção dos pacotes de dados em classes.

Cada equipamento de rede tem uma *qdisc* associada que controla como é tratada a informação contida nas filas de dados. Como se pode ver na Figura 39, uma *qdisc* muito simples consiste numa fila única com dados armazenados onde estes são enviados pela ordem por que foram introduzidos (FIFO), à velocidade máxima que o equipamento suporta. Por omissão a *qdisc* implementado pelo *kernel* Linux é o *pfifo_fast,* que é semelhante ao FIFO mas um pouco mais elaborado.

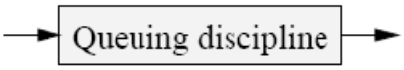

Figura 39: Uma qdisc simples [50]

Uma *queueing discipline,* como o próprio nome indica, funciona com base em disciplinas, estas podem conter ou não classes [46]. Existem dois grandes grupos de *qdisc* [46] [55]: as *classful qdisc* (*qdisc* com classes) e as *classless qdisc* (*qdisc* sem classes). Existem vários tipos de disciplinas adoptadas nas *classful qdisc*, são elas o *Hierarchical Token Bucket* (HTB), *Hierarchical Fair Service Curve* (HFSC)*, Priority Scheduler* (PRIO) e o *Class Based Queuing* (CBQ). Em [46] e [55] são igualmente apresentadas as *classless qdisc* (*qdisc* sem classes), nomeadamente o *First-In-First-Out* (FIFO), o *Packet First In First Out Fast* (pfifo_fast) (o *qdisc* utilizado por omissão em Linux), o *Stochastic Fair Queuing* (SFQ), o *Extended Stochastic Fair Queuing* (ESFQ), o *Generic Random Early Drop* (GRED) e o *Token Bucket Filter* (TBF).

As *qdisc* mais elaboradas podem conter filtros. O filtro é o componente mais complexo do sistema de controlo de tráfego em Linux, e pode ser associado a *qdisc* ou a classes dentro de *qdisc*. O filtro disponibiliza dois dos mecanismos mais importantes no controlo de tráfego: a classificação de pacotes e o policiamento.

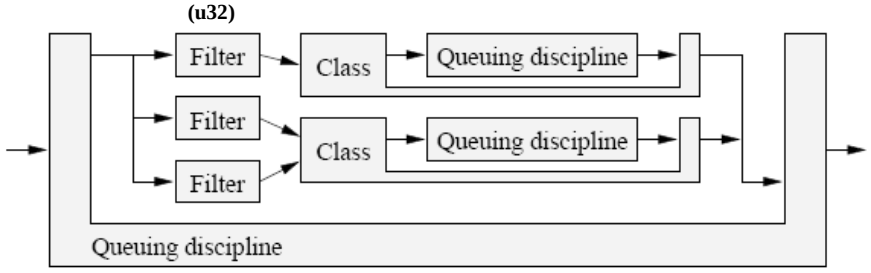

Figura 40: Exemplo de uma *qdisc* com várias classes e filtros [50]

Em [55] são apresentados alguns dos classificadores mais utilizados, são eles o *fw*, o *u32*, o *route*, o *handle* e o *protocol*. Com o *fw* faz-se a filtragem de pacotes com base na marcação efectuada pela *firewall*. Com o *u32* faz-se a filtragem com base nos campos do pacote de dados, como por exemplo o endereço IP de destino. Com o *route* faz-se a filtragem com base na rota que o pacote de dados vai seguir). O *handle* tem diferentes significados para diferentes filtros, por exemplo, identifica o tráfego associado a uma class. Com o *protocol* faz-se a filtragem ao nível do protocolo (por exemplo, o protocolo IP).

Os filtros Linux permitem ao utilizador classificar os pacotes que se encontram nas filas de saída. Os pacotes identificados pelos filtros como preenchendo os critérios de selecção são encaminhados para a respectiva classe, os restantes são enviados pelo método *pfifo_fast*, que é o método utilizado por omissão. A título de exemplo, na Figura 40, utilizando o filtro u32 é possível identificar as tramas de dados que têm no campo IP de destino o valor 10.0.7.61, e desta encaminhar os pacotes para a respectiva classe.

O policiamento (*policing*) é o mecanismo que permite limitar a taxa de transmissão ou recepção de dados. O *ingress qdisc* não suporta classes, mas suporta filtros. Desta forma só é possível associar o mecanismo de policiamento ao *ingress qdisc* para limitar a quantidade de tráfego de entrada na interface de rede. Como a *egress qdisc* suporta classes, é possível tratar o tráfego em função das suas características. Portanto, quando associado à *egress qdisc*, o "polícia" pode limitar a taxa de transmissão de dados para diferentes clientes, descartando ou reclassificando o tráfego excedente. Por exemplo, um filtro pode classificar as tramas de dados que têm no campo IP de destino o valor 10.0.7.61 e em seguida o polícia garantir que a taxa de transmissão de pacotes para esse IP não ultrapassa 72Mbps. Na Figura 37 está representada a interacção do polícia com os demais elementos do *tc*.

As classes são muito flexíveis e podem sempre conter múltiplas classes filhas, ou mesmo um *classful qdisc*. Como se pode ver na Figura 40 as classes podem ter um número arbitrário de filtros associados, o que permite, por exemplo, encaminhar os dados para uma classe filha ou descartar o tráfego que está a entrar numa classe. Existem depois as *leaf classes* que são as últimas classes da árvore duma *qdisc*. Como se pode ver na Figura 41, qualquer classe que contém uma classe filho é denominada *inner class* ou *root class*.

Hierarchichal Token Bucket (HTB)

O *Hierarchichal Token Bucket* (HTB) permite modelar o tráfego (*traffic shaping*). É um mecanismo de gestão de filas que é implementado em classes (*classfull qdisc*). Foi escrito por Martin Devera e introduzido no kernel 2.4. [47]

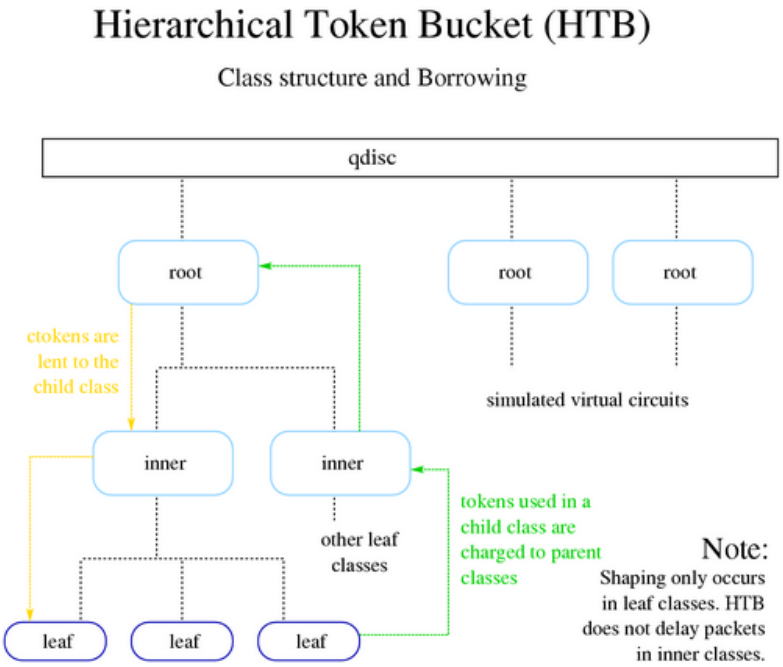

Figura 41: Estrutura de classes no HTB [47]

Conceptualmente, o HTB contém um número arbitrário de *token buckets*, organizados de forma hierárquica. Considere-se o seguinte cenário baseado na Figura 41: uma *root qdisc* contém somente uma classe (cenários complexos podem ter múltiplas classes associados à *root qdisc*); esta classe funciona segundo a disciplina HTB, configurada apenas com os parâmetros *rate* e *ceil*; estes parâmetros devem ter o mesmo valor na classe directamente associada à *root qdisc*, e representam o total da largura de banda disponível para aquela ligação. No HTB, a taxa de transferência (*rate*) significa o valor mínimo de largura de banda reservado para uma determinada classe. O valor de *ceil* é o valor máximo de largura de banda que uma classe pode utilizar, ou seja, caso exista largura de banda disponível esta pode ser utilizada (não é reservada). Qualquer largura de banda utilizada entre os valores de rate e ceil é "roubada" à classe hierarquicamente superior (*parent class*). [47]

Cada "classe pai" (*inner class*) pode conter várias "classes filhas" (*leaf classes*), cada uma destas pode alocar uma porção da largura de banda existente na "classe pai". Nas "classes

filhas" os valores de *rate* e *ceil* não têm de ser iguais. As configurações das "classe pai" e "classes filhas", permitem ao HTB não só alocar largura de banda para uma determinada classe, mas também calcular o rácio de utilização da largura de banda. Com base no rácio de utilização da largura de banda de cada classe, o HTB distribui a largura de banda que não está a ser utilizada (até ao valor de *ceil*).

O acrónimo "tc" representa o programa utilizado em Linux para aceder às funcionalidades de controlo de tráfego existentes no kernel Linux. Em [55] são apresentados com elevado pormenor os parâmetros de utilização do software "tc".

6.4. RESUMO

Neste capítulo faz-se uma análise crítica à utilização das ferramentas de controlo de tráfego em Linux, referindo algumas das principais vantagens e desvantagens. Uma das vantagens da utilização de ferramentas de controlo de tráfego está associada à introdução do conceito de *stateful* nas redes de comutação de pacotes. Deste modo torna mais previsível e menos volátil a utilização da rede, e permite o melhoramento dos parâmetros de QoS (por exemplo, o *delay* e o *throughput*). A complexidade é uma das principais desvantagens da utilização dos mecanismos de controlo de tráfego. Nesse contexto apresentou-se a *Framework Iproute2* que é um conjunto de ferramentas em Linux vocacionadas para a gestão de redes. As principais ferramentas da Framework Iproute2 são o *ip* e o *tc*. A ferramenta *ip* controla a configuração das redes IPv4 e IPv6, ao passo que o *tc* é aplicado no controlo de tráfego.

O *Linux Traffic Control* (*tc*) é apresentado e descrito o seu funcionamento. O *tc* actua ao nível das filas de dados do interface de rede, quer seja a fila de entrada (*igress queue*) ou a fila saída de dados (*egress queue*), de modo a executar as configurações definidas pelo programador. O *tc* tem por base os seguites conceitos: as disciplinas de tratamento de filas de pacotes (qdisc), as classes (dentro duma qdisc), a filtragem e o policiamento (policing). A qdisc define o algoritmo utilizado no tratamento dos dados (por exemplo, o HTB, o TBF, o CBQ, etc). Nas classes configuram-se os parâmetros que definem o modo como as tramas de dados serão processadas (por exemplo, limitar a taxa de transferência). Os filtros permitem identificar os pacotes de dados pelas características do respectivo cabeçalho, encaminhando-os para as respectivas classes. O policiamento (*policing*) é o mecanismo que permite limitar a taxa de transmissão ou recepção de dados. Dos elementos do *tc*

apresentados, assume destaque a disciplina de tratamento de dados *Hierarchichal Token Bucket* (HTB). O HTB é o substituto de CBQ. O HTB é muito utilizado porque suporta classes e filtros, sendo mais fácil de configurar que o CBQ. A título de exemplo, o HTB permite limitar o débito máximo de dados com destino a qualquer *host*, deste modo torna mais previsível e menos volátil a utilização da rede de dados.

7. BANDWIDTH / TRAFFIC CONTROL (BWTC)

O DEG disponibiliza, através da sua rede de dados, vários serviços informáticos aos seus docentes e discentes. O DEG fornece o serviço de página de internet, o serviço de contas de utilizadores em SAMBA, as salas de informática, o acesso à internet, entre outros. Esporadicamente é necessário repor as imagens de software nos computadores das salas de informática por várias razões: devido à existência de vírus; instalação de novos módulos de software; cotas de disco no limite e outras. O processo de reposição de imagens de software nos computadores poupa muito tempo, caso contrário seria necessário formatar um computador de cada vez e instalar todo o seu software. No entanto a reposição de imagens tem o inconveniente de utilizar elevada largura de banda, o que resulta no congestionamento de tráfego nas intersecções de rede, e consequente diminuição da qualidade dos restantes serviços informáticos (que também utilizam a rede de dados).

Como solucionar este problema? Fazendo a instalação das imagens de software durante a noite? E se algum aluno ou docente se encontra a aceder ao servidor WEB? E se os computadores não suportarem arranque remoto por WOL e/ou PXE? E se for urgente formatar uma sala de informática durante a manhã para estar disponível à tarde? …

São demasiadas perguntas e o gestor da rede deve ter mais com que se preocupar. O controlo do débito de dados do servidor de imagens provém da necessidade de melhorar a qualidade dos serviços prestados aos docentes e alunos do DEG, quando se encontra a decorrer a instalação de imagens de software em computadores do departamento. O controlo de débito de dados enquadra-se no tema controlo de tráfego.

BWTC - BandWidth / Traffic Control

O *BandWidth / Traffic Control* (BWTC) foi criado com o objectivo de controlar o débito de dados, do servidor de imagens para os computadores clientes.

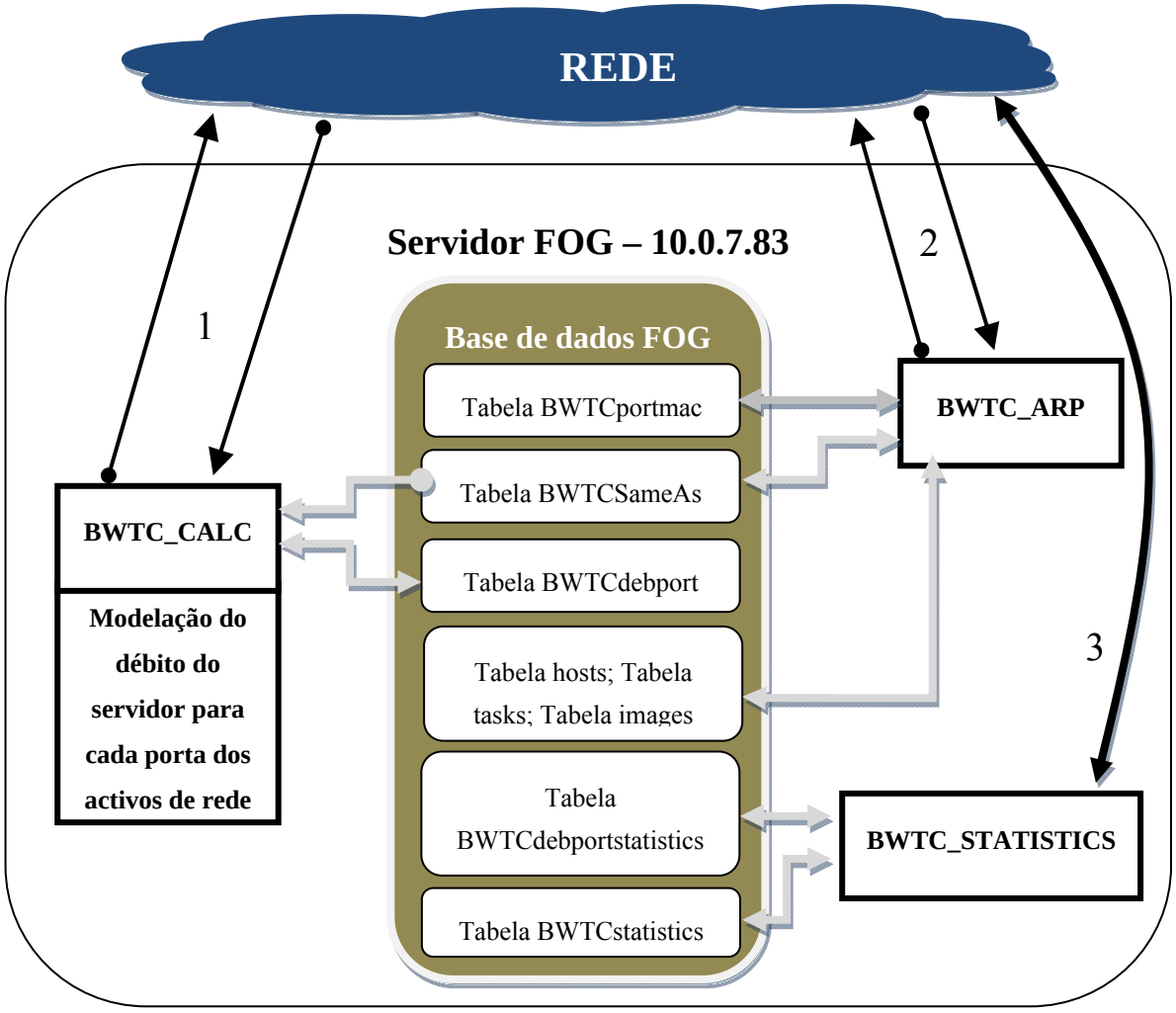

Figura 42: Arquitectura de software do sistema BWTC

Na Figura 42, as setas numeradas com o número 1 representam os pedidos SNMP do módulo BWTC_CALC aos activos de rede e o tráfego de dados do servidor de imagens para os computadores clientes. As setas numeradas com 2 e 3 representam os pedidos

SNMP dos módulos BWTC_ARP e BWTC_STATISTICS aos activos de rede. As restantes setas (a cinzento) representam a troca de informação com a base de dados FOG.

Com o BWTC controla-se o débito do servidor de imagens, evita-se o congestionamento de tráfego nas intersecções de rede, e garante-se que existe largura de banda disponível para os restantes serviços informáticos. Este sistema funciona em cooperação com o software implementado anteriormente, o *Network Management System with Imaging Support* (NMSIS).

O BWTC funciona graças à interacção entre os vários módulos que o constituem: o BWTC_CALC, o BWTC_ARP e o BWTC_STATISTICS. A Figura 42 fornece uma perspectiva global da arquitectura deste sistema.

Os vários módulos do software BWTC funcionam em intervalos de tempo discretos, com períodos diferentes consoante as tarefas que desempenham. O BWTC_CALC funciona com um período de 10 segundos, o BWTC_ARP com um período de 7 segundos e o BWTC_STATISTICS com um período de 36 segundos.

O funcionamento global do BWTC divide-se em três grandes fases:

- Obter as estatísticas de tráfego das portas dos activos de rede;
- Identificar a lista de IP's associados a cada porta dos activos de rede, e que estão marcados na base de dados para o processo de instalação de imagem de software;
- Aplicar o controlo do débito de dados do servidor de imagens para cada computador, de modo a que a percentagem de utilização de cada porta não ultrapasse o valor de *ceil* (valor definido pelo gestor da rede).

Em seguida serão apresentados os pormenores de funcionamento de cada módulo, bem como a interacção entre eles.

7.1. BWTC_ARP – *BANDWIDTH / TRAFFIC CONTROL ARP*

O controlo do tráfego do servidor de imagens é feito com a ferramenta *Linux Traffic Control* (*tc*). Esta ferramenta contém uma infindável quantidade de parâmetros de configuração. No contexto do controlo do débito, o tráfego de saída do servidor de imagens pode ser classificado em função dos parâmetros que caracterizam cada uma das tramas de

dados, seja: ao nível do protocolo, do porto, do IP de destino, da prioridade do pacote, entre outros. Nas comunicações *Ethernet* sob o protocolo TCP/IP, descartando alguns casos particulares como o *Network Address Translation* (NAT), os pacotes de dados contêm sempre o mesmo endereço IP de destino e são encaminhados pela rede até chegarem ao equipamento correspondente. Por isto optou-se por fazer o controlo do débito do servidor para cada computador com base no IP atribuído pelo DHCP (estático).

Nos cenários de teste ao software BWTC, realizados na rede do DEG, não se teve acesso a *routers*, somente a um *switch* com gestão SNMP (HP Procurve 2510-24). As características deste *switch* estão apresentadas no Anexo A. É de salientar que o HP Procurve 2510-24 suporta a Bridge MIB [RFC1493], portanto mantém esta MIB actualizada com a informação dos endereços MAC, por porta, para os quais o *switch* encaminhou ou filtrou informação.

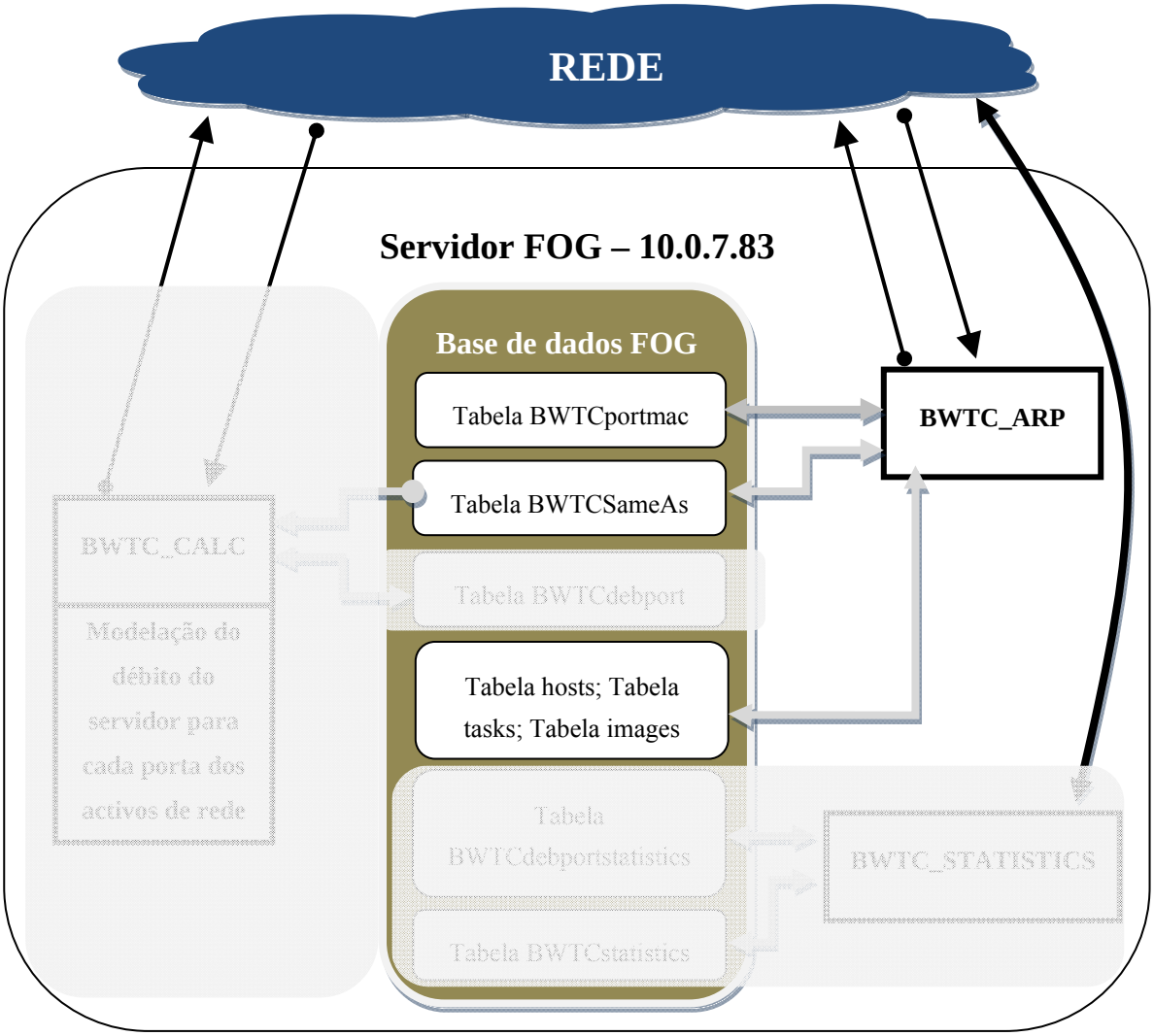

Figura 43: Interacção entre o BWTC_ARP, a base de dados e a rede

O BWTC_ARP utiliza a informação contida nas Bridge MIB dos activos de rede para identificar os computadores (pelo endereço MAC) em comunicação com cada porta. Como está ilustrado na Figura 43, o BWTC_ARP não interage somente com a rede de dados, mas também com a base de dados de gestão (base de dados FOG).

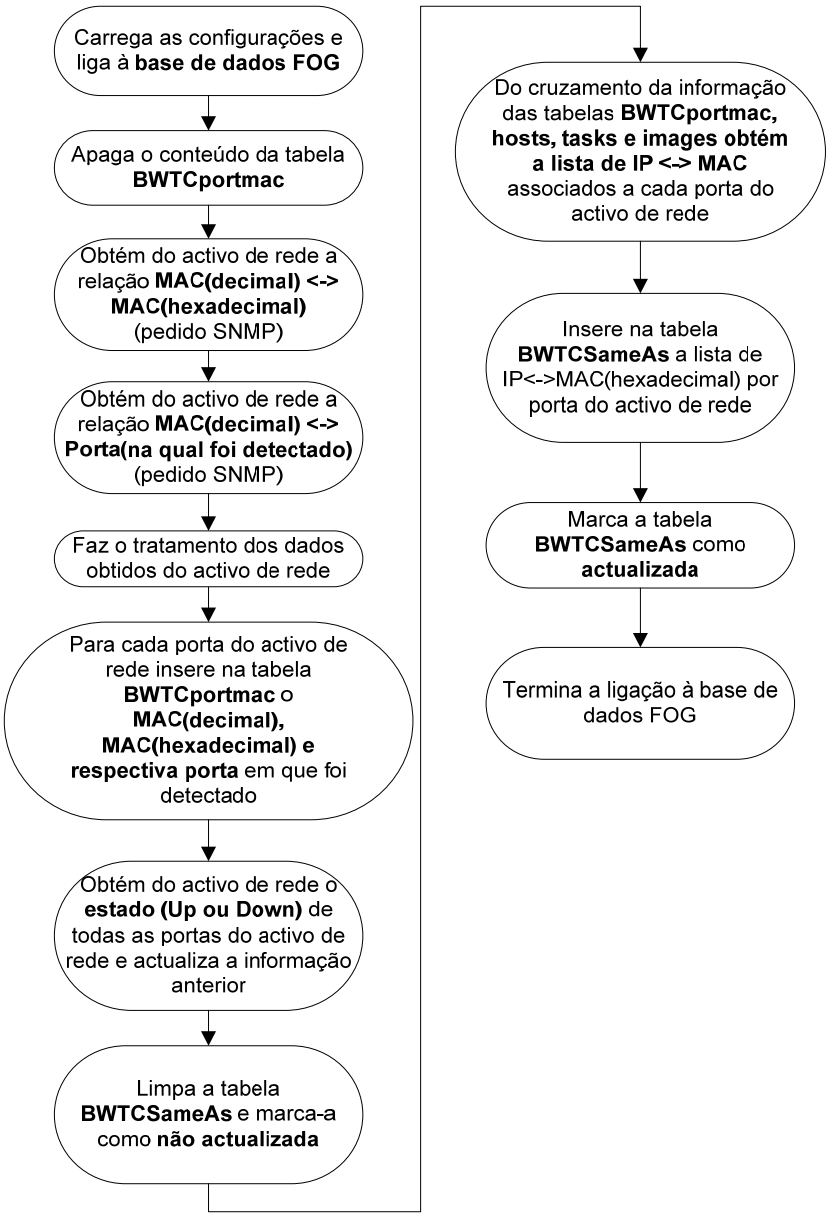

Figura 44: Algoritmo do BWTC_ARP

Como está ilustrado no algoritmo do BWTC_ARP, na Figura 44, é necessário realizar duas leituras do activo de rede e tratar os dados recebidos para se obter a relação MAC(hexadecimal) <-> Porta.

O processo de obtenção do endereço MAC nos formatos hexadecimal e decimal (para cada porta do activo de rede com o IP "10.0.7.245" e senha SNMP "deg") está representado no seguinte extracto de código:

```
/*Obtenção dos endereços MAC por Porta do Activo de rede,
recorrendo à informação da BRIDGE-MIB (RFC 1493)*/
/*http://www.oidview.com/mibs/0/BRIDGE-MIB.html*/

//Obter a relação MAC(decimal) <-> MAC(hexadecimal)
//(01)iso(03)org(06)dod(01)internet(02)mgmt(01)mib-
//2(17)dot1dBridgeMIB(04)dot1dTp(03)dot1dTpFdbTable(01)dot1dTpFdbE
//ntry(01)dot1dTpFdbAddress

$b5 = snmpwalkoid("10.0.7.245", "deg", "1.3.6.1.2.1.17.4.3.1.1");

//Obtenção da relação MAC(decimal) <-> Porta do activo de rede
//(01)iso(03)org(06)dod(01)internet(02)mgmt(01)mib-
//2(17)dot1dBridgeMIB(04)dot1dTp(03)dot1dTpFdbTable(01)dot1dTpFdbE
//ntry(02)dot1dTpFdbPort

$b6 = snmpwalkoid("10.0.7.245","deg","1.3.6.1.2.1.17.4.3.1.2");
```

Depois de executadas as linhas de código anteriores, os dados são tratados e introduzidos na tabela BWTCportmac (ver a Figura 43 e a Figura 44). Em seguida é apresentada a *query* que permite obter a lista de IP's associados a cada porta do activo de rede e que estão marcados na base de dados FOG como destinatários para a instalação de imagem de software:

```
$sql = "SELECT hosts.hostIP, hosts.hostMAC,BWTCportmac.portID FROM
tasks,hosts,images,BWTCportmac WHERE tasks.taskHostID=hosts.hostID
AND hosts.hostImage=images.imageID AND tasks.taskState in (0,1)
AND BWTCportmac.hostMAChex=hosts.hostMAC AND
BWTCportmac.portID='$k'";

$res = mysql_query( $sql, $conn ) or die( mysql_error() );
```

Depois da execução do código anterior, os dados obtidos são introduzidos na tabela BWTCSameAs (representada na Figura 43 e na Figura 44). Quando a tarefa de instalação da imagem de software termina, o sistema operativo embebido a correr na memória RAM do computador cliente informa o servidor FOG de que esta terminou (tasks.taskState passa a 2). Assim, como o BWTC_ARP corre periodicamente ao longo do tempo, garante-se que a

largura de banda que ficou disponível é redistribuída pelos restantes IP's associados à mesma porta.

Uma situação habitual numa rede de dados é existir um HUB ou um SWITCH sem capacidades de gestão e/ou de fraca capacidade de processamento, ligados por *uplink* a um SWITCH com gestão e de elevada capacidade de processamento. Esta solução é economicamente rentável e na prática bastante funcional. No contexto da modelação do débito do servidor de imagens em função do estado da rede, os activos de rede sem gestão são um problema porque não permitem localizar físicamente os computadores clientes. O BWTC_ARP permite ultrapassar este problema, uma vez que identifica quais os computadores em comunicação com todas as portas de um *switch* com gestão (que suporte a Bridge MIB). A título de exemplo: um *switch* sem gestão, ligado pela sua porta de *uplink* à porta 10 de um *switch* com gestão SNMP, MIB II e Bridge MIB; o BWTC_ARP identifica todos os computadores (pelo endereço MAC obtém o endereço IP atribuído por DHCP) associados à porta 10. Posteriormente a informação recolhida pelo BWTC_ARP é utilizada pelo BWTC_CALC.

7.2. BWTC_CALC – *BANDWIDTH / TRAFFIC CONTROL CALCULATE*

O módulo mais importante do sistema BWTC é o BWTC_CALC. Este módulo é responsável por modelar o débito do servidor de imagens de software, em função da utilização da rede de dados. Funciona em plena cooperação com o módulo de software BWTC_ARP, pois é este que fornece a listagem de computadores (IP e MAC) que se encontram em comunicação com cada porta do activo de rede (com gestão), e que estão marcados na base de dados de gestão para o processo de instalação de imagens de software. Na Figura 45 está representado o algoritmo do software BWTC_CALC. Neste algoritmo identifica-se as principais fases de operação deste módulo: 1) Obter as estatísticas de tráfego de todas as portas do activo de rede (iterações 1 a 14 da Figura 45); 2) Obter a lista de computadores (IP e MAC) associados a cada porta do activo de rede, e que estão marcados na base de dados de gestão para o processo de instalação de imagens de software (ler da tabela BWTCSameAs) (iterações 15 a 18 da Figura 45); 3) Aplicar o controlo do débito do servidor para cada uma das portas do activo de rede, de forma a garantir que não é ultrapassado o valor de *ceil* (iterações 19 a 31 da Figura 45).

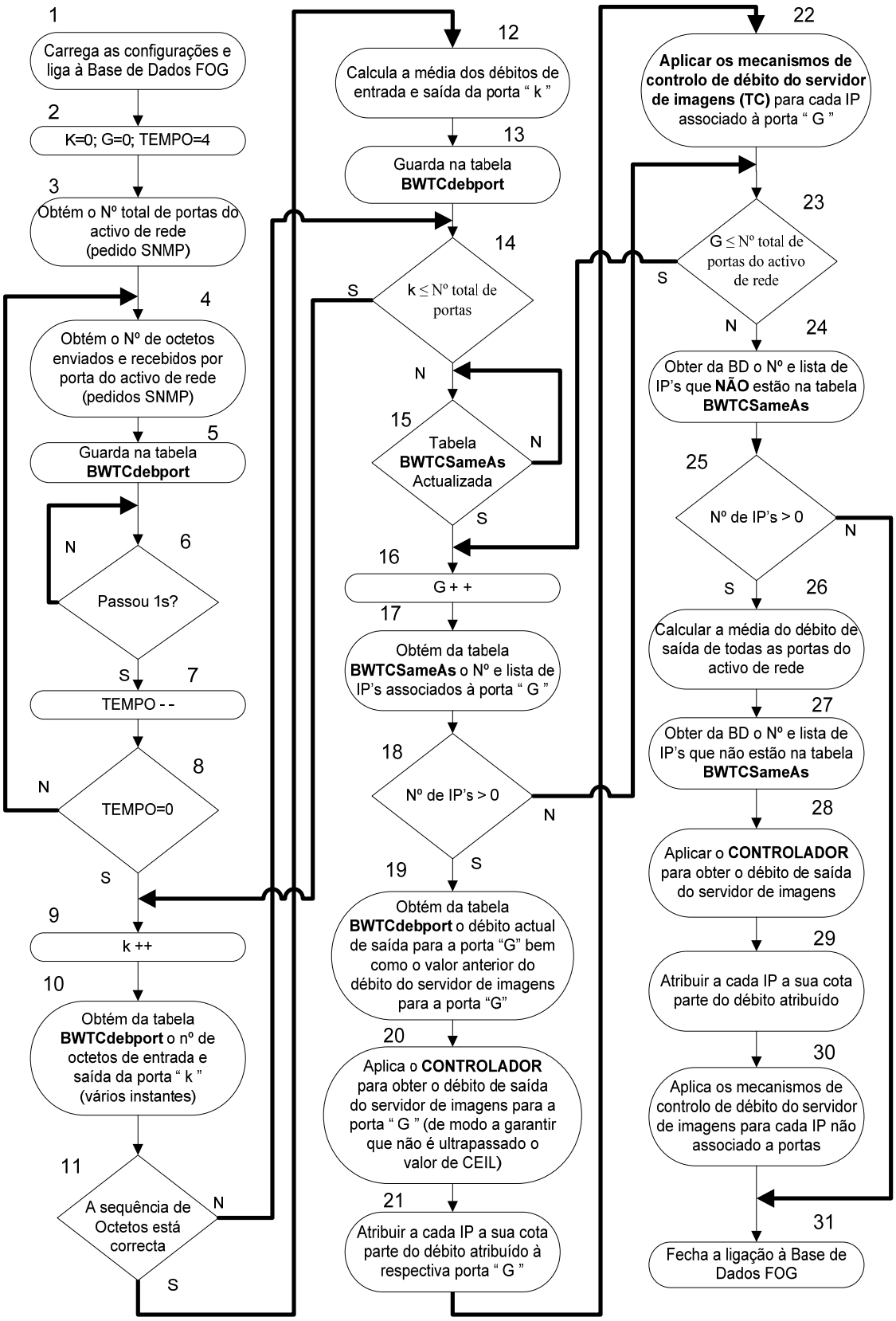

Figura 45: Algoritmo do BWTC_CALC

Como está ilustrado na Figura 46, o BWTC_CALC interage com a rede de dados, com a base de dados FOG e com o BWTC_ARP. O BWTC_CALC obtém informação dos activos de rede (com suporte para o protocolo SNMP), através de pedidos SNMP.

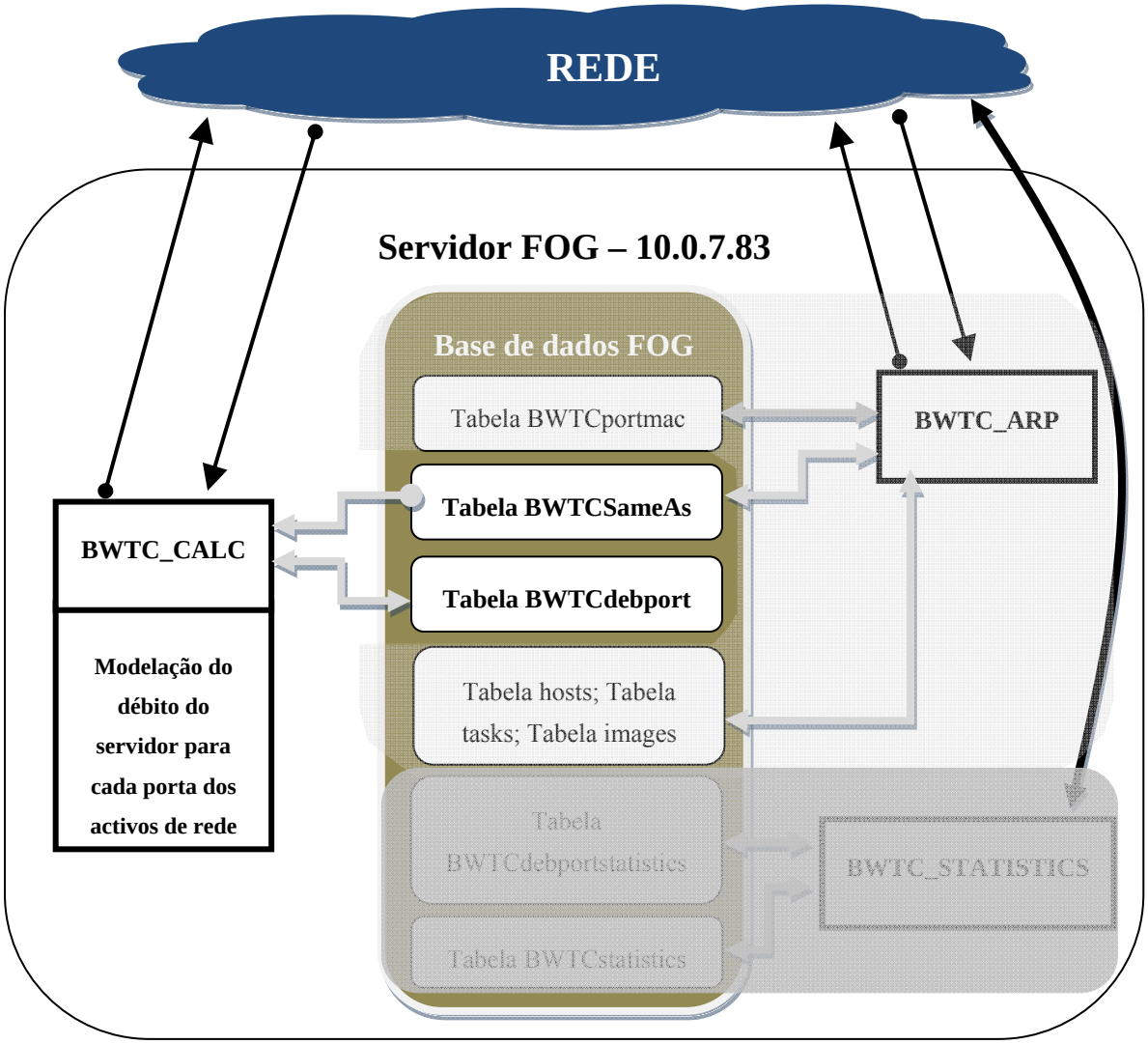

Figura 46: Interacção entre o BWTC_ARP, a base de dados e a rede

Características técnicas e funcionais do BWTC_CALC:

Na Figura 47 está representado um cenário simples de instalação de imagens de software a decorrer numa rede de dados, numa altura em que tantos os docentes como os alunos se encontram a aceder à internet. Num cenário de instalação de imagens sem controlo do débito de dados do servidor de imagens é fácil ocorrerem situações de congestionamento de tráfego nas intersecções de rede (a encarnado), basta para isso que cada um dos computadores clientes faça o *download* da imagem de software à taxa de 50Mbps, perfazendo os 100Mbps da rede de dados.

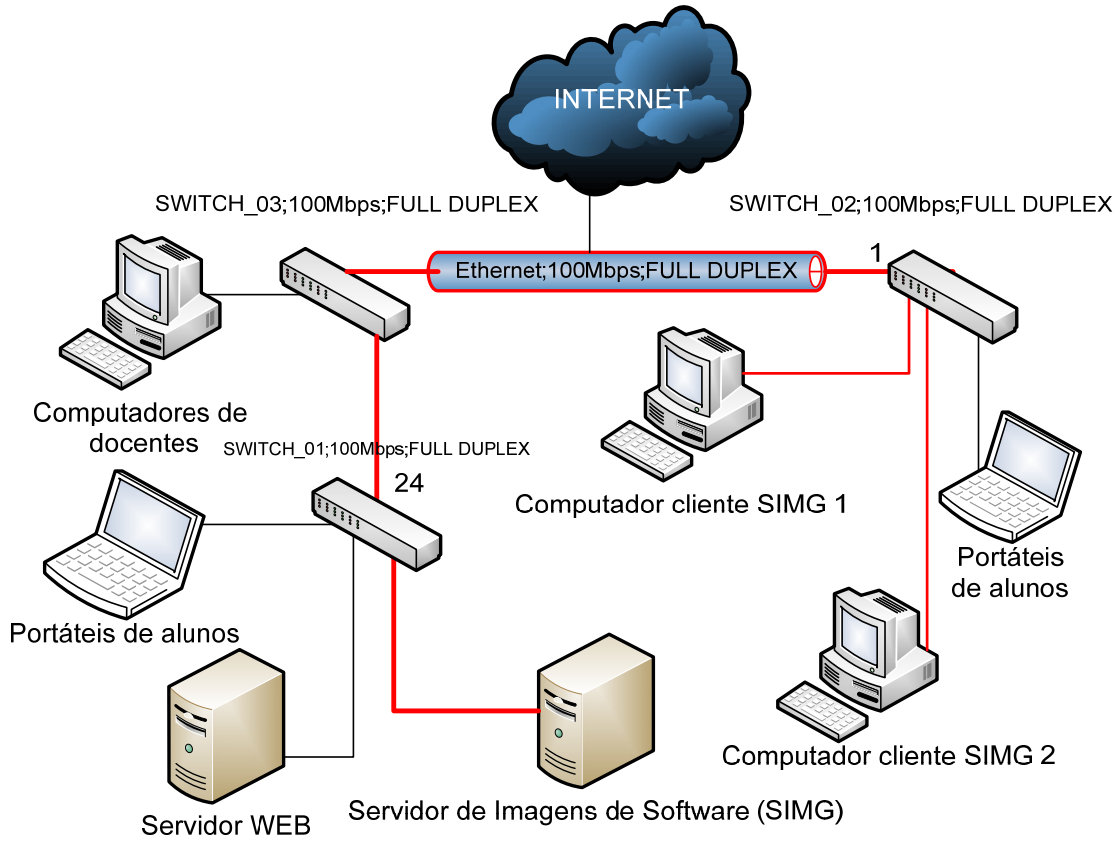

Figura 47: Cenário de instalação de imagens de software (exemplo 1)

Neste caso, tanto os docentes como os alunos iriam ter dificuldades no acesso à internet, uma vez que o caminho dos dados do servidor de imagem é o mesmo que permite o acesso à internet. Com o módulo de software BWTC_CALC pretende-se minimizar este problema. O BWTC_CALC trabalha à volta do valor de *ceil* definido. O valor *ceil* é o valor máximo da carga de tráfego em cada porta dos activos de rede, definido pelo gestor da rede. Ou seja, nas intersecções de rede, o tráfego proveniente do servidor de imagens (SIMG) mais o tráfego resultante dos restantes serviços de rede (tais como o acesso à internet e o acesso ao servidor Web) não deve ultrapassar o valor de *ceil*. Para que isto se verifique o débito do servidor de imagens é modelado em função da rede de dados, ao longo do tempo.

1) Obtenção das estatísticas de tráfego do activo de rede:

Para obter as estatísticas de tráfego nas portas dos activos de rede é condição necessária que os activos de rede suportem o protocolo SNMP e a MIB II. O agente SNMP que corre no activo de rede actualiza a MIB II com vária informação, tal como: o número de octetos enviados (ifOutOctets) e recebidos (ifInOctets) em cada porta do activo de rede

(disponíveis pelos OID subsequentes a (01)iso(03)org(06)dod(02)mgmt(01)mib-2(02)interfaces(02)ifTable); o número de portas do activo de rede (disponíveis no OID (01)iso(03)org(06)dod(02)mgmt(01)mib-2(02)interfaces(01)ifNumber). A MIB II não contém o valor do débito de dados, é necessário calculá-lo de forma indirecta. Para isso é necessário fazer no mínimo duas leituras consecutivas do número de octetos enviados e recebidos num determinado intervalo de tempo. Depois de vários testes, optou-se por realizar 4 medições dos valores de ifInOctets e ifOutOctets, com um intervalo de tempo de 1 segundo entre estas.

Em seguida é apresentada uma pequena amostra do código fonte do módulo BWTC_CALC, correspondente ao processo de obtenção do número de octetos enviados em cada porta de um activo de rede, e correspondente armazenamento na base de dados BWTCdebport. Na amostra fazem-se leituras do activo de rede com o IP "10.0.7.245" e senha SNMP "deg". Foram retiradas as linhas de código correspondentes à obtenção e armazenamento do número de octetos de entrada em todas as portas do activo de rede, por serem análogas às existentes.

```
/* Obtenção, tratamento e armazenamento do número de octetos enviados em todas as portas do activo de rede (4 medições desfasadas de 1s entre elas) */

$tempo=4;//4s - 4 medições - 3 variações

$j=1;

$k=1;//número da leitura

/* Obter ifOutOctets de todas as portas do activo de rede */

echo "<br />Obter ifOutOctets do Activo de Rede<br />";
while($j<=$tempo)//Início do ciclo "while – tempo"
{
 //(1)Regista o instante de tempo em que se começa a medição
 $time_start=microtime(true);//tempo em micro segundos

 //(2)Obter o número de octetos enviados em todas as portas
 $b6 = snmpwalkoid("10.0.7.245", "deg", "ifOutOctets");

 //(3)Tratar e inserir os dados na tabela BWTCdebport
 for (reset($b6); $i = key($b6); next($b6) ){
   //(4)Tratar os dados obtidos do activo de rede
   $port = substr(strrchr($i, "."), 1);//guarda a porta
   $port=(int)$port;
   $count_out = substr(strrchr($b6[$i], ":"),2);//valor do número
de octetos de saída
```

```
 //(5)Insere o ifOutOctets da porta $port na tabela BWTCdebport
   if($k <2 && $state<1){//(6)A tabela BWTCdebport está vazia
      $old=1; //insere em tds as portas e indicação de que tem
valores antigos
      $query="INSERT INTO BWTCdebport(portID,count_out_n_$j,state)
value('$port','$count_out','$old') ";
      if ( mysql_query( $query, $conn ) ){
           //echo "INSERIU $k+1 vezes!<br /><br />";
      }
       else{echo "ERRO nao adicionou<br />";}
   }

   if($k>1 || $state>0){//(7)A tabela BWTCdebport já foi
utilizada, insere os novos dados
      $query="UPDATE BWTCdebport SET count_out_n_$j='$count_out'
WHERE portID='$port' ";
       if ( mysql_query( $query, $conn ) ){
           //echo "Adicionou a BD de Calculo!<br /><br />";
       }
       else{echo "ERRO nao adicionou<br />";}
   }

 }//fim ciclo “for – b6”

//(8)Regista o instante de tempo em que se termina a medição
$time_end = microtime(true);//retorna no formato 1.6 (por ex.
0.352348 segundos)
$time = $time_end - $time_start;//calcula o tempo que demorou a
obter o numero de octetos
//echo "Demorou $time us a realizar a leitura $j<br />";

//(9)Calcular o tempo que falta para perfazer 1s
$hold=1-$time;
//echo "O ciclo $j demorou $time a ler octetos e guardar na BD-
>tem de espera:$hold para fazer 1s<br />";

$hold=$hold*1000000;
//(10)Esperar o tempo necessário para perfazer um segundo
usleep($hold);//por ex. para esperar 0.700000 segundo, $hold
deveria ser 700000
$k++;$j++;

}//Fim ciclo “while - tempo”
```

O extracto de código anterior corresponde aos pontos 4 a 8 do algoritmo BWTC_CALC, ilustrado na Figura 45.

No extracto de código anterior começa-se por registar o instante de tempo em que começa a medição (1), depois obtém-se o número de octetos enviados para todas as portas do activo de rede (2). Trata-se a informação obtida em (3 e 4), e guarda-se na tabela BWTCdebport o número de octetos (ifOutOctets) para cada porta do activo de rede (5). Como foi referido anteriormente, na tabela BWTCdebport são registadas 4 leituras consecutivas do número de octetos para cada porta do activo de rede, para posteriormente fazer uma média aritmética. Assim, como o *script* funciona em ciclo, para distinguir as diferentes portas e medições são aplicados filtros (6 e 7). Para garantir que o tempo entre cada medição é de 1 segundo, regista-se o instante de tempo em que termina o processo de obtenção e armazenamento do número de octetos enviados (8). Finalmente, calcula-se o tempo que falta para perfazer 1 segundo (9) e no párasse a execução do *script* durante esse tempo (10).

Como se encontra ilustrado nas iterações 5 e 13 do algoritmo BWTC_CALC (Figura 45), tanto o número de octetos como os débitos de entrada e saída de dados para cada porta são guardados na tabela BWTCdebport. As quatro medições efectuadas no extracto de código anteriore resultam em três débitos de dados para cada um destes. Com os três débitos é feita uma média aritmética, o que permite atenuar as variações bruscas na rede de dados. Funciona como um filtro passa baixo, uniformizando as medições obtidas.

Foram realizados vários testes e pôde-se constatar que o programa BWTC_CALC demora aproximadamente 0,3 segundos a efectuar a leitura do número de octetos de entrada e saída de cada porta do activo de rede e armazenar a informação na base de dados. Portanto são gastos aproximadamente 0,7 segundos com a função *usleep*.

2) Obtenção da lista de computadores (IP e MAC) associados a cada porta do activo de rede, e que estão marcados na base de dados do FOG para o processo de instalação de software:

O BWTC_ARP actualiza periodicamente a tabela BWTCSameAs com a lista de IP <–> MAC dos computadores para os quais o activo de rede encaminhou dados, para cada porta. Em seguida o BWTC_CALC utiliza essa informação para modelar o débito de dados para cada porta do activo de rede. Através do processo de sincronização que fora desenvolvido o BWTC_CALC não utilizada a informação da tabela BWTCSameAs enquanto o BWTC_ARP estiver a actualizá-la.

No algoritmo BWTC_CALC o processo de obtenção da lista IP <–> MAC está representado nos pontos 15 a 18 da Figura 45.

3) Aplicar o controlo do débito do servidor para cada porta do activo de rede:

Como está ilustrado nos pontos 19 a 23 da Figura 45, o algoritmo do BWTC_CALC leva em consideração não só o débito de saída de dados de cada porta do activo de rede, mas também o número de computadores que lhe estão associados (e que estão integrados no processo de instalação de imagens de software). O BWTC_CALC obtém esta informação a partir das tabelas BWTCSameAs e BWTCdebport (consultar a Figura 46) e realiza a modelação do controlo de débito do servidor de imagens.

Existem vários métodos de aplicar o controlo de tráfego. Nas publicações científicas "*ABR Traffic Control over ATM Network Using Fuzzy Immune-PID Controller"* [56], "*Implementation and PID tuning of network-based control systems via Profibus polling network"* [57] e "*A Queue Management Scheme for Improving Network Resilience Under worm Attacks"* [58] são apresentadas várias implementações de controladores PID em aplicações de controlo de tráfego. No artigo cientifico "*Approaches to Congestion Control in Packet Networks"* [59] são estudados vários algoritmos de controlo de tráfego, como por exemplo o *Additive Increase and Multiplicative Decrease with Fast Convergence* (AIMD-FC), ou o *Binary Increase Congestion Control Protocol* (BIC-TCP), no entanto é necessário fazer um estudo muito aprofundado da rede de dados para que se possa utilizar estes algoritmos. Para utilizar os algoritmos apresentados em [59] seria necessário fazer um estudo pormenorizado da rede do DEG, uma vez que são utilizados parâmetros como a eficiência (*efficiency*), a equidade (*fairness*), a velocidade de convergência (*convergence speed*), a uniformidade/suavidade (*smoothness*) e receptividade (*responsiveness*) de uma rede de dados.

A eficiência é o valor médio de *flows* que se encontram a efectuar trocas de informação [59].

A equidade de uma rede descreve a capacidade de distribuir, de forma justa, os recursos pelos *flows*, numa situação de congestionamento de tráfego [59].

A velocidade de convergência descreve o tempo decorrido até ser atingido a estado de equilíbrio da rede [59].

A suavidade reflecte a amplitude da oscilação do tráfego na rede durante a fase de "*multiplicative-decrease*". O "*multiplicative-decrease*" faz parte do mecanismo de controlo de congestionamento *Additive Increase and Multiplicative Decrease* (AIMD), existente no protocolo TCP. Com o AIMD o emissor aumenta de forma aditiva a taxa de transmissão de dados, mas diminui a taxa de transmissão de forma multiplicativa (pelo factor 1/2) quando detecta uma situação de congestionamento.

A receptividade descreve o número de etapas necessárias, ou *Round-Trip Times* (RTTs), para chegar ao estado de equilíbrio. A diferença entre a receptividade e a convergência é que a primeira está relacionada com um fluxo de informação (*flow*) individual, a segunda está relacionada com todo o sistema.

Apesar de existirem outros algoritmos que provavelmente levariam a melhores resultados, com o tempo que se dispunha optou-se por utilizar um controlador proporcional, em malha fechada, com memória. O controlador que está ilustrado na Figura 48 e na Figura 49 é proporcional porque das componentes de um controlador Proporcional, Integrativo e Derivativo somente se aplica a proporcional, é em malha fechada porque corre periodicamente, em torno do valor de *ceil*. Tem memória porque no cálculo do débito de saída do servidor de imagens para cada porta do activo de rede no instante *n*, utiliza os valores calculados para a iteração *n*-1, e que tinham sido guardados na base de dados.

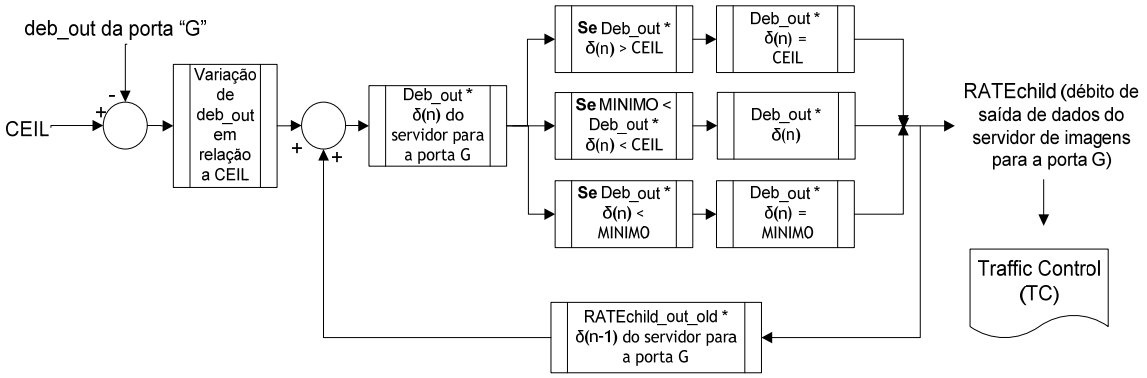

Figura 48: Controlador do débito de saída do servidor de imagens

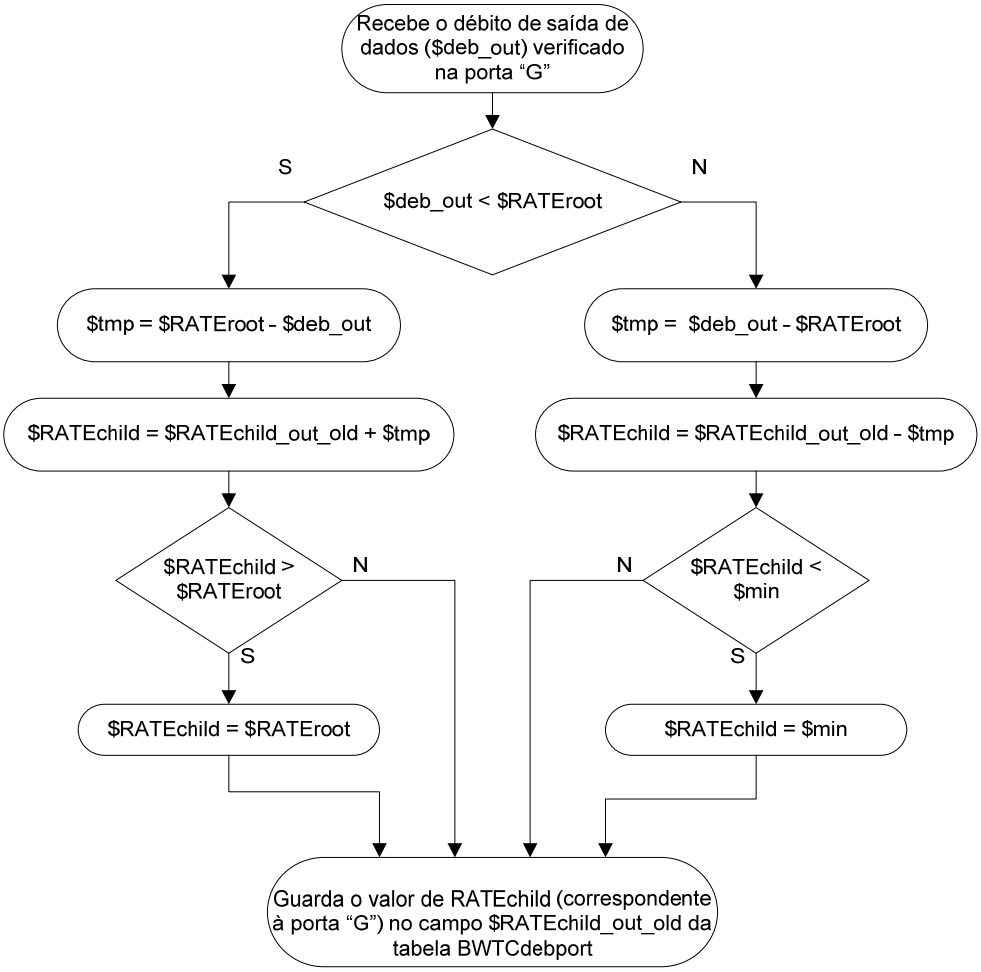

Figura 49: Algoritmo do controlador integrado no BWTC_CALC

Significado das variáveis do algoritmo de controlo do BWTC_CALC:

- **$deb_out** - débito de dados da porta "G" (instante actual);
- **$RATEroot** - valor de *ceil* definido (o valor do débito de saída da porta "G", não deve ultrapassar o valor de *ceil* definido pelo gestor);
- **$RATEchild** - valor calculado para o débito de dados de saída do servidor de imagens para a porta "G" (instante actual ou *n*);
- **$RATEchild_out_old** - valor calculado para o débito de dados de saída do servidor de imagens para a porta "G" (instante anterior ou *n*-1);
- **$tmp** - variável temporária;
- **$cnt** - nº de IP's associados à porta "G";
- **$min** - valor mínimo do débito do servidor de imagens para todos os IP's associados à porta "G", para que a ligação a cada computador não termine por "time out" (**$min = $cnt * 50KB**);

O facto do controlador ter memória é muito importante, porque o débito de saída de dados de uma qualquer porta "G" do activo de rede no instante *n* pode incluir o tráfego introduzido pelo servidor de imagens no instante *n*-1.

Optou-se por começar com um controlador do tipo proporcional e se os resultados não fossem satisfatórios partir para as abordagens PD, PI ou PID. O controlador é uma das peças chave da modelação do débito de dados, pois é este que procura manter a carga de tráfego em cada porta do activo de rede perto do valor de *ceil* definido.

Para se perceber qual é a actuação do controlador apresentado na Figura 49 suponham-se alguns cenários baseados na Figura 50, em que os computadores 1 e 2 estão inseridos no processo de instalação de imagens de software:

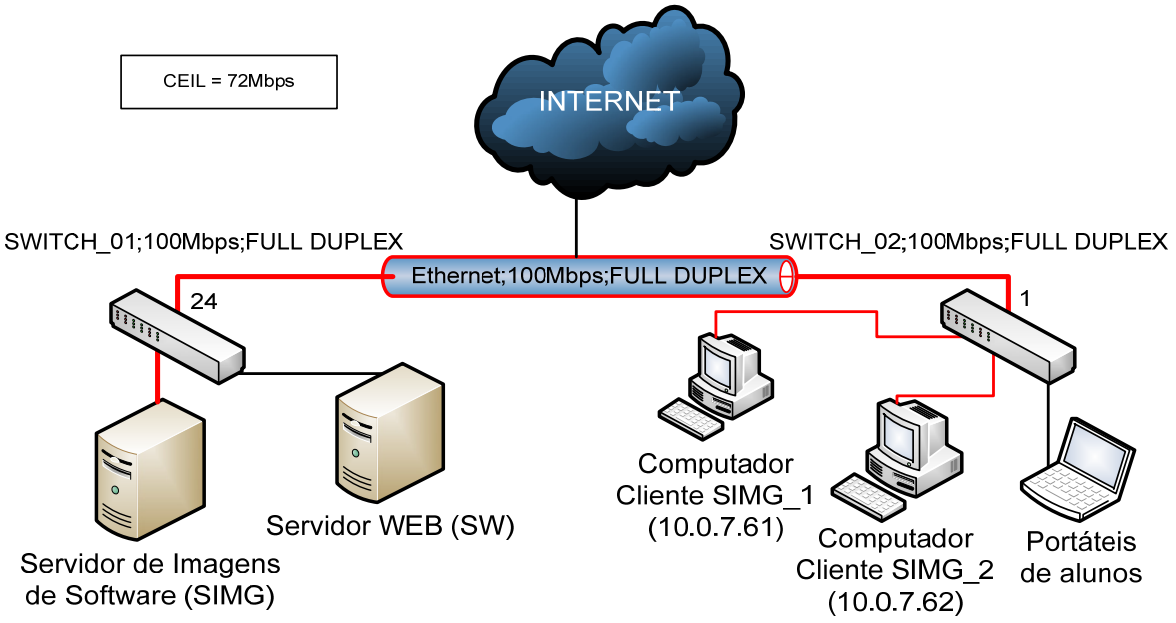

Figura 50: Cenário de instalação de imagens de software (exemplo 2)

Cenário 1: O valor de *ceil* é 72Mbps. Em qualquer instante o tráfego de saída de dados da porta 24 do SWICTH_01 tem duas fontes, o servidor de imagens e o servidor Web.

No instante *n* o débito do SW é de 22Mbps e o débito do SIMG é de 50Mbps, o que perfaz 72Mbps. No instante *n*+1 o débito do SW passa de 22Mbps para 32Mbps. Então o controlador diminui o débito de dados do servidor de imagens com destino à referida porta, neste caso de 50Mbps para 40Mbps, perfazendo os 72Mbps;

Cenário 2: O valor de *ceil* é 72Mbps. Em qualquer instante o tráfego de saída de dados da porta 24 do SWICTH_01 tem duas fontes, o servidor de imagens e o servidor Web.

No instante *n* o débito do SW é de 22Mbps e o débito do SIMG é de 50Mbps, o que perfaz 72Mbps. No instante *n*+1 o débito do SW passa de 22Mbps para 12Mbps. Então o controlador aumenta o débito de dados do servidor de imagens com destino à referida porta, neste caso de 50Mbps para 60Mbps, perfazendo os 72Mbps;

Cenário 3: Mesmo que o tráfego existente na porta 24 seja só o proveniente do servidor de imagens, então este não deve ultrapassar o valor de *ceil* (por exemplo 72Mbps), de modo a garantir largura de banda disponível para aplicações que entretanto entrem em operação;

Cenário 4: Caso o débito de dados na porta 24 do SWITCH_01, excluindo os dados provenientes do servidor de imagens, seja superior ao valor de *ceil*, é definido o valor mínimo de 50kBps para cada computador. Deste modo o processo de instalação de imagens não termina por perda de ligação ao servidor de imagens ou *time out*.

Em qualquer um dos cenários anteriores, depois do controlador calcular o valor do débito de dados do servidor de imagens para a porta 24 do SWITCH_01, este valor é dividido pelo número de clientes em comunicação com a referida porta e que se encontram registados no processo de instalação de imagens (registados na tabela BWTCSameAs). No caso da Figura 47 são os computadores clientes 1 e 2. Foram dados vários exemplos para a porta 24 do SWICTH_01 e computadores 1 e 2, mas o BWTC_CALC funciona de forma automática e de forma análoga para qualquer porta do activo de rede, qualquer que seja o número de computadores associados à porta do activo de rede.

Finalmente, para aplicar a modelação do débito de saída do servidor de imagens para cada IP recorre-se à ferramenta de software *Linux Traffic Control* (TC). No caso de estudo é necessário limitar o tráfego de saída do servidor em função do IP de destino. Para isto é necessário utilizar o *tc* a operar com uma disciplina que permita escalonar o envio dos pacotes de dados (*scheduler*), e deste modo limitar o débito de dados. Ao mesmo tempo deve possuir filtros para encaminhar os pacotes de dados para as respectivas classes. Neste contexto, as disciplinas mais utilizadas são o CBQ e o HTB. O CBT foi introduzido no kernel 2.2, é muito complexo, com inúmeros parâmetros de configuração [47][65]. O HTB foi introduzido no kernel 2.4 [47], é o substituto do CBQ, é mais simples, com uma configuração mais acessível [65]. O extracto de código apresentado a seguir é um exemplo

de implementação do controlo do débito de dados no cenário 3. Neste extracto de código faz-se o controlo de débito da interface "eth0" do SIMG (Figura 50) para os computadores clientes SIMG_1 (com o IP 10.0.7.61) e SIMG_2 (com o IP 10.0.7.62).

```
1) tc qdisc add dev eth0 root handle 1: htb default 30
2) tc class add dev eth0 parent 1: classid 1:1 htb rate 72mbit
burst 922k
3) tc class add dev eth0 parent 1:1 classid 1:11 htb rate 36Mbit
burst 922k
4) tc class add dev eth0 parent 1:1 classid 1:12 htb rate 36Mbit
ceil 72Mbit burst 922k
5) tc qdisc add dev eth0 parent 1:11 handle 11: sfq perturb 10
6) tc qdisc add dev eth0 parent 1:12 handle 12: sfq perturb 10
7) tc filter add dev eth0 protocol ip parent 1:0 prio 1 u32 match
ip dst 10.0.7.61 flowid 1:11
8) tc filter add dev eth0 protocol ip parent 1:0 prio 1 u32 match
ip dst 10.0.7.62 flowid 1:12
```

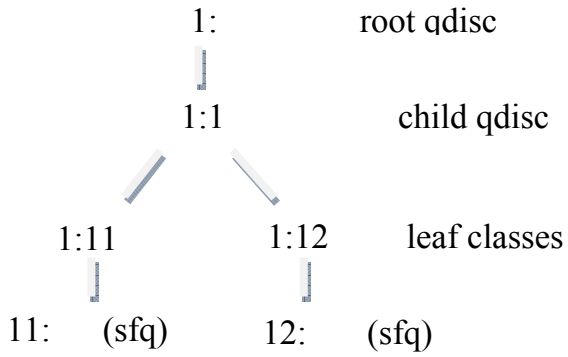

Figura 51: Árvore de dependências do *qdisc*

A Figura 51 representa a árvore da *root qdisc* (*qdisc* de saída de dados) que é criada depois de executado o extracto de código anterior. Em **1)** definiu-se que a estrutura anexada ao *root qdisc* do interface de rede "eth0" é o HTB. Em **2)** introduziu-se uma classe à *root qdisc*, nesta define-se a taxa de transferência máxima pretendida (*rate*) que é 72Mbps. O valor de *burst* é o tamanho do *bucket* (*buffer*) de transmissão, e está definido na documentação [55] como sendo aproximadamente 80 vezes mais pequeno que a taxa de transmissão que se pretende. Em **3)** e **4)** são definidas as classes filhas com os respectivos parâmetros. Em **5)** e **6)** são associados a cada uma das classes **3)** e **4)** respectivamente, o mecanismo de controlo de tráfego SFQ. Este é aconselhado pelos autores de [55] para garantir o limite superior das respectivas classes. Em **7)** e **8)** são definidos os filtros, estes garantem que os pacotes de dados que vão para a *root qdisc* (1: é equivalente a 1:0) são

classificados e encaminhados para as respectivas classes. Estes filtros utilizam os seguintes critérios: só aceitam pacotes provenientes do protocolo IP (camadas superiores da pilha protocolar), e encaminham para as respectivas classes os pacotes cujo IP de destino é o definido. Por exemplo, no script anterior o *tc* é configurado para encaminhar os pacotes de dados que têm como destino o IP 10.0.7.61 (*flowid* 1:11) para a classe com *classid* 1:11, e nesta o débito máximo permitido é de 36Mbps. Os filtros são as estruturas mais complexas do processo de controlo de tráfego mas, na óptica do programador, são muito flexíveis. Os filtros podem funcionar em cooperação com a firewall do Linux (*ipchains* ou *iptables*) [64].

Período de funcionamento do BWTC:

Um parâmetro muito importante para o controlo de débito do servidor de imagens é a periodicidade de operação dos vários módulos de software que constituem o BWTC. O período de operação do módulo BWTC_CALC serve de referência aos outros módulos, uma vez que neste são realizadas as operações de modelação do débito do servidor de imagens, e como tal tem influência directa na comunicação deste com os computadores clientes. Existe um *delay* entre o instante em que é alterado o débito de saída do servidor de imagens e o instante em que os computadores clientes passam à nova taxa de *download*. Assim, o período de execução do módulo BWTC_CALC deve ser tal que permita aos computadores clientes actualizarem a sua taxa de *download* da imagem de software do servidor, entre cada iteração do BWTC_CALC. Por exemplo, na iteração *n* o BWTC_CALC define que a taxa máxima de *download* do cliente A a partir do servidor de imagens é de 20Mbps, na iteração *n*+1 define que a taxa máxima de *download* do mesmo cliente passa de 20Mbps para 70Mbps. Na prática verifica-se que a actualização por parte do cliente A para a nova taxa de *download* não é instantânea.

A comunicação entre o servidor de imagens (emissor) e os clientes (receptores) funciona por TCP/IP, pelo que existem dois conceitos a ter em consideração: o protocolo de comunicação TCP não possui ferramentas de QoS; o TCP tem mecanismos de controlo de fluxo e controlo de congestionamento [60] [22], se bem que deficitários, uma vez que só funcionam quando já se estão a perder dados na rede. O controlo de fluxo evita que o emissor congestione o receptor (método da janela deslizante), ao passo que o controlo de congestionamento evita que o emissor congestione a rede (método da janela de congestionamento).

De forma muito simples, no método da janela deslizante o receptor vai "informando" o emissor (com os ACK's) do estado da janela de recepção (*buffer* de recepção), e o emissor envia os dados enquanto "couberem" na janela do receptor.

O método da janela de congestionamento, representado na Figura 52, funciona do seguinte modo: começa a transmissão de dados do emissor para o receptor, a uma taxa que aumenta exponencialmente (duplica a cada iteração) até ser atingido o valor de *threshold*; atingido o valor de *threshold* a taxa de transmissão passa a aumentar linearmente; na transmissão 7 ocorre *timeout* (que pode significar congestionamento na rede); o valor de *threshold* passa para metade do valor da janela de congestionamento e esta é reinicializada. Na transmissão 11 é atingindo o novo valor de *threshold* e, caso não ocorra novamente *timeout*, a janela de congestionamento cresce linearmente até ao tamanho da janela deslizante do receptor. Ou seja, existe uma certa inércia associada ao processo de actualização da taxa de transmissão de dados do servidor para os clientes.

O BWTC_CALC modela o débito de dados do servidor de imagens em função da taxa de utilização da rede de dados. Neste contexto o processo de comunicação entre o servidor de imagens (emissor) e os clientes (receptores), é directamente influenciado pelos mecanismos de controlo de fluxo e congestionamento do protocolo TCP. Assim, o período de execução do BWTC_CALC deve ser tal que permita aos computadores clientes actualizarem para a nova taxa de *download*, antes do BWTC_CALC definir uma taxa de *download* diferente para os respectivos clientes.

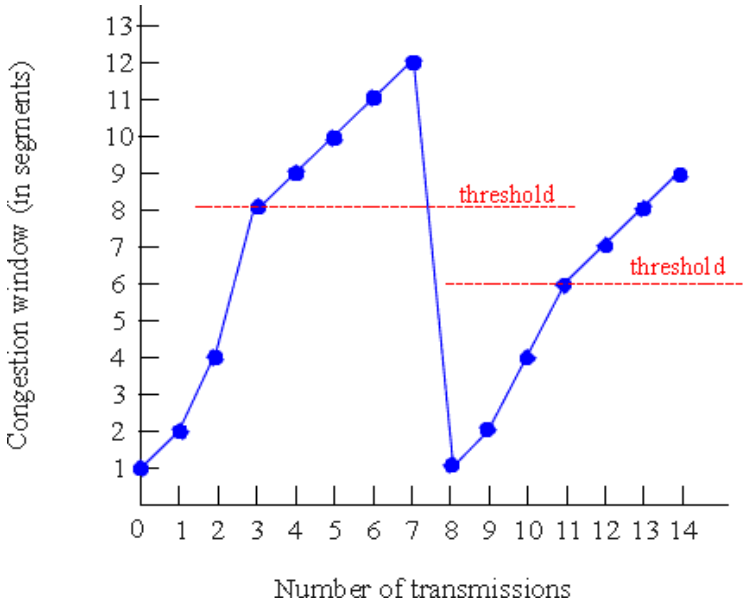

Figura 52: Janela de congestionamento do protocolo TCP

Para definir o valor do período de funcionamento do BWTC_CALC recorreu-se à seguinte análise prática:

1) Os *switch* HP Procurve 2510-24 incluem uma aplicação embebida, acessível por interface Web, que apresenta gráficos da percentagem de utilização de todas as portas do activo de rede;

2) Os computadores clientes 01 e 02 da Figura 53 acedem aos SF e SW respectivamente, aumentando a tráfego na porta 24 do SWITCH_01;

3) Em seguida introduziu-se o tráfego do SIMG para os respectivos clientes, com modulação do débito do SIMG (para a taxa máxima de utilização da porta 24 na ordem dos 72%);

4) Realizaram-se vários testes para diferentes peridos de funcionamento do BWTC_CALC.

Com base nos testes efectuados pode-se afirmar que para valores baixos do período de funcionamento do BWTC_CALC, na ordem dos 5 segundos, o tráfego na porta que apresentava congestionamento de tráfego (porta 24 do SWITCH_01) variava bruscamente. Isto está associado aos processos de controlo de fluxo e controlo de congestionamento do TCP, uma vez que a adaptação por parte dos computadores clientes (03 e 04) para a nova taxa de *download* não é instantânea. Depois de realizar vários testes ao módulo de software BWTC_CALC, o valor em que se verificou o comportamento mais estável em torno do valor 72% é de 10 segundos.

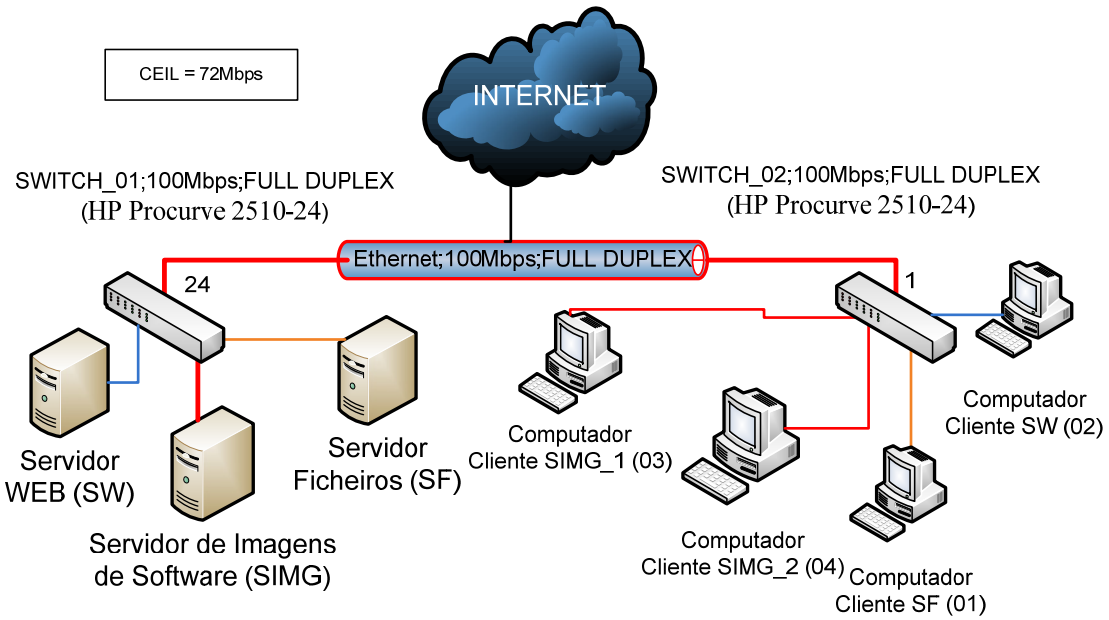

Figura 53: Cenário de testes para obtenção do período de funcionamento do BWTC_CALC

7.3. BWTC_STATISTICS - *BANDWIDTH / TRAFFIC CONTROL STATISTICS*

O módulo de software *Bandwidth-Traffic Control Statistics* faz medições do débito de dados de entrada e saída de cada porta dos activos de rede, em instantes discretos. Na Figura 54 está representada a interacção do BWTC_STATISTICS com a rede de dados e a base de dados FOG.

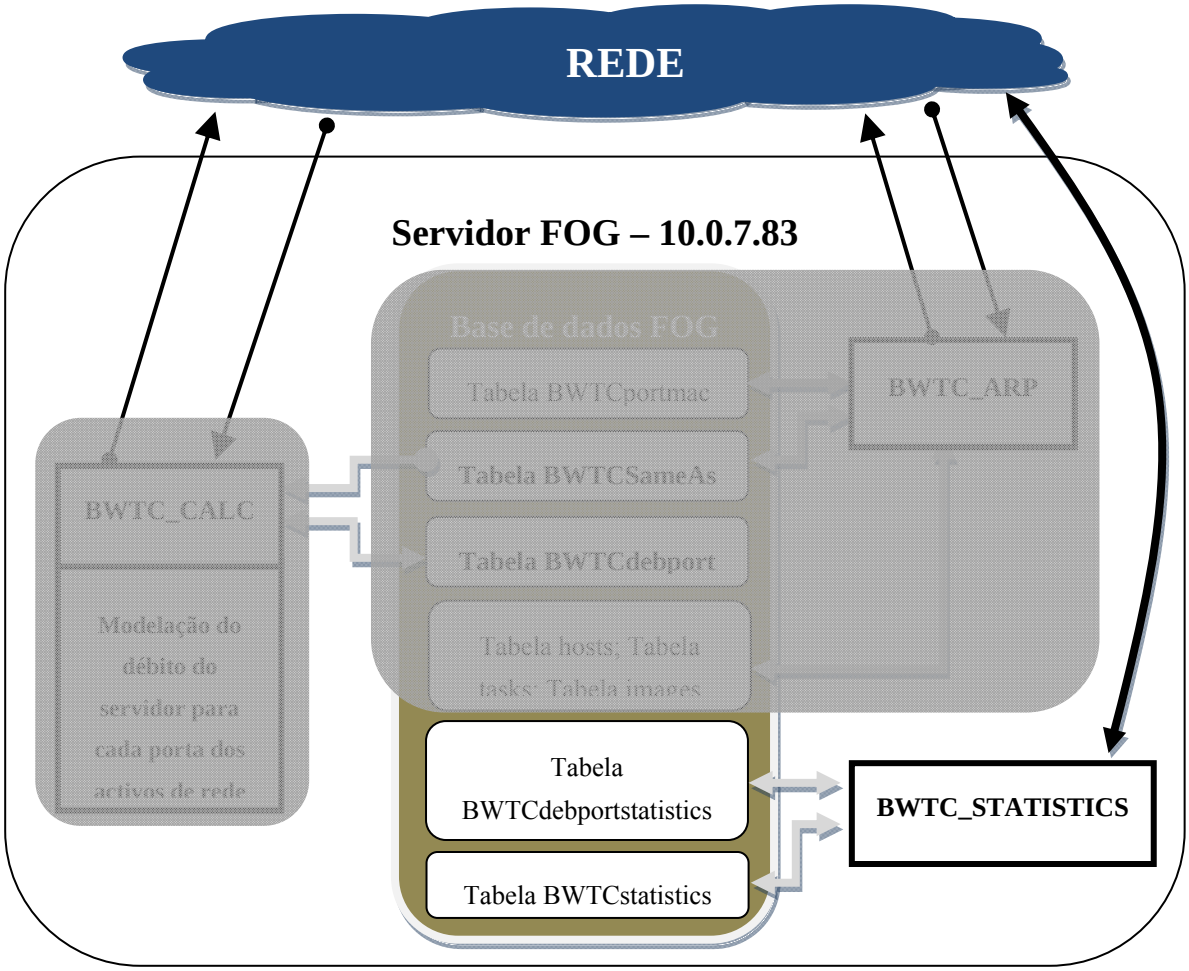

Figura 54: Interacção entre o BWTC_ARP, a base de dados e a rede

Já existem ferramentas que permitem fazer leituras do débito das portas dos activos de rede com gestão SNMP, tal como o MRTG e o Cricket. No entanto grande parte destas ferramentas faz leituras dos activos de rede e fornece a informação do débito de dados na forma de valores médios, na base de tempo de cinco minutos. Com o intuito de obter os débitos de dados em instantes discretos, sem médias, foi desenvolvido o módulo de software BWTC_STATISTICS. O algoritmo do BWTC_STATISTICS está representado na Figura 55.

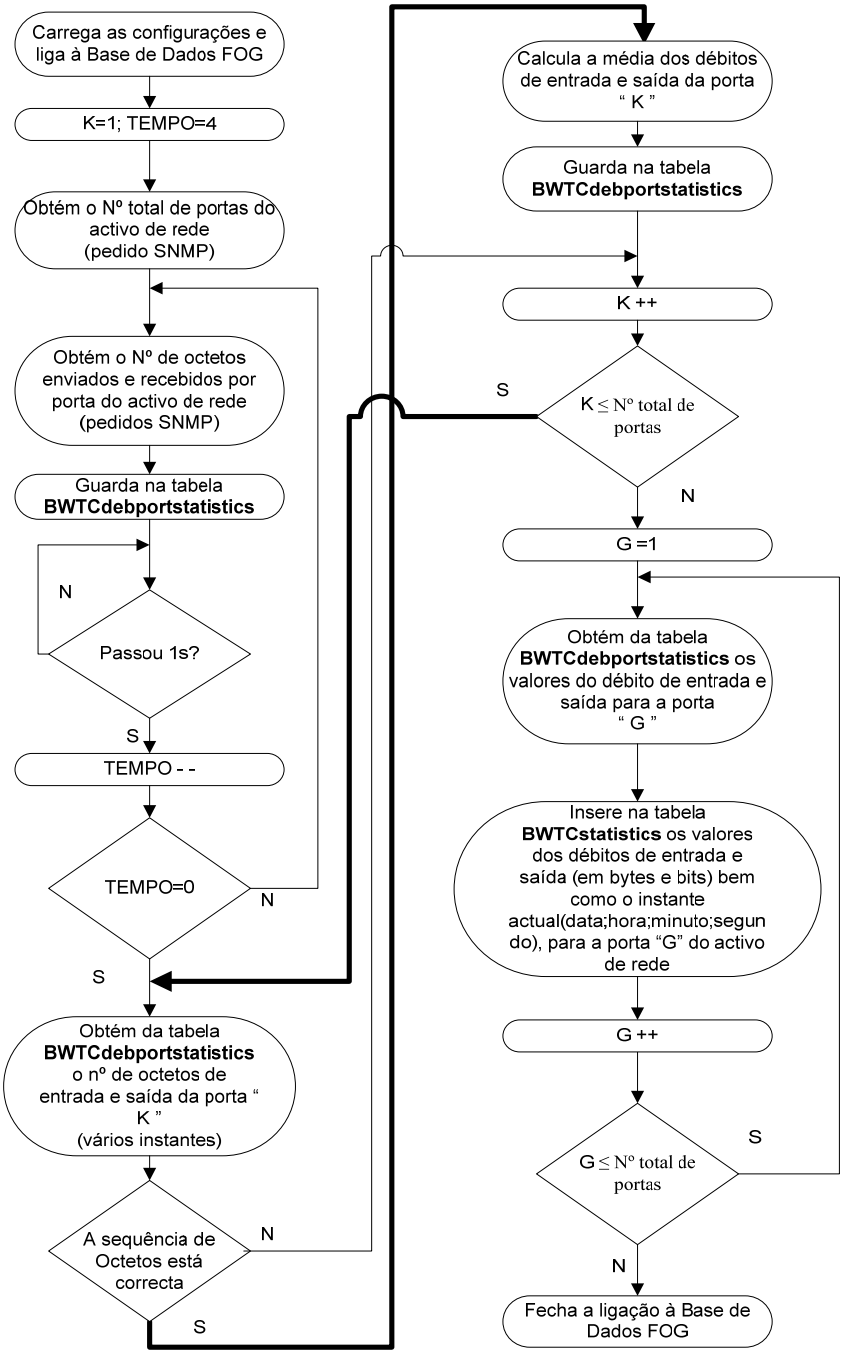

Figura 55: Algoritmo do BWTC_STATISTICS

O software foi projectado para funcionar periodicamente, em intervalos de tempo que podem ser definidos pelo programador. Depois de realizar alguns testes foi utilizado o período de 36 segundo, deste modo evita-se não só as colisões com os restantes módulos do sistema BWTC mas também a saturação dos activos de rede com pedidos SNMP.

Como se pode ver na Figura 55, não são realizadas somente as leituras do número de octetos enviados e recebidos por cada porta do activo de rede, mas também é detectado se

a sequência de octetos está correcta, sendo descartadas as medições erradas. A detecção de sequências de octetos correcta resulta da seguinte constatação: cada porta do *switch* HP 2510-24 tem associados dois contadores de 32 bits (cujo valor se pode obter por SNMP) que contêm o número de octetos enviados e recebidos, o que significa que pode contar até 4Gbytes de informação em ambos os sentidos. A uma taxa de transmissão de 72Mbps o contador de saída de octetos reinicia a cada 7,6 minutos, sendo necessário identificar se a sequência de octetos lida não contém valores de transição.

A operação de acesso à informação da MIB e ao seu armazenamento na base de dados é realizado por um *script* em PHP. Um extracto do código que permite guardar na base de dados o valor dos contadores de octetos enviados e recebidos (ifOutOctets e ifInOctets) de cada porta do activo de rede é apresentado em seguida:

```
/***Guardar o valor de ifOutOctets para cada porta (índice $j)***/

if($k>1 || $state>0){
$query="UPDATE BWTCdebportstatistics SET count_out_n_$j='$count_out'
WHERE portID='$port' ";
                  if ( mysql_query( $query, $conn ) ){
                    //echo "Adicionou a BD do Statistics!<br /><br />";
                  }
                  else{echo "ERRO nao adicionou<br />";}
/*************************************************************/

/***Guardar o valor de ifOutOctets para cada porta (índice $j)*****/

$query="UPDATE BWTCdebportstatistics SET count_in_n_$j='$count_in' WHERE
portID='$port' ";
           if ( mysql_query( $query, $conn ) ){
            //echo "Adicionou a BD do Statistics!<br /><br />";
           }
           else{echo "ERRO nao adicionou<br />";}

/*************************************************************/
```

7.4. RESUMO

Este capítulo descreve o *BandWidth-Traffic Control* (BWTC). O BWTC foi desenvolvido com o intuito de atenuar o impacto negativo da instalação de imagens de software através da rede, em particular o congestionamento de tráfego nas intersecções de rede. O BWTC é constituído por três elementos modulares: o BWTC_ARP, o BWTC_CALC e o BWTC_STATISTICS.

O BWTC_ARP foi desenvolvido para identificar as portas do activo de rede em que cada cliente do servidor de imagens se encontra ligado. O BWTC_ARP acede às MIB's dos activos de rede e recolhe a lista de endereços MAC dos computadores em comunicação com cada porta do activo de rede. Cruza os endereços MAC obtidos com a base de dados de equipamento e obtém os respectivos IP's. Introduz na base de dados a relação IP <-> MAC dos clientes do servidor de imagens, bem como a porta do activo de rede em que se encontra ligado. O BWTC_CALC utiliza a informação introduzida pelo BWTC_ARP nas operações de modelação.

O BWTC_CALC é o módulo mais importante do BWTC. Este módulo é responsável pela modelação do débito do servidor de imagens de software, em função da utilização da rede. O BWTC_CALC recolhe informação das MIB's dos activos de rede com gestão, calcula o débito de dados de cada porta e guarda a informação na base de dados. Com base no débito de dados de cada porta do activo de rede e da localização de cada cliente do servidor de imagens (informação obtida pelo BWTC_ARP), o BWTC_CALC modela o débito do servidor de imagens com destino a cada porta, de modo a que a taxa de utilização não ultrapasse o valor de *ceil* (valor máximo de utilização definido pelo gestor). A peça central de modelação do débito do servidor é o controlador. O controlador é proporcional, em malha fechada e com memória. O controlador define o débito de dados do servidor de imagens com destino a cada porta dos activos de rede, de modo a que o tráfego total (tráfego de imagens mais o tráfego já existente) em cada porta não ultrapasse o valor de *ceil* (por exemplo, 72Mbps). O facto do controlador ter memória é muito importante, porque o débito de saída de dados de uma qualquer porta "G" do activo de rede no instante *n* pode incluir o tráfego introduzido pelo servidor de imagens no instante *n*-1. Deste modo evita-se o congestionamento de tráfego nas intersecções de rede, a garante-se que existe largura de banda disponível para novas aplicações que entretanto sejam activadas.

O BWTC_STATISTICS foi desenvolvido para permitir uma análise detalhada do débito de dados (valores discretos) das portas dos activos de rede. Com um período de 36 segundos, este módulo acede às MIB's dos activos de rede, recolhe o número de octetos enviados e recebidos de cada porta, calcula o débito e guarda-o na base de dados (juntamente com o instante em que ocorreu a leitura). Posteriormente, pode-se aceder à base de dados do BWTC_STATISTICS, recolher a informação e criar gráficos (por exemplo, no Excel).

Com cada um dos elementos do BWTC pretende-se dar resposta a vários problemas relacionados com a tarefa de modelação do débito do servidor de imagens em função da utilização da rede de dados. Os problemas prendem-se não só com a modelação do débito do servidor de imagens, mas também em identificar a porta (do activo de rede) em que os computadores clientes estão ligados e qual a taxa de utilização das mesmas. Todos os módulos do BWTC podem ser activados remotamente, através da interface Web do NMSIS. O BWTC funciona ao longo do tempo, de forma autónoma e dinâmica.

8. ANÁLISE DE DESEMPENHO DO BWTC – TESTES E RESULTADOS

A ferramenta BWTC permite modelar o débito de dados do servidor de imagens em função da percentagem de utilização da rede de dados. Este software foi desenvolvido para atenuar os problemas de congestionamento de tráfego nas intersecções de rede, quando se encontra a decorrer a instalação de imagens de software. Deste modo garante-se que existe largura de banda disponível para os restantes serviços informáticos suportados pela rede do DEG (serviço de internet, página Web, servidor de ficheiros, entre outros). Para verificar a fiabilidade do BWTC foram elaborados vários testes à aplicação, em ambiente real. Com estes testes pretende-se avaliar: 1) o desempenho do BWTC em situações de descongestionamento e congestionamento de tráfego, quando está a decorrer a instalação de imagens de software em computadores remotos; 2) a capacidade do BWTC optimizar a utilização das portas dos activos de rede para 72% (valor definido pelo gestor) da capacidade máxima. Deste modo garante-se que existe largura de banda disponível para aplicações que entretanto arranquem; 3) o impacto nos serviços informáticos do DEG, resultante da instalação imagens de software em computadores remotos (através da rede de

dados), sem o BWTC; 4) o impacto nos serviços informáticos do DEG, resultante da instalação imagens de software em computadores remotos, usando o BWTC.

Características dos cenários de testes:

Para que seja plena a percepção dos cenários de teste que foram projectados para avaliar o BWTC, é importante ter vários factores em consideração:

1) O cenário 1 serve de referência para o cenário 2, o cenário 3 serve de referência para o cenário 4.

2) Os gráficos de tráfego com fundo azul foram recolhidos a partir do NAV. O NAV recolhe, com o período de 5 minutos, o número de octetos enviados e recebidos para cada porta dos activos de rede (por exemplo, o SWITCH_01 na Figura 56), essa informação é armazenada na ferramenta RRDtools, e apresentada na forma de gráficos com a ferramenta Cricket. Os gráficos provenientes do Cricket são gráficos de utilização média dos últimos 5 minutos;

3) Como se pode ver no Anexo A e na Tabela 10, os *switches*, os servidores e os computadores utilizados nos vários cenários de teste são *full-duplex*. Deste modo cada porta dos activos de rede suporta o débito de dados a 100Mbps, nos dois sentidos, em simultâneo;

4) Alguns dos activos de rede do DEG são actualmente geridos pelo Gabinete de Gestão de Sistemas Informáticos do ISEP (GSI), pelo que os cenários de teste foram projectados e executados com base no nível de acesso que foi facultado. Para a relização dos testes o GSI permitiu o acesso a dois *switches* HP Procurve 2510-24 com gestão SNMP, cujas características estão apresentadas no Anexo A. Os *switches* estão localizados fisicamente nas salas C207 e C213 da rede do DEG, consoante a representação na Figura 10. Entre o SWITCH_01 e o SWITCH_02 da Figura 10 existe o SWITCH_03, que é um *switch* HP Procurve 2626, 100Mbps, *full-duplex*;

5) Será utilizada a seguinte nomenclatura: servidor da página de internet – SW, servidor de ficheiros – SF, servidor de imagens de software – SIMG, cliente do servidor de imagens de software – CSIMG, cliente do servidor Web – CSW, cliente do servidor de ficheiros - CSF;

6) Nos cenários de teste, os computadores clientes fazem *downloads* dos servidores. O cliente do servidor de ficheiros faz *downloads* aleatórios (*download* de muitos ficheiros, de

tamanho variável), os clientes dos servidores Web e de imagens fazem *downloads* lineares (*download* de poucos ficheiros, de elevado tamanho);

7) A inclusão do servidor Web nos cenários de teste também permite estudar directamente: qual é o impacto da instalação de imagens, nos serviços de contas de utilizadores SAMBA e serviço de licenças de software, uma vez estes correm no mesmo servidor físico;

Tabela 10: Descrição técnica dos servidores e computadores

Nome do Servidor/ Computador	Tipo de Processador	Memória RAM	Disco Rígido	Placa de rede	Ano
Servidor da página de internet (SW)	Intel; Pentium 4; 3,0GHz	768MB	80Gb; 100Mbps; 7200rpm; 2MB cache;	Gigabit Ethernet; 1Gbps; Full-Duplex	2004
Servidor de ficheiros (SF	Intel; Pentium 4; 3,0GHz	768MB	80Gb;100Mbps; 7200rpm; 2MB cache;	Gigabit Ethernet; 1Gbps; Full-Duplex	2004
Servidor de Imagens (SIMG)	Intel; Pentium 4 Hyper Threading; 3GHz	512MB	80Gb; 100Mbps; 7200rpm; 8MB cache;	Fast Ethernet; 100Mbps; Full-Duplex	2007
Servidor NAV (SNAV)	Intel; Pentium 4; 1,7GHz	768MB	40Gb; 100Mbps; 7200rpm; 2MB cache;	Fast Ethernet; 100Mbps; Full-Duplex	2002
Computadores Clientes	Intel; Pentium 4; 1,7GHz	512MB	40Gb; 100Mbps; 7200rpm; 2MB cache;	Fast Ethernet; 100Mbps; Full-Duplex	2002

8) Na Tabela 10 são apresentadas as caracteristicas técnicas dos servidores e computadores; é de salientar que o SW e o SF têm características físicas idênticas, sendo estas diferentes do SIMG. Os computadores clientes têm características físicas idênticas;

9) Nos cenários de teste 1, 2 e 3 o BWTC_STATISTICS não foi utilizado, uma vez que ainda não existia. O BWTC_STATISTICS é introduzido no cenário 4, permitindo uma análise minuciosa nos cenários de teste em que ocorrem situações de congestionamento de tráfego eminente. Em seguida serão apresentados os vários cenários de teste, apresentados os resultados experimentais e posterior análise dos resultados.

8.1. CENÁRIO_1_REFERÊNCIA

Com este cenário de testes pretende-se medir qual é a carga de tráfego resultante do acesso contínuo ao servidor de ficheiros e servidor Web, de vários utilizadores em simultâneo, ao longo do tempo. Como é ilustrado na Figura 56, os computadores clientes estão directamente ligados ao mesmo *switch* (SWITCH_01) que os servidores. Este cenário não tem tráfego com destino à instalação de imagens de software, através da rede.

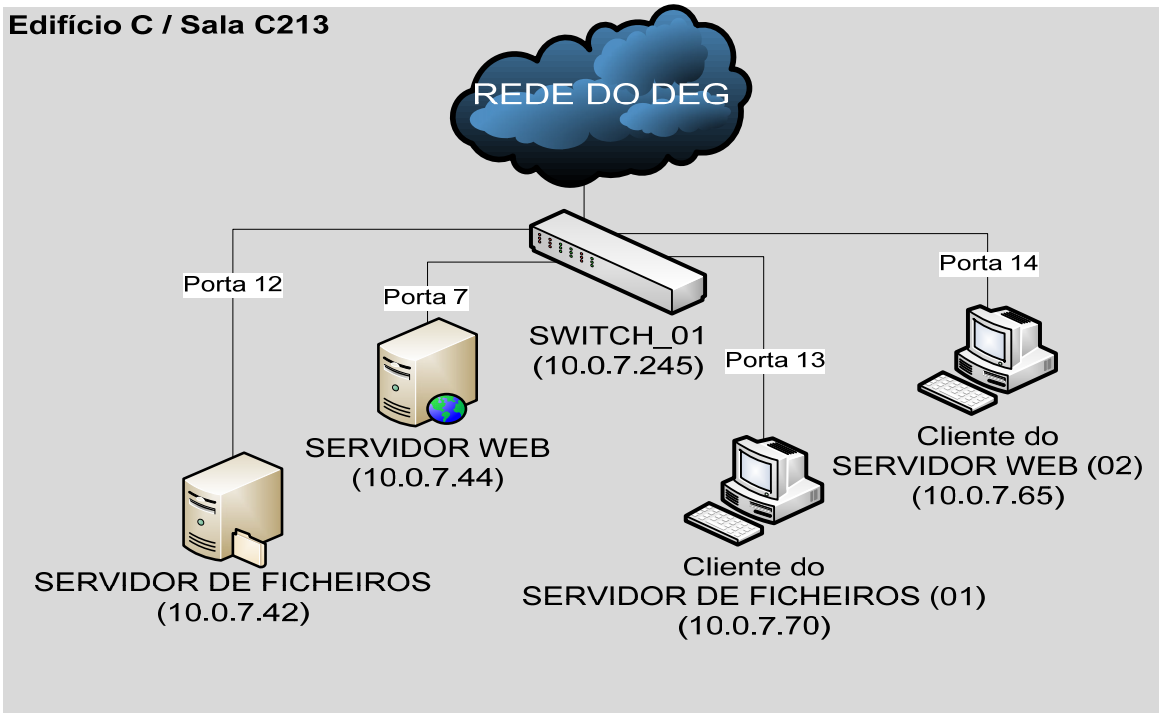

Figura 56: Esquemático do cenário_1_referência de testes

Tabela 11: Resumo do cenário_1_referência

Clientes	Cliente Do SW	Cliente Do SF	O Tráfego é processado no SWITCH_01	Tamanho do Download
01		X	X	59Gb (7.45GB)
02	X		X	58.16Gb (7.27GB)

8.1.1. RESULTADOS EXPERIMENTAIS – CENÁRIO_1_REFERÊNCIA

Os resultados experimentais são apresentados na forma de gráficos, com o resumo na Tabela 12. Marcado com a caixa a vermelho, na Figura 57 e na Figura 58, estão representados os débitos do servidor Web (SW) e do servidor de ficheiros (SF), para os respectivos clientes.

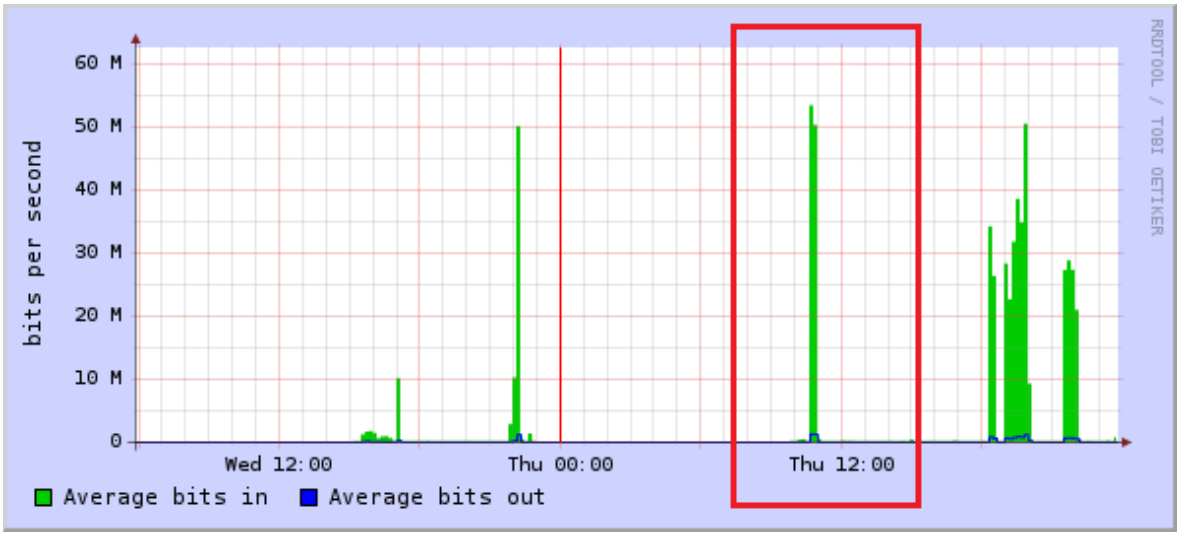

Figura 57: Gráfico de débito de dados de entrada e saída na porta 07 do SWITCH_01 (SW)

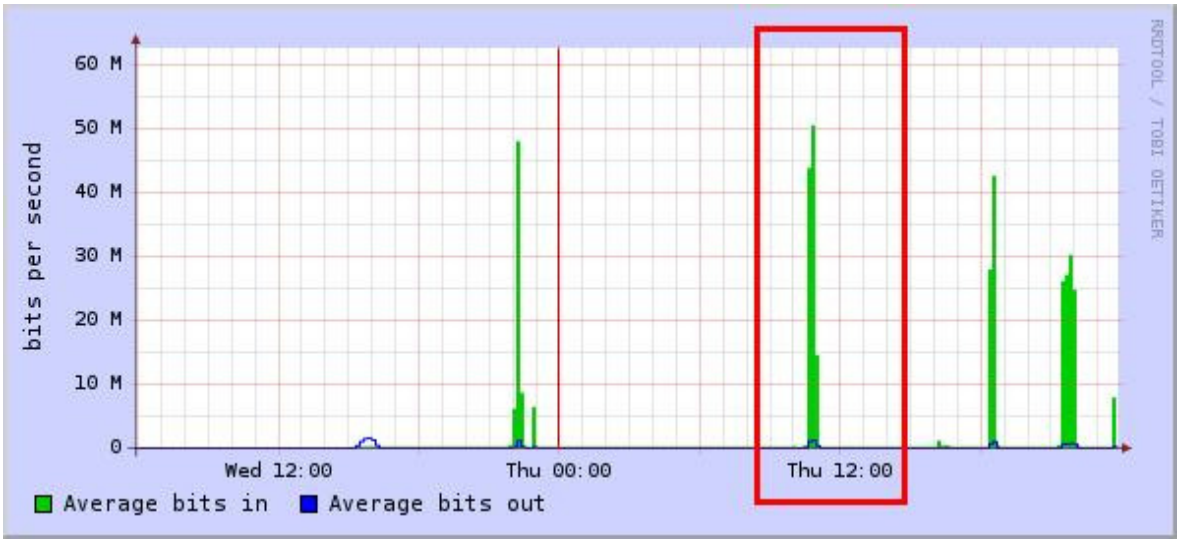

Figura 58: Gráfico de débito de dados de entrada e saída na porta 12 do SWITCH_01 (SF)

Tabela 12: Resultados experimentais do cenário_1_referência[2]

Servidor	Número da porta	Duração Da tarefa	Débito máximo Do SW	Débito máximo Do SF
SW	07	**00H19**	**53Mbps**	
SF	12	**00H23**		**50Mbps**

8.1.2. ANÁLISE DOS RESULTADOS – CENÁRIO_1_REFERÊNCIA

Neste cenário de teste pretende-se verificar qual é o débito máximo de dados do servidor Web e do servidor de ficheiros, resultante do acesso de vários utilizadores em simultâneo. Este cenário serve de referência para o cenário_2_sem_BWTC e para o cenário_2_com_BWTC.

Como é apresentado na Tabela 10, os sistemas SF e SW têm placa de rede de 1000Mbps (*Gigabit Ethernet*) *Full-Duplex* e disco rígido de 100Mbps. Apesar da placa de rede a 1Gbps, a rede funciona a 100Mbps, pelo que a placa de rede dos servidores também. A partir dos resultados experimentais (Figura 58, Figura 57 e Tabela 12) depreende-se que o débito de dados, também conhecido por *capacity ou bandwidth* [22], dos servidores Web (SW) e de ficheiros (SF), fica abaixo das características do *hardware*. Os valores apresentados são o resultado de vários factores, uns relacionados com as características do *hardware* em si, outros com o protocolo TCP.

Ao nível do *hardware*, uma das característica que mais influência o débito de dados de um equipamento de rede é a capacidade (*capacity*) do equipamento de transmissão. A capacidade tem em consideração vários factores, como o tipo de equipamento de transmissão, o *driver* do equipamento [147], a performance do seu processador, o *delay*, os erros de transmissão, o congestionamento, o desgaste do equipamento, bem como qualquer factor que afecte a transmissão. Em [22] afirmam que depois de todos estes factores, um equipamento do tipo *Fast Ethernet* com a capacidade de 100Mbps *half-duplex*, pode ter a performance reduzida em 50% [22], nos equipamentos do tipo *full-duplex,* os efeitos destes factores sobre a capacidade de transmissão é minimizado [22].

[2] Os valores apresentados na tabela foram obtidos da análise da Figura 57, Figura 58 e do vídeo contido no anexo "Teste26062008_10H40_sem_BWTC", disponível no CD anexo à tese

Ao nível do protocolo TCP os principais factores que influenciam a taxa máxima de transmissão de dados são: os mecanismos de controlo de fluxo, os mecanismos de controlo de congestionamento e a máquina de estados TCP.

Como está ilustrado na Figura 56, cada computador e servidor deste cenário de testes está ligado a uma porta do *switch*, pelo que não há congestionamento de tráfego.

O controlo de fluxo (método da janela deslizante) [22][60] limita a taxa máxima de transferência do emissor para cada um dos receptores, através do tamanho da janela (tamanho máximo de bytes transmitidos sem receber um ACK). Deste modo torna-se possível a comunicação entre equipamentos com diferentes janelas (*buffers*) de envio e/ou recepção, e evita-se que um só cliente (ou *flow*) utilize toda a largura de banda do emissor [143].

A máquina de estados TCP tem alguma influência na capacidade efectiva de transmitir dados, nomeadamente ao nível das operações que executa sobre os dados (por exemplo, o suporte para várias ligações em simultâneo, o encapsulamento dos dados, o controlo de erros, etc.). [145][146]

Este cenário de teste serve de referência para os próximos testes, que serão realizados no âmbito do cenário_1_referência, introduzindo a instalação de imagens de software através da rede de dados.

8.2. CENÁRIO 2

Com este cenário de testes pretende-se mostrar qual é a influência de uma instalação de imagens de software através da rede, no cenário_1_referência. Numa primeira fase (cenário_2_sem_BWTC) o débito de dados do servidor de imagens é gerido pelo mecanismo *pfifo_fast* (utilizado por omissão em sistemas operativos Linux). Numa segunda fase (cenário_2_com_BWTC) o débito de dados é modelado utilizando o BWTC. A arquitectura utilizada é a representada na Figura 59, os computadores clientes e os servidores estão fisicamente ligados ao mesmo *switch*.

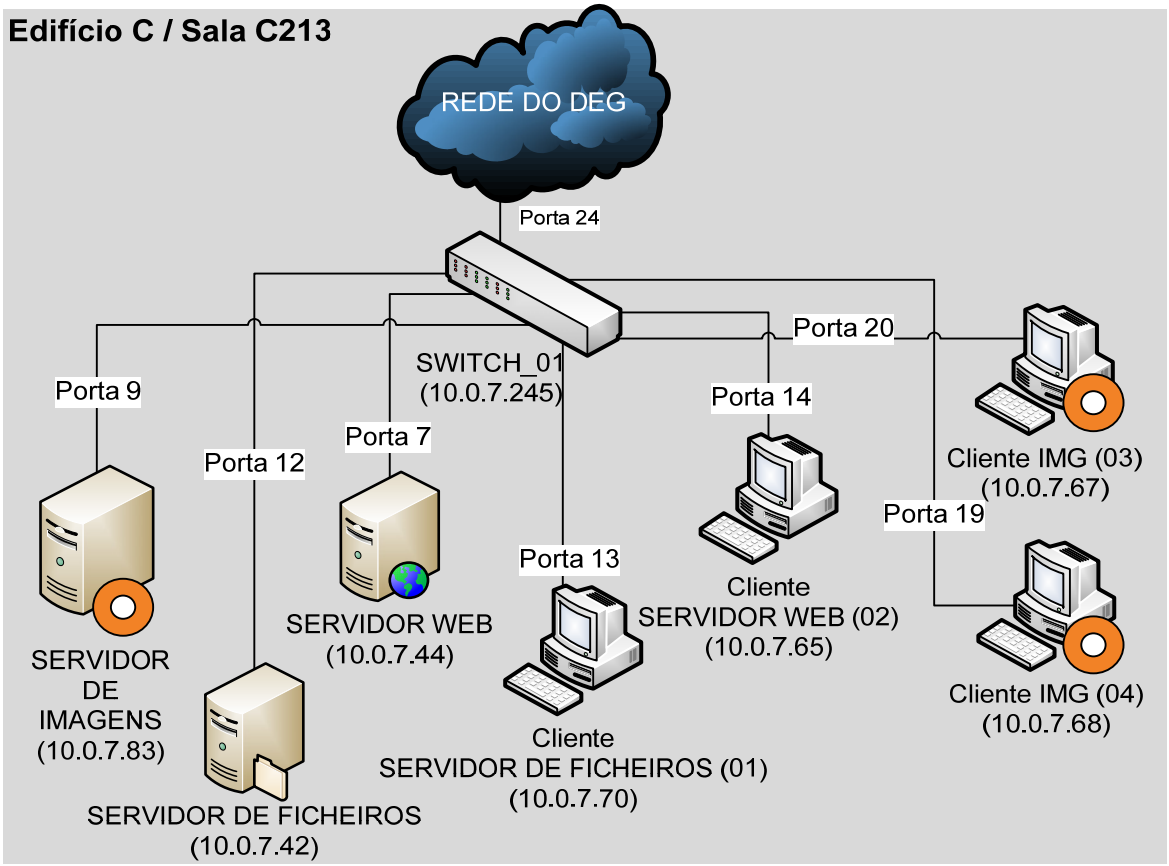

Figura 59: Esquema do cenário 2 de teste

Tabela 13: Resumo do cenário 2

Clientes	Cliente Do SW	Cliente Do SF	Cliente Do SIMG	Tráfego processado no SWITCH_01	Tamanho do Download
01		X		X	59Gb (7.45GB)
02	X			X	58.16Gb (7.27GB)
03			X	X	60.8Gb (7.6GB)
04			X	X	60.8Gb (7.6GB)

8.2.1. RESULTADOS EXPERIMENTAIS – CENÁRIO_2_SEM_BWTC

Neste cenário de testes é utilizado o *scheduler* de pacotes de dados utilizado por omissão em Linux, o *pfifo_fast*. Os gráficos do débito de saída de dados dos servidores SW e SF, representados nas Figura 60 e Figura 61 respectivamente, são iguais aos gráficos de entrada de dados nos respectivos clientes. A Figura 62 representa o débito de dados do servidor de imagens (SIMG). Os clientes do SIMG estão ligados nas portas 19 e 20 do SWITCH_01, e os gráficos de entrada de dados nestes estão representados na Figura 63 e na Figura 64. A Tabela 14 contém o resumo dos resultados experimentais.

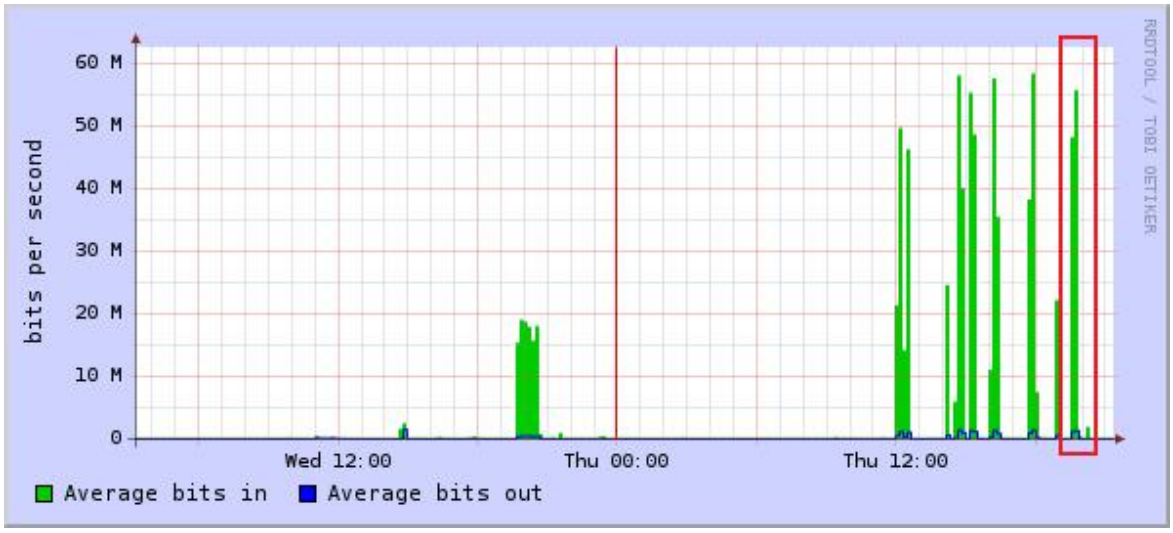

Figura 60: Débito de dados do servidor Web para o respectivo cliente (Porta 7).

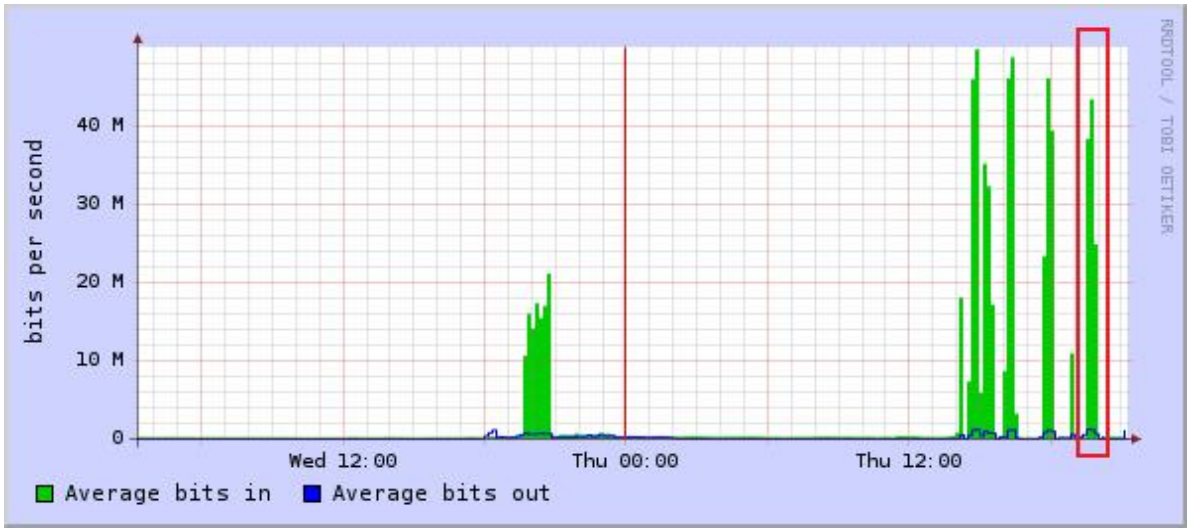

Figura 61: Débito de dados do servidor de ficheiros para os respectivos clientes (Porta 12).

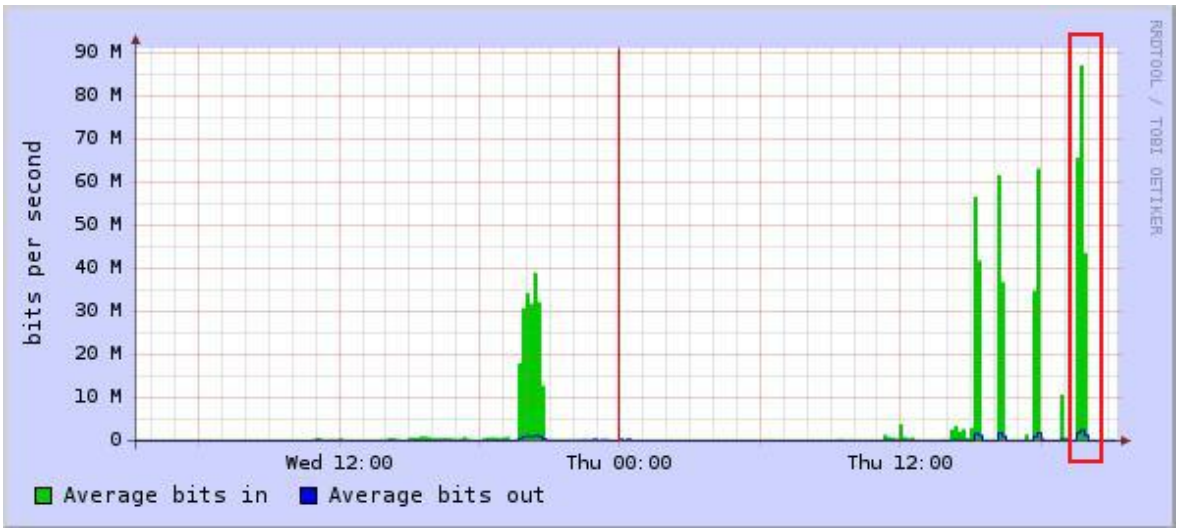

Figura 62: Débito total de dados do servidor de imagens para os respectivos clientes (Porta 9).

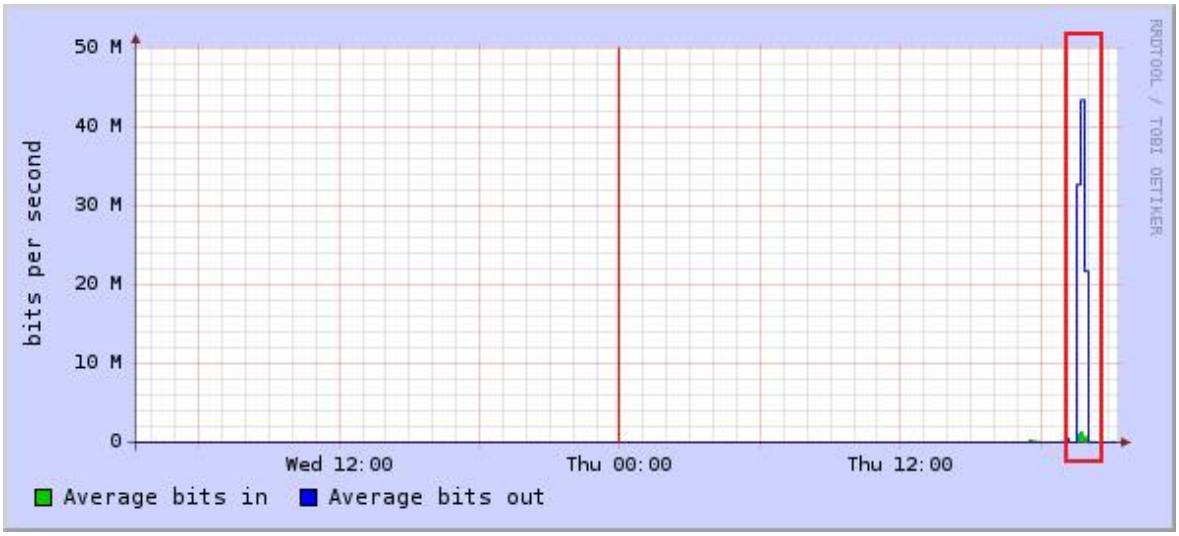

Figura 63: Débito de dados do servidor de imagens para o cliente 04 (Porta 19).

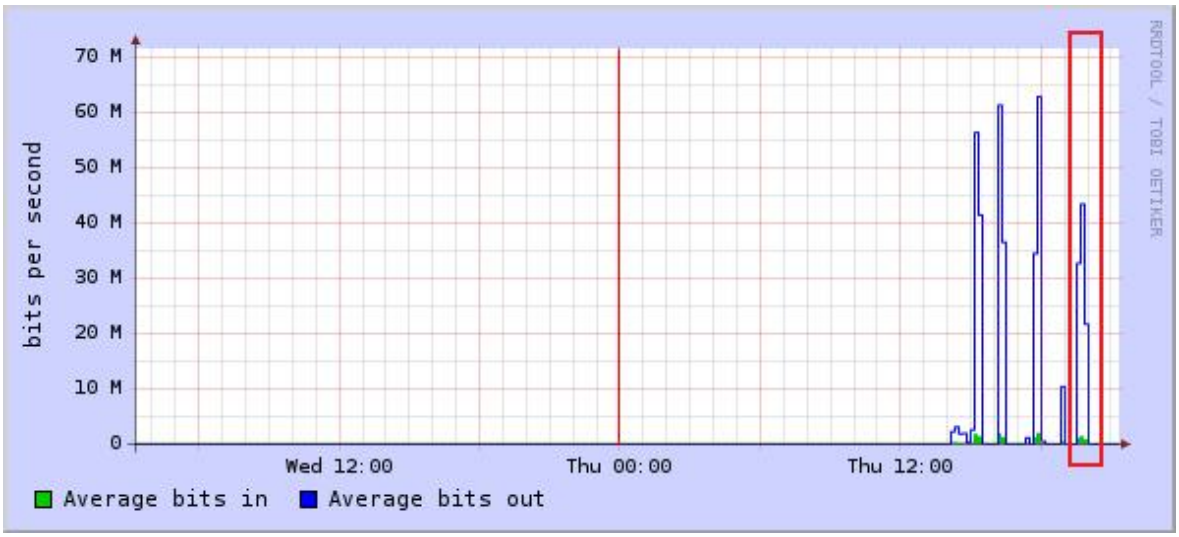

Figura 64: Débito de dados do servidor de imagens para o cliente 03 (Porta 20).

Tabela 14: Resultados experimentais do cenário_2_sem_BWTC[3]

Servidor/ Cliente	Número da porta	Duração Da tarefa	Débito máximo SW	Débito máximo SIMG	Débito máximo SF
SW	07	00H19	56Mbps		
SIMG	09	00H25		87Mbps	
SF	12	00H28			43.5Mbps
CSIMG	19	00H25		43.5Mbps	
CSIMG	20	00H25		43.5Mbps	

8.2.2. ANÁLISE DOS RESULTADOS – CENÁRIO_2_SEM_BWTC

Neste cenário de teste pretende-se verificar qual é a influência da instalação de imagens de software (através da rede de dados) em computadores localizados fisicamente no mesmo *switch* que o servidor de imagens.

Comparando os resultados experimentais do cenário actual (cenário_2_sem_BWTC) com os resultados do cenário anterior (cenário_1_referência) constata-se que: 1) a tarefa associada ao SW decorreu no mesmo período de tempo que no cenário anterior, com o aumenta do débito máximo na ordem dos 5,66%; 2) a tarefa associada ao SF demorou mais 5 minutos que no cenário anterior, com um decréscimo no débito máximo de dados, na ordem dos 13%; 3) o servidor de imagens (SIMG) registou o débito máximo (médio) de dados. Tal como foi descrito na análise de resultados do cenário anterior (secção 8.1.2), o TCP tem ferramentas de controlo de fluxo que limitam o débito máximo de dados do emissor para cada receptor. Uma vez que o SIMG transmite dados para dois clientes em simultâneo o débito deste é significativamente superior ao dos SW e SF. No entanto o SIMG tem limitações de hardware: placa de rede *Fast Ethernet* (100Mbps) e disco IDE (100Mbps). Como se pode constatar na Tabela 14, estas limitações reflectem-se no débito máximo de download dos computadores clientes, 43,5Mbps em ambos os casos.

Com base nas constatações apresentadas, verifica-se que a introdução da instalação de imagens, num cenário como o da Figura 56, não afecta significativamente a performance dos serviços associados ao SW e SF. Para ultrapassar as limitações da placa de rede do

[3] Os valores apresentados na tabela foram obtidos da análise das Figura 60 à Figura 64 e do vídeo contido no anexo "Teste03072008_19H30_sem_BWTC", disponível no CD anexo à tese

SIMG e da rede do DEG (100Mbps, *full-duplex*) faz sentido estudar a solução de instalação das imagens de software por *multicasting* [22].

8.2.3. RESULTADOS EXPERIMENTAIS – CENÁRIO_2_COM_BWTC

Neste cenário de testes foi utilizado o software de modelação do débito do servidor de imagens – BWTC. Os gráficos do débito de saída de dados dos servidores SW e SF, representados nas Figura 66 e Figura 65 respectivamente, são iguais aos gráficos de entrada de dados nos respectivos clientes. A Figura 67 representa o débito de dados do servidor de imagens. Os clientes do SIMG estão ligados nas portas 19 e 20 do SWITCH_01, e os gráficos de entrada de dados nestes estão representados na Figura 68 e na Figura 69. A Tabela 15 contém o resumo dos resultados experimentais.

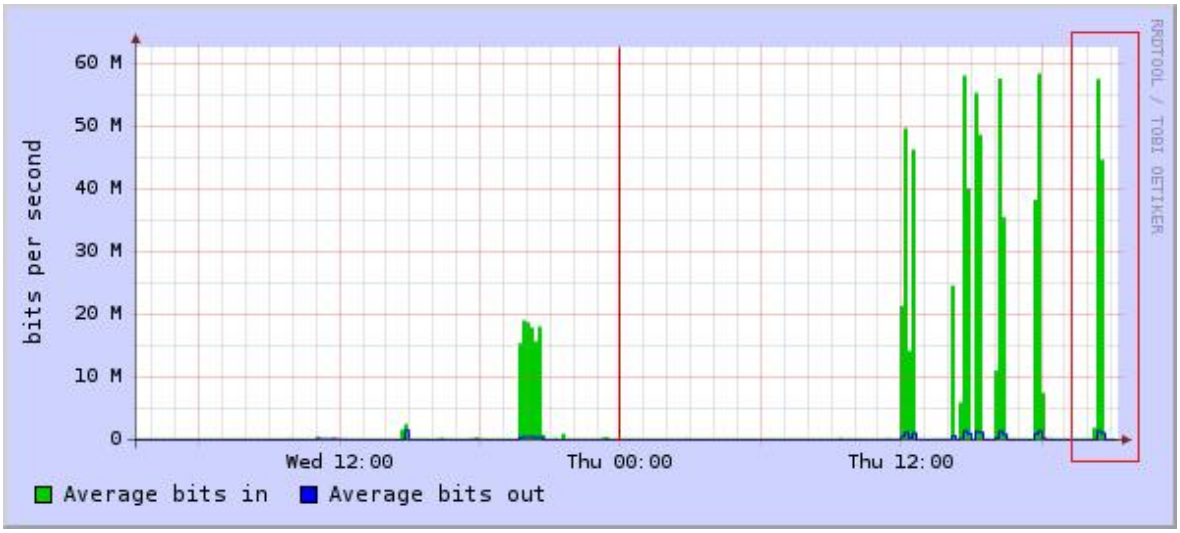

Figura 66: Débito de dados do servidor Web para o respectivo cliente (Porta 7).

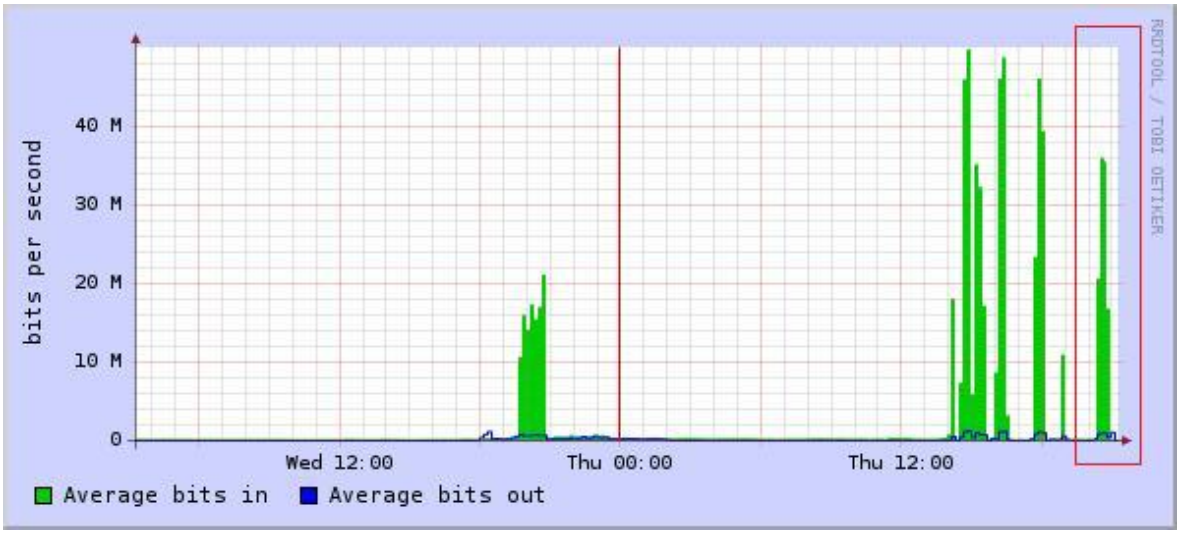

Figura 65: Débito de dados do servidor de ficheiros para o respectivo cliente (Porta 12).

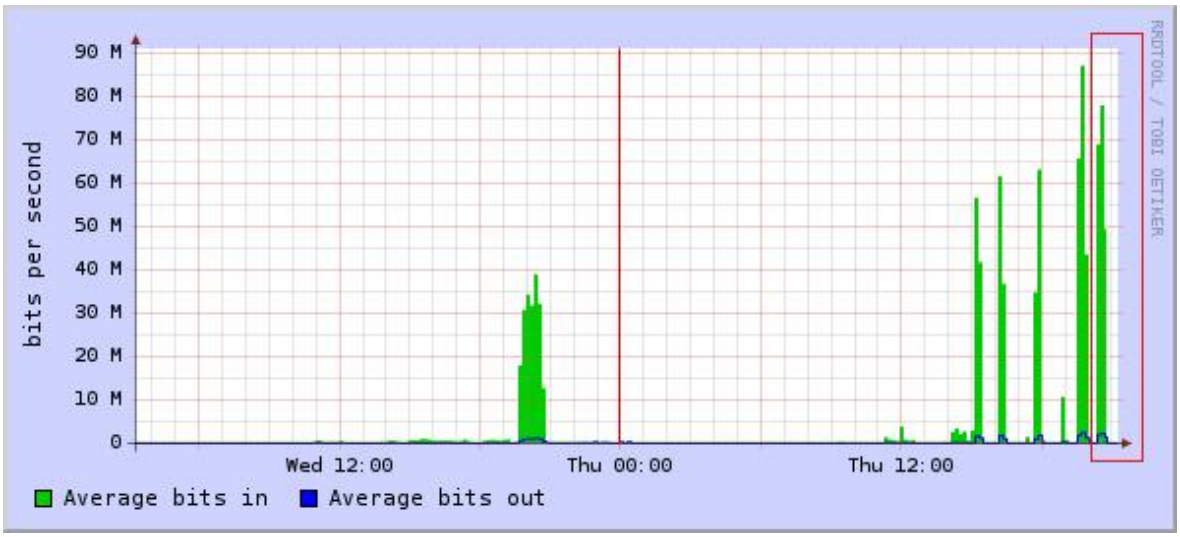

Figura 67: Débito total de dados do servidor de imagens para os clientes 03 e 04 (Porta 9)

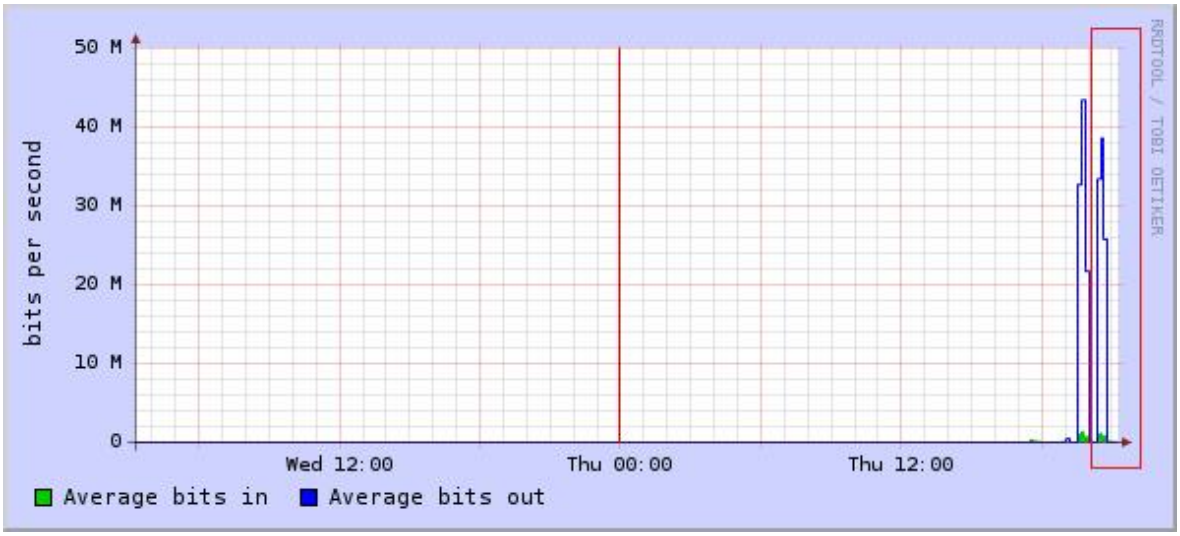

Figura 68: Débito de dados do servidor de imagens para o cliente 04 (Porta 19).

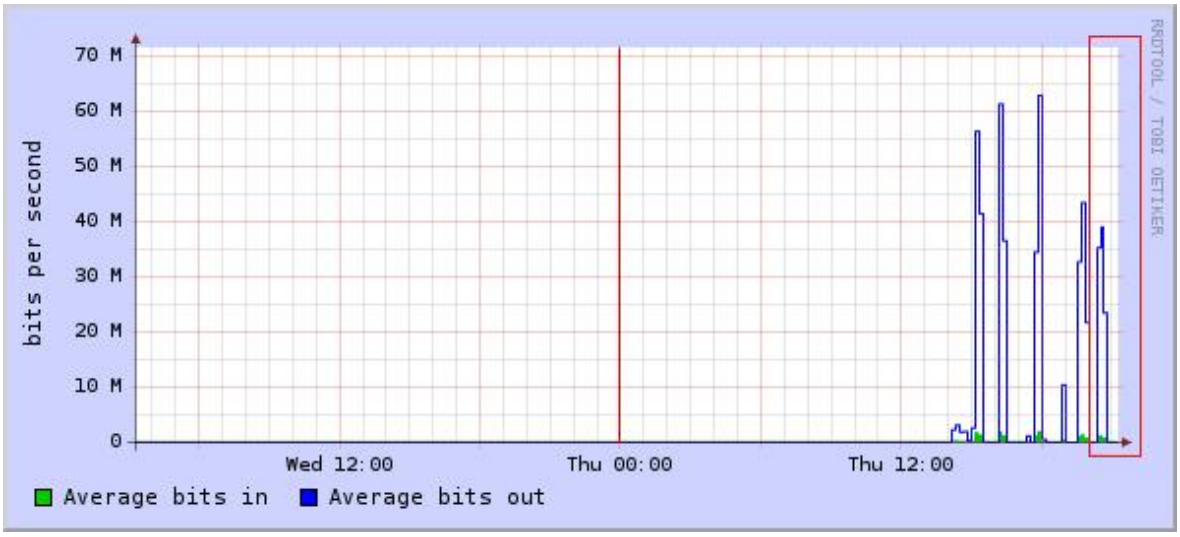

Figura 69: Débito de dados do servidor de imagens para o cliente 03 (Porta 20).

Tabela 15: Resultados experimentais do cenário_2_com_BWTC[4]

Servidor/ Cliente	Número da porta	Duração do teste	Débito Máximo SW	Débito Máximo SIMG	Débito Máximo SF
SW	**07**	**00H19**	**57.5Mbps**		
SIMG	**09**	**00H28**		**77Mbps**	
SF	**12**	**00H29**			**36Mbps**
CSIMG	**19**	**00H28**		**38.5Mbps**	
CSIMG	**20**	**00H28**		**38.5Mbps**	

8.2.4. ANÁLISE DOS RESULTADOS – CENÁRIO_2_COM_BWTC

Neste cenário de teste pretende-se averiguar qual a influência da modelação do débito do servidor de imagens, comparando o cenários actual (cenário_2_com_BWTC) com o cenário anterior (cenário_2_sem_BWTC).

Analisando os resultados do cenário actual (Tabela 15) e os resultados do cenário anterior (Tabela 14) verifica-se que: o tempo de execução da tarefa associada ao SW volta a manter-se igual (19 minutos), no entanto o débito máximo atingido por este subiu dos 56Mbps, no cenário anterior, para 57.5Mbps no cenário actual; o tempo de execução das tarefas associadas ao SIMG aumentou de 25 para 28 minutos, com um decréscimo no débito máximo de 87Mbps para 77Mbps no cenário actual (decréscimo de 12%); o débito máximo de dados do SF desceu dos 43.5Mbps, no cenário anterior, para os 36Mbps no cenário actual, um decréscimo de 17%. O tempo de execução da respectiva tarefa aumentou em apenas 1 minuto.

O BWTC define um valor máximo de utilização (*ceil*) de cada porta, definido pelo gestor, que nestes cenários de teste é 72% (ou 72Mbps). Como se pode ver na Figura 67 (encaixilhado a vermelho), o valor máximo de utilização da porta 24 do SWICTH_01 no cenário actual foi de 77Mbps. Assim, é notória a influência do BWTC, uma vez que o valor máximo passou de 87Mbps (Figura 62), no cenário anterior, para 77Mbps no cenário actual, o que indica um decréscimo de 11%. Apesar do valor máximo actual (cenário com

[4] Os valores apresentados na tabela foram obtidos da análise das Figura 66 a Figura 69 e do vídeo contido no anexo "Teste03072008_20H18_com_BWTC", disponível no CD anexo à tese

BWTC) não ser exactamente o pretendido, a aproximação obtida é um indicador positivo do desempenho do BWTC.

Com base na comparação dos cenários 2, sem e com BWTC, pode-se concluir que:

1) Mesmo em situações de descongestionamento de tráfego, o BWTC "obriga" o servidor de imagens a não utilizar toda a largura de banda existente. Deste modo, garante-se que existe largura de banda disponível para as aplicações de rede que sejam activadas entretanto;

2) A inserção da tarefa de instalação de imagens de software num cenário como o da Figura 56 (um só activo de rede), não influência de forma significativa a performance dos restantes serviços informáticos. Isto acontece porque o tráfego dos diferentes servidores não circula nas mesmas portas do activo de rede;

3) Nas situações de descongestionamento de tráfego, as limitações de hardware do SIMG prejudicam gravemente os tempos de instalação de imagens de software. A largura de banda do SIMG (100Mbps) é dividida por dois clientes que estão ligados em diferentes portas do SWITCH_01. Portanto, mesmo que o servidor apresentasse o *throughput* de 100Mbps, resultava em 50Mbps para cada cliente, uma vez que não se está a utilizar uma instalação por *multicasting*. No *multicasting* o emissor envia os pacotes de dados para o activo de rede (que suporta *multicasting*) e este encaminha-os para vários receptores. Deste modo poupa-se a largura de banda do emissor, mas implica que os activos de rede suportem *multicasting*.

Assim, os cenários de teste 1 e 2 são necessários, mas não suficientes, para provar a utilidade do software BWTC em cenários de congestionamento de tráfego.

Os cenários de teste que se seguem permitem estudar o comportamento do software BWTC, em situações de congestionamento eminente.

8.3. CENÁRIO_3_REFERÊNCIA

Com este cenário de testes pretende-se avaliar a carga de tráfego resultante do acesso contínuo ao servidor de ficheiros e servidor Web, de vários utilizadores em simultâneo, ao longo do tempo. Como se pode constatar na Figura 70, os computadores clientes não estão directamente ligados ao mesmo *switch* que os servidores. Note-se que este cenário não tem tráfego com destino à instalação de imagens de software, através da rede.

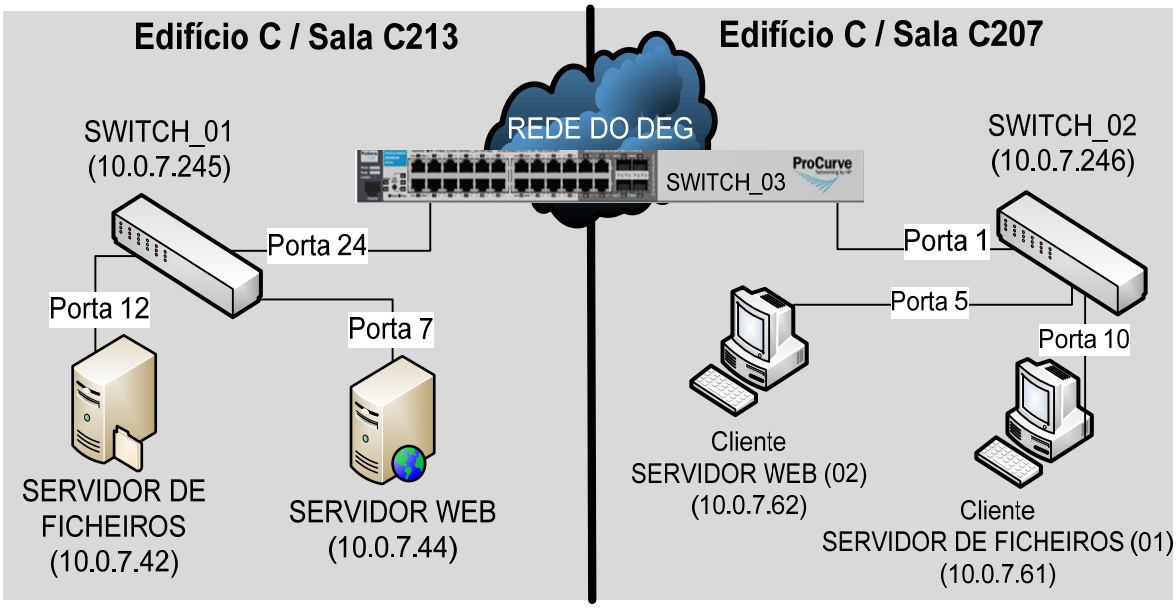

Figura 70: Esquema do cenário 3 de teste

Tabela 16: Resumo do cenário 3

Cliente	Cliente do SW	Cliente do SF	Tráfego processado pelo SWITCH_01	Tráfego processado pelo SWITCH_02	Tamanho do Download
01		**X**	**X**	**X**	**59Gb (7.45GB)**
02	**X**		**X**	**X**	**58.16Gb (7.27GB)**

8.3.1. RESULTADOS EXPERIMENTAIS – CENÁRIO_3_REFERÊNCIA

Os gráficos seguintes representam o débito de dados do SW, do SF e da porta de *uplink* do SWITCH_01 (porta 24) para os respectivos computadores clientes, sem BWTC.

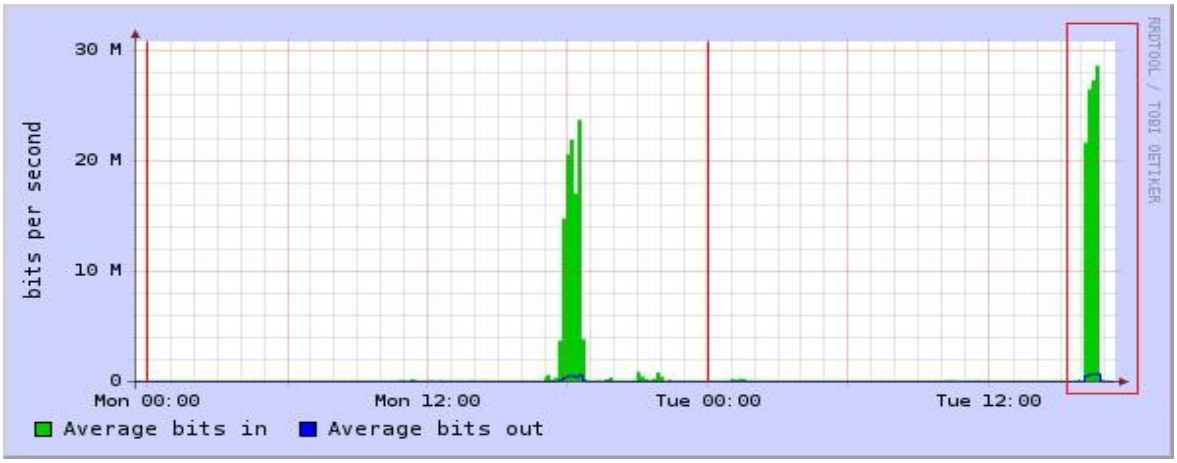

Figura 71: Débito de dados do servidor Web para o respectivo cliente (Porta 7).

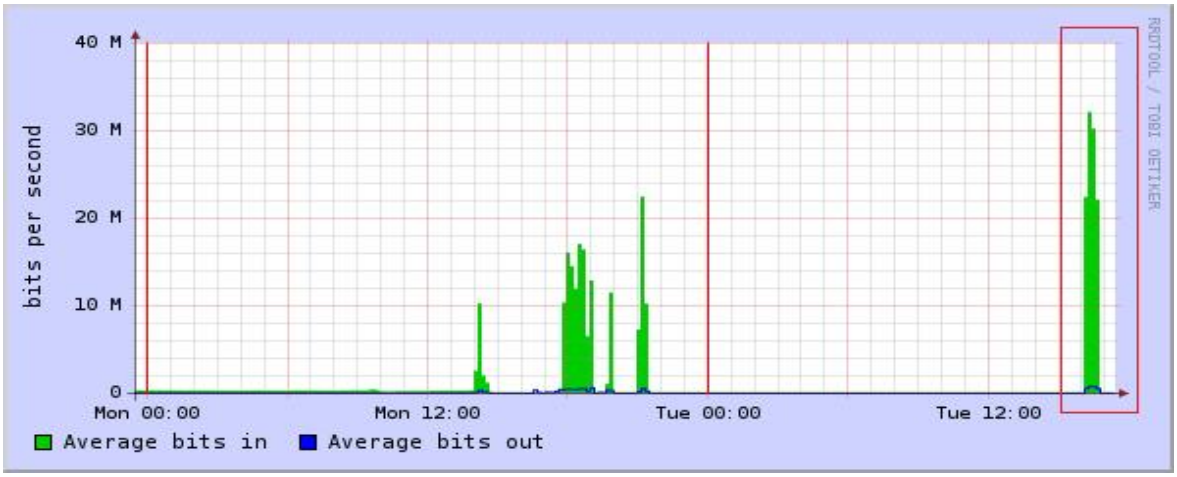

Figura 72: Débito de dados do servidor de ficheiros para o respectivo cliente (Porta 12).

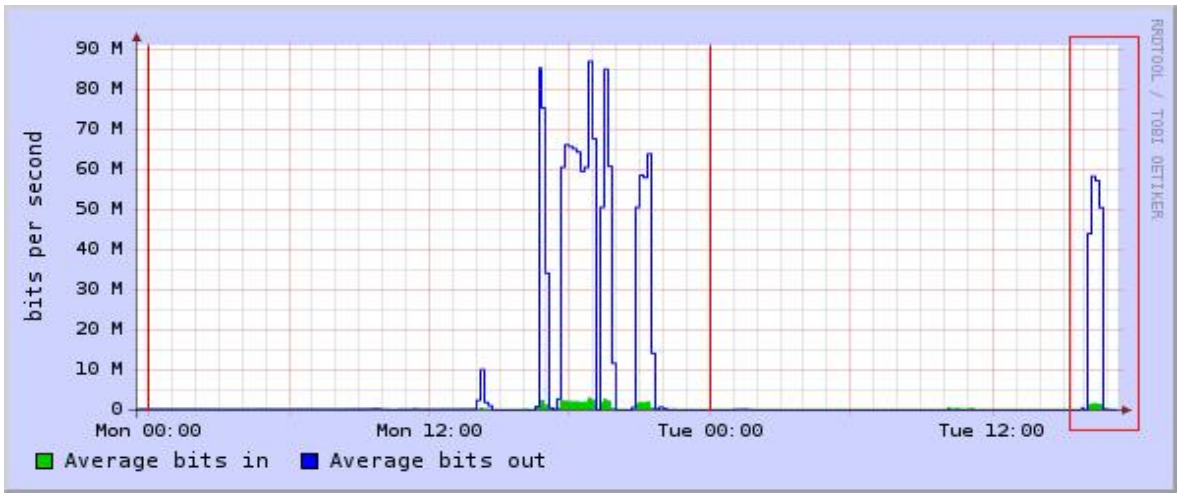

Figura 73: Débito de saída de dados da porta 24 (uplink) do SWITCH_01

Tabela 17: Resultados experimentais do cenário_3_referência[5]

Servidor/ Porta	Número da porta	Duração do teste	Débito máximo Do SW	Débito máximo do SF
SW	**07**	**00H37**	**28,5Mbps**	
SF	**12**	**00H35**		**32Mbps**
UPLINK	**24**	**00H37**	**59.5Mbps**	

8.3.2. ANÁLISE DOS RESULTADOS – CENÁRIO_3_REFERÊNCIA

Neste cenário introduz-se um factor muito importante para demonstrar a robustez e performance do software BWTC. Ao contrário dos cenários 1 e 2, em que o tráfego de dados para os clientes era distribuído por diferentes portas do SWITCH_01, neste caso todo o tráfego com destino aos clientes percorre o mesmo trajecto físico até chegar ao SWITCH_02. Como se pode constatar na Figura 70, todo o tráfego do SW e SF sai pelo *uplink* do SWITCH_01, porta 24, e entra no SWITCH_02 pela porta 1. Deste modo cria-se a situação de congestionamento de tráfego eminente.

Os resultados do cenário actual (cenário_3_referência, Tabela 17) podem ser comparados com os do 1º cenário (cenário_1_referência, Tabela 12), porque os cenários têm características semelhantes. No cenário actual verifica-se que o débito máximo no *uplink* do SWITCH_01 é 59.5Mbps, muito inferior ao que seria de esperar. O tempo de execução da tarefa associada ao SW passou de 19 minutos, no cenário anterior, para 37 minutos (aumento de 95%), com um decréscimo no débito máximo de 53Mbps, no cenário anterior, para 28,5Mbps (decrescimo de 46%). O tempo de execução da tarefa associada ao SF passou de 23 para 35 minutos (aumento de 52%), com um decréscimo no débito máximo de 50Mbps para 32Mbps (decrescimo de 36%). O decrescimo de performance em todos os serviços está relacionado não só com os mecanismos de controlo de fluxo e congestionamento do protocolo TCP, mas também com o *delay* na entrega dos pacotes de dados resultante da necessidade dos vários *switches* enviarem o dobro da informação através da mesma porta.

Este cenário de testes serve de referência para os cenários seguintes, nos quais será introduzida a tarefa de instalação de imagens de software. Na análise dos próximos

[5] Os valores apresentados na tabela foram obtidos da análise das Figura 72, Figura 73 e Figura 73 e do vídeo contido no anexo "Teste08072008_16h12_sem_BWTC", disponível no CD anexo à tese

cenários de teste serão introduzidos os gráficos e informação obtidos com o módulo de software BWTC_STATISTICS.

8.4. CENÁRIO 4

Com este cenário de testes pretende-se demonstrar qual é a influência da instalação de imagens de software através da rede, no cenário_3_referência. Numa primeira fase (cenário_4_sem_BWTC) o débito de dados do servidor de imagens é gerido pelo mecanismo *Packet First In First Out Fast* (*pfifo_fast* [144]). O *pfifo_fast* é utilizado por omissão em sistemas operativos Linux. Na segunda fase (cenário_4_com_BWTC) o débito de dados é modelado utilizando o BWTC. Como se ilustra na Figura 74, os computadores clientes e os servidores não estão fisicamente ligados ao mesmo *switch*.

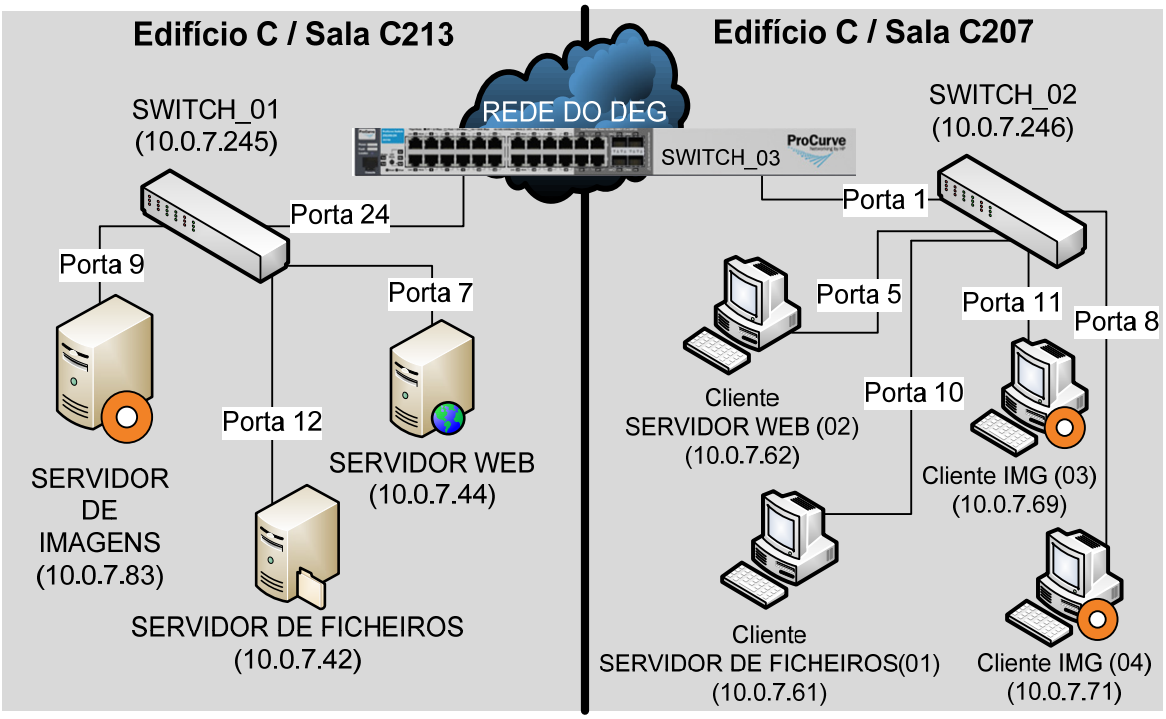

Figura 74: Esquema do cenário de testes 4

Nos cenários 4 e 5 introduziu-se a informação obtida a partir do módulo de software BWTC_STATISTICS. *Qual é a vantagem da utilização desta informação?* Todos os gráficos de tráfego de dados apresentados até agora (com fundo azul), foram obtidos a partir da ferramenta Cricket, disponível no software NAV. Estes foram suficientes para a profundidade do estudo que se pretendia realizar até ao cenário 3. No entanto, os vários pontos que constituem os gráficos gerados pelo Cricket são criados com base na média de

utilização, nos últimos 5 minutos. No BWTC_STATISTICS os dados são obtidos em instantes discretos, com a periodicidade de 36 segundos, o que permite fazer um estudo mais detalhado.

Como se pode ver nos resultados do cenário_3_referência (Tabela 17), o débito máximo de dados na porta de *uplink* do SWITCH_01 é de 59.5Mbps, este valor indicia de forma precoce problemas de congestionamento quando for adicionado o tráfego de dados com destino à instalação de imagens de software.

Introduzindo o tráfego para a instalação de imagens de software, permite testar o software BWTC numa situação de congestionamento eminente. Com os cenários 4 e 5 pretende-se demonstrar a robustez e fiabilidade do software BWTC.

Na secção 8.4.1 são apresentados os resultados experimentais do cenário_4_sem_BWTC, com o resumo na Tabela 23. Na secção 8.4.2 é feita a respectiva análise de resultados.

Na secção 8.4.3 são apresentados os resultados experimentais do cenário_4_com_BWTC, com o resumo na Tabela 28. Na secção 8.4.4 é feita a respectiva análise de resultados.

Tabela 18: Resumo do cenário 4

Cliente	Cliente do SW	Cliente do SF	Cliente do SIMG	Tráfego processado no SWITCH_01	Tráfego processado no SWITCH_02	Tamanho Do Download
01		**X**		**X**	**X**	**59Gb (7.45GB)**
02	**X**			**X**	**X**	**58.16Gb (7.27GB)**
03			**X**	**X**	**X**	**60.8Gb (7.6GB)**
04			**X**	**X**	**X**	**60.8Gb (7.6GB)**

8.4.1. RESULTADOS EXPERIMENTAIS – CENÁRIO_4_SEM_BWTC

Na Figura 75 e na Figura 76 está representado o débito de dados do SW para o respectivo cliente. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 19 contém os resultados experimentais relativamente à 2ª figura.

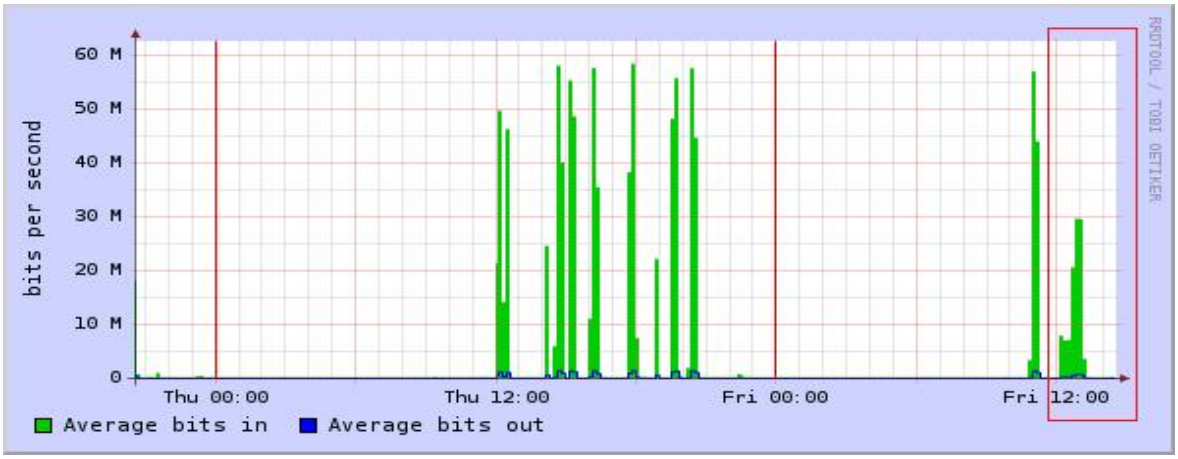

Figura 75: Débito de dados do SW (Porta 7; SWITCH_01) – média de 5 minutos de utilização

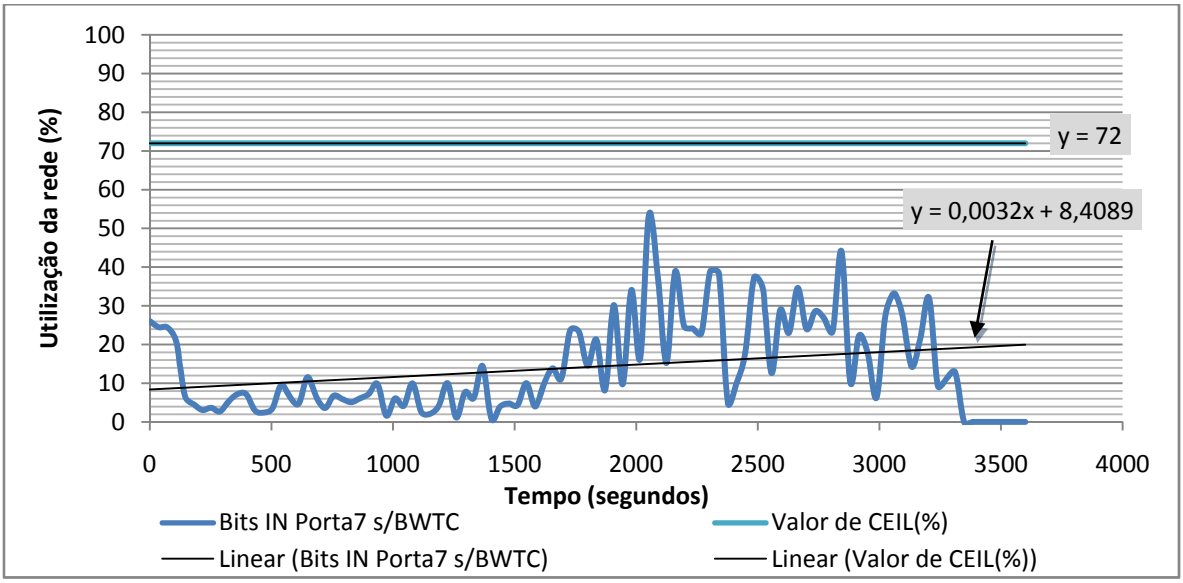

Figura 76: Débito de dados do SW (Porta 7; SWITCH_01) – valores discretos

Tabela 19: Resultados experimentais do SW do cenário_4_sem_BWTC (valores discretos)

Variáveis	Bits IN Porta7 s/BWTC(%)	Valor de CEIL (%)
Utilização média total (0-3600 segundos)	9,949272156	72
Utilização média de 0-1692 (28.2 minutos)	6,08613205	72
Utilização média de 1728-3600 (31.2 minutos)	22,26513672	72
Valor Máximo (valor instantâneo)	53,43422699	72

Na Figura 77 e na Figura 78 está representado o débito de dados do SIMG para os respectivos clientes. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 20 contém os resultados experimentais, relativamente à 2ª figura.

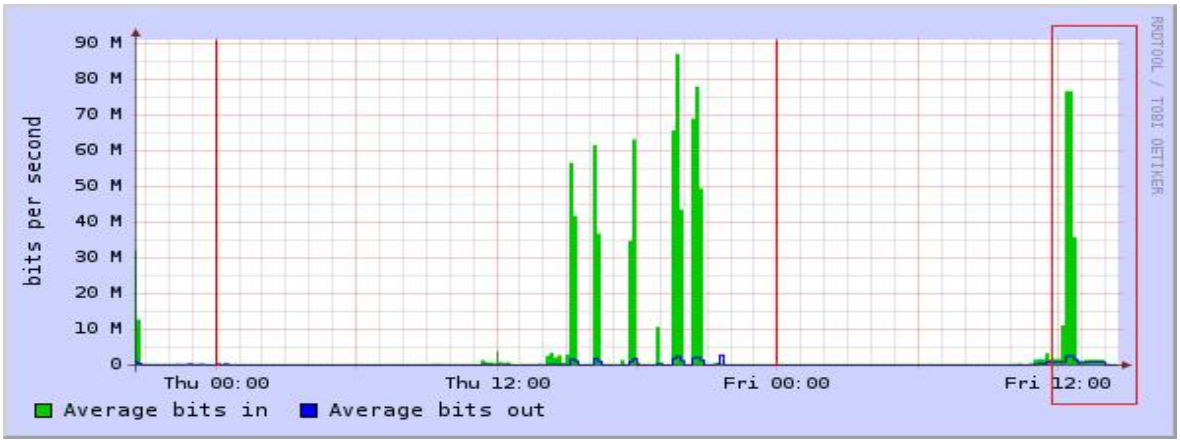

Figura 77: Débito de dados do SIMG (Porta 9; SWITCH_01) – média de 5 minutos de utilização

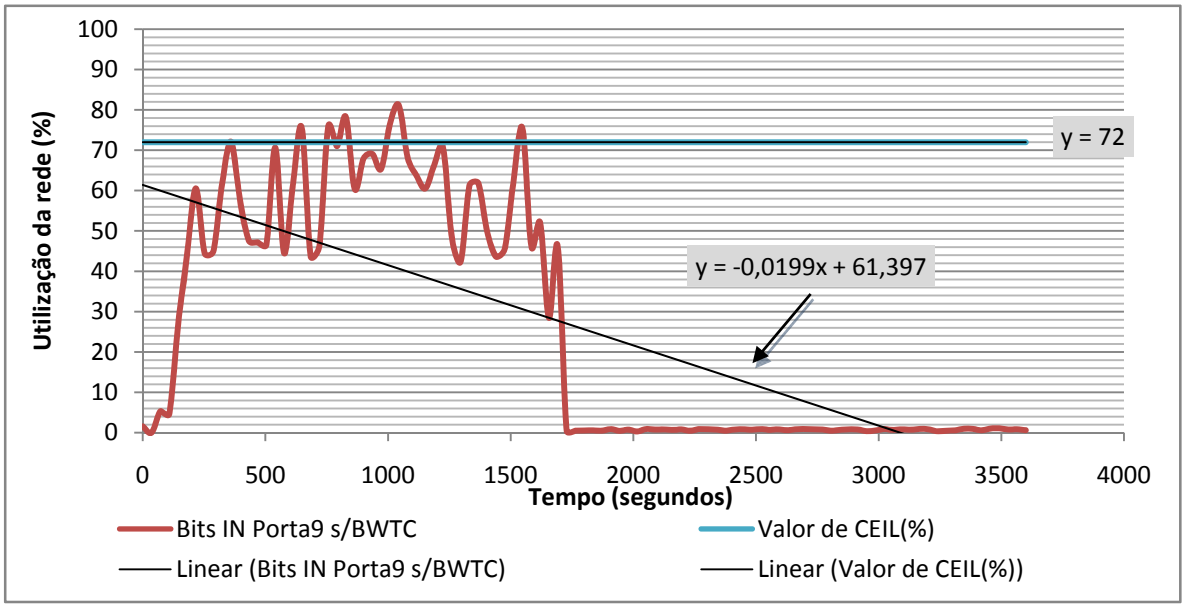

Figura 78: Débito de dados do SIMG (Porta 9; SWITCH_01) – valores discretos

Tabela 20: Resultados experimentais do SIMG no cenário_4_sem_BWTC (valores discretos)

Variáveis	Bits IN Porta9 s/BWTC(%)	Valor de CEIL (%)
Utilização média total (0-3600 segundos)	0,949607849	72
Utilização média de 0-1692 (28.2 minutos)	59,08335114	72
Utilização média de 1728-3600 (31.2 minutos)	0,785453796	72
Valor Máximo (valor instantâneo)	81,15612793	72

Na Figura 80 e na Figura 79 está representado o débito de dados do SF para o respectivo cliente. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 21 contém os resultados experimentais, relativamente à 2ª figura.

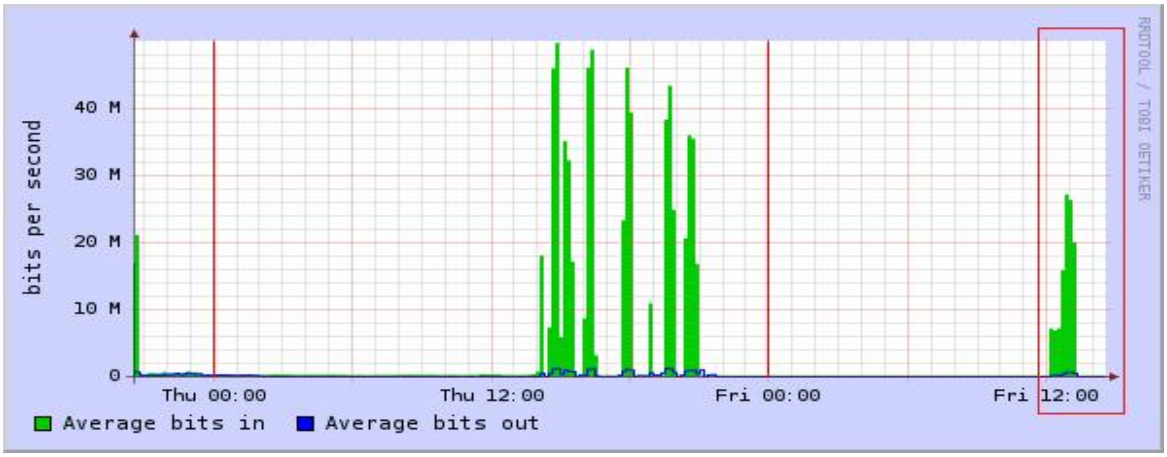

Figura 80: Débito de dados do SF (Porta 12; SWITCH 01) - média de 5 minutos de utilização

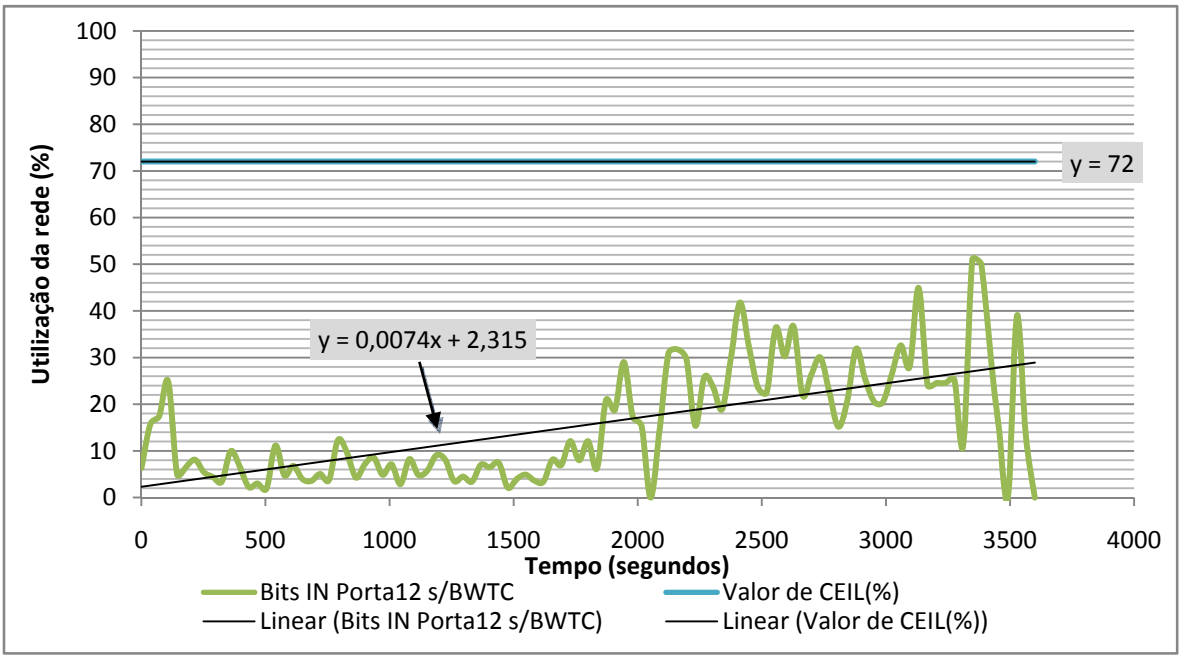

Figura 79: Débito de dados do SF (Porta 12; SWITCH_01) – valores discretos

Tabela 21: Resultados experimentais do SF no cenário_4_sem_BWTC (valores discretos)

Variáveis	Bits IN Porta12 s/BWTC (%)	Valor de CEIL (%)
Utilização média total (0-3600 segundos)	12,06345367	72
Utilização média de 0-1692 (28.2 minutos)	5,586791992	72
Utilização média de 1728-3600 (31.2 minutos)	24,10869598	72
Valor Máximo (valor instantâneo)	50,94473267	72

Na Figura 81 e na Figura 82 está representado o débito de dados no *uplink* do SWITCH_01. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 22 contém os resultados experimentais, relativamente à 2ª figura.

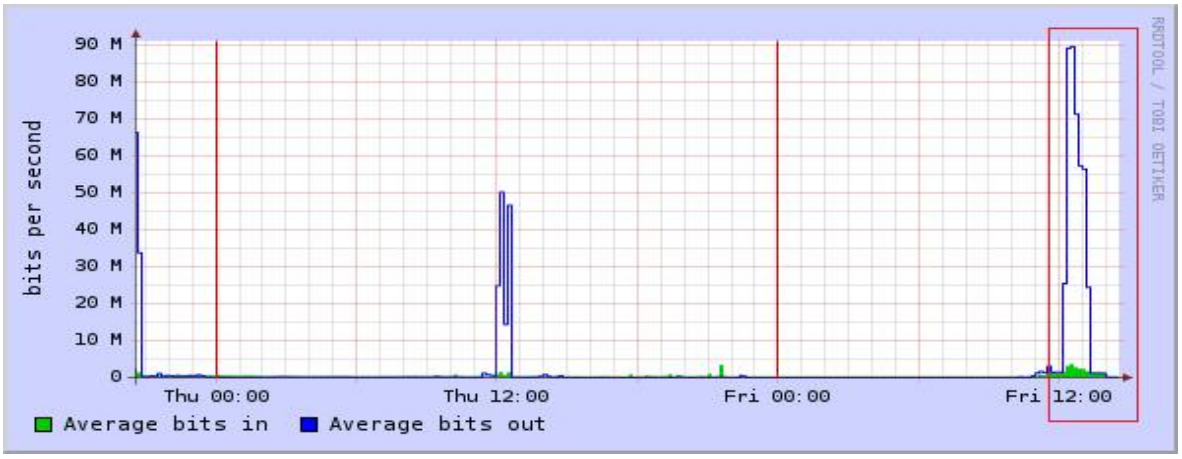

Figura 81: Débito de saída de dados da porta 24 (uplink) do SWITCH_01 - média de 5 minutos

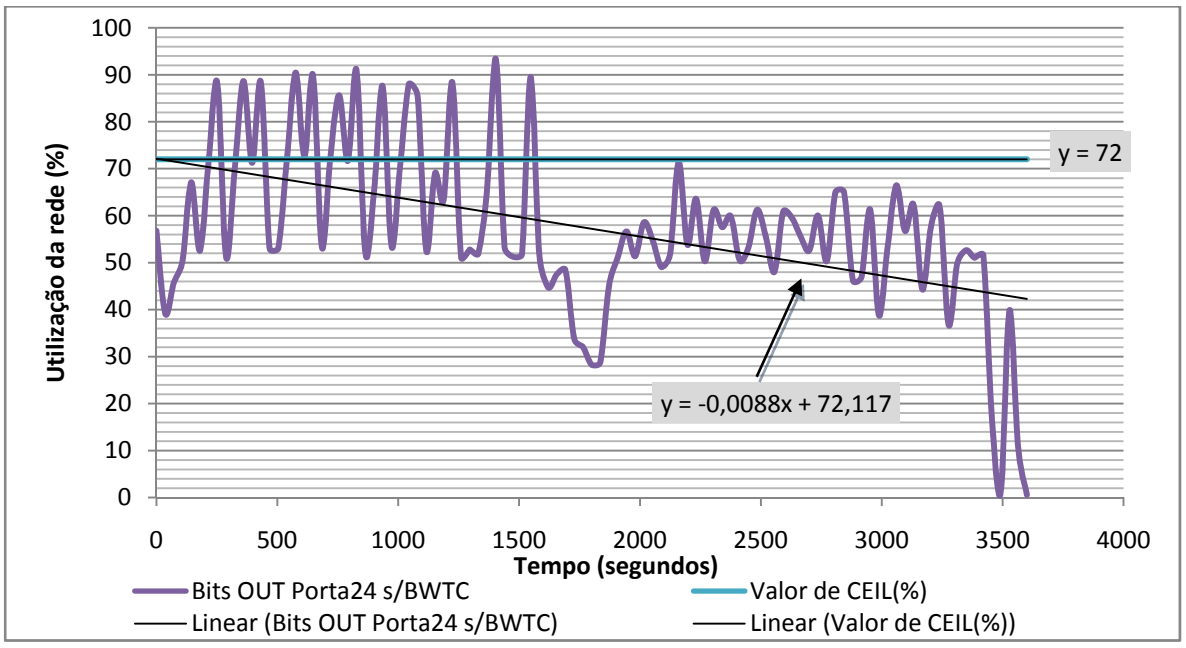

Figura 82: Débito de entrada de dados na porta 24 do SWITCH_01

Tabela 22: Resultados experimentais do UPLINK no cenário_4_sem_BWTC (valores discretos)

Variáveis	Bits OUT Porta24 s/BWTC (%)	Valor de CEIL (%)
Utilização média total (0-3600 segundos)	53,49953461	72
Utilização média de 0-1692 (28.2 minutos)	65,61542511	72
Utilização média de 1728-3600 (31.2 minutos)	52,58904266	72
Valor Máximo (valor instantâneo)	93,30987549	72

Tabela 23: Resultados experimentais do cenário_4_sem_BWTC[6]

Características			Débito Máximo (Mbps)			Débito Médio (Mbps) (0 a 3600 segundos)		
Servidor/ UPLINK	Número da porta	Duração do teste	Débito máximo do SW (Mbps)	Débito máximo do SIMG	Débito máximo do SF (Mbps)	Débito médio do SW (Mbps)	Débito médio do SIMG (Mbps)	Débito médio do SF (Mbps)
SW	07	00H59	30			9.95		
SIMG	09	00H29		76			0.95	
SF	12	00H59			27			12.06
UPLINK	24	00H59	90			53,5		

8.4.2. ANÁLISE DOS RESULTADOS – CENÁRIO_4_SEM_BWTC

Neste cenário de teste, cenário_4_sem_BWTC, pretende-se estudar qual é a influência da introdução do tráfego de imagens no cenário_3_referência. Ao mesmo tempo, este cenário de teste permite verificar as diferenças com o cenário_2_sem_BWTC, uma vez que têm características semelhantes (Tabela 13 e Tabela 18).

Comparando os resultados experimentais dos cenários 2 e 4, ambos sem BWTC, facilmente se verifica que no cenário 4 as tarefas associadas ao SW e SF sofreram elevadas penalizações relativamente ao cenário 2. O tempo de execução passou de 19 minutos no SW e 28 minutos no SF, para 59 minutos em ambos os casos, reflectindo um aumento na ordem dos 210% para o caso do SW e 110% no caso do SF. O mesmo se verifica nos débitos máximos atingidos por ambos, o SW teve uma diminuição de 56Mbps no cenário_2_sem_BWTC para 30Mbps no cenário_4_sem_BWTC (46%), ao passo que o SF verificou uma diminuição de 43,5Mbps para 27Mbps (38%). Pode-se constatar que a tarefa associada ao SF (na qual são feitos *downloads* aleatórios, de ficheiros de diferentes dimensões), sofreu menor impacto do que a tarefa associada ao SW (na qual são feitos *downloads* lineares, de poucos ficheiros de elevadas dimensões). Estas constatações são fortalecidas com os valores obtidos para o débito médio dos diferentes servidores, disponíveis na Tabela 23. A tarefa associada ao SIMG teve uma diminuição do débito máximo, de 87Mbps no cenário_2_sem_BWTC, para 76Mbps no cenário_4_sem_BWTC

[6] Os valores apresentados na tabela, relativamente aos débitos máximos, foram obtidos dos gráficos provenientes do Cricket (fundo azul); os débitos médios foram obtidos do módulo de software BWTC_STATISTICS, relativamente ao tempo total de execução do cenário de testes. Os dados obtidos pelo BWTC_STATISTICS estão nos ficheiros Excel do anexo "Teste04072008_12H15_sem_HTB", do CD anexado à tese. Os tempos de execução foram obtidos do vídeo contido no anexo "Teste04072008_12H15_sem_HTB", do CD anexado à tese.

(12,6%). Mesmo assim é menor do que os 46% do SW, e 38% do SF. No SIMG, o tempo de execução teve um aumento de apenas 16%, ao contrário dos 210% e 110% dos SW e SF, respectivamente.

Comparando os resultados experimentais dos cenários 3 (referência, sem tráfego de imagens) e 4 (com tráfego de imagens), ambos sem BWTC, é claro e notório o impacto negativo nas tarefas associadas aos SF e SW, resultante da introdução da tarefa de instalação de imagens de software. Apesar dos débitos máximos atingidos pelos SF e SW terem sofrido pequenas alterações, verifica-se que os tempos de execução aumentaram de 37 minutos para 59 minutos no SW (60%) e de 35 minutos para 59 minutos no SF (69%). Verificou-se também um aumento de 51% (de 59.5Mbps para 90Mbps) no débito máximo atingido na porta de *uplink* (porta 24) do SWITCH_01. Este cenário de teste é revelador da situação de congestionamento de tráfego na porta de *uplink* (porta 24) do SWITCH_01, uma vez que o débito máximo verificado é de 90Mbps.

Os gráficos obtidos através do BWTC_STATISTICS (fundo branco) fornecem um nível de detalhe na análise do cenário que não é possível de obter nos gráficos do Cricket (com fundo azul).

Com base nos gráficos da Figura 76, da Figura 78, da Figura 79 e da Figura 82 pode-se dividir o cenário 4 em duas fases: a primeira fase, do inicio até ao minuto 28,2 (1692 segundos), a segunda fase até ao final. Como se pode ver nas Figura 78 e Figura 82, o débito máximo de dados (90Mbps) na porta 24 (*uplink*) do SWITCH_01 verifica-se até ao minuto 28,2 (1692 segundos), período de tempo este em que ocorre a tarefa de instalação de imagens de software. Como se pode ver nas Figura 76 e Figura 79, durante o mesmo período de tempo, as tarefas associadas ao SW e SF são gravemente prejudicadas. As Tabela 19 à Tabela 22 contêm os valores dos débitos médios de dados nas duas fases do cenário. Durante a primeira fase do cenário, o SW apresenta um débito médio de dados de apenas 6,1Mbps, e o SF de 5,6Mbps. Ao contrário destes, o SIMG apresenta o débito médio de dados de 59,1Mbps. Durante a segunda fase do cenário a situação tende a inverter-se: como a tarefa associada ao SIMG terminou, este apresenta o débito médio de apenas 0,79Mbps; o *uplink* (porta 24) passa a ter apenas tráfego das tarefas associadas aos SW e SF, e estes atingem os débitos médios de 22,3Mbps e 24,1Mbps, respectivamente.

Com base nos resultados obtidos conclui-se que durante o período de instalação das imagens de software (28,2 minutos), os utilizadores do SW e do SF vêm limitadas a 6,1Mbps e 5,6Mbps as taxas médias de download. A situação agrava-se se for feita a extrapolação deste cenário com 2 computadores clientes do SIMG para um cenário com 10 clientes, no qual as limitações se mantinham durante 2H30.

O cenário retratado nesta secção serve de referência para o seguinte, no qual é introduzido o BWTC para a modelação do débito do servidor de imagens (SIMG) em função da utilização da rede de dados.

8.4.3. RESULTADOS EXPERIMENTAIS – CENÁRIO_4_COM_BWTC

Na Figura 83 e na Figura 84 está representado o débito de dados do SW para o respectivo cliente. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 24 contém os resultados experimentais, relativamente à 2ª figura.

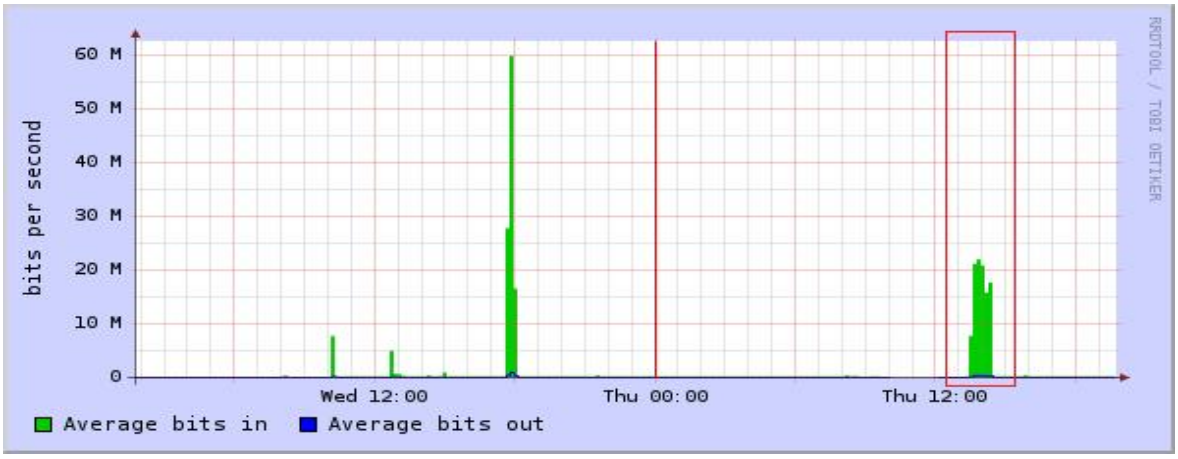

Figura 83: Débito de dados do SW (Porta 7; SWITCH_01) – média de 5 minutos de utilização

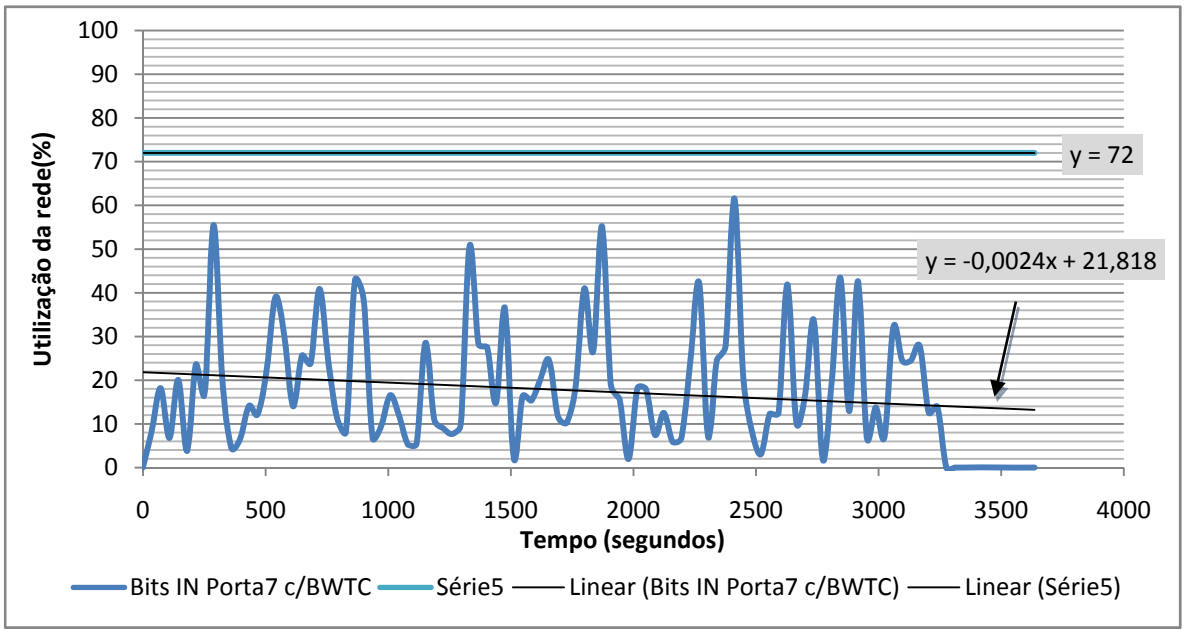

Figura 84: Débito de dados do SW (Porta 7; SWITCH_01) – valores discretos

Tabela 24: Resultados experimentais do SW no cenário_4_com_BWTC (valores discretos)

Variáveis	Bits IN Porta7 c/BWTC(%)	Valor de CEIL(%)
Utilização média total (0-3636 segundos)	14,06095505	72
Utilização média de 0-1692 (28.2 minutos)	15,80331039	72
Utilização média de 1728-3636 (31.8 minutos)	12,76485443	72
Valor Máximo (valor instantâneo)	61,55148315	72

Na Figura 85 e na Figura 86 está representado o débito de dados do SIMG para os respectivos clientes. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 24 contém os resultados experimentais, relativamente à 2ª figura.

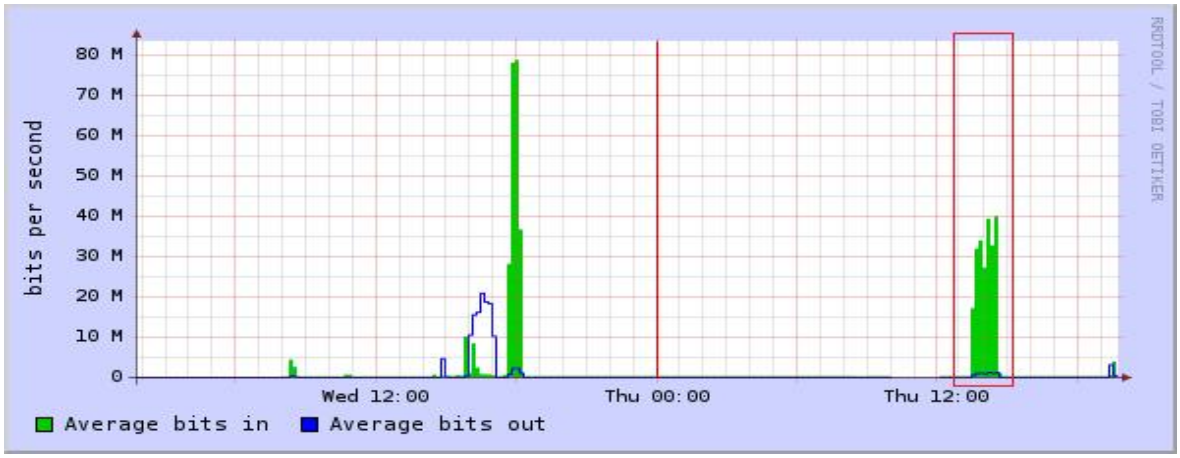

Figura 85: Débito de dados do SIMG (Porta 9; SWITCH_01) – média de 5 minutos de utilização

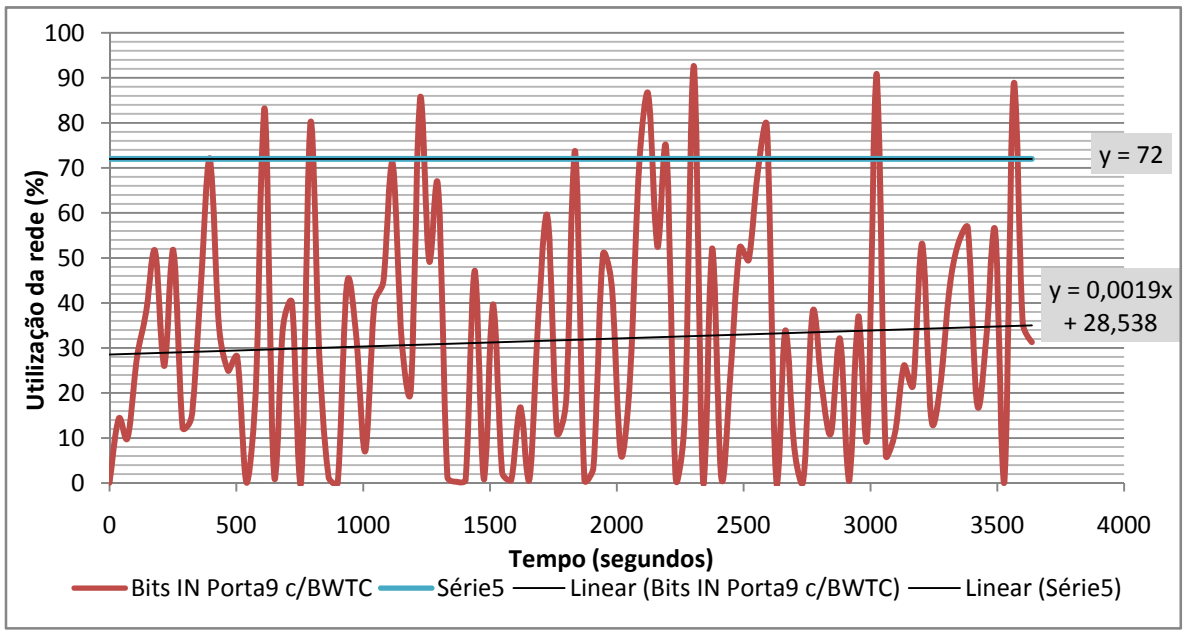

Figura 86: Débito de dados do SIMG (Porta 9; SWITCH_01) – valores discretos

Tabela 25: Resultados experimentais do SIMG no cenário_4_com_BWTC (valores discretos)

Variáveis	Bits IN Porta9 c/BWTC(%)	Valor de CEIL(%)
Utilização média total (0-3636 segundos)	27,81153107	72
Utilização média de 0-1692 (28.2 minutos)	27,62350082	72
Utilização média de 1728-3636 (31.8 minutos)	31,55436707	72
Valor Máximo (valor instantâneo)	92,57241821	72

Na Figura 87 e na Figura 88 está representado o débito de dados do SF para o respectivo cliente. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 26 contém os resultados experimentais, relativamente à 2ª figura.

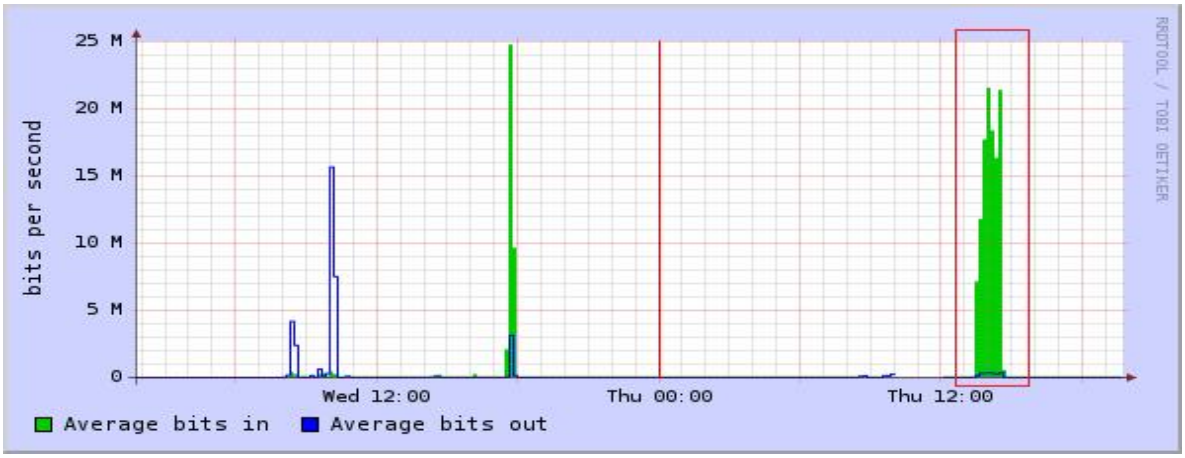

Figura 87: Débito de dados do SF (Porta 12; SWITCH_01) – média de 5 minutos de utilização

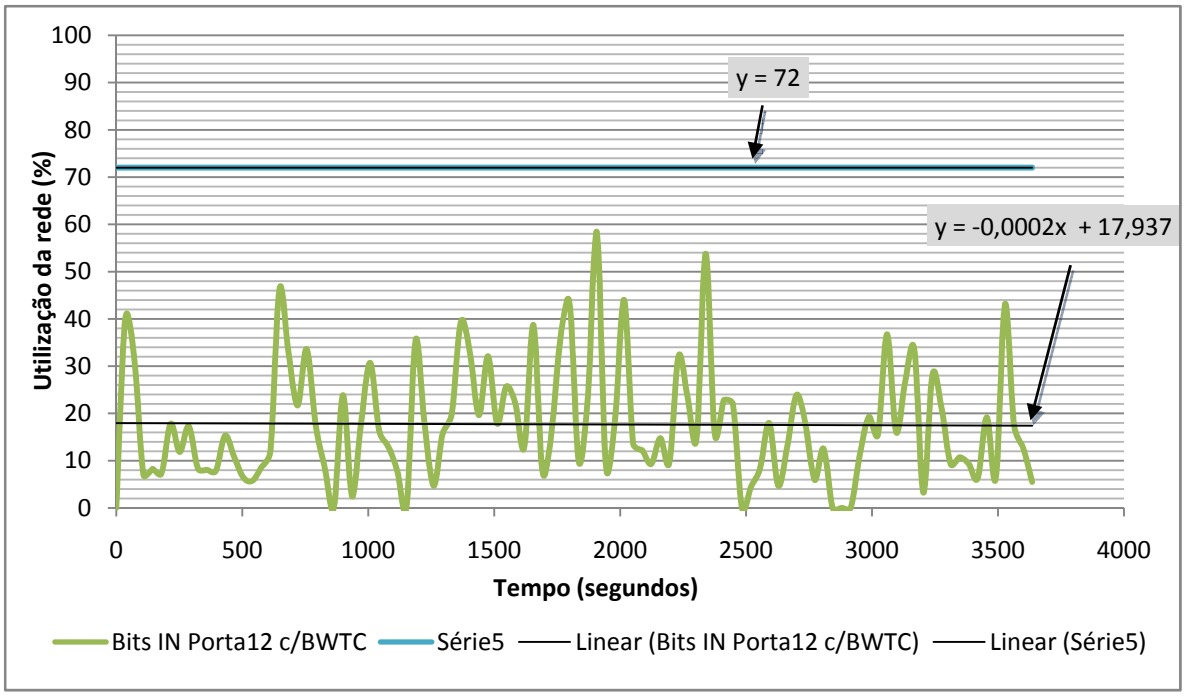

Figura 88: Débito de dados do SF (Porta 12; SWITCH_01) – valores discretos

Tabela 26: Resultados experimentais do SF no cenário_4_com_BWTC (valores discretos)

Variáveis	Bits IN Porta12 c/BWTC(%)	Valor de CEIL(%)
Utilização média total (0-3636 segundos)	15,55102158	72
Utilização média de 0-1692 (28.2 minutos)	16,03796005	72
Utilização média de 1728-3636 (31.8 minutos)	15,14861679	72
Valor Máximo (valor instantâneo)	58,32743835	72

Na Figura 89 e na Figura 90 está representado o débito de dados do SF para o respectivo cliente. Na 1ª figura cada ponto do gráfico resulta da média de utilização dos últimos 5 minutos, ao passo que na 2ª figura os valores são discretos, obtidos com o período de 36 segundos. A Tabela 27 contém os resultados experimentais relativos à 2ª figura.

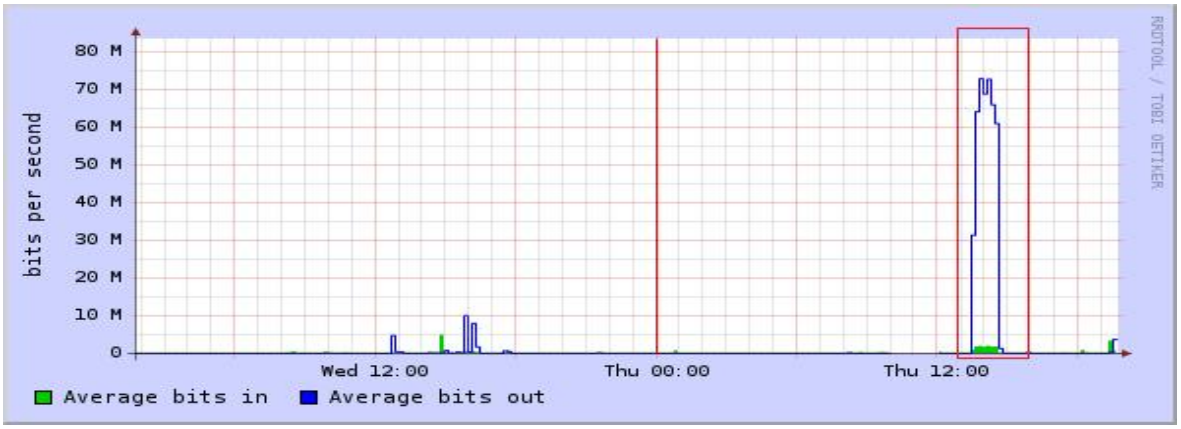

Figura 89: Débito de dados do UPLINK (Porta 24; SWITCH_01) – média de 5 minutos de

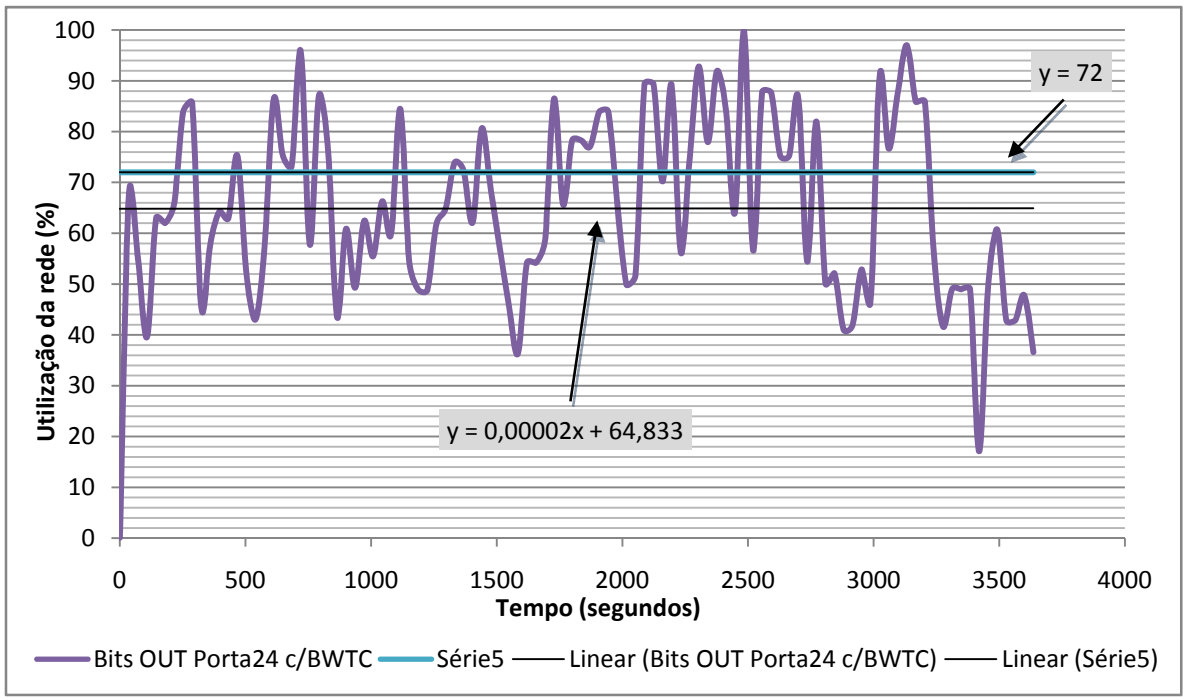

Figura 90: Débito de dados do UPLINK (Porta 24; SWITCH_01) – valores discretos

Tabela 27: Resultados experimentais do UPLINK no cenário_4_com_BWTC (valores discretos)

Variáveis	Bits OUT Porta24 c/BWTC(%)	Valor deCEIL(%)
Utilização média total (0-3636 segundos)	62,99459839	72
Utilização média de 0-1692 (28.2 minutos)	61,94683075	72
Utilização média de 1728-3636 (31.8 minutos)	72,47900772	72
Valor Máximo (valor instantâneo)	99,90219116	72

Tabela 28: Resultados experimentais do cenário_4_com_BWTC[7]

Características			Débito Máximo (Mbps)			Débito Médio (Mbps) (0 a 3636 segundos)		
Servidor/ UPLINK	Número da porta	Duração do teste	Débito máximo Do SW (Mbps)	Débito máximo do SIMG	Débito máximo do SF (Mbps)	Débito médio Do SW (Mbps)	Débito médio do SIMG (Mbps)	Débito médio Do SF (Mbps)
SW	07	00H59	21			14.1		
SIMG	09	00H53		40			27.8	
SF	12	01H01			21,5			15.6
UPLINK	24	01H01	72,5			62,99		

8.4.4. ANÁLISE DOS RESULTADOS – CENÁRIO_4_COM_BWTC

A comparação do cenário actual (cenário_4_com_BWTC) com o cenário anterior (cenário_4_sem_BWTC) deve ser feita em períodos homólogos, ou seja, em intervalos de tempo com características semelhantes. O tempo total de execução dos dois cenários é aproximadamente igual a uma hora o que permite dividir os cenários em duas fases, a 1ª fase do inicio até ao minuto 28,2 (1692 segundos), e deste até ao final representa a 2ª fase. Na 1ª fase ocorre a instalação de imagens de software em ambos os cenários, na 2ª fase só existe no cenário_4_com_BWTC. No Anexo D são apresentados os gráficos do cruzamento de resultados dos testes em análise, o cenário 4, com e sem BWTC.

Na primeira fase do cenário actual, cenário_4_com_BWTC, verifica-se que o débito médio de dados dos SW e SF aumentou em relação ao cenário anterior (cenário_4_sem_BWTC). No SW o débito médio de dados enviados aumentou de 6,1Mbps para 15,8Mbps, um aumento na ordem dos 159%. No SF o débito médio de dados enviados também aumentou, de 5,6Mbps para 16,04Mbps, um aumento de 186%. Estes valores demonstram que o BWTC melhora significativamente a performance dos serviços associados ao SW e SF. Efectivamente, comparando os valores do débito médio de dados dos SW e SF, na segunda fase dos dois cenários (do minuto 28,2 até ao fim), os valores do cenário anterior são superiores aos do cenário actual. Esta diferença resulta do facto de que na 2ª fase do

[7] Os valores apresentados na tabela, relativamente aos débitos máximos, foram obtidos dos gráficos provenientes do Cricket (fundo azul); os débitos médios foram obtidos do módulo de software BWTC_STATISTICS, relativamente ao tempo total de execução do cenário de testes. Os dados obtidos pelo BWTC_STATISTICS estão nos ficheiros Excel do anexo "Teste28082008_13H35_com_HTB", do CD anexado à tese. Os registos podem ser consultados no anexo "Teste28082008_13h35_com_HTB", do CD anexado à tese.

cenário anterior (cenário_4_sem_BWTC), não se encontra a decorrer a instalação de imagens de software, permitindo aos SF e SW atingir débitos superiores.

Como se pode ver nos gráficos e tabelas relativas ao cenário actual, o débito médio de dados dos SW e SF mantém-se praticamente constante ao longo de todo o cenário, ao contrário do que acontece no cenário anterior. Esta constatação está evidente nas rectas de regressão linear traçadas nos gráficos, uma vez que o declive da recta reflecte a convergência do respectivo gráfico, ao longo do tempo. Verifica-se também que a recta com menor declive correspondente ao gráfico do débito de dados na porta de *uplink* do SWITCH_01 (ver a Figura 90). O declive é de 0.002% na Figura 90 do cenário actual, ao passo que na Figura 82 do cenário anterior este é de 0,88%.

Se for considerado o período de tempo total de execução, que é cerca de uma hora, também se verificam melhorias de performance (consultar da Tabela 19 à Tabela 28). Em relação ao cenário anterior, o cenário actual apresenta várias melhorias no valor do débito médio de dados: um aumento de 9.95Mbps para 14,06Mbps do SW (47%); um aumento de 0.95Mbps para 27,8Mbps no SIMG (2829%); um aumento de 12,06Mbps para 15,55Mbps no SF (29%); um aumento de 53,5Mbps para 62,99Mbps na utilização do *uplink* (porta 24) do SWITCH_01 (18%).

O valor definido para o parâmetro *ceil*, valor máximo de utilização de cada porta do activo de rede, é de 72Mbps. No cenário anterior o débito máximo na porta de *uplink* é de 90Mbps. Neste cenário o débito máximo na porta de *uplink* desce para 72,5Mbps (decréscimo de 19,4%), reflectindo uma diferença de apenas 0,5Mbps para o valor pretendido (72Mbps). O controlador integrado no BWTC é do tipo proporcional (com memória, em malha fechada) pelo que a aproximação é bastante razoável.

Perante os resultados dos vários cenários de teste realizados ao software BWTC, pode-se afirmar que este permite atenuar significativamente o impacto negativo da instalação de imagens de software, nos restantes serviços informáticos do DEG. O desempenho destes serviços melhora significativamente, 159% no SW e 186% no SF. Ao mesmo tempo, verifica-se uma utilização mais previsível e menos volátil dos recursos de rede. Com estes valores pode-se afirmar que foi utilizada uma boa abordagem no desenvolvimento do BWTC, o que levará a bom porto os futuros desenvolvimentos.

8.5. RESUMO

O *BandWidth Traffic Control* (BWTC) foi criado com o intuito de atenuar um dos problemas associados à instalação de imagens de software, o congestionamento de tráfego nas intersecções de rede. Para avaliar os resultados aplicacionais do BWTC, foram projectados e realizados vários testes em diversos cenários. Com estes cenários de teste avaliou-se: 1) o desempenho do BWTC em situações de descongestionamento e congestionamento de tráfego, quando está a decorrer a instalação de imagens de software em computadores remotos; 2) a capacidade do BWTC optimizar a utilização das portas dos activos de rede para 72% (valor definido pelo gestor) da capacidade máxima, garantindo deste modo a disponibilidade imediata de largura de banda para aplicações que entretanto arranquem; 3) o impacto nos serviços informáticos do DEG, resultante da instalação imagens de software em computadores remotos (através da rede de dados), sem o BWTC; 4) o impacto nos serviços informáticos do DEG, resultante da instalação imagens de software em computadores remotos, utilizando o BWTC.

O cenário_1_referência é composto por 2 servidores (servidor Web (SW) e servidor de ficheiros (SF)) e 2 computadores clientes (CSW e CSF), todos ligados no mesmo *switch*. Este cenário serve de referência para o cenário_2_sem_BWTC onde é introduzida a instalação de imagens de software. Com base nos resultados experimentais do cenário_1_referência constata-se que o débito máximo de dados dos servidores para os respectivos clientes ronda em média os 50Mbps (50% da capacidade máxima da interface de rede). Esta limitação está relacionada não só com o desgaste do equipamento, mas também com os seguintes factores: i) a capacidade efectiva da interface de rede (*capacity*), que depende do congestionamento de tráfego, do *delay* na entrega dos pacotes de dados, do hardware, etc; ii) os mecanismos de controlo de fluxo do protocolo TCP (por exemplo, o mecanismo da janela deslizante, utilizado no TCP para evitar o congestionamento ponto a ponto); iii) os mecanismos de controlo de congestionamento do protocolo TCP (por exemplo, o mecanismo da janela de congestionamento, utilizado para evitar o congestionamento da rede); e iv) a máquina de estados do protocolo TCP (por exemplo, introduz algum *delay* no encapsulamento das tramas de dados ao longo da pilha protocolar).

O cenário_2_sem_BWTC é composto por 3 servidores (servidor de imagens (SIMG), servidor Web (SW) e servidor de ficheiros (SF)), 4 computadores clientes (1 CSW, 1 CSF

e 2 CSIMG), todos ligados ao mesmo *switch*. Este cenário permite analisar qual é o impacto nos restantes serviços informáticos (SW e SF), resultante da introdução das tarefas de instalação de imagens no cenário_1_referência. Com base nos resultados experimentais pode-se afirmar que não é expressivo o impacto da instalação de imagens de software nos restantes serviços informáticos. No entanto, pode-se constatar que o débito de dados da porta em que está ligado o SIMG atinge o débito de 87Mbps, pelo que no período de tempo em que este máximo ocorre existe apenas 13% de largura de banda livre para novas aplicações que entretanto arranquem. Na porta do SIMG é atingido o valor 87Mbps porque não existe controlo de tráfego (isto é, o mecanismo utilizado no envio dos pacotes é o *pfifo_fast* (utilizado por omissão em Linux)).

O cenário_2_com_BWTC é igual ao cenário anterior, acrescentando-se o controlo de tráfego com o BWTC. Este cenário permite analisar qual o impacto nos restantes serviços informáticos (SW e SF), resultante da introdução do BWTC no cenário_2_sem_BWTC. Relativamente ao cenário anterior pode-se constatar que a introdução do BWTC não influencia significativamente os restantes serviços informáticos (SW e SF). No entanto, observa-se que o débito máximo de dados, na porta em que está ligado o SIMG, diminui de 87Mbps no cenário_2_sem_BWTC para 77Mbps neste cenário (diminuição de 11%). O objectivo da introdução do BWTC neste cenário é limitar a 72% a percentagem máxima de utilização das portas do activo de rede, mesmo numa situação de descongestionamento de tráfego. Apesar do valor obtido (77%) não ser exactamente o pretendido (72%), a aproximação obtida é um indicador positivo do desempenho do BWTC. O BWTC substitui o *pfifo_fast* pelo HTB.

O cenário_3_referência é composto por 2 servidores (SW e SF) e 2 computadores clientes (CSW e CSF), em que os servidores e os clientes estão ligados em *switches* diferentes. Este cenário serve de referência para o cenário 4 onde é introduzida a instalação de imagens de software. Relativamente aos cenários 1 e 2, neste cenário introduz-se o factor congestionamento de tráfego, uma vez que todo o tráfego com destino aos clientes percorre o mesmo trajecto. Com base nos resultados experimentais do cenário_3_referência verifica-se que o débito máximo de dados do SW ronda os 28,5Mbps ao passo que no SF ronda os 32Mbps. Esta limitação está relacionada não só com o desgaste de hardware dos servidores e clientes, mas também com os factores: capacidade efectiva de uma interface

de rede; mecanismos de controlo de fluxo e congestionamento do protocolo TCP; máquina de estados do protocolo TCP.

O cenário_4_sem_BWTC é composto por 3 servidores (SIMG, SW e SF), 4 computadores clientes (1 CSW, 1 CSF e 2 CSIMG), em que os servidores e os clientes estão ligados em *switches* diferentes. Este cenário permite analisar qual o impacto nos restantes serviços informáticos (SW e SF), resultante da introdução das tarefas de instalação de imagens no cenário anterior (cenário_3_referência). Com base nos resultados experimentais verifica-se que as tarefas associadas aos SW e SF sofreram elevadas penalizações, perceptíveis nos tempos de execução que aumentaram de 37 e 35 minutos no cenário anterior, para 59 minutos neste cenário (aumento de 60% no SW e 69% no SF). Como era de prever, o débito máximo verificado na porta de *uplink* (porta 24) passou de 59,5Mbps no cenário anterior, para 90Mbps no cenário actual, revelador de uma situação de congestionamento de tráfego.

O cenário_4_com_BWTC é igual ao cenário anterior (cenário_4_sem_BWTC), com a introdução do BWTC. O BWTC faz a modelação do débito do SIMG, de acordo com a utilização da rede. Este cenário permite analisar qual é o impacto nos restantes serviços informáticos (SW e SF), resultante da introdução do BWTC no cenário anterior. Para a comparação entre os dois cenários mencionados dividiu-se os cenários de teste em duas fases. A fase mais importante é a 1ª fase, em que ocorre a instalação de imagens de software em ambos os casos. Na 1ª fase do cenário actual verifica-se que o débito médio de dados dos SF e SW aumentou em relação à mesma fase do cenário anterior. No SW o débito médio de dados enviados aumentou de 6,1Mbps para 15,8Mbps, um aumento na ordem dos 159%. No SF o débito médio de dados enviado também aumentou, de 5,6Mbps para 16,04Mbps, um aumento de 186%. O débito máximo verificado na porta de *uplink* baixou de 90Mbps no cenário anterior para 72,5Mbps no cenário actual.

Considerações finais:

1) Perante os cenários de teste 1 e 2 e as respectivas análises de resultados pode-se concluir que o BWTC modela o tráfego com destino às portas dos activos de rede, mesmo em situações de descongestionamento de tráfego. Assim, existe largura de banda disponível para as aplicações que entretanto entrem em funcionamento.

2) Perante os cenário de teste 3 e 4 e as respectivas análises de resultados, conclui-se que a introdução das tarefas de instalação de imagens de software afecta drasticamente a performance dos restantes serviços de rede (SW e SF). A introdução do BWTC permite atenuar significativamente o impacto negativo da instalação de imagens, verificando-se melhorias de performance na ordem dos 159% na tarefa associada ao SW, e 186% na tarefa associada ao SF. O BWTC também optimizou a utilização da porta de *uplink* (porta 24 do SWITCH_01), descendo o débito máximo de dados de 90Mbps, no cenário_4_sem_BWTC, para os 72,5Mbps, no cenário_4_com_BWTC (19.4%). Isto reflecte um erro de apenas 0,5Mbps relativamente ao valor máximo de utilização da porta definido pelo gestor (72Mbps). O valor médio de utilização da porta de *uplink* passa de 53,5Mbps para 62,99Mbps (aumento de 18%).

9. CONCLUSÕES

Ao longo desta tese/dissertação foram sendo apresentadas conclusões que permitiram sustentar as opções que foram tomadas. Esta tese surgiu da necessidade de resolver vários problemas de gestão e monitorização da rede informática do Departamento de Engenharia Geotécnica do ISEP.

Para dar resposta às necessidades de gestão e monitorização da rede do DEG desenvolveu-se o *Network Management System with Imaging Support* (NMSIS). Perante a nova aplicação de gestão e monitorização, o NMSIS, pode-se afirmar com segurança que a solução obtida dá resposta não só às necessidades de gestão de uma rede de pequenas dimensões como a do DEG, mas também a redes de maiores dimensões como a do IPP. Esta aplicação incorpora algumas características que lhe fornecem um valor acrescido: a interface Web de gestão, a gestão de contas de utilizadores no acesso à aplicação de gestão, a ferramenta de apoio à manutenção, a ferramenta de gestão de alarmes de rede, a gestão de imagens de software, e o facto de ser modular, permitindo integrar módulos que venham a ser necessários.

A interface Web tem a grande vantagem de permitir o acesso à aplicação de gestão, em qualquer lugar a partir da internet. Um gestor de rede pode estar em viagem e realizar várias operações de gestão sobre a rede, sem que para isso tenha de se deslocar fisicamente

ao local. A título de exemplo, o gestor pode a partir de qualquer parte do mundo, aceder ao módulo de gestão de imagens de software. Depois disto e com poucos cliques no seu PDA, pode criar a tarefa de formatar um edifício inteiro de computadores, ou simplesmente fazer a cópia de segurança de um qualquer computador da LAN. Os computadores serão ligados remotamente pela aplicação de gestão, a tarefa é executada e os computadores desligam-se.

A gestão de contas de utilizadores permite dividir as diferentes tarefas associadas à gestão e monitorização da rede, por vários gestores, com diferentes níveis de acesso. Por exemplo, permite que o Gestor Principal tenha acesso a todas funcionalidades da aplicação de software. Este limita o acesso do Gestor B às estatísticas de tráfego e aos alarmes relacionados com os excessos na rede, e limita o Gestor C a ter acesso ao módulo de apoio à manutenção da rede e à aplicação de gestão de imagens de software. Os níveis de acesso são totalmente configuráveis, de acordo com as necessidades pretendidas. A gestão de contas é especialmente útil em redes de elevadas dimensões, em que existem vários gestores.

A ferramenta de apoio à manutenção permite organizar as tarefas de manutenção que normalmente devem ser executadas numa rede de dados, tal como a limpeza e verificação do sistema de refrigeração de equipamentos críticos (servidores, *routers*, *switches, etc.*), a verificação da cablagem de alimentação, a verificação das fontes de alimentação, entre outros. Esta ferramenta funciona em cooperação com a ferramenta de gestão de alarmes, de modo a que no período de tempo em que está definida uma tarefa de manutenção num qualquer equipamento de rede, não são lançados alertas relativos ao mesmo.

A ferramenta de gestão de alarmes do NMSIS é um dos principais trunfos deste sistema. Esta ferramenta permite, ao gestor principal, definir quais os alarmes que os gestores secundários podem configurar e receber.

Com a gestão de imagens de software poupa-se muito tempo nas tarefas de instalação/reinstalação de software em computadores remotos, evitando a configuração dos computadores um a um. No entanto a instalação de imagens pode facilmente congestionar o tráfego nas intersecções de rede. O *Bandwidth-Traffic Control* (BWTC) é um módulo do NMSIS desenvolvido com o intuito de atenuar o impacto da instalação de imagens de software, na performance dos serviços informáticos de uma rede. A instalação de imagens provoca problemas de congestionamento de tráfego nas intersecções de rede. O BWTC

permite modelar o débito de dados do servidor de imagens, em função da utilização da rede de dados.

Com o intuito de aferir as qualidades e vantagens efectivas da solução desenvolvida, foram projectados e executados vários cenários de teste em ambiente real, procurando simular situações de múltiplos acessos aos servidores Web (SW) e de ficheiro (SF), sem BWTC e com BWTC. Observando os resultados obtidos com o BWTC verificam-se melhorias de desempenho na ordem dos 159%, nos acessos ao servidor Web, e 186%, nos acessos ao servidor de ficheiros. Ao mesmo tempo, o débito médio de utilização da porta de *uplink* do activo de rede em estudo aumenta em 18% e o débito máximo na mesma porta diminui de 90Mbps (no cenário sem BWTC) para 72,5Mbps, reflectido um erro de apenas 0,5Mbps relativamente ao valor pretendido (72Mbps). Uma vez que os cenários decorreram em situações de congestionamento eminente, os valores obtidos são bastante satisfatórios, provam a aplicabilidade do BWTC, e são reveladores do esforço que foi empregue no desenvolvimento desta solução.

9.1. DESENVOLVIMENTOS FUTUROS

Apesar da aplicação desenvolvida cumprir com os objectivos definidos, existem alguns parâmetros que poderão servir de base para o desenvolvimento de trabalhos futuros.

Em relação ao NMSIS. O NAV e o FOG estão funcionalmente integrados, mas não graficamente. É possível aceder ao FOG a partir de *links* existentes na interface Web do NAV. Apesar de ambos terem interface Web, a integração gráfica é importante para criar uma ferramenta totalmente centralizada.

A integração das funcionalidades de correlação de eventos e alarmes seria muito útil no auxílio à detecção, identificação e resolução de problemas de rede.

Os desenvolvimentos futuros no módulo BWTC do NMSIS passam por melhorar o controlador de débito, avaliando os efeitos da introdução de uma componente diferencial, integrativa ou ambas, à componente proporcional já existente.

No BWTC pode-se também analisar o impacto da utilização de serviços diferenciados, bem como a instalação de imagens de software por *multicasting*, sempre que os activos de rede os suportarem.

Outro desenvolvimento importante no módulo BWTC seria o de extrapolar a sua arquitectura para permitir a sua utilização numa rede de quaisquer dimensões. Esta tarefa seria facilitada pelo facto do BWTC já suportar a modelação do débito de dados com destino a qualquer porta de qualquer activo de rede, seja qual for o número de computadores clientes que lhe estão associados (note-se que o activo de rede tem de suportar o protocolo SNMP, a MIB II e a Bridge MIB). Esta extrapolação é importante, uma vez que o tráfego de imagens de software pode percorrer várias intersecções de rede até chegar aos computadores clientes, resultando na introdução de *delay* e *jitter* no encaminhamento dos pacotes. Para identificar o trajecto dos dados até chegar ao computador cliente poderiam explorar-se ferramentas como o *traceroute*, de modo a identificar todas as intersecções de rede até ao computador cliente.

Ao nível do hardware e localização do servidor de imagens, o servidor de imagens do DEG tem uma placa de rede *Fast Ethernet* (100Mbps) e um disco IDE (100Mbps), o que limita as capacidades de instalação de imagens de software. Uma vez que o protocolo TCP é conhecido por apresentar alguns problemas em redes de elevada velocidade [59], a escolha do hardware do servidor deveria ser feita também ela com base em estudos. Por exemplo, numa primeira análise utilizava-se um só servidor de imagens, com uma placa de rede *Gigabit Ethernet* (1Gbps) e um disco da nova geração dos SATA (6.0Gbps). A localização do servidor de imagens deveria ser mudada para um sítio estratégico da rede de dados, com acesso à rede *Gigabit Ethernet* (1Gbps). Numa segunda análise utilizavam-se vários servidores de imagens, com placa de rede *Fast Ethernet* (100Mbps), com um disco da nova geração dos SATA (6.0Gbps), distribuindo a tarefa de instalação de imagens de software por vários servidores estrategicamente localizados na rede, em cooperação uns com os outros.

Para as redes de dados em que o acesso à rede 1Gbps é limitado ou inexistente, seria interessante desenvolver um sistema que realizasse a instalação de imagens de forma distribuída, hierárquica e ramificada, no sentido em que o servidor principal instala a imagem em determinados computadores clientes e estes passavam a servidores de outros clientes, e assim sucessivamente. Deste modo evita-se a saturação do servidor de imagens central, diminui o tráfego de imagens a circular na rede, e acelera o processo de instalação de imagens.

Referências Documentais

[1] LEINWAND, Allan, FANG-CONROY, Karen — Network management, A practical Perspective. 2ªed. Massachusetts. ADDISON-WESLEY PUBLISHING COMPANY, INC. 1996.

[2] MAXWELL, Steve — Unix Network Management Tools. 1ªed. USA. MacGraw-Hill. 1999.

[3] STALLINGS, William — SNMP and SNMPv2: The Infrastruture for Network Management. IEEE Communications Magazine. Março 1998.

[4] HYPERIC, http://www.hyperic.com.

[5] MRTG, http://oss.oetiker.ch/mrtg.

[6] NAGIOS, http://www.nagios.org.

[7] SNORT, http://www.snort.org.

[8] ZABBIX, http://www.zabbix.com

[9] NETDISCO, http://www.netdisco.org.

[10] NAV, http://metanav.uninett.no.

[11] ProCurve Manager, http://www.hp.com/rnd/products/management/ProCurve_Manager_Plus/overview.htm.

[12] OpenNMS, http://www.opennms.org.

[13] Kappler, Cornelia - Building Distributed Management Applications with the IETF Script MIB, IEEE, 2000.

[14] ZIMMERMANN, Hubert, OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection.

[15] CISCO, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/introint.htm.

[16] Arquitectura Hierárquica - http://nms.gdd.net/index.php/Enterprise_NMS_Architectures.

[17] CISCO Press – Network Management Fundamentals, 2007.

[18] ISO/IEC – ISO/IEC 7498-1. 2ªed., 1996.

[19] E. COMER, Douglas - Internetworking with TCP/IP Principles, Protocols, and Architectures, 4ª ed., 2000.

[20] LINUX MAGAZINE – Open Source, Open Standards, 02 Setembro, 2006.

[21] HILL, McGraw – CCNA Cisco Certified Network Associate Study Guide.

[22] HILL, McGraw – Encyclopedia of Networking Telecommunications, May 24, 2001.

[23] CHERKAOUI, Omar, RICO, Nathalie – SNMPv3 can still be simple? , Canada, França, IEEE.

[24] RFC 3584 – Coexistence between version 1, version 2, and version 3 of the Internet – Standard Network Management Framework.

[25] RFC 3577, Remote Network Monitoring, IETF.

[26] RFC 959, File Transfer Protocol, IETF.

[27] IEC, http://www.iec.org/online/tutorials/ems/topic03.html.

[28] ITU-T M3010, International Telecommunication Union.

[29] STALLINGS, William – Data and computer communications. 8ª ed. New Jersey. Person Education, 2007.

[30] TANENBAUM, Anfrew – Computer Networks. 4ª ed. New Jersey. Personal Education International. 2003.

[31] Linhas de orientação para configurar *routers* no NAV – http://metanav.uninett.no/subnetsandvlans#guide_lines_for_configuring_router_interface_descriptions.

[32] HELD, Gilbert – Ethernet Networks: Design, Implementation, Operation, Management.4ª ed. JOHN WILEY & SONS, LTD. 2003.

[33] CHADDA, Ankur – Quality of Service Testing Methodology, Thesis, University of Bombay(Mumbai), India, 1999.

[34] Clonezilla, http://clonezilla.org.

[35] Free Open Source Ghost, http://freeghost.no-ip.org.

[36] Ghost For Unix, http://www.feyrer.de/g4u/.

[37] Linbox Rescue Server, http://www.lrs.linbox.org.

[38] Mondo Rescue, http://www.mondorescue.org.

[39] Partimage, http://www.partimage.org.

[40] Trinity Rescue Kit, http://trinityhome.org/trk/.

[41] Assent Management Software, HTTP://WWW.BESTPRICECOMPUTERS.CO.UK/GLOSSARY/ASSET-MANAGEMENT-SOFTWARE.HTM.

[42] INTEL, SYSTEMSOFT – Preboot Execution Environment (PXE) Specification, Version: 2.1. Setembro 1999.

[43] 3COM Corporation – What is PXE? , 3COM Corporation, Corporation Headquarter, Santa Clara, CA 95052-8145, 2001.

[44] Wired for Management (WfM) Technology Specifications, http://www.intel.com/design/archives/wfm/index.htm.

[45] GUID, http://msdn.microsoft.com/en-us/library/cc246027.aspx.

[46] BROWN, Martin A. – Traffic Control HOWTO. Version: 1.0.2, Outubro de 2006.

[47] BROWN, Martin A. – Traffic Control using tcng and HTB HOWTO. Version: 1.0.1, Abril de 2006.

[48] ALMESBERGER, Werner – Linux Network Traffic Control – Implementation Overview, 23 de Abril de 1999.

[49] STANIC, Milan P. – TC, Traffic Control, Linux QoS Control Tool, http://www.rns-nis.co.yu/~mps/linux-tc.html.

[50] ALMESBERGER, Werner – Linux Network Traffic Control, EPFL ICA, April 23, 1999.

[51] CLARK, David.; SHENKER, Scott; ZHANG, Lixia - Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism, Proceedings of SigComm'92, Baltimore, MID, August 1992, http://ana-www.lcs.mit.edu/anaweb/ps-papers/csz.ps

[52] IETF, Integrated Services (*intserv*) working group. http://www.ietf.org/html.charters/OLD/intserv-charter.html

[53] IETF, Differentiated Services (*diffserv*) working group. http://www.ietf.org/html.charters/OLD/diffserv-charter.html

[54] Bernet, Yoram; Yavatkar, Raj; Ford, Peter; Baker, Fred; Zhang, Lixia; Nichols, Kathleen; Speer, Michael; Braden, Bob - Interoperation of RSVP/Int-Serv and Diff-Serv Networks (work in progress), Internet Draft draft-ietf-diffserv-rsvp-02.txt, February 1999.

[55] HUBERT, Bert, GRAF, Thomas, MAXWELL, Gregory, VAN OOTERHOUT, Martijn, SCHROEDER, Paul, SPAANS, Jasper, LARROY, Pedro – Linux Advanced Routing & Traffic Control HOWTO, October 2003.

[56] REN, Tao, DIMIROVSKI, Georigi, JING, Yuan-wei – ABR Traffic Control over ATM Network Using Fuzzy Immune-PID Controller, Minneapolis, Minnesota, USA, June 2006.

[57] CHANG LEE, Kyung, LEE, Suk, HEE LEE, Hong - Implementation and PID tuning of network-based control systems via Profibus polling network, Elsevier B.V., 2003.

[58] XIONG, Jintao - PID: A Queue Management Scheme for Improving Network Resilience Under worm Attacks, School of Engineering, University of Turabo, IEEE 2007.

[59] MAMATAS, Lefteris, HARKS, Tobias, TSAOUSSIDIS, Vassilis - Approaches to Congestion Control in Packet Networks, JOURNAL OF INTERNET ENGINEERING, VOL. 1, NO. 1, JANUARY 2007.

[60] VIANA, Paula – Interligação e Gestão de Redes – Protocolos de Transporte, 2004;

[61] Net:Iproute2 - http://www.linuxfoundation.org/en/Net:Iproute2;

[62] IPROUTE2 Utility Suite Howto - http://www.policyrouting.org/iproute2.doc.html;

[63] Linux Advanced Routing & Traffic Control HOWTO - http://lartc.org/howto/;

[64] Firewall and Advanced Routing Under Linux - http://ornellas.apanela.com/dokuwiki/pub:firewall_and_adv_routing;

[65] ARRIZABALAGA, Saio, SALTERAIN, Antonio – QoS Management: Differentiating Services and Practical Views, CEIT and Tecnum (University of Navarra), Spain, IEEE, 2006.

[66] SEMERIA, Chuck - Supporting Differentiated Service Classes: Queue Scheduling Disciplines Juniper Networks. December 2001.

[67] Intel, Understanding WOL - http://www.intel.com/support/network/sb/CS-000084.htm, October, 2006.

[68] LIEBERMAN, Philip - Wake on LAN Technology - http://www.liebsoft.com/index.cfm/whitepapers/Wake_On_LAN, 2002.

[69] Intel Centrino Mobile Technology Wake on Wireless LAN (WoWLAN) Feature - http://www.intel.com/network/connectivity/resources/doc_library/tech_brief/wowlan_tech_brief.pdf.

[70] Throughput, http://www.babylon.com/definition/throughput/Portuguese.

[71] Stateless and Stateful - http://whatis.techtarget.com/definition/0,,sid9_gci213051,00.html.

[72] Proposta de tese de Hélder Mendes - http://ave.dee.isep.ipp.pt/~jtm/Teses/JBM1.pdf

[73] FEIBEL, Werner - Encyclopedia of Networking, Second Edition, 1996.

[74] Sourcefire - http://investor.sourcefire.com/phoenix.zhtml?c=204582&p=irol-newsArticle&ID=1041607.

[75] HP Network Node Manager (NNM) - https://h10078.www1.hp.com/cda/hpms/display/main/hpms_content.jsp?zn=bto&cp=1-11-15-119^1155_4000_100__.

[76] Tivoli NetView - http://www-01.ibm.com/software/tivoli/products/netview/.

[77] Intel LANDesk Management Suite - http://www.landesk.com/solutionservices/default.aspx.

[78] Modelos de Referência – OSI e TCP/IP, Prof. Pedro Assis.

[79] Diskless Remote Boot in Linux (DRBL) - http://drbl.sourceforge.net/.

[80] PARTCLONE - http://partclone.nchc.org.tw/.

[81] DD - http://www.gnu.org/software/coreutils/manual/html_node/dd-invocation.html.

[82] NTFSCLONE - http://man.linux-ntfs.org/ntfsclone.8.html.

[83] UDPCAST - http://udpcast.linux.lu/.

[84] MARTINS, Ricardo – Arquitecturas e Ferramentas de Gestão de Redes e Sistemas, Universidade de Aveiro, 2001.

[85] DRBL-WINROLL - http://drbl-winroll.sourceforge.net/.

[86] UNINETT - http://www.uninett.no/index.en.html.

[87] NAV / Technical Documentation - http://metanav.uninett.no/navtechdoc.

[88] JBoss Application Server - http://www.jboss.org/jbossas/.

[89] An Overview and Example of the Buffer-Overflow Exploit - http://iac.dtic.mil/iatac/download/Vol7_No4.pdf.

[90] Stealth Port Scans - http://www.icir.org/vern/papers/norm-usenix-sec-01-html/node8.html.

[91] Vulnerability in Server Message Block - http://www.securiteam.com/windowsntfocus/5XP092KEUG.html.

[92] Block OS Fingerprinting - http://oreilly.com/pub/h/1347.

[93] SnortSnarf - http://www.securityfocus.com/tools/1603.

[94] Sguil - http://sguil.sourceforge.net/.

[95] Open Source Security Information Management (OSSIM) - http://www.ossim.net/.

[96] Basic Analysis and Security Engine (BASE) - http://base.secureideas.net/.

[97] Clam AntiVirus (ClamAV) - http://www.clamav.net/

[98] SPADE - http://www.spade.com/linux/

[99] MySQL - http://www.mysql.com/.

[100] PostgreSQL - http://www.postgresql.org/.

[101] AJAX - http://www.uie.com/articles/ajax/.

[102] MASON - http://www.masonhq.com/.

[103] SQLITE - http://www.sqlite.org/.

[104] ORACLE - http://www.oracle.com/index.html.

[105] Cisco Discovery Protocol (CDP) - http://www.cisco.com/en/US/docs/ios/12_1/configfun/configuration/guide/fcd301c.html.

[106] NAV Users - http://metanav.uninett.no/navusers.

[107] Procurve - http://www.procurve.com/.

[108] NSClient - http://nsclient.ready2run.nl/.

[109] RRDTool - http://oss.oetiker.ch/rrdtool/.

[110] Cricket - http://cricket.sourceforge.net/.

[111] Partimage Live CD - http://www.sysresccd.org/.

[112] Logical Volume Management (LVM) - http://docs.hp.com/en/B2355-90684/lvm.7.html.

[113] TightVNC - http://www.tightvnc.com/.

[114] SAMBA - http://us6.samba.org/samba/.

[115] XFS – A High-performance journaling filesystem - http://oss.sgi.com/projects/xfs/.

[116] ReiserFS - http://www.funtoo.org/en/articles/linux/ffg/1/.

[117] QTPARTED - http://qtparted.sourceforge.net/.

[118] Partition-Image - http://www.osalt.com/partition-image.

[119] Midnight Commander - http://www.ibiblio.org/mc/.

[120] FREEMAN, Roger L. – Telecommunication System Engineering, 4ªed., John Wiley & Sons, Inc., Hoboken, New Jersey, 2004.

[121] RFC 3176, SFLOW, IETF.

[122] COMODO Firewall, http://www.personalfirewall.comodo.com/.

[123] NAV – Machine Tracker, http://metanav.uninett.no/machinetracker.

[124] NAV – Maintenance, http://metanav.uninett.no/maintenancetasks.

[125] NAV – Arnold, http://metanav.uninett.no/arnold.

[126] NAV – User Administration, http://metanav.uninett.no/useradminpanel.

[127] NAV – Alert Profiles, http://metanav.uninett.no/alertprofiles.

[128] NAV – Network Explorer, http://metanav.uninett.no/networkexplorer.

[129] OCS-Inventory, http://ocsinventory-ng.org/.

[130] Unix FileSystem (UFS), http://www.apl.jhu.edu/Misc/Unix-info/workshop/fstour.html.

[131] Hierarchical FileSystem (HFS), http://www.computerhope.com/jargon/h/hierfile.htmlepContactLinksDictionaryForums.

[132] Project "/etc/net", http://etcnet.org/.

[133] Transit Networks - http://www.computerhope.com/jargon/t/trannetw.htm.

[134] Definition of the Differentiated Services Field (DS field) in the IPv4 and IPv6 Headers - http://www.ietf.org/rfc/rfc2474.txt.

[135] Lightweight Directory Access Protocol (LDAP) - http://www.ietf.org/rfc/rfc1487.txt?number=1777.

[136] New RENO - http://www.ietf.org/rfc/rfc3782.txt.

[137] BIC – http://netsrv.csc.ncsu.edu/twiki/bin/view/Main/BIC.

[138] TSANG, Estella, CHANG, Rocky – A Simulation Study on the Throughput Fairness of TCP Vegas, The Hong Kong Polytechnic University, IEEE, 2001.

[139] VEGAS – http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.1088.

[140] HA, Sangtae, RHEE, Injong, XU Lisong – CUBIC: A New TCP-Friendly High-Speed TCP Variant, North Carolina State University, University of Nebraska, 2005.

[141] WESTWOOD – http://www.cs.ucla.edu/NRL/hpi/tcpw/.

[142] H-TCP – http://www.hamilton.ie/net/htcp/index.htm.

[143] KUZNETSOV, Alexey, SAROLAHTI, Pasi – Congestion Control in Linux TCP, *Institute of Nuclear Research at Moscow and University of Helsinki, Department of Computer Science.*

[144] Pfifo_fast – http://linux.die.net/man/8/tc-pfifo_fast.

[145] TCP Finite State Machine (FSM) – http://www.tcpipguide.com/free/t_TCPOperationalOverviewandtheTCPFiniteStateMachineF.htm.

[146] LEFEBVRE, J.H., TREURNIET, J. – A Finite State Machine Model of TCP Connections in the Transport Layer, Defence R&D Canada – Ottawa, November 2003.

[147] Theoretical and Real-World Throughput – http://www.tcpipguide.com/free/t_TheoreticalandRealWorldThroughputandFactorsAffecti-2.htm.

Anexo A. Características técnicas do HP Procurve 2510-24

Connectivity

- **10/100 and 10/100/1000 connectivity:** provides customers with the choice to select the network connectivity speed that best meets their needs, with a consistent user experience

- **Gigabit uplinks:**
  - **2510-24 and 2510-48** the 2510-24 has two dual-personality ports for either 10/100/1000 or mini-GBIC connectivity; the 2510-48 has four Gigabit ports, which can all be used concurrently with two 10/100/1000 ports and two open mini-GBIC slots
  - **2510G-24 and 2510G-48** four dual personality for 10/100/1000 or SFP ports for optional fiber connectivity such as Gigabit-SX, -LX, -LH, or 100-FX

- **ProCurve Auto-MDIX:** automatically adjusts for straight-through or crossover cables

Ports

- 24 RJ-45 auto-sensing 10/100 ports (IEEE 802.3 Type 10Base-T, IEEE 802.3u Type 100Base-TX)
- Media Type: Auto-MDIX
- Duplex: half or full
- 2 dual-personality ports
- each port can be used as either an RJ-45 10/100/1000 port (IEEE 802.3 Type 10Base-T; IEEE 802.3u Type 100Base-TX; IEEE 802.3ab 1000Base-T Gigabit Ethernet) or an open mini-GBIC slot (for use with mini-GBIC transceivers)
- 1 RJ-45 serial console port and 10/100/1000 copper ports

Memory and Processor

- MIPS 32 @ 264 MHz, 8 MB flash, 64 MB SDRAM; packet buffer size: 384 KB

on all 10/100

Resiliency and high availability

- **IEEE 802.1s Multiple Spanning Tree:** provides high link availability in multiple VLAN environments by allowing multiple spanning trees; provides legacy support for IEEE 802.1d and IEEE 802.1w

- **IEEE 802.3ad Link Aggregation Control Protocol (LACP) and ProCurve trunking:**
  - 2510-24 supports up to two 10/100 trunks each with four links/ports plus one Gigabit trunk
  - 2510-48 supports up to 24 10/100 trunks with eight links/ports per trunk

- 2510G *switches* support up to 24 trunks with eight links/ports per trunk

Layer 2 switching

- **VLAN support and tagging:** support up to 64 port-based VLANs and dynamic configuration of IEEE 802.1Q VLAN tagging, providing security between workgroups

- **GARP VLAN Registration Protocol:** allows automatic learning and dynamic assignment of VLANs

- **Jumbo packet support:** supports up to 9,216-byte frame size to improve performance of large data transfers

Security

- **Protected ports:** provides increased security by allowing specified ports to be isolated from all other ports on the switch; the protected port or ports can only communicate with the uplinks or shared resources

- **Multiple user authentication methods:**
  - **IEEE 802.1X** industry-standard way of user authentication using an IEEE 802.1X supplicant on the client in conjunction with a RADIUS server
  - **Web-based authentication** similar to IEEE 802.1X, provides a browser-based environment to authenticate clients that do not support the IEEE 802.1X supplicant
  - **MAC-based authentication** client is authenticated with the RADIUS server based on the client's MAC address

- **Multiple IEEE 802.1X users per port:** provides authentication of up to two IEEE 802.1X users per port; prevents user "piggybacking" on another user's IEEE 802.1X authentication

- **BPDU port protection:** blocks Bridge Protocol Data Units (BPDUs) on ports that do not require BPDUs, preventing forged BPDU attacks

- **Secure management access:** all access methods--CLI, GUI, or MIB--are securely encrypted through SSHv2, SSL, and/or SNMPv3

- **TACACS+:** eases switch management security administration by using a password authentication server

Convergence

- **IP multicast (data-driven IGMPv3):** automatically prevents flooding of IP multicast traffic (except 2510-24)

Quality of Service (QoS)

- **IEEE 802.1p prioritization:** delivers data to devices based on the priority and type of traffic

Manageability

- **IEEE 802.1AB Link Layer Discovery Protocol (LLDP):** automated device discovery protocol for easy mapping by network management applications
- **RMON:** provides advanced monitoring and reporting capabilities for statistics, history, alarms, and events
- **Friendly port names:** allow assignment of descriptive names to ports
- **Full-featured console:** provides complete control of the switch with a familiar command-line interface (CLI)
- **Web interface:** allows configuration of the switch from any Web browser on the network
- **Stacking capability:** single IP address management for a virtual stack of up to 16 *switches*, including the ProCurve 2500 series, 2510 series, 2600 series, 2610 series, 2800 series, 2810 series, 2900 series, 3400cl series, 3500yl series, 4200vl series, 6108, 6200yl-24G-mGBIC, and 6400cl series
- **Find-Fix-and-Inform:** finds and fixes common network problems automatically, then informs administrator
- **Dual flash images:** provides independent primary and secondary operating system files for backup while upgrading
- **Software updates:** free downloads from the Web

Flexibility

- **Fanless design:** reduces noise and distractions when deployed in open spaces (2510-24 only)
- **Multiple port density and connectivity speed options:** provide choice and flexibility with a consistent user experience

Management

[redacted]

ProCurve Manager Plus; ProCurve Manager; command-line interface; Web browser; out-of-band management

Standards and protocols

[redacted]

Device Management
HTML and telnet management

General Protocols
IEEE 802.1p Priority
IEEE 802.1Q VLANs
IEEE 802.1s Multiple Spanning Trees
IEEE 802.3ad Link Aggregation Control Protocol (LACP)
IEEE 802.3x Flow Control
RFC 768 UDP
RFC 783 TFTP Protocol (revision 2)
RFC 792 ICMP
RFC 793 TCP
RFC 826 ARP
RFC 854 TELNET
RFC 951 BOOTP
RFC 1542 BOOTP Extensions
RFC 2030 Simple Network Time Protocol (SNTP) v4

IP Multicast
RFC 3376 IGMPv3

MIBs
RFC 1213 MIB II
RFC 1493 Bridge MIB
RFC 1573 SNMP MIB II
RFC 2021 RMONv2 MIB
RFC 2096 IP Forwarding Table MIB
RFC 2613 SMON MIB
RFC 2618 RADIUS Client MIB
RFC 2620 RADIUS Accounting MIB
RFC 2665 Ethernet-Like-MIB
RFC 2668 802.3 MAU MIB
RFC 2674 802.1p and IEEE 802.1Q Bridge MIB
RFC 2737 Entity MIB (Version 2)
RFC 2863 The Interfaces Group MIB

Network Management
IEEE 802.1AB Link Layer Discovery Protocol (LLDP)
RFC 2819 Four groups of RMON: 1 (statistics), 2 (history), 3 (alarm) and 9 (events)
SNMPv1/v2c/v3

Security
IEEE 802.1X Port Based Network Access Control
RFC 1492 TACACS+
RFC 2138 RADIUS Authentication
RFC 2866 RADIUS Accounting
Secure Sockets Layer (SSL)
SSHv1/SSHv2 Secure Shell

Anexo B. Rede de dados do rés-do-chão do DEG

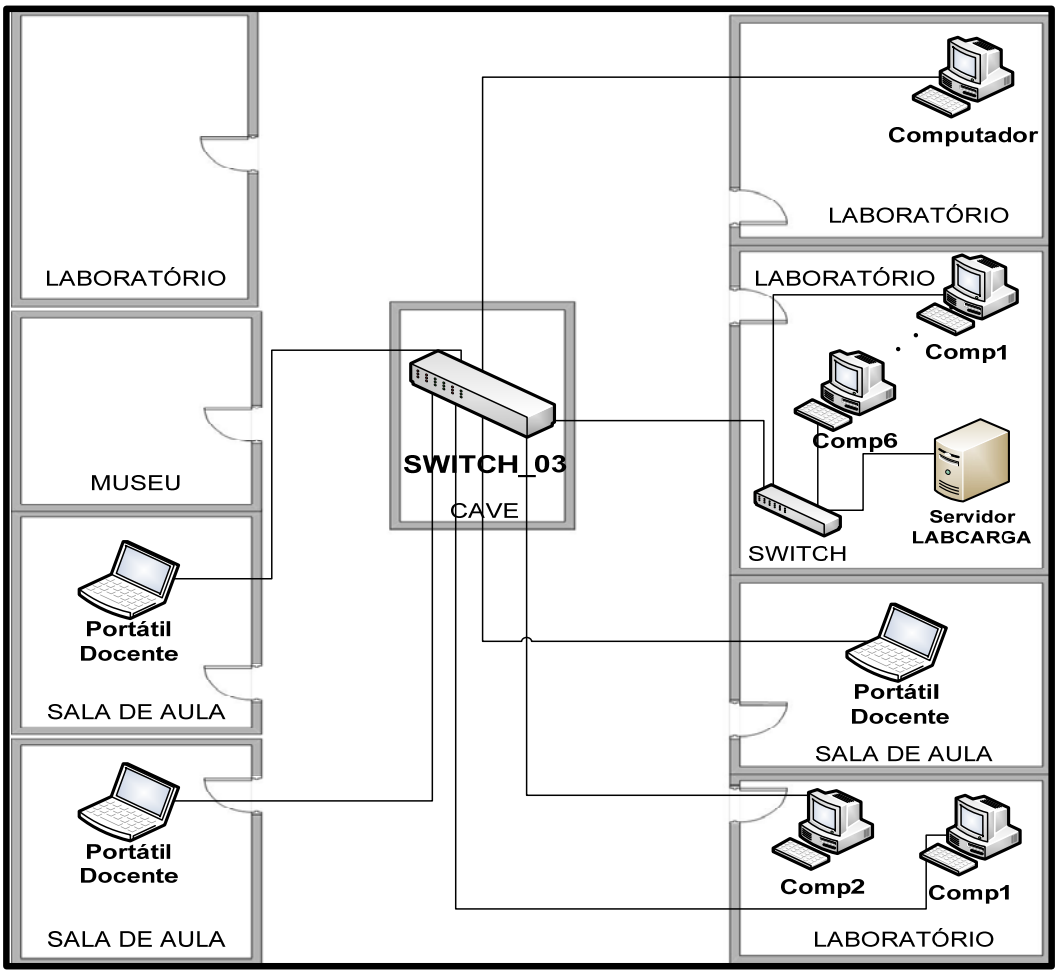

Figura 91: Esquema da rede de dados do rés-do-chão do DEG

Anexo C. Requisitos de instalação e funcionamento do NAV

A lista de dependências de instalação do NAV é a seguinte:

- Make; Autoconfg; Ant; Perl ≥ 1.4.0;Tomcat ≥ 4 (ou um Java Servlet API compatível); JavaSnmp ≥ 1.4 (A.K.A Drexel SNMP); Cheetah Templates;

Para que o NAV funcione têm de estar disponíveis:

- Apache 2; Mod_python; Mod_php ≥ 4; Cricket; PostgreSQL ≥ 7.4; Rrdtool; Cheetah Templates; Tomcat ≥ 4 (ou um Java Servlet API compatível); Java 2 SDK ≥ 1.4.0; Perl ≥ 5.6.0; Python ≥ 2.3.0;
- As seguintes extensões PHP são necessárias: Php-gd; Php-pgsql;
- Os módulos Perl necessários são: DBD::Pg; Pg; SNMP_Session (com SNMP_util, BER)
- As bibliotecas seguintes do Java: PostgreSQL JDBC driver; JavaSnmp >= 1.4 (A.K.A Drexel SNMP);
- Os seguintes módulos Python: pysnmp2; egenix-mxdatetime; forgethtml; forgetsql; Ipy; *Lightweight Directory Access Protocol* (LDAP); psycopg; rrd;
- Para obter alertas por SMS, através de um telefone móvel, são necessários os pacotes: Gammu e python-gammu;
- Ao nível do hardware, a capacidade de processamento de um sistema que suporte o NAV depende principalmente da dimensão da rede, uma vez que varia o número de elementos a processar. Na rede do DEG [2.2] funciona, fluentemente, num Pentium 4 (1,7GHz), com 768MB de memória RAM, 40Gb de disco e placa de rede *Fast Ethernet* (100Mbps).

Principais ficheiros de configuração

Todos os ficheiros de configuração encontram-se na pasta /etc/nav. Antes de realizar a inicialização dos `daemons` e `cron jobs` é necessário configurar os ficheiros `nav.conf` e `db.conf`:

1. No `nav.conf` definir o `ADMIN_MAIL` que será a conta de correio electrónico do administrador NAV;

2. No `nav.conf` definir `DOMAIN_SUFFIX` para o domínio da sub-rede, no caso `deg.isep.ipp.pt`. Definir também o `JAVA_HOME` caso não esteja no `PATH`;
3. No `db.conf` definir `userpw_nav,` que é a senha utilizada para o acesso à base de dados PostgreSQL pelo NAV, mas também no debbugging realizado pelo utilizador numa shell através do `psql manage nav`. Definir mas deixar em branco `userpw_navread` e `userpw_navwrite`.
4. As alterações do aspecto da página de abertura do NAV não são obrigatórias, mas podem realizar-se nos ficheiros contidos na pasta `/etc/nav/webfront.` Podem ser alterados: mensagem de boas vindas, ligações externas, definir atalhos para módulos do software, contactos, entre outros.

Iniciar os daemons e Cron jobs do NAV

Para iniciar os processos back-end do NAV faz-se `nav start` numa shell com permissão de administrador. Para verificar os processos que estão a correr executa-se `nav status.`

```
NAV3:/etc/init.d/nav status
Up: alertengine cricket eventengine getDeviceData
iptrace      logengine mactrace maintengine
networkDiscovery pping servicemon smsd thresholdMon

Down: snmptrapd
```

Se algum dos processos não arrancar pode-se iniciar com `nav start <nome_processo>`. Neste exemplo o snmptrapd não se encontra a correr, pode-se inicia-lo com nav start snmptrapd. Para mais informações sobre os processos *back-end* do NAV consultar [87].

Anexo D. Gráficos de débito de dados no cenário 4

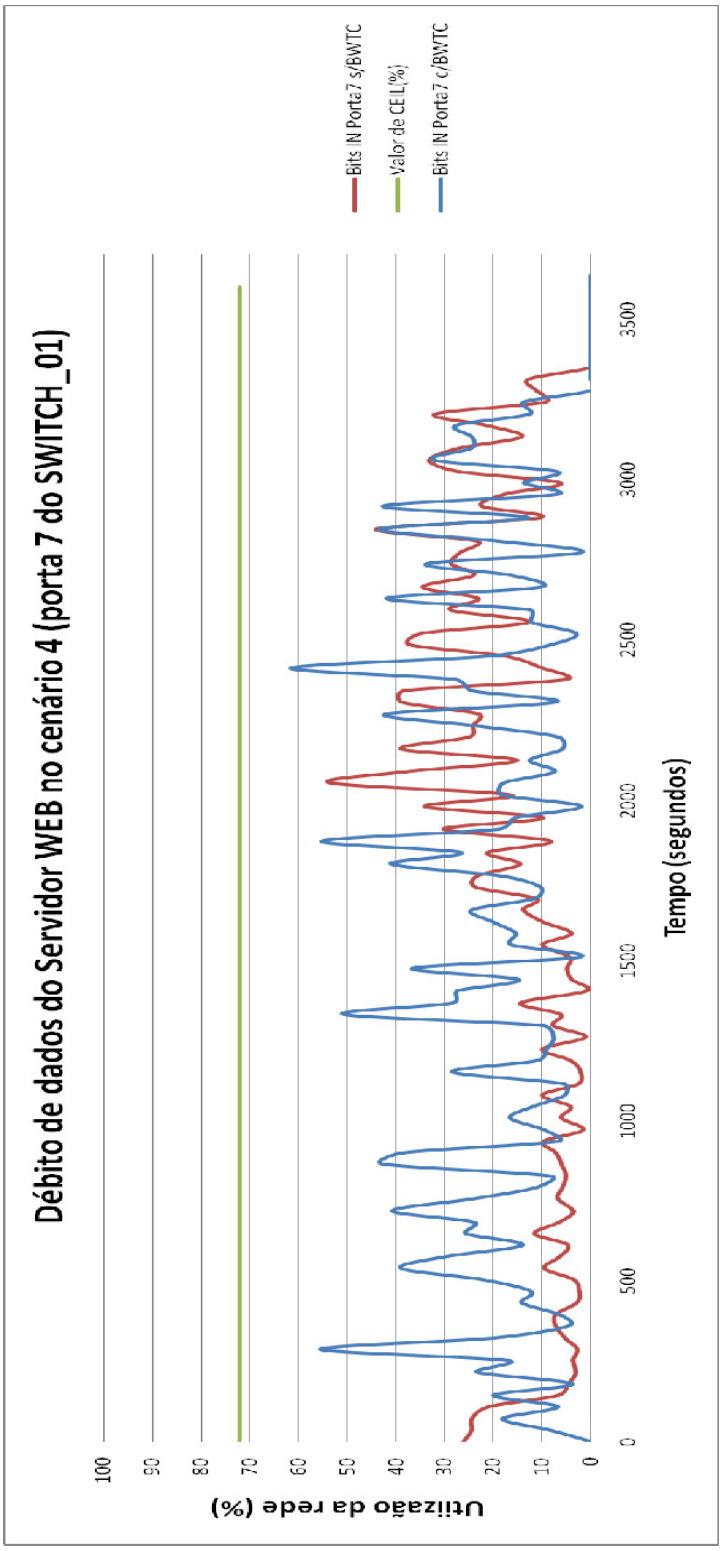

Figura 92: Gráfico do débito de dados do servidor Web (SW) no cenário 4, sem e com BWTC

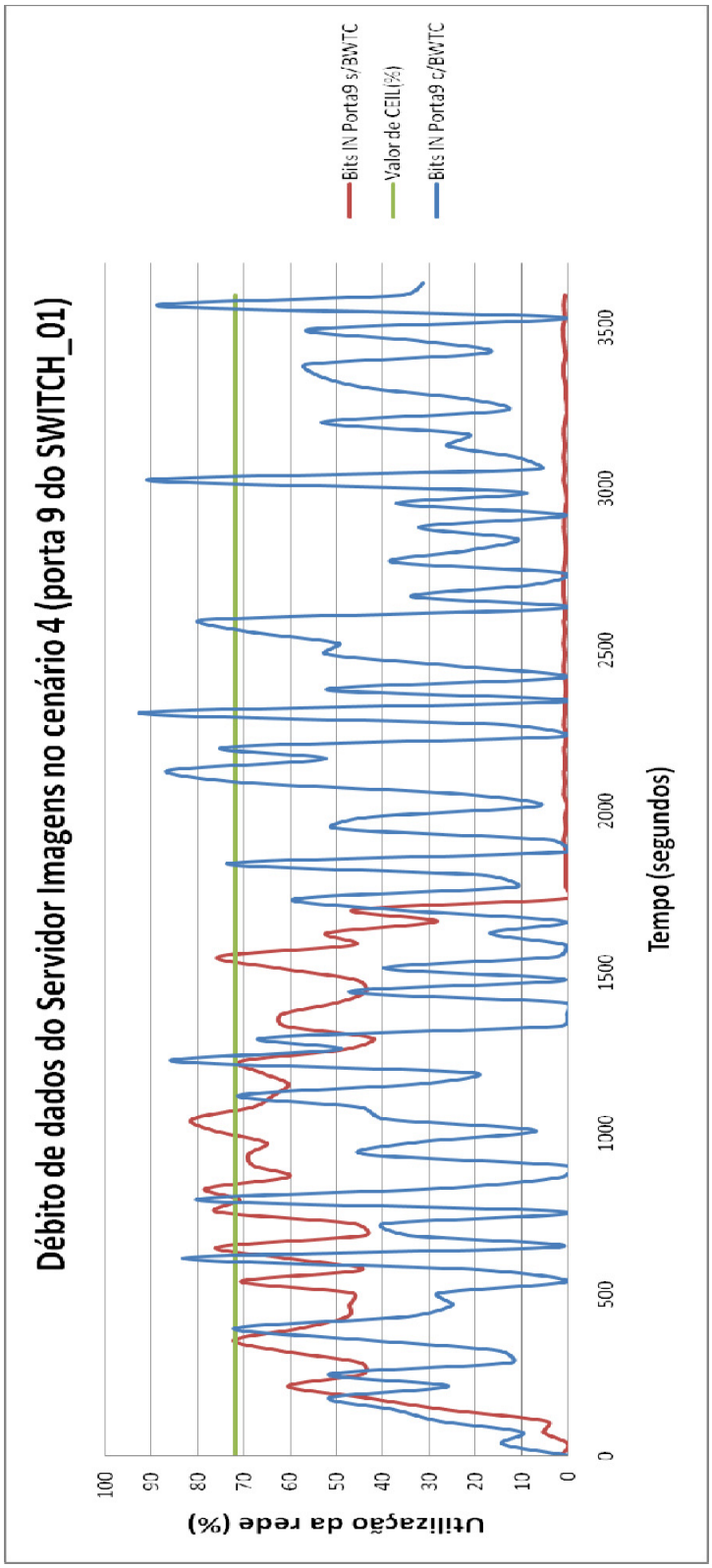

Figura 93: Gráfico do débito de dados do servidor de imagens (SIMG) no cenário 4, sem e com BWTC

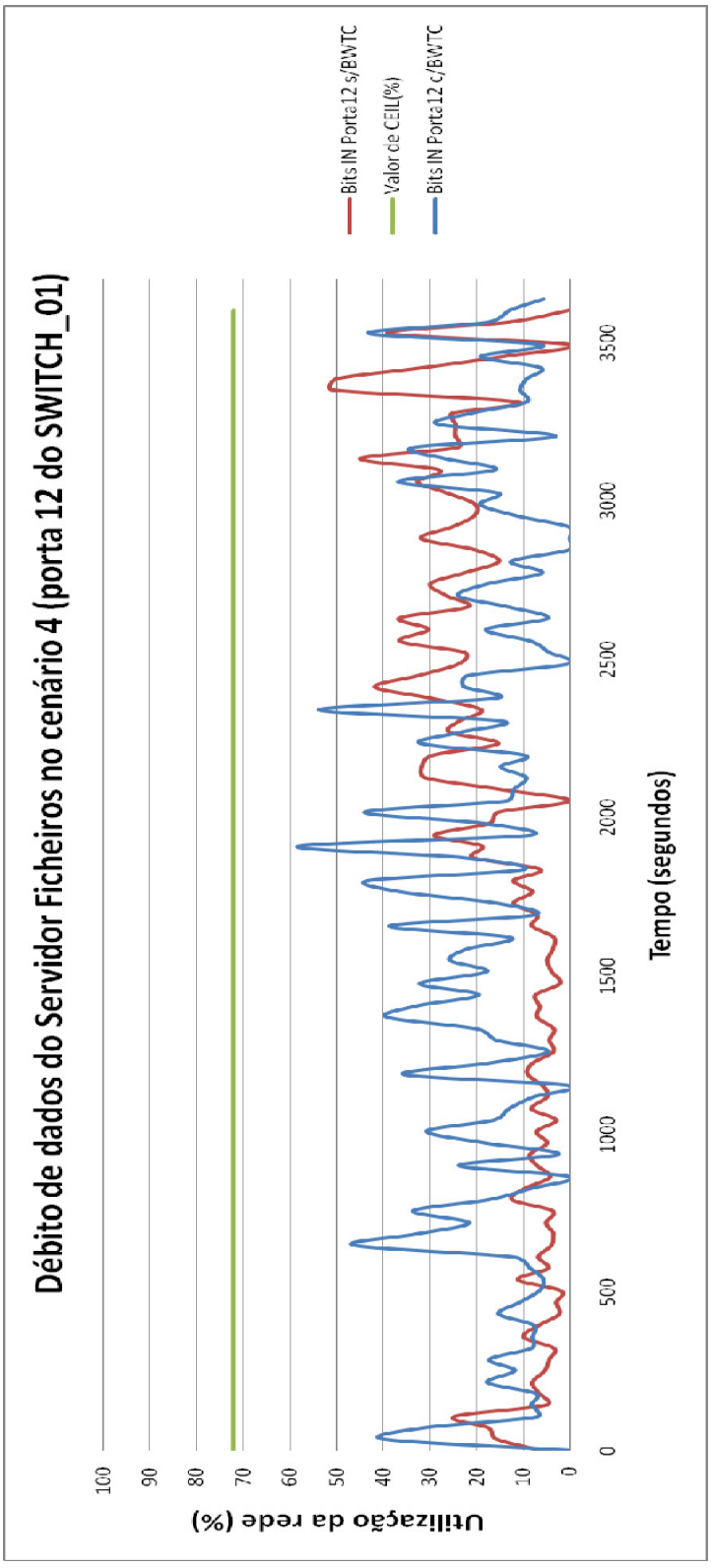

Figura 94: Gráfico do débito de dados do servidor de ficheiros (SF) no cenário 4, sem e com BWTC

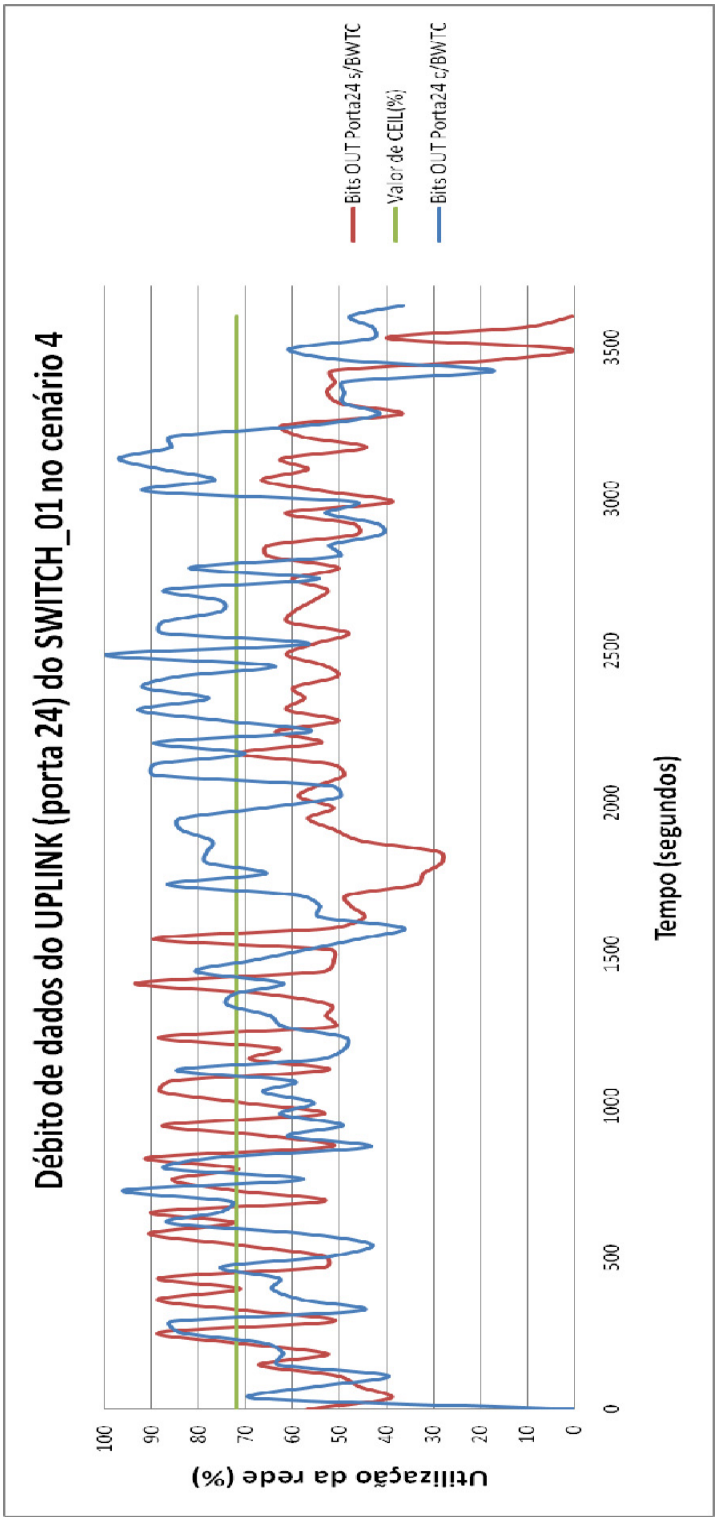

Figura 95: Gráfico do débito de dados na porta de UPLINK do SWITCH_01, no cenário 4, sem e com BWTC

Histórico

- 6 de Setembro de 2008, Versão 1.0, mailto:hvm@isep.ipp.pt
- 15 de Dezembro de 2008, Versão 1.1, mailto:hvm@isep.ipp.pt