Faculdade de Engenharia da Universidade do Porto

IPBrick - Controlo e Monitorização do Parque Informático

André Osório de Castro Ferreira

VERSÃO FINAL

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Telecomunicações

Orientador: Prof. Dr. João Manuel Couto das Neves
Proponente: iPortalMais - Serviço de Internet e Redes Lda.

Julho de 2011

Resumo

O presente trabalho visa a construção de um módulo que, ao ser integrado na IPBrick, permita efectuar a monitorização de todo o parque informático.

A IPBrick é uma solução para servidores de comunicações e de intranet, desenvolvido pela empresa iPortalMais, que se caracteriza por apresentar um sistema operativo baseado no kernel do Linux e oferecer uma interface Web multifuncional através da qual o utilizador consegue configurar de um modo simples e intuitivo todo o sistema.

Neste contexto, surgiu a necessidade de lhe adicionar mecanismos que garantissem a possibilidade não só de monitorizar, mas também de personalizar as opções de monitorização.

Das opções implementadas destacam-se o facto do utilizador poder configurar a IPBrick para funcionar como cliente ou como servidor, definir quais as máquinas e quais os serviços a monitorizar, especificar servidores de monitorização e ainda configurar mecanismos de alertas que permitem a recepção de emails quando ocorre algum evento crítico.

O seu desenvolvimento está inserido num projecto proposto pela empresa iPortalMais à Faculdade de Engenharia da Universidade do Porto, sendo que o grande objectivo deste trabalho é oferecer a qualquer administrador da IPBrick a capacidade de monitorizar uma rede informática de um modo simples e funcional.

Abstract

Enclosed academic work, within professional environment and background support, aims the construction of a module to be integrated at IPBrick thus allowing the monitoring of the entire network.

The IPBrick is a solution for intranet and communication servers developed by iPortalMais enterprise, offering a multifunctional Web interface where the user is able to configure the entire system in an efficient, simple and intuitive way.

Within these settings, comes into sight the need to add mechanisms to the operative system guaranteeing its ability to, not only monitor, but also customize the monitoring options.

From the available options, one enlightens the capability for the user to fully configure the IPBrick to work as either a client or a server, define which machines and services to monitor, specify the monitoring servers as well as configuring the alert mechanisms that display email templates when any sort of problem arises.

Its development and potential implementation is merged in a project proposed by iPortalMais to Faculdade de Engenharia da Universidade do Porto with its major goal being the possibility to offer any IPBrick administrator the capability of monitoring a computer network in an efficient and instinctive manner.

Agradecimentos

Este espaço é dedicado a todas as pessoas que, de uma maneira ou de outra, ofereceram a sua contribuição ao longo de todo este trabalho.

Em primeiro lugar, agradeço a toda a minha família pelo seu apoio, dedicação e paciência ao longo desta dissertação, bem como de todo o percurso académico. Sem o seu sacrifício, dificilmente teria sido possível chegar até aqui.

À minha namorada pelo apoio, força, motivação, amizade e companheirismo.

Ao meu orientador, Prof. Doutor João Neves, por todo o apoio, compreensão, sabedoria, competência e exigência.

A toda a equipa da empresa iPortalMais pela forma como me acompanhou e auxiliou ao longo de todo este trabalho.

Índice

Lista de Figuras

Lista de Tabelas

Abreviaturas e Acrónimos

Lista de abreviaturas e acrónimos:

AJAX	Asynchronous Javascript And XML
API	Application Programming Interface
CGI	Common Gateway Interface
CPU	Central Processing Unit
CSS	Cascading Style Sheets
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Denial of Service
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
gcc	GNU Compiler Collection
GLPI	Gestion Libre de Parc Informatique
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IM	Instant Messaging
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
MIB	Management Information Base
NRPE	Nagios Remote Plugin Executor
OCS Inventory	Open Computer and Software Inventory
PHP	Hypertext Preprocessor

POP3	Post Office Protocol 3
QoS	Quality of Service
RAM	Random Access Memory
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SQL	Structured Query Language
SSL	Secure Sockets Layer
SSH	Secure Shell
TCP/IP	Transmission Control Protocol/Internet Protocol
URL	Uniform Resource Locator
VPN	Virtual Private Network
Web	World Wide Web
XHTML	eXtensible Hypertext Markup Language
XML	Extensible Markup Language

Capítulo 1

Introdução

1.1 - Motivação

Actualmente, existe uma enorme diferença entre as várias organizações ao nível do seu funcionamento interno. No entanto, e para todas elas, um bom desempenho do parque informático e uma boa gestão dos seus recursos são fundamentais para o sucesso do seu negócio. Na verdade, há medida que um determinado negócio cresce, a sua rede aumenta não só em tamanho e complexidade, mas também em importância.

Sendo assim, com um recurso tão valioso, torna-se necessário garantir o seu bom funcionamento através de um controlo e de uma monitorização constante de modo a que seja possível agir proactivamente para evitar o aparecimento de problemas que comprometam a segurança e o desempenho de todo o sistema.

Por estas razões, a motivação para o desenvolvimento deste trabalho tem por base o interesse do autor na área da monitorização de redes, tendo como objectivo perceber as vantagens resultantes da escolha de uma ferramenta deste tipo por parte de uma empresa com negócios dependentes do seu parque informático.

1.2 - Importância da monitorização

A monitorização aqui referida está relacionada com todas as actividades, métodos, procedimentos e ferramentas que são responsáveis pela operação, administração, manutenção e controlo do sistema da rede.

Tal como já foi dito anteriormente, existe uma enorme dependência entre os negócios das empresas e as suas redes informáticas, na medida em que estas desempenham um papel fundamental não só no plano económico, podendo garantir uma redução nos custos de todos os recursos utilizados, como também no plano produtivo, influenciando directamente o

desempenho da empresa no mercado e criando assim vantagens competitivas em relação à concorrência.

No entanto, de modo a maximizar este tipo de vantagens, torna-se necessária uma gestão eficaz e eficiente. Caso contrário, observa-se um impacto negativo no desempenho da empresa, que se pode manifestar no aumento dos custos ou até mesmo em prejuízos significativos.

De entre as várias razões que justificam uma monitorização destacam-se as seguintes:

- Capacidade de saber aquilo que está a acontecer – as ferramentas responsáveis pelo controlo e monitorização mantêm o administrador da rede informado acerca da operacionalidade e conectividade de todos os equipamentos e recursos da rede;
- Planeamento de upgrades ou mudanças – se um dispositivo está constantemente a falhar, ou se a largura de banda de uma determinada ligação está frequentemente a atingir o seu limite, é sinal que algo deve mudar. As aplicações responsáveis pela monitorização permitem detectar este tipo de situações, procedendo-se assim às alterações necessárias;
- Diagnosticar problemas rapidamente – havendo um problema na ligação a um servidor, sem uma monitorização eficiente, torna-se impossível saber se o problema está no servidor, no router ou no switch ao qual o servidor está ligado;
- Visão geral da rede – com este tipo de aplicações há a possibilidade de obter gráficos, diagramas e até estatísticas que resumem todo o funcionamento do sistema, fornecendo uma visão mais pormenorizada do seu estado;
- Garantia de que os sistemas de segurança estão a funcionar correctamente – sem uma monitorização e sem um controlo intensivo, torna-se impossível saber se as ferramentas de segurança (firewalls, antivírus, etc.) estão actualizadas e a funcionar perfeitamente;
- Informação do estado da rede – a grande maioria das aplicações e ferramentas permitem um controlo remoto, tornando assim possível ao administrador ter conhecimento do estado da rede em qualquer altura e em qualquer lugar.

1.3 - Objectivos

A IPBrick é um sistema operativo para servidores de comunicações e de intranet, estando já representado em inúmeros países e que conta com várias participações em eventos internacionais[1].

Este sistema operativo tem a capacidade de fornecer a configuração de rede para todos os equipamentos existentes num parque informático, para os quais muitas das vezes se necessita

de conhecer o software que têm instalado, qual o hardware que utilizam, o estado dos seus serviços e também o tráfego que geram na rede. Todo este tipo de informação é bastante importante para um administrador, nomeadamente quando diz respeito a servidores, routers ou switches.

O objectivo deste trabalho consiste em responder às situações referidas anteriormente, integrando de forma nativa no sistema operativo IPBrick um conjunto de ferramentas que lhe permitam obter as tais informações de hardware e software sobre os equipamentos da rede, sobre o estado dos serviços e sobre o tráfego gerado. De referir que deverá ser garantida a total integração e parametrização automática entre as ferramentas seleccionadas e a IPBrick.

Como resultado final, pretende-se que um administrador da IPBrick disponha de uma solução que lhe permita um controlo e uma monitorização eficiente de todo o parque informático.

1.4 - Enquadramento

O projecto cujo objectivo está em cima mencionado foi proposto pela empresa iPortalMais à Faculdade de Engenharia da Universidade do Porto e, no final, será incorporado no sistema operativo IPBrick desenvolvido pela mesma.

A iPortalMais é uma empresa que se dedica ao fabrico de sistemas de comunicações, sendo a IPBrick um dos seus principais produtos[2]. Neste contexto, surgiu a necessidade de dar continuação ao seu desenvolvimento, criando um módulo responsável pelo controlo e monitorização que pudesse ser implementado no sistema operativo.

1.5 - Estrutura da dissertação

Enquanto o presente capítulo pretende apenas fornecer informação introdutória sobre o projecto, o capítulo seguinte tem como objectivo apresentar as ferramentas e tecnologias que foram estudadas e analisadas por apresentarem uma potencial importância para este trabalho.

O terceiro capítulo apresenta as diferentes abordagens ao actual problema.

O capítulo quatro descreve todos os detalhes envolvidos na solução implementada.

Por fim, o quinto capítulo apresenta uma síntese do trabalho desenvolvido e indica um conjunto de sugestões futuras que poderão ser implementadas para melhorar o módulo criado.

Capítulo 2

Estado da arte

Este capítulo tem como objectivo apresentar uma visão geral dos conceitos, das tecnologias e das ferramentas que estão relacionadas com este projecto.

2.1 - Ferramentas de monitorização

Hoje em dia, as ferramentas de monitorização desempenham um papel fundamental nas organizações. Permitem uma observação dos dispositivos que pertencem ao parque informático, dos seus recursos e ainda de todos os serviços, proporcionando assim uma melhoria significativa no desempenho da rede.

Os conceitos base da monitorização envolvem a execução de verificações periódicas aos equipamentos, a recepção dos resultados e o tratamento dessa informação para depois ser apresentada aos utilizadores, a grande maioria das vezes através de uma interface Web simples e amigável[3].

Existem duas formas distintas de efectuar a monitorização de um equipamento: com ou sem agentes.

A monitorização sem agentes é, por norma, realizada com recurso a protocolos de comunicação como o SNMP (Simple Network Management Protocol) ou o SSH (Secure Shell). Embora seja um processo mais rápido, uma vez que o utilizador não necessita de instalar e configurar agentes nas máquinas remotas, não oferece grandes detalhes nas verificações que efectua. Normalmente, este tipo de avaliação apenas permite obter informações sobre a disponibilidade do equipamento.

Por outro lado, existe a monitorização com recurso a agentes. Os agentes são aplicações que são instaladas nas máquinas remotas e que têm a capacidade de recolher dados localmente e disponibilizá-los ao servidor responsável pela gestão. É um mecanismo bastante

mais trabalhoso, mas o facto de proporcionar grande detalhe sobre a máquina onde estão instalados faz com que este tipo de monitorização seja adoptado a grande maioria das vezes.

Para além das funcionalidades que já foram referidas, este tipo de ferramentas apresenta outro tipo de potencialidades. De um modo geral, possuem mecanismos que permitem enviar notificações (via email, pager ou até SMS (Short Message Service)) quando surgem problemas na rede, têm a capacidade de gerar tabelas e gráficos personalizados sobre determinado serviço ou recurso e ainda a possibilidade de definir horários de funcionamento para os vários equipamentos.

2.1.1 - Nagios

O Nagios é uma ferramenta open source desenvolvida por Ethan Galstad que permite efectuar a monitorização de um sistema apresentando todos os resultados obtidos numa interface Web. Basicamente, isto significa que a aplicação tem a capacidade de verificar constantemente o estado das máquinas presentes na rede, bem como os respectivos serviços.

Inicialmente surgiu com o nome de NetSaint e foi desenvolvido para ambiente Linux, mas com o decorrer do tempo, o autor do programa e a sua equipa de colaboradores procederam às mudanças necessárias para que esta aplicação se tornasse compatível com outros sistemas operativos.

Actualmente, a ferramenta encontra-se na versão 3.2.3 e está disponível para download na Internet[4].

2.1.1.1 - Requisitos

Os dois principais requisitos para utilizar o Nagios são uma máquina com um sistema operativo UNIX e um compilador de C. Além disso, é necessário a pilha TCP/IP (Transmission Control Protocol/Internet Protocol) estar devidamente configurada uma vez que a grande maioria das verificações são feitas pela rede[5].

Não é obrigatório usar os CGIs (Common Gateway Interface) disponibilizados pelo Nagios, mas caso se pretenda a sua utilização torna-se necessário instalar um servidor Web (Apache) e uma biblioteca gd (GD Graphics Library).

2.1.1.2 - Arquitectura

A arquitectura do Nagios é baseada no conceito de cliente e servidor. O servidor representa a máquina responsável por efectuar todo o processo de monitorização, enquanto o cliente constitui a máquina remota que é monitorizada.

Apesar de o Nagios ter a capacidade de controlar qualquer equipamento ou serviço que possa ser contactado via TCP/IP, os objectos sujeitos a este controlo dividem-se em duas

categorias: os hosts, que representam as máquinas físicas (servidores, routers, switches, estações de trabalho, impressoras, etc.) e os serviços, como por exemplo: SMTP (Simple Mail Transfer Protocol), POP3 (Post Office Protocol 3), HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), SNMP e ICMP (Internet Control Message Protocol).

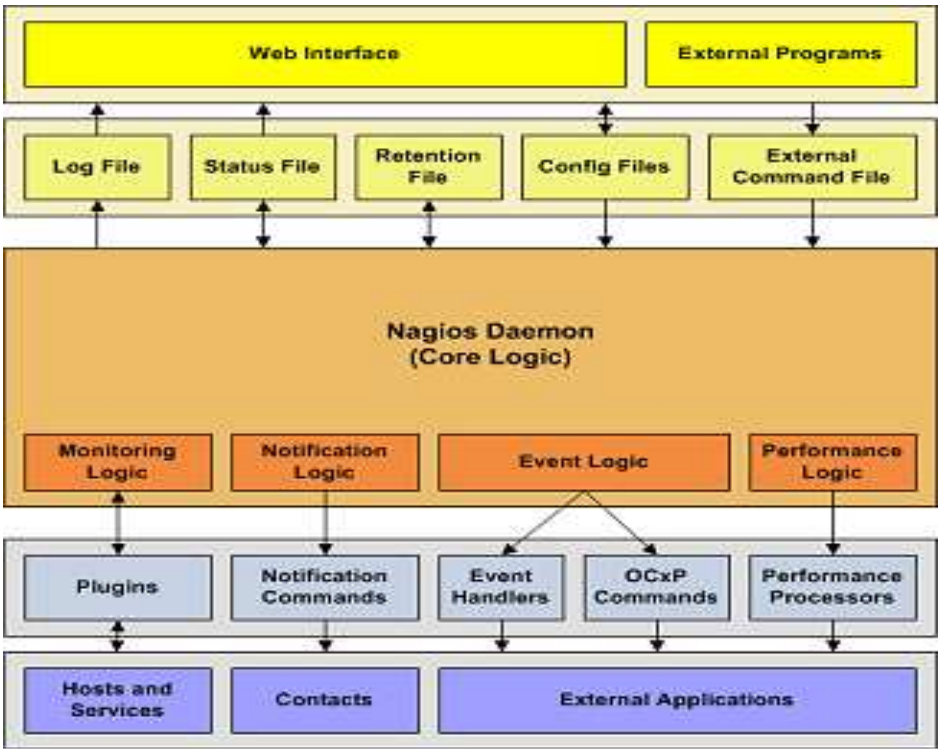

Figura 2.1 - Arquitectura do Nagios

A arquitectura do Nagios, como é possível observar na figura, está dividida em cinco camadas. A camada central constitui o principal serviço do Nagios, responsável pela lógica de monitorização, de notificação e de eventos. Abaixo desta camada está a camada de comunicação, onde se destaca a presença dos plugins. Esta camada tem como principal objectivo garantir a comunicação entre a aplicação e as máquinas exteriores, representadas na última camada. Além disso, é também neste último nível onde estão representados os contactos que são utilizados para o envio de notificações. As duas camadas superiores são igualmente importantes, sendo que a primeira é constituída pela interface Web responsável pela apresentação de resultados e a segunda é caracterizada por apresentar todos os ficheiros de configuração do programa[6].

2.1.1.3 - Funcionalidades

Em relação às suas funcionalidades, e para além de permitir a monitorização de alguns serviços já referidos, o Nagios tem a capacidade de realizar uma avaliação contínua sobre os

recursos dos hosts, como por exemplo: o espaço em disco, a utilização da memória física e virtual, a carga actual do processador, o uptime da máquina, o número de processos em execução e muito mais.

Suporta um serviço de notificações (em tempo real) em caso de falhas na rede através de email, pager, SMS ou qualquer outro método que seja definido e configurado por parte do utilizador.

Além disso, o Nagios possibilita a monitorização remota usando SSH ou SSL (Secure Sockets Layer), a geração de logs de forma automática, a apresentação de gráficos e relatórios e ainda tratadores de eventos que respondem de modo automático a problemas na rede[7].

2.1.1.4 - Funcionamento

Relativamente ao processo de monitorização, o Nagios não funciona por si só uma vez que depende de uma série de plugins que são responsáveis por efectuar as verificações necessárias e analisar os resultados recebidos. Este modo de funcionamento torna-o uma ferramenta bastante poderosa, até porque os utilizadores têm a capacidade de desenvolver os seus próprios plugins para efectuar as mais variadas tarefas, recorrendo a linguagens de programação como o C, Perl, Python, PHP (Hypertext Preprocessor) e C#. Além disso, têm ainda à sua disposição um conjunto de plugins oficiais desenvolvidos pela equipa do Nagios[8].

Esta aplicação permite fazer uma monitorização distribuída, o que significa a possibilidade de existirem várias estações descentralizadas enviando os seus resultados para uma máquina central, que fica assim responsável por apresentar o panorama geral da rede. Há também a possibilidade de ser configurada para efectuar monitorizações redundantes, ie, em caso de haver duas estações a controlar os mesmos serviços, uma delas envia uma notificação e a outra assume essa tarefa no caso de falha da primeira[9].

Outra das grandes vantagens no funcionamento do Nagios, e ainda no que diz respeito à monitorização, é que esta funciona de modo paralelo. Este tipo de funcionamento garante que no caso de haver um número muito elevado de itens a controlar, não há qualquer risco de alguns não serem observados por falta de tempo.

Além disso, é importante realçar que o Nagios também tem a capacidade de definir a rede hierarquicamente (hosts pais e hosts filhos), permitindo a distinção entre equipamentos indisponíveis e equipamentos inalcançáveis. Isto significa que a partir do servidor de monitorização é possível construir uma árvore hierárquica, onde este fica no topo e os hosts seguintes serão posicionados ao longo das ramificações.

2.1.1.5 - Agentes

O NRPE (Nagios Remote Plugin Executor) é uma ferramenta, também desenvolvida por Ethan Galstad, que permite ao utilizador executar plugins em máquinas remotas cujo sistema operativo seja baseado no UNIX.

Para este processo se desenrolar, para além do agente NRPE que necessita de ser instalado na máquina que se pretende monitorizar, há também a necessidade de instalar o plugin check_nrpe no servidor. Este plugin vai ser responsável por interagir com o agente e está disponível para download no site oficial do Nagios[4].

Para efectuar uma consulta, o Nagios executa o plugin check_nrpe e indica-lhe qual o comando que pretende executar (check_x). Este, por sua vez, vai entrar em contacto com o agente NRPE e transmite-lhe o respectivo comando. Por fim, e após a tarefa ter sido executada, ocorre o processo inverso onde os resultados são passados pelo agente ao plugin, que por sua vez os indica ao Nagios que acaba por disponibilizá-los na interface Web. De notar que o agente NRPE tem um ficheiro de configuração – nrpe.cfg – onde são definidos todos os comandos que são possíveis executar[10]. Todo este processo está demonstrado na figura seguinte.

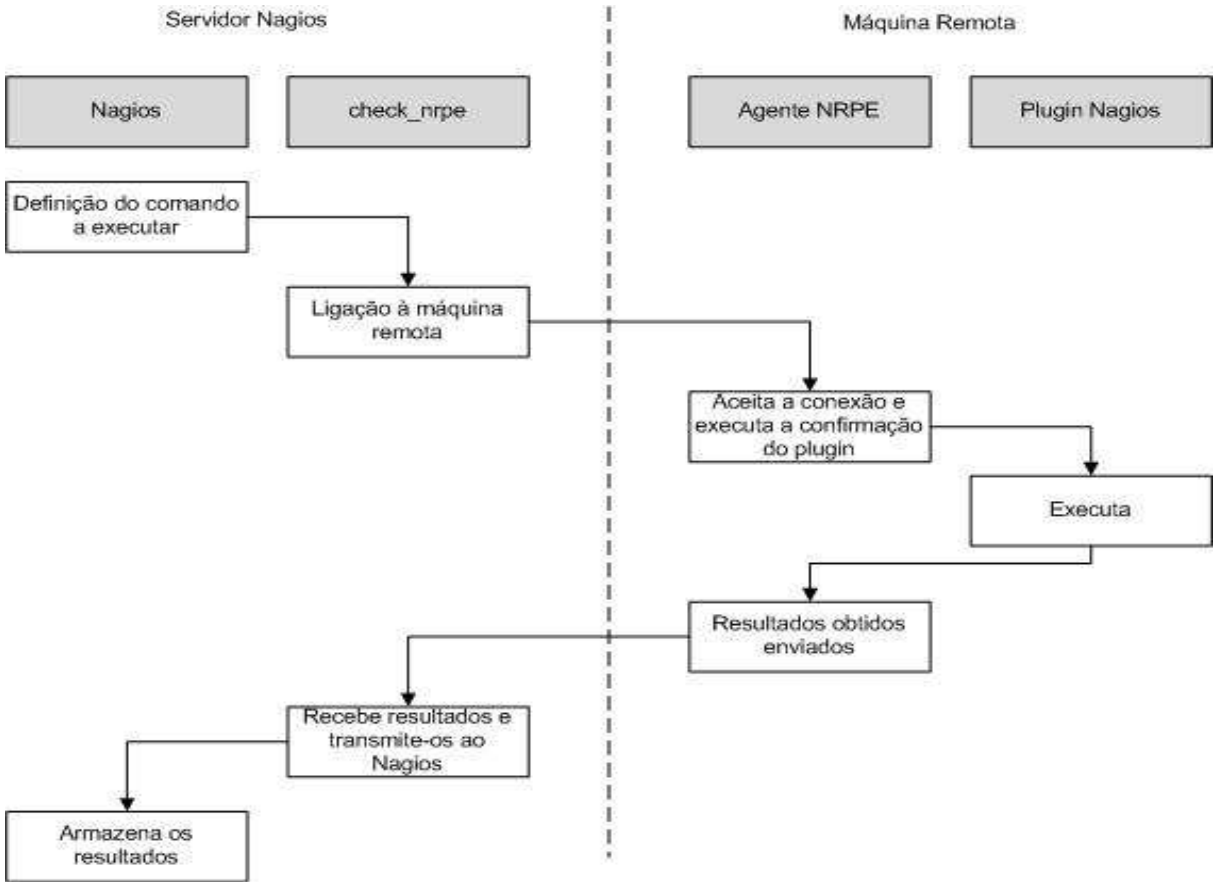

Figura 2.2 – Processo de monitorização com o agente NRPE

Quanto ao modo de funcionamento, o NRPE pode efectuar dois tipos de observações: observação directa e observação indirecta. Na primeira, o agente tem como função monitorizar serviços e recursos locais como a carga do CPU (Central Processing Unit), a memória que está a ser utilizada, o número de utilizadores, etc. No segundo caso, o agente tem a capacidade de consultar um servidor Web remoto que não é alcançável directamente pela máquina de monitorização. Basicamente, funciona como uma proxy.

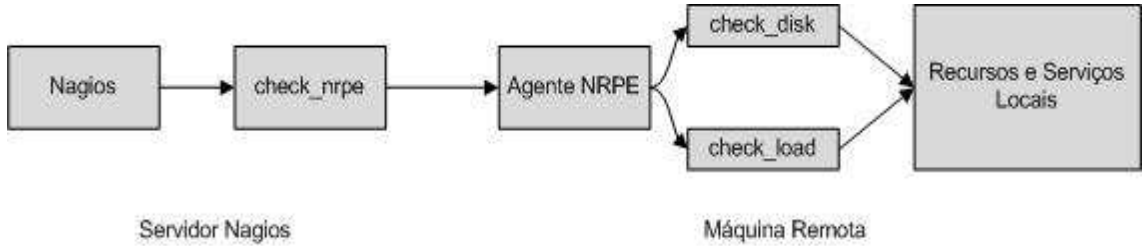

Figura 2.3 - Observação directa do agente NRPE

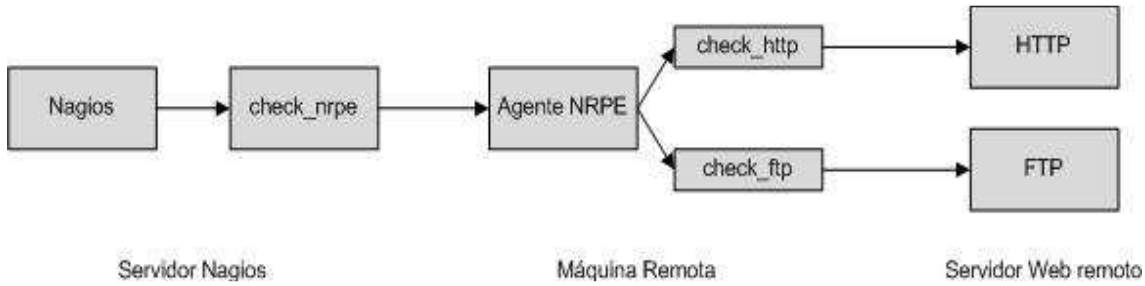

Figura 2.4 - Observação indirecta do agente NRPE

Este tipo de monitorização é caracterizado por apresentar uma série de vantagens, nomeadamente ao nível dos mecanismos de segurança e encriptação que consegue oferecer. Além disso, o overhead na comunicação entre servidor e agente é inferior quando comparado com outro tipo de comandos, o que significa que o servidor Nagios e a máquina remota utilizam menos recursos do CPU para desempenhar as suas tarefas, situação bastante importante quando estão a ser monitorizadas um número elevado de máquinas.

O NSClient++ é uma ferramenta análoga ao NRPE, mas é utilizada pelos administradores para monitorizar máquinas Windows. Funciona como um serviço e vem constituído com uma série de módulos e funções que permitem a monitorização de recursos.

Esta ferramenta pode operar em dois modos distintos: o primeiro é idêntico ao programa original NSClient e utiliza o comando check_nt para monitorizar a máquina remota; o segundo modo é uma implementação do NRPE, onde o servidor utiliza o comando check_nrpe para

efectuar todo o tipo de verificações[11]. A figura seguinte apresenta um exemplo de monitorização com o agente NSClient++.

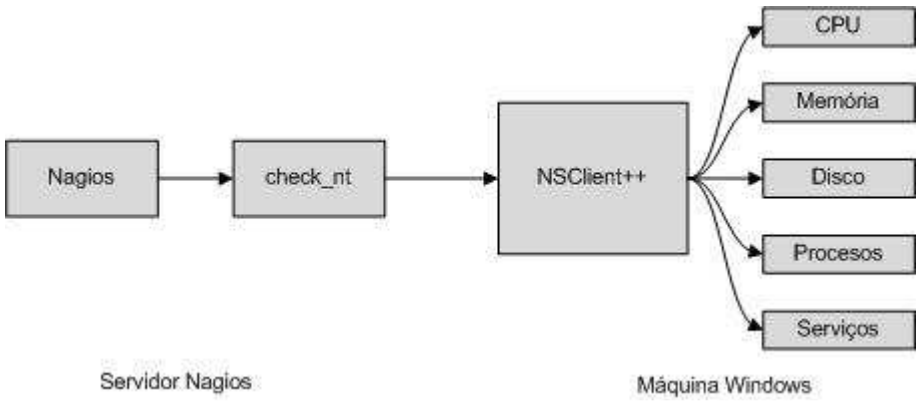

Figura 2.5 - Monitorização com o agente NSClient++

2.1.2 - Zabbix

O Zabbix apresenta-se como uma boa alternativa ao Nagios. É considerado um sistema de monitorização semi-distribuído com gestão centralizada que oferece uma solução para monitorizar a disponibilidade e o desempenho de servidores, equipamentos de rede e aplicações.

Apesar das suas características, funcionalidades e modo de operação serem semelhantes aos do Nagios, veremos que existem algumas diferenças entre estas duas ferramentas.

2.1.2.1 - Requisitos

Os requisitos mínimos para utilizar o Zabbix são o sistema de compilação gcc (GNU Compiler Collection), o automake e o sistema de gestão de base de dados MySQL. No entanto, dependendo da distribuição que é utilizada e das funcionalidades que são pretendidas, poderá ser necessária a instalação de pacotes adicionais.

Relativamente aos requisitos de hardware, uma máquina com 128MB de RAM (Random Access Memory) e um processador Pentium II consegue suportar a aplicação de uma forma bastante satisfatória. De qualquer das maneiras, com o objectivo de visualizar gráficos mais complexos, será necessário hardware mais avançado para que o programa opere a uma velocidade aceitável.

2.1.2.2 - Arquitectura

Tal como o Nagios, também o Zabbix funciona segundo o modelo de servidores e agentes. Os agentes são instalados nas máquinas remotas com o objectivo de recolher informações,

enquanto os servidores reúnem um conjunto de componentes que constituem o núcleo do programa. A figura seguinte representa uma rede segundo a perspectiva do Zabbix.

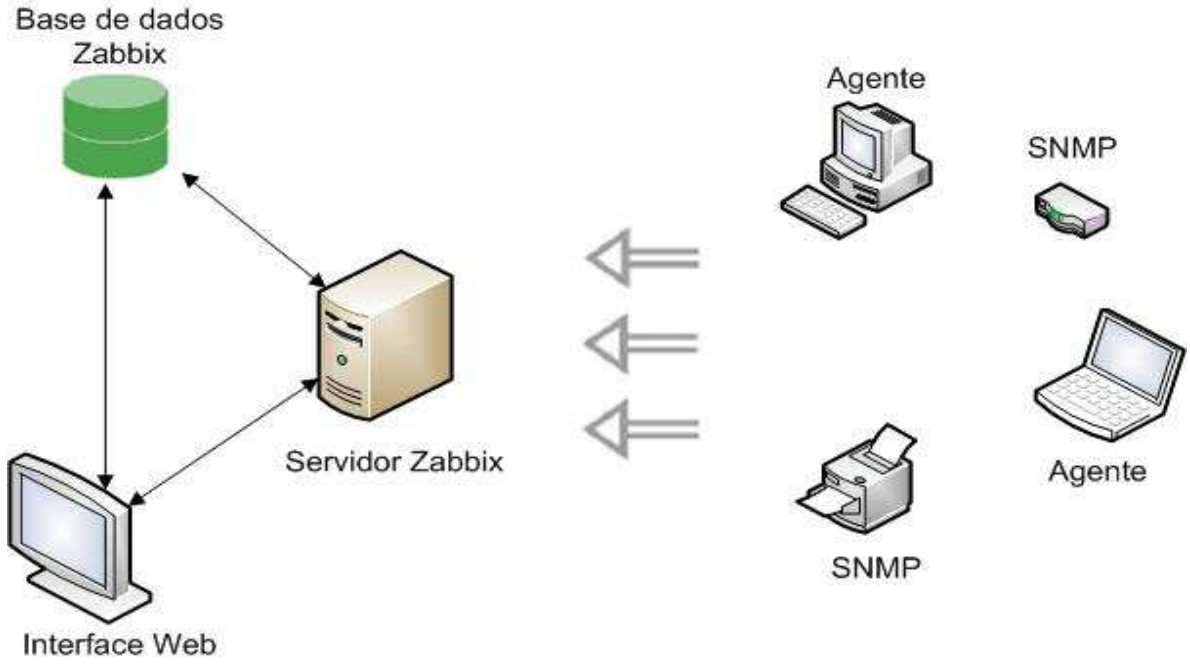

Figura 2.6 - Arquitectura do Zabbix

O componente fundamental é a base de dados Zabbix, responsável por guardar todo o tipo de informações, não só ao nível das configurações do utilizador como também os dados do desempenho. A base de dados está directamente ligada a um servidor e a uma interface Web (desenvolvida em PHP e Javascript). O servidor, para além de permitir a esta ferramenta efectuar verificações periódicas aos equipamentos da rede, também se destaca como o componente central ao qual os agentes reportam informações e estatísticas sobre a disponibilidade e sobre o desempenho. Por sua vez, a interface Web permite ao utilizador ver o estado da rede e efectuar as configurações necessárias na aplicação. Convém referir que estes dois últimos componentes não têm necessariamente que estar na mesma máquina que a base de dados. No entanto, e caso isso aconteça, é obrigatório que lhe consigam aceder, como mostra a figura seguinte[12].

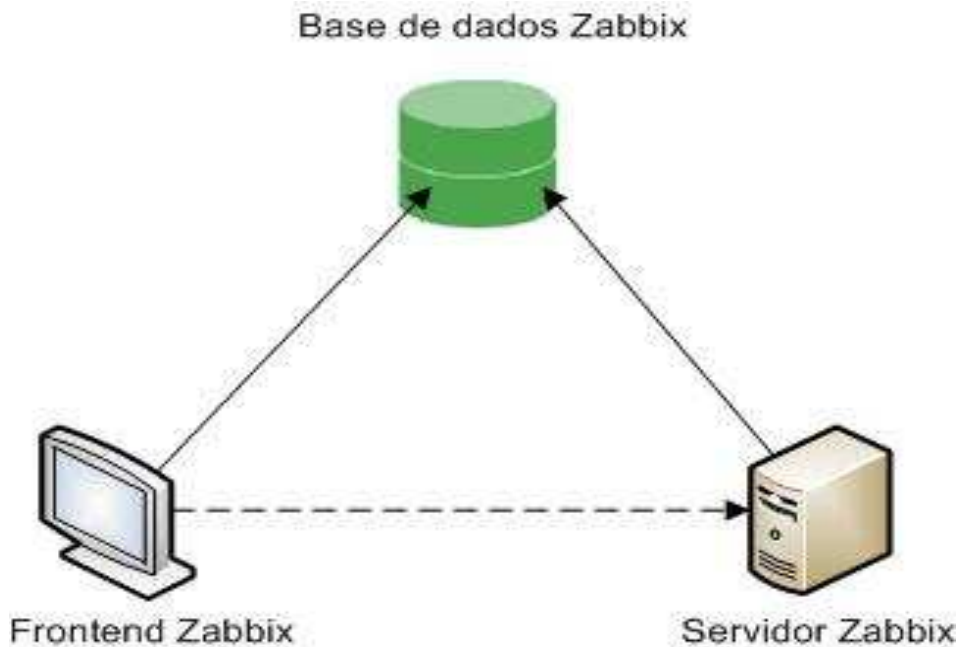

Figura 2.7 - Separação física do servidor e interface Web do Zabbix

Embora não estando representado na figura 2.6, a arquitectura pode também ser composta por uma proxy. A proxy desempenha um papel fundamental quando há a presença de uma firewall entre o servidor e os agentes, tendo como principal objectivo recolher e reunir dados localmente e só depois transferi-los para o servidor Zabbix ao qual está associada, destacando-se assim como a solução ideal para a monitorização centralizada.

2.1.2.3 - Funcionalidades

Relativamente aos recursos que podem ser controlados, não existe grande diferença em relação ao que acontece com as outras ferramentas de monitorização. Os servidores têm a capacidade de avaliar o espaço em disco, a quantidade de memória em utilização, os processos do sistema, informações sobre o processador, etc. Além disso, o Zabbix também permite a monitorização de serviços como o FTP, SSH, SMTP, HTTP e DNS (Domain Name System).

Entre as muitas funcionalidades que o Zabbix apresenta aos seus utilizadores, destaca-se ainda o facto de suportar IPv4 (Internet Protocol version 4) e IPv6 (Internet Protocol version 6), ser compatível com uma grande variedade de sistemas operativos, permitir a execução de comandos remotos e a descoberta automática de servidores e equipamentos de rede, suportar uma configuração flexível e um bom sistema de notificações e ainda o facto de ser compatível com o protocolo SNMP v1, 2 e 3[13].

2.1.2.4 - Funcionamento

No Zabbix, a monitorização das máquinas pode ser activa ou passiva. No primeiro caso, o servidor envia directamente ao host que está a controlar um pedido sobre determinado recurso, obtendo depois a respectiva resposta. No caso da monitorização passiva, é o agente que toma a iniciativa, enviando ao servidor um pedido sobre todos os comandos disponíveis, sendo que após obter a respectiva informação lhe envia os dados correspondentes[12].

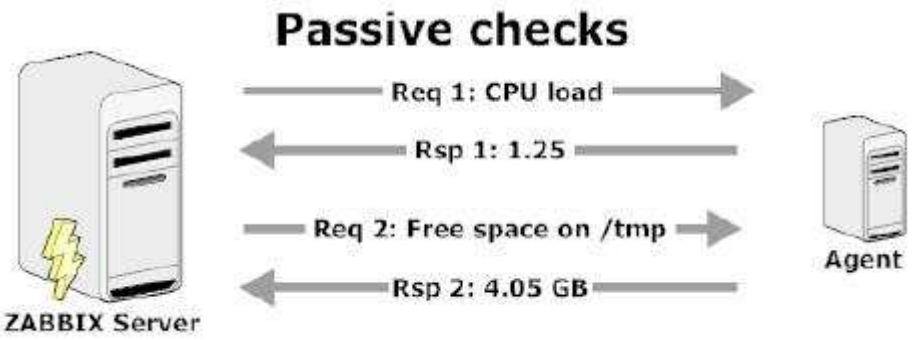

Figura 2.8 - Monitorização passiva com o Zabbix

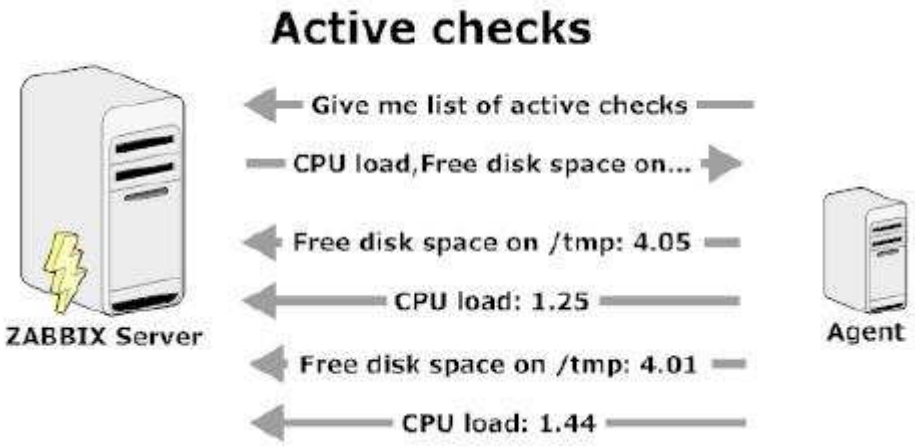

Figura 2.9 - Monitorização activa com o Zabbix

2.2 - Ferramentas de Gestão

Para além das ferramentas de monitorização que foram avaliadas, existe um outro tipo de ferramentas igualmente importantes para garantir um bom desempenho e uma boa gestão de um parque informático. Essas ferramentas são designadas por ferramentas de gestão de inventário e, como o próprio nome indica, têm como principal objectivo reunir todo um inventário ao nível de software e do hardware dos equipamentos constituintes de uma rede.

2.2.1 - OCS Inventory NG

O OCS Inventory NG (Open Computer and Software Inventory Next Generation) é um dos programas que permite gerir e controlar um parque informático.

O principal objectivo do OCS é reunir o inventário de todas as máquinas pertencentes à rede e disponibilizá-lo ao administrador, através de uma interface Web simples e transparente.

É um programa baseado no paradigma cliente-servidor, onde os clientes correspondem a agentes que são instalados nas máquinas que se pretendem controlar. Quanto ao servidor responsável pela gestão, é constituído por quatro componentes:

- Servidor de base de dados – armazena informações sobre o inventário das máquinas;
- Servidor de comunicações – responsável pela comunicação entre as bases de dados e os agentes;
- Consola de administração – permite aos administradores consultarem as bases de dados usando um browser;
- Servidor de implementação – armazena a configuração dos pacotes implementados (necessita de HTTPS)[14].

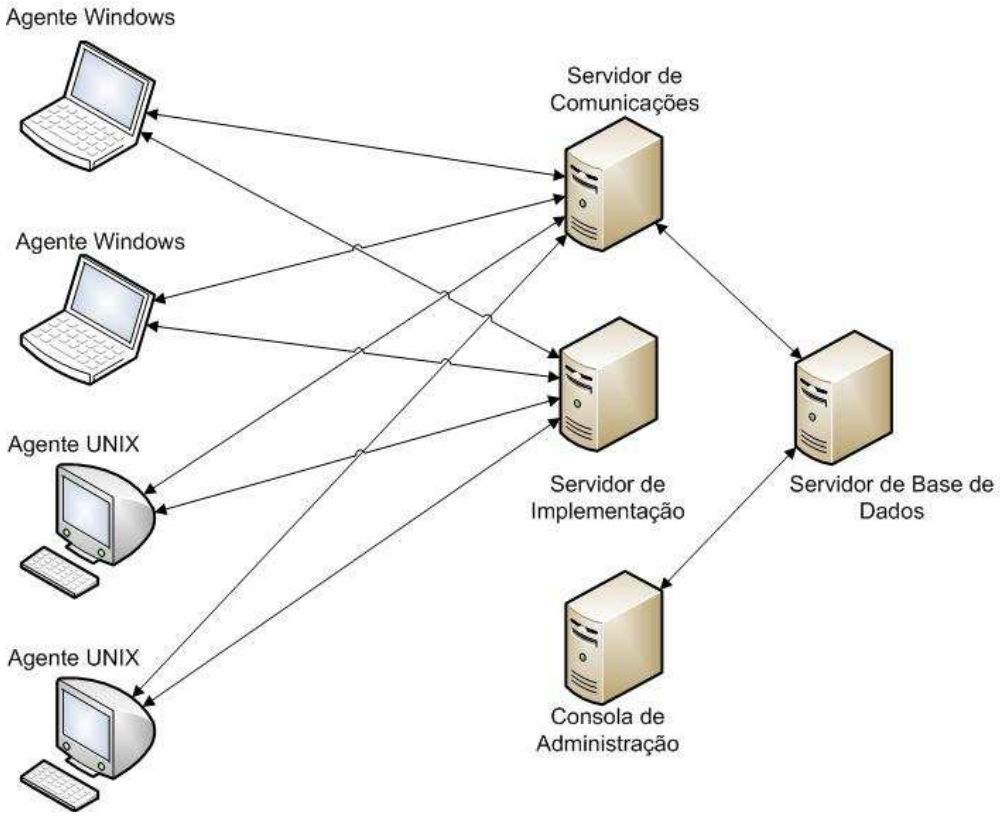

Figura 2.10 - Arquitectura do OCS Inventory

A comunicação entre os agentes e o servidor de comunicações é feita recorrendo ao protocolo HTTP, sendo que o formato dos dados trocados é o XML (eXtensible Markup Language). Além disso, esta aplicação suporta os mais variados sistemas operativos (Microsoft Windows, Linux, *BSD, Sun Solaris, MacOS X, etc.), não só para os agentes como também para os servidores. É importante referir que os dados coleccionados de uma máquina ocupam no máximo 5KB, o que implica não haver congestionamento da rede nem sobrecarga no servidor, condição essencial e necessária quando há um elevado número de hosts para gerir. Caso isto aconteça, existe sempre a possibilidade de fazer a separação dos quatro componentes em várias máquinas, ie, juntar o servidor de base de dados e o servidor de comunicações numa máquina, enquanto a consola e o servidor de implementação ficariam ao cargo de uma máquina independente.

Esta ferramenta permite obter informações sobre o software instalado (nome, versão), sobre a memória (capacidade, tipo), sobre os processadores (tipo, número, fabricante, velocidade), sobre o hardware (adaptadores de vídeo, periféricos), sobre dispositivos amovíveis e muito mais[15].

Uma das grandes vantagens do OCS é que tem a capacidade única de reunir o inventário de máquinas que não pertencem à rede. Na realidade, esta ferramenta possui um algoritmo que lhe permite avaliar se determinado host representa uma estação de trabalho ou uma impressora (por exemplo). Depois de chegar a uma conclusão, e caso esteja perante uma estação de trabalho, é sinal que um agente tem que ser implementado, tarefa que o servidor entrega às restantes máquinas da rede às quais está directamente ligado.

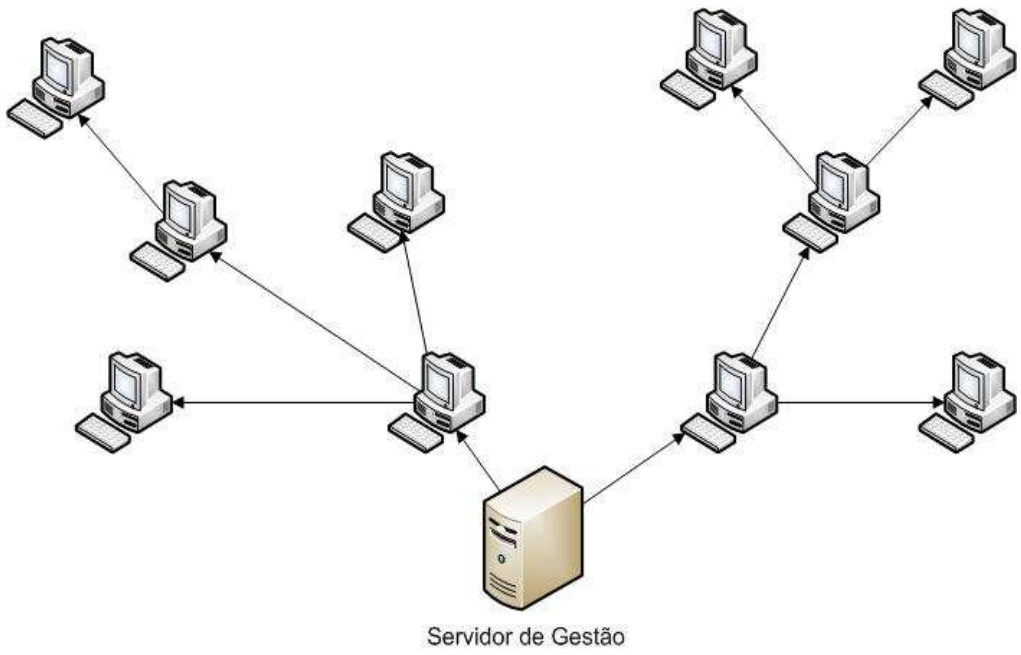

Figura 2.11 - Capacidade de auto-discover no OCS Inventory

2.2.2 - GLPI

A análise do funcionamento do OCS Inventory NG e do modo como é feita a interacção com os agentes, permite ter uma ideia da forma como todo o inventário é recolhido. No entanto, existem outras aplicações que funcionam em conjunto com o OCS e que o tornam uma ferramenta mais completa e robusta.

O GLPI (Gestion Libre de Parc Informatique) é um programa open source que consegue tirar partido das informações já recolhidas e armazenadas pelo OCS e assim proporcionar ao administrador do parque informático uma melhor organização e gestão de todos os recursos.

É importante referir que esta aplicação não muda rigorosamente nada no funcionamento do OCS uma vez que este pode ser utilizado individualmente. Aquilo que o GLPI faz é importar a base de dados do OCS Inventory para a sua própria estrutura, sendo que a partir desse momento funciona de modo independente. À medida que novos dados vão sendo recolhidos pelo OCS, o GLPI tem a capacidade de se sincronizar ficando automaticamente actualizado.

Para além do inventário que também é disponibilizado, o GLPI permite associar a todos os itens vários tipos de informação extra, como por exemplo: a sua localização, o seu custo, os técnicos responsáveis pela sua manutenção, os vendedores, os fabricantes, entre outros.

Ter acesso à localização dos equipamentos é fundamental em organizações com vários edifícios ou mesmo vários andares, enquanto informações sobre o custo, vendedores e fabricantes também são bastante úteis quando há a necessidade de encomendar material, uma vez que facilitam todo este processo.

Permite fazer uma monitorização ao estado dos equipamentos presentes na rede, ie, saber se determinado equipamento está operacional, se está avariado, se está em manutenção, etc. Esta funcionalidade transmite aos administradores uma perspectiva de todo o parque.

Tem incorporado um sistema de helpdesk, compatível com o facto de ser multiutilizador (suporta administradores, técnicos e utilizadores). Os utilizadores reportam problemas em relação aos equipamentos, fazendo com que os administradores recebam um aviso e entreguem a tarefa aos técnicos. Após estes procedimentos, a evolução do estado do equipamento pode ser acompanhada pelos técnicos, pelo administrador e ainda pelo utilizador que reportou a situação.

Apresenta um sistema de notificação bastante desenvolvido que desempenha um papel crucial, não só no sistema de helpdesk, mas também em situações em que é necessário alertar os responsáveis devido a um determinado stock estar a acabar.

Como é sabido, num parque informático com muitas máquinas, existe sempre uma grande variedade de software instalado e com vários tipos de licenças associadas. Algumas são vitalícias, enquanto outras são limitadas. O GLPI suporta o tracking de licenças o que permite saber quando estas se encontram prestes a expirar, e assim tomar as medidas apropriadas.

Tem a capacidade de gerar relatórios (informações financeiras, histórico do hardware, software instalado, etc.) e estatísticas (número total de tickets, média de problemas resolvidos por dia, número de tickets não resolvidos, tempo médio para resolver um problema, histórico dos utilizadores, etc.).

Por fim, possui também um sistema de pesquisa bastante avançado que permite aos utilizadores pesquisarem por equipamento, por estado, por vendedor, por fabricante e muito mais, proporcionando assim uma solução onde consigam obter o desejado de uma forma rápida e eficaz[16].

2.2.3 - Fusion Inventory

O Fusion Inventory é mais uma ferramenta que oferece uma solução para reunir o inventário das máquinas que pertencem a um parque informático. No entanto, apresenta determinadas características e funcionalidades que o distinguem do OCS Inventory NG.

Uma das diferenças é que este programa funciona como um plugin para a aplicação de gestão de software GLPI, permitindo a comunicação directa com os vários agentes Fusion Inventory. Isto significa que o GLPI em vez de usar a base de dados disponibilizada pelo OCS Inventory e funcionar de modo independente, tem assim um plugin a ele associado responsável por desempenhar essas funções.

O funcionamento do Fusion Inventory depende fundamentalmente de três componentes: Fusion Inventory para GLPI (servidor), agentes Fusion Inventory e ainda uma biblioteca PHP.

O primeiro, como já foi dito, representa um servidor de comunicações cujo principal objectivo é comunicar com os agentes via HTTP ou HTTPS.

Os agentes, para além de reunirem o inventário das máquinas onde estão instalados, são caracterizados por serem multi-plataforma (vários sistemas operativos são suportados, tais como Windows, Linux, Mac OS X, FreeBSD, OpenBSD e Solaris), multi-servidor (possibilidade de enviar o inventário para vários servidores de gestão, bastando para isso editar o ficheiro de configuração e definir quais os servidores com quem comunica), compatíveis com servidores OCS Inventory, permitirem a execução periódica de determinados comandos e suportarem os módulos Netdiscovery e Snmpquery. O Netdiscovery permite através do SNMP, nmap e Netbios a descoberta automática de dispositivos que pertencem à rede, enquanto o módulo Snmpquery permite reunir o inventário de impressoras, routers e switches através de pedidos SNMP.

Por último, a biblioteca PHP permite incorporar esta ferramenta noutro tipo de aplicações.

2.3 - Comparação de resultados das ferramentas

Neste subcapítulo é feita uma comparação entre as várias ferramentas de monitorização e de gestão que foram analisadas.

2.3.1 - OCS Inventory NG vs. Fusion Inventory

As diferenças mais significativas entre o OCS Inventory NG e o Fusion Inventory estão especificadas nas tabelas em baixo apresentadas, não só ao nível dos agentes, mas também dos servidores[17].

Tabela 2.1 – Agentes

Característica	Detalhe	Agente Fusion Inventory	Agente OCS Inventory
Output em HTML	Gerar o inventário em HTML	+	-
Wake up Remoto	Controlar o agente a partir do servidor	+	-
Wake up Local	Forçar a execução local do agente	+	Apenas Windows

Tabela 2.2 – Inventário no Agente

Característica	Detalhe	Agente Fusion Inventory	Agente OCS Inventory
Software - Nome	Obter nome	+	+
Software - KB	Obter KB	+	+-
Software - Número de série	Obter nº série	-	-
Software – Data de instalação	Obter data de instalação	+	+

Impressora – Número de série	Obter número de série	+	+
Impressora - USB	Obter informações de USB	+	-
Disco rígido – Número de série	Obter número de série	+	+
Disco rígido - Interface	Obter nome da interface	+	+
BIOS – Data de fabrico	Obter data de fabrico	+	+
BIOS - Fabricante	Obter fabricante	+	+
BIOS - Modelo	Modelo computador	+	+
Processador – Frequência CPU	Obter frequência CPU	+	-
Processador - Memória	Obter memória física e virtual	+	+

Tabela 2.3 - Inventário no Servidor

Característica	Detalhe	Servidor Fusion Inventory	Servidor OCS Inventory
Processador	Informações do processador	+	+-
Memória	Informações da memória	+	+
Controlador	Apresenta controladores	+	+
Slot	Apresenta Slots	-	+
Portas	Apresenta Portas	-	+
Placa gráfica	Placas gráficas	+	+
Monitor	Apresenta Monitores	-	+
Software	Lista de software	-	+
Informações de rede	Informações de rede	+	+
Impressora	Informações de impressoras	+	+-
Placa de som	Apresenta placas de som	+	+
BIOS	Informações da BIOS (número	+	+

	de série, fabricante, etc.)		
Baterias	Apresenta baterias	+	-
Antivírus	Apresenta antivírus	+	-

2.3.2 - Nagios vs. Zabbix

Esta secção pretende demonstrar as principais diferenças existentes entre o Nagios e o Zabbix[18].

2.3.2.1 - Configuração

A ferramenta Nagios guarda todas as suas configurações em simples ficheiros de texto. Este facto acaba por se tornar uma vantagem em várias situações, na medida em que permite aos utilizadores efectuarem alterações radicais em toda a sua rede recorrendo a várias linguagens de scripting. Por exemplo, de uma maneira relativamente simples, torna-se possível gerar a configuração de centenas de hosts a partir de uma base de dados recorrendo a um simples script Python.

Em relação ao Zabbix, todas as configurações são armazenadas numa base de dados. Sendo assim, as alterações nas configurações têm obrigatoriamente que ser feitas pela interface gráfica ou através de comandos XML. Uma das desvantagens em relação a este modo de funcionamento é que a base de dados, caso não seja configurada apropriadamente, pode crescer demasiado depressa e sobrecarregar o sistema.

2.3.2.2 - Escalabilidade

Relativamente à escalabilidade também existe uma ligeira diferença entre estas duas ferramentas. O Nagios, apenas com a utilização de um addon a funcionar como proxy, consegue efectuar verificações a um grande número de equipamentos. Já o Zabbix, apresenta características que lhe permitem monitorizar um elevado volume de hosts sem recurso a programas adicionais.

No entanto, o Zabbix também suporta o funcionamento através de proxies, apesar de existirem algumas diferenças relativamente ao modo como este processo funciona. No caso do Nagios, todas as proxies executam verificações arbitrárias para todos os hosts. Por outras palavras, isso significa que hosts específicos não podem ser associados a proxies específicas. Este método apresenta uma vantagem, na medida em que caso se perca uma proxy não significa perder a capacidade de monitorizar um determinado conjunto de hosts. A

desvantagem é que não se pode configurar uma simples proxy para monitorizar um subconjunto de nós.

No Zabbix isso já não se verifica, pois uma proxy é associada a um host. Isto permite associar proxies a determinadas sub-redes, mas por outro lado cria pontos de falha mais evidentes.

2.3.2.3 - Relatórios

Por defeito, o Nagios guarda um histórico de alertas para todos os hosts e serviços. Relatórios de disponibilidade podem ser criados para servidores individuais, grupos de hosts, serviços específicos, etc. Através de addons, tem a capacidade de exportar estas informações para outro tipo de formato (RRD por exemplo) que ferramentas como o Cacti podem utilizar. Além disso, o Nagios permite aos administradores fazerem comentários com timestamps de modo a que consigam fazer um histórico do trabalho e das avaliações efectuadas para hosts em particular.

O Zabbix também tem incorporado um vasto conjunto de gráficos e relatórios, mas por outro lado não permite que sejam adicionados comentários com timestamps para os hosts.

2.3.2.4 - Interface

O Nagios tem uma interface embutida que já vem com o pacote standard. Esta interface permite monitorizar hosts, ver alertas, fazer a gestão da agenda, criar relatórios, monitorizar serviços, etc. A interface não tem a capacidade de adicionar ou modificar qualquer tipo de informação em relação a hosts ou serviços. Todo este tipo de modificações é feito editando os ficheiros de texto das configurações do Nagios. De qualquer maneira, convém referir que existem aplicações que permitem configurar o Nagios através de uma interface gráfica.

A interface do Zabbix permite efectuar todo o tipo de configurações sem recorrer aos ficheiros, mas por outro lado tem suporte muito limitado à linha de comandos ou mudanças por scripts.

2.4 - IPBrick

A IPBrick é um sistema operativo baseado no Oracle Enterprise Linux que se apresenta como uma solução para servidores de comunicações e de intranet. É um servidor integrado completo, desenvolvido e comercializado pela empresa iPortalMais, sendo caracterizado por oferecer uma instalação bastante rápida (aproximadamente dez minutos) e sem qualquer intervenção por parte do utilizador, uma interface Web funcional onde se podem efectuar todo o tipo de configurações necessárias (acedida a partir de qualquer browser) e ainda mecanismos que permitem a recuperação do sistema em caso de avarias e problemas[2].

É um sistema operativo bastante funcional uma vez que permite uma administração simples e intuitiva graças à sua interface gráfica inovadora (menus orientados a uma "Lógica de Negócios"), tornando assim possível uma fácil configuração de todos os serviços sem grandes conhecimentos prévios de Linux e/ou redes.

Possui uma série de características que lhe garantem uma enorme estabilidade ao nível da intranet e ao nível das comunicações, tais como: gestão de ataques DoS (Denial of Service), qualidade de serviço (Traffic Shaping e QoS) e ainda filtragens de conteúdos.

Além disso, apresenta uma enorme portabilidade na medida em que é independente do hardware. No caso de haver algum tipo de falha, pode ser utilizada a IPBrick.D para repor o sistema em qualquer outro servidor. Todas as configurações são preservadas, gravando-se inclusive a data e hora das mesmas para permitir a recuperação de qualquer configuração.

Quanto a questões de segurança, a IPBrick está integrada com ferramentas antivírus e antispam Kaspersky que lhe conferem toda a protecção necessária.

Para responder aos vários segmentos do mercado a IPBrick desenvolveu várias appliances, cada uma com determinado tipo de características e que têm o objectivo de satisfazer as necessidades dos seus clientes[19].

2.4.1 - IPBrick.IC

A IPBrick.IC representa uma plataforma de comunicações para empresas e constitui um dos vários produtos associados à IPBrick. Funcionalidades como correio electrónico, ferramentas colaborativas, serviços de firewall e proxy para acesso à Internet fazem com que a IPBrick.IC seja capaz de proporcionar uma comunicação segura e completamente integrada.

Divide-se em dois componentes: IPBrick.I (servidor intranet) e IPBrick.C (servidor de comunicações).

2.4.1.1 - IPBrick.I

A IPBrick.I é um servidor de intranet caracterizado por suportar as aplicações de negócio essenciais para a gestão de uma empresa. Fornece ferramentas colaborativas (email, livro de endereços e agenda/calendário), serviços de suporte à rede como o protocolo DHCP (Dynamic Host Configuration Protocol) e DNS, é um servidor de domínio, de ficheiros, de impressoras e de base de dados. Para além disso, tem a capacidade de garantir o backup das áreas de trabalho, questão fundamental para garantir a segurança dos dados.

2.4.1.2 - IPBrick.C

A IPBrick.C é um servidor de comunicações e segurança: fornece protecção por firewall, Intrusion Detection System (IDS), servidor VPN (Virtual Private Network), antispam e antivírus

Kaspersky, servidor de IM (Instant Messaging), serviços de telefonia e ainda Fax2Mail, Mail2Fax e Mail2SMS.

2.4.2 - IPBrick.GT

A IPBrick.GT é uma appliance que fornece voz, vídeo, fax, email, Web, SMS e Instant Messaging numa única appliance.

Uma das suas grandes vantagens é o facto de possuir mecanismos para gravação de comunicações: chamadas, faxes, emails e conversas instantâneas. Além disso, oferece uma gama de serviços que passam pela gestão de filas de espera, escalonamento, parqueamento e transferência de chamadas, conferências de chamadas, entre outros.

2.4.3 - IPBrick.H

A IPBrick.H é uma appliance que permite gerir o aceso à Internet. Através da IPBrick.H é possível ter um HotSpot à imagem de qualquer negócio, configurando as tarifas que vão ser cobradas pela ligação, personalizando a página de acesso à Internet (informações, contactos, publicidade, etc.) e ainda controlando as páginas que são consultadas.

Esta appliance gera dois tipos de cartões (vouchers) que indicam o login e palavra-chave necessários para um utilizador se validar no sistema: pré-pagos e pós-pagos. Os primeiros são caracterizados por ter um crédito em dinheiro que vai sendo descontado tendo em conta a tarifa actual, enquanto nos segundos o utilizador utiliza livremente a Internet sendo que no final lhe é facturado esse uso.

A IPBrick.H permite fazer a configuração das tarifas a serem cobradas, tendo em conta o dia ou a hora do dia, gerar estatísticas para ver o desempenho do HotSpot, controlar o tráfego, filtragem de conteúdos e ainda visualizar o estado do utilizador.

2.4.4 - IPBrick.KAV

A IPBrick.KAV é uma appliance que garante o acesso seguro à Internet e aumenta o nível de segurança das redes empresariais. Basicamente, consegue impedir que estações de trabalho se liguem directamente à Internet, evitando que programas “trojan” estabeleçam túneis com o exterior ou abram backdoors que permitam o acesso à rede da empresa a partir do exterior.

Possui serviços que garantem a segurança de email (antivírus e antispam da Kaspersky, responsáveis por limpar os conteúdos perigosos enviados por correio electrónico), segurança da rede (firewall que impede acessos não desejados à rede interna e ainda mecanismos de detecção de intrusão) e segurança da intranet (detecção de máquinas infectadas na rede interna).

Além disto, também oferece funcionalidades para melhorar a operação da empresa na interacção entre a rede interna e a Internet: webmail para poder consultar a caixa de correio a partir de qualquer lugar e de forma segura; VPN SSL para conseguir aceder com segurança aos dados e aplicações da empresa a partir do exterior; VoIP para telefonar via Internet e um servidor Web para disponibilizar conteúdos desde a rede local para Internet.

2.4.5 - IPBrick.LIVE

A IPBrick.LIVE é uma solução que permite configurar painéis informativos multimédia com filmes, notícias e meteorologia. É composta por duas interfaces: a que configura os parâmetros da IPBrick.LIVE e a interface gráfica que permite reproduzir os filmes, mostrar as notícias e previsão do tempo em LCD, monitores, televisões plasma ou quaisquer outros equipamentos que possam ser conectados à saída VGA.

Existem duas formas distintas de configuração - uma permite configurar manualmente toda a informação que vai passar no painel informativo multimédia, predefinindo uma playlist de filmes e outros conteúdos exibidos ininterruptamente, enquanto a outra forma de configuração foi pensada para instituições maiores com painéis em diferentes localizações, onde há a possibilidade de configurar várias IPBrick.LIVE a partir de uma IPBrick.LIVE principal.

2.4.6 - IPBrick.SCHOOL

A IPBrick.SCHOOL é uma tecnologia revolucionária que se apresenta como uma solução que fornece um desktop multi-sessão e está pensada para o ambiente escolar do futuro, baseado num conjunto de terminais ligeiros ligados a um servidor IPBrick. Com esta tecnologia, qualquer escola pode instalar um servidor em 30 minutos ligado a mais de 64 terminais com dois ou mais sistemas operativos.

Oferece enormes vantagens como poupança de energia e dinheiro, uma gestão fácil automática e configuração simples, ambiente colaborativo onde alunos e professores partilham os mm recursos (email, agenda, contactos e ficheiros) e suporta a plataforma educacional Moodle.

2.4.7 - IPBrick.SOHO

A IPBrick.SOHO é uma solução que fornece em apenas um equipamento duas ferramentas essenciais em todos os escritórios – PBX e FAX. Alem disso, oferece também email, SMS, servidor Web e mensagens instantâneas, todos integrados num único aparelho.

A IPBrick.SOHO proporciona todos os serviços essenciais para comunicações empresariais – PBX IP, faz, SMS, email, mensagens instantâneas e Web. Todos estes serviços são unificados

sobre o IP da IPBrick, proporcionando email e faz, correio de voz, correio de mensagens e webphone webchat. Todos estes serviços tão disponíveis a partir de qualquer computador, smartphone e evidentemente, integrados em telefones IP comuns.

Capítulo 3

Cenários de configuração automática do Nagios

Este capítulo tem como objectivo apresentar e caracterizar todas as abordagens que foram analisadas com o objectivo de resolver o actual problema.

3.1 - Ferramentas de configuração do Nagios

Uma vez que o Nagios é considerado uma das principais ferramentas no que diz respeito à monitorização de um parque informático, surgiu a necessidade de desenvolver aplicações que disponibilizassem uma interface Web para efectuar a configuração desta ferramenta de uma forma completamente automática. Sendo assim, de seguida são apresentadas um conjunto de ferramentas que foram testadas e analisadas e que desempenham este tipo de funções, para além de possuírem características que lhes garantam uma possibilidade de serem integradas com o sistema operativo IPBrick.

3.1.1 - Nconf

O Nconf (Nagios Configurator), lançado no ano de 2009, apresenta-se como uma solução capaz de configurar o Nagios de um modo automático. Basicamente, através da sua interface o utilizador consegue alterar todos os ficheiros de configuração dessa ferramenta, tanto os ficheiros globais como os ficheiros específicos das máquinas, de uma forma mais fácil e rápida, tarefa que se revela bastante complicada e morosa de realizar directamente no Nagios, que requer a alteração manual dos mesmos ficheiros[20].

3.1.1.1 - Requisitos

Os principais requisitos para utilizar o Nconf e usufruir de todas as suas capacidades são: um servidor Web Apache, o software de base de dados MySQL (>= 5.0.2), PHP (>= 5) e o pacote php-mysql, Perl (>= 5.6) com os pacotes perl-DBI e perl-DBD-MySQL e ainda o Nagios 3.x (o binário é estritamente necessário para testar a configuração).

3.1.1.2 - Arquitectura

A arquitectura do Nconf é constituída por três componentes fundamentais: uma interface gráfica, uma base de dados MySQL e uma API (Application Programming Interface) desenvolvida em Perl[21].

Relativamente às funcionalidades presentes na interface gráfica, todas elas são implementadas em PHP. Além disso, também os módulos de autenticação são desenvolvidos nesta linguagem de programação.

A base de dados funciona como um repositório intermediário, onde as actualizações e modificações feitas pelo utilizador são armazenadas antes de serem encaminhadas para os ficheiros de configuração do Nagios.

A API tem como principal objectivo permitir a ligação às funcionalidades internas do Nconf, nomeadamente a sua base de dados. Todas as operações que são executadas em modo silencioso, tais como a importação e exportação dos ficheiros do Nagios, são implementadas em Perl.

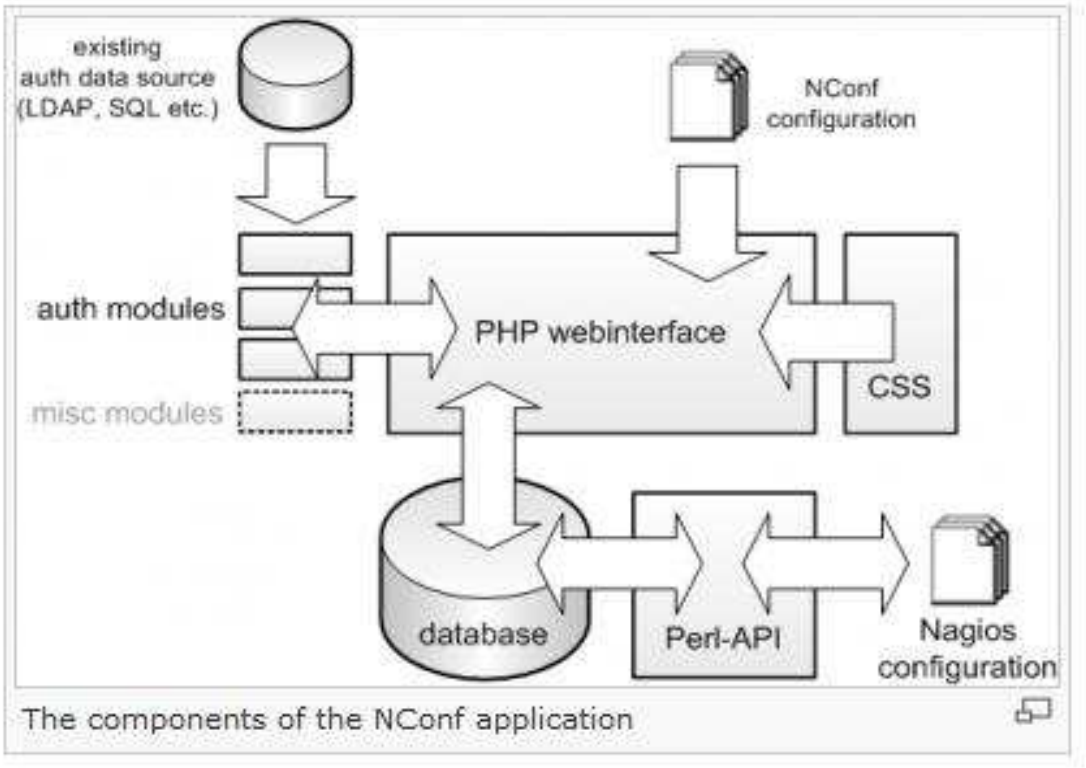

Figura 3.1- Arquitectura do Nconf

3.1.1.3 - Funcionalidades

Tal como já foi dito anteriormente, a principal funcionalidade do Nconf é simplificar todo o processo de configuração do Nagios. Através da sua interface, é possível adicionar e remover máquinas/grupos de máquinas, serviços, contactos, comandos e muito mais. Além disso, antes de gerar os ficheiros de configuração, esta ferramenta tem a capacidade de efectuar testes de sintaxe aos respectivos ficheiros, evitando assim que sejam gerados ficheiros de configuração com erros que comprometam toda a estrutura.

Sendo que o Nagios apresenta um limite máximo de dispositivos que consegue monitorizar, outra das vantagens desta ferramenta é possibilidade de suportar uma monitorização distribuída e assim contornar esta situação.

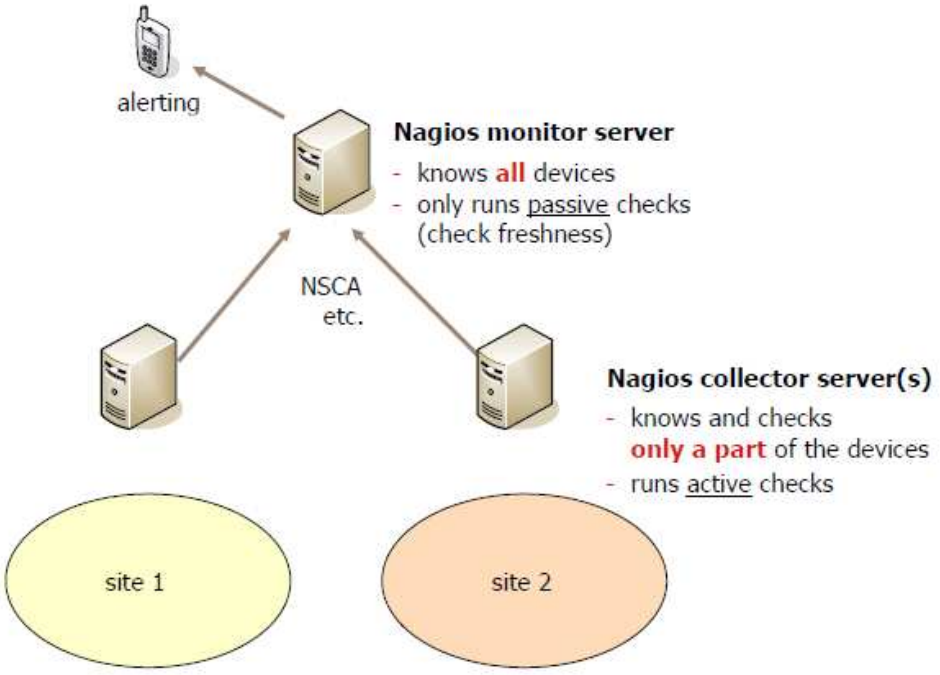

Figura 3.2 - Monitorização distribuída suportada pelo Nconf

Esta estrutura é diferente da estrutura centralizada na medida em que existem dois tipos de servidores: servidor de recolha e servidor de monitorização.

O primeiro efectua monitorização activa e tem como principal objectivo fazer a recolha de informação em ambientes remotos, sendo que depois a informação é transmitida ao servidor de monitorização que a processa e apresenta numa interface Web. Diz-se que este último efectua uma monitorização passiva.

Esta estrutura apresenta como grande vantagem não só a capacidade de conseguir monitorizar um maior número de dispositivos, mas também o facto de os servidores de recolha conseguirem diminuir a carga existente no servidor principal.

Outra vantagem do Nconf é a possibilidade de importar ficheiros CSV (Comma-Separated Values) com informação sobre novos dispositivos, serviços, contactos que se pretendam adicionar. Embora este método tenha obrigatoriamente executado que ser através da linha de comandos, é uma mais-valia para a ferramenta nomeadamente nos casos em que se pretende implementar uma arquitectura de rede já proveniente de outra configuração do Nagios.

Esta ferramenta tem a preocupação de garantir uma rápida configuração de todo o sistema, sendo que para isso possui uma opção de clonagem de serviços. Os serviços podem ser clonados entre dispositivos, o que significa que caso um determinado serviço esteja associado a uma máquina e se pretenda associá-lo a outra, a ferramenta permite copiar essa informação de modo automático poupando assim todo o tempo de inserção e garantido a uniformidade na monitorização da rede.

Por fim, o Nconf possui um sistema de permissões que permite fazer uma separação entre utilizadores normais e administradores. Sendo assim, os utilizadores apenas têm acesso a configurar um determinado conjunto de parâmetros, enquanto os administradores podem parametrizar tudo aquilo que a aplicação oferece.

3.1.1.4 - Funcionamento

O modo de funcionamento do Nconf é bastante linear e consistente. O utilizador usa a interface gráfica para configurar o Nagios tendo em conta as suas preferências, sendo que depois tem à sua disposição uma opção que permite exportar as actualizações efectuadas. Ao executar esta opção, a informação é introduzida na base de dados e simultaneamente são gerados os ficheiros cfg responsáveis por caracterizar todas as parametrizações do Nagios. O processo inverso também é possível, ou seja, existe a possibilidade de utilizar uma configuração já existente do Nagios e importá-la para a base de dados do Nconf.

Sendo assim, esta aplicação torna o processo de configuração da ferramenta Nagios bastante mais simples, para além se tornar ajuda crucial para todos os administradores que não dominam o Nagios na totalidade.

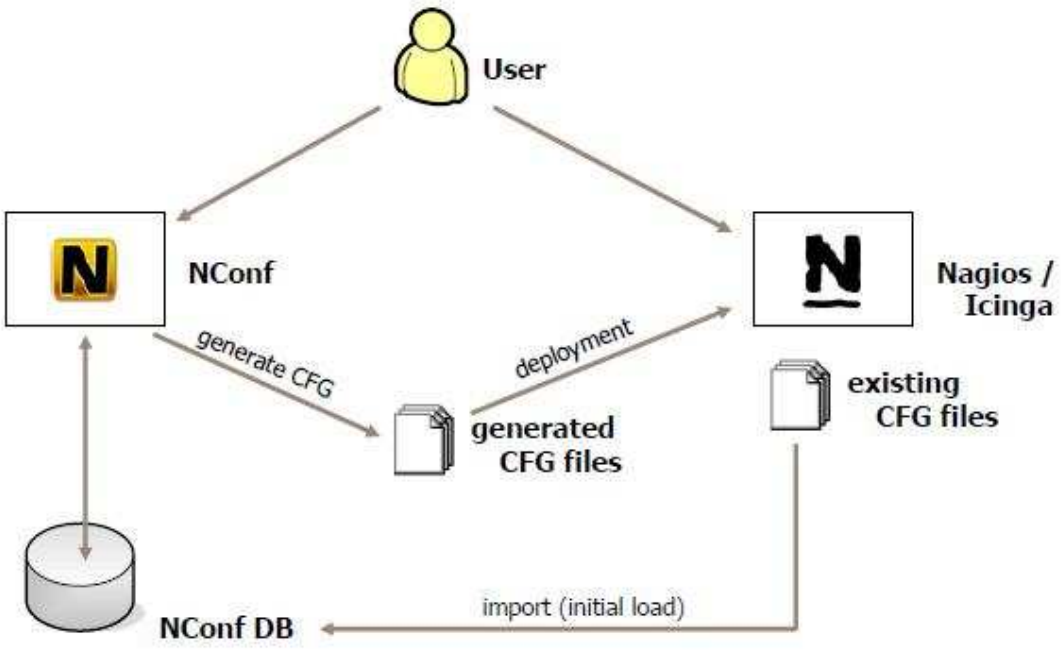

Figura 3.3 - Funcionamento do Nconf

3.1.1.5 - Análise de resultados

Tendo em conta todas as características observadas ao longo da utilização do Nconf, pode-se concluir que se apresenta como uma aplicação que permite reduzir em parte todo o processo de monitorização. No entanto, e embora possibilite a criação de uma nova configuração através da interface Web, acaba por criar apenas um ficheiro compactado com todos os ficheiros de configuração e por não possibilitar a implementação automática do mesmo.

O facto de suportar a monitorização distribuída apresenta-se como uma solução bastante interessante no cenário de grandes redes de monitorização. A possibilidade de definir novos objectos, importados através de ficheiros CSV, é outra das características que é igualmente importante para aligeirar o processo de configuração em grandes redes.

A grande falha do Nconf é a ausência de representação, tanto em termos de tabelas como gráficos, do estado dos dispositivos na rede e desempenho dos mesmos e dos seus recursos. Acaba por ser apenas uma ferramenta de configuração do Nagios. Torna-se portanto indispensável aceder a interface do Nagios para poder ver o estado de toda a informação relativa a rede. Uma nova versão do Nconf seria esperado corrigir este problema.

3.1.2 - GroundWork Monitor

O GroundWork Monitor é uma das soluções de monitorização mais completas e sofisticadas de todo o mercado. Esta aplicação reúne um conjunto de tecnologias "open source" com um subsistema de gestão de informação. Também ele apresenta a capacidade de monitorizar sistemas, aplicações, bases de dados, servidores e equipamentos de rede.

3.1.2.1 - Requisitos

O ficheiro de instalação do GroundWork instala automaticamente todo o tipo de ferramentas extra que sejam necessárias para o funcionamento do programa. Instala o Nagios, Cacti, Ntop, servidor MySQL e Apache, Ganglia, Net SNMP, NRPE, NSCA, Perl e até PHP, fazendo com que não sejam necessários quaisquer tipos de requisitos prévios.

3.1.2.2 - Arquitectura

A arquitectura do GroundWork Monitor é dividida em quatro camadas: camada de configuração, camada de instrumentação, camada de recolha e camada de visualização. Apesar de cada uma das camadas desempenhar um determinado tipo de funções, todas elas são igualmente importantes para o funcionamento do programa[22].

Relativamente às tarefas de configuração, embora não sejam parte formal do processo de recolha de dados, desempenham um papel fundamental no fluxo de informação. Esta camada constitui o local onde os recursos da rede a monitorizar são definidos, o que por sua vez vai influenciar directamente todos os outros componentes do sistema. Além disso, controla o modo como os dados são recolhidos pela camada de instrumentação e indica às ferramentas que processam esses mesmos dados o que fazer após terem sido recolhidos.

É na camada de instrumentação que são definidas as ferramentas responsáveis pela monitorização. Embora o GroundWork Monitor suporte várias ferramentas para recolher dados, ele utiliza o software Nagios como aplicação primária pois este se apresenta como uma solução que permite trabalhar com qualquer outra aplicação via linha de comandos.

Por sua vez, a camada de recolha funciona como uma ponte de ligação entre a camada de instrumentação e a camada de visualização. Mais especificamente, contém as bases de dados fundamentais para o sistema e as ferramentas de integração que ligam os componentes independentes ao sistema global. Para o armazenamento de dados, esta camada usa uma base de dados MySQL que armazena informação importante (hosts e serviços monitorizados por exemplo) que é importante pois representa uma localização comum que pode ser acedida por múltiplas ferramentas; conjunto de scripts de controlo e bibliotecas de programação (Perl, PHP, Java e SOAP) que fornecem interfaces para aplicações quando tem que interagir com sistema directamente.

A camada superior, de visualização, contém as aplicações Web usadas para disponibilizar os dados recolhidos pela camada inferior. Existem uma dúzia de aplicações, mas o componente principal é o conjunto Guava.

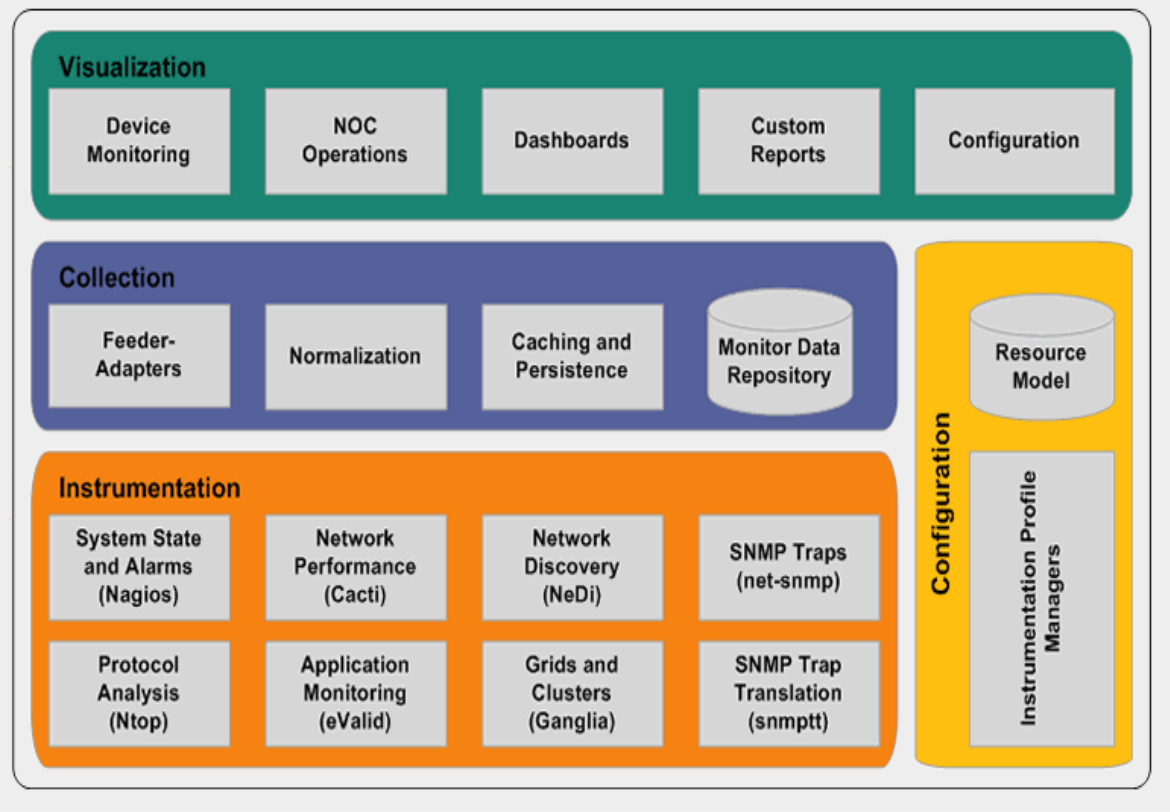

Figura 3.4 - Arquitectura do GroundWork Monitor

3.1.2.3 - Funcionalidades

Relativamente às funcionalidades suportadas pelo GroundWork, tudo aquilo que é possível efectuar alterando manualmente os ficheiros do Nagios, esta ferramenta também o permite.

Possibilita ao utilizador a capacidade de adicionar hosts, serviços, comandos, templates, contactos, etc. Uma característica importante desta ferramenta, é que após a definição de qualquer comando, é possível testar a sua sintaxe de modo a saber se o processo ocorreu conforme planeado.

A própria interface do GroundWork tem incorporado o frontend disponibilizado pelo Nagios, podendo o utilizador consultar directamente todas as alterações efectuadas sem ter que abandonar o programa.

3.1.2.4 - Funcionamento

Segundo uma perspectiva de alto nível, o modo de funcionamento do GroundWork pode ser explicado da seguinte maneira: a camada de configuração define os componentes da rede que necessitam de ser monitorizados e o modo como os dados são processados após terem sido recolhidos; após os dados serem recolhidos, os componentes na camada de instrumentação são utilizados para recolher amostragens de dados através de uma enorme variedade de interfaces de monitorização; após isso, as ferramentas de integração na camada intermédia reúnem os dados para processamento e armazenamento. Por fim, as aplicações Web na camada de cima disponibilizam a informação para os utilizadores[22].

3.1.2.5 - Análise de resultados

Depois de analisar, pode-se concluir que na teoria é uma solução mais completa e com mais qualidades em relação aos seus concorrentes. Ao contraio de muitos frontends, esta ferramenta é mais do que uma mera ferramenta de configuração. Pode-se pensar no GroundWork como uma ferramenta independente de monitorização de tão completa que é.

Apresenta um layout bastante limpo e organização muito boa, mas o facto de querer cobrir todas as funcionalidades disponibilizadas pelo Nagios, acaba por atingir um nível de complexidade elevado para quem tem o primeiro contacto com a ferramenta.

O facto de se basear numa ferramenta "open source", torna esta solução também gratuita. No entanto se um utilizador desejar aumentar as potencialidades da solução e utilizar as extensões disponíveis, só poderá utilizar extensões desenvolvidas pela comunidade. As extensões comerciais desenvolvidas pela equipa do GroundWork Monitor são exclusivas a quem por elas pagar.

Capítulo 4

Implementação e teste do módulo de monitorização

Este capítulo tem como objectivo descrever pormenorizadamente todas as tarefas realizadas, procedimentos efectuados e implementações executadas no desenvolvimento deste projecto. Sendo assim, a primeira secção justifica a opção tomada e apresenta um conjunto de especificações gerais relacionadas com o trabalho. De seguida, é feita uma descrição da plataforma de desenvolvimento e do ambiente em que o trabalho foi realizado. A terceira secção apresenta a arquitectura idealizada e o papel desempenhado por cada um dos seus componentes. A quarta e última secção descreve os detalhes de toda a implementação necessária, com principal destaque para as funcionalidades disponíveis e para a base de dados utilizada como suporte ao módulo desenvolvido.

4.1 - Introdução

Apesar das ferramentas analisadas no capítulo anterior possuírem mecanismos que lhes permitem uma integração com a IPBrick e funcionalidades para desempenharem as funções pretendidas, acabou por ser adoptada uma metodologia ligeiramente diferente.

O Nconf foi descartado devido a dois pormenores: em primeiro lugar, e como já foi referido anteriormente, é uma ferramenta apenas compatível com bases de dados MySQL. Sendo a IPBrick um sistema operativo que utiliza uma base de dados PostgreSQL para armazenar todo o tipo de informações, esta característica tornou-se um factor impeditivo à sua utilização. Para além disso, mostrou-se uma ferramenta que apresenta algumas deficiências ao nível da sua API e alguns erros no seu funcionamento. Em relação ao GroundWork, para além de sofrer do mesmo problema em relação à base de dados, o seu ficheiro de instalação engloba um conjunto de programas adicionais que são automaticamente

instalados no sistema, tais como: Nagios, Apache, MySQL, Cacti, Ntop, entre outros. Uma vez que a IPBrick já possui um servidor Apache e a grande maioria dos outros programas não são necessários para aquilo que se pretende, esta aplicação também acabou por ser excluída.

Para todos os efeitos, a solução encontrada para resolver o actual problema acaba por reunir algumas características das duas ferramentas mencionadas. Basicamente, a solução passou por idealizar um sistema com uma arquitectura semelhante àquela encontrada no Nconf, sendo que a base de dados utilizada seria implementada directamente na IPBrick e com uma estrutura idêntica à do GroundWork Monitor. Isto significa que tudo aquilo que está relacionado com a implementação e comunicação entre ficheiros do sistema, bases de dados e interface foi realizado de modo completamente independente.

Uma vez que o objectivo do trabalho está relacionado com a monitorização de todo o parque informático, houve a necessidade de garantir que as IPBricks consigam não só monitorizar, mas também serem monitorizadas. Em virtude deste pormenor, foram implementados mecanismos que lhes garantissem não só funcionar em modo servidor, mas também em modo cliente.

Além disso, houve também a necessidade de acrescentar à interface da IPBrick um conjunto de menus e de opções que permitissem ao utilizador configurar e definir todos os parâmetros relacionados com a monitorização do parque informático.

4.2 - Plataforma de desenvolvimento

Todo este projecto foi desenvolvido nas instalações da empresa iPortalMais, sendo que os seus responsáveis se encarregaram de facultar todo o material necessário ao seu desenvolvimento. Foram também fornecidos os ficheiros que constituem o código fonte da IPBrick, de modo a serem analisados e assim servirem como base à implementação do módulo de monitorização.

Todos os componentes foram desenvolvidos a partir de um terminal com sistema operativo Ubuntu e testados em simultâneo numa IPBrick v5.3.

4.2.1 - Ferramentas de suporte ao desenvolvimento

O desenvolvimento deste módulo teve como suporte essencialmente duas ferramentas: o editor Bluefish e o phpPgAdmin.

Relativamente ao Bluefish, é um editor bastante utilizado para o desenvolvimento de páginas dinâmicas, suportando linguagens como HTML, XHTML, CSS, PHP, SQL e Javascript. Uma das suas grandes vantagens é a capacidade que o programa apresenta de se adaptar à linguagem que está a ser utilizada, oferecendo sugestões ao utilizador tornando assim mais fácil a escrita do código[23].

O phpPgAdmin é uma aplicação Web, desenvolvida em PHP, que oferece uma interface gráfica para a gestão e manipulação de bases de dados PostgreSQL. Embora a sua utilização não fosse obrigatoriamente necessária, acabou por se tornar uma ferramenta preponderante pois permitiu uma mais fácil gestão das bases de dados e uma visão mais concreta do modo como o sistema está organizado. Além disso, esta aplicação tem a capacidade de simplificar as tarefas de criação, alteração e pesquisa de tabelas ou bases de dados[24].

4.2.2 - Estrutura da IPBrick

Uma vez que o objectivo do projecto consiste em implementar mecanismos de monitorização na IPBrick, este sistema operativo apresenta-se como o componente principal de todo este trabalho. Sendo assim, esta secção pretende caracterizar a sua estrutura e os seus componentes que, de uma maneira ou de outra, estão directamente ligados ao projecto desenvolvido. Será analisada a interface da IPBrick, a estrutura e constituição da sua base de dados e o modo como a comunicação entre estes dois componentes é realizada.

4.2.2.1 - Interface

A interface da IPBrick é baseada em HTML, AJAX e PHP e pode ser acedida a partir de qualquer browser, sendo apenas necessário colocar o endereço IP ou FQDN da máquina como URL e introduzir os respectivos dados (utilizador e palavra-chave) de autenticação. Basicamente, é constituída por um conjunto de menus, formulários e campos de selecção que permitem ao utilizador configurar todo o sistema operativo, como por exemplo: adicionar máquinas e grupos de máquinas, configurar as propriedades das interfaces de rede (endereço IP, máscara, endereço de rede e broadcast), configurar o servidor de email, DHCP ou DNS, configurar a firewall, configurar o serviço VPN (Virtual Private Network) e muito mais. A interface possui uma opção que permite ao utilizador aplicar as alterações efectuadas, sendo que neste caso é feita uma sincronização com a base de dados e todas as modificações efectuadas são lá armazenadas.

4.2.2.2 - Base de dados

O funcionamento de todo este sistema operativo é suportado por duas bases de dados distintas: a ipbricksoft e a ipbox.

A primeira é responsável por armazenar todas as informações que estão directamente ligadas à construção da interface gráfica com a qual o utilizador interage. É constituída por quatro tabelas: menu, pages, links e liga_pages_links. A tabela menu contém as referências a todos os menus da interface que disponibilizam o acesso às diferentes páginas; a tabela pages contém as páginas a serem incluídas na interface; por sua vez, a tabela links representa a

localização dos ficheiros de cada uma das páginas no sistema e a liga_pages_links faz a ligação da página com o respectivo ficheiro.

Relativamente à base de dados ipbox, é onde são armazenadas as configurações de todos os serviços do sistema operativo. Máquinas adicionadas ao sistema, grupos de máquinas existentes, domínio e endereços IP são exemplos de informações presentes nesta base de dados.

4.2.2.3 - Classes de acesso à base de dados

A ligação entre a interface e a base de dados é feita através de um conjunto de classes localizadas na pasta PHP. Convém realçar que existe uma classe para cada serviço da IPBrick, ie, para cada um dos serviços existe um ficheiro que contém todas as "queries" que efectuam os pedidos de informação alojada na base de dados.

4.3 - Arquitectura

A arquitectura idealizada para implementar o módulo de monitorização na IPBrick é constituída fundamentalmente por quatro componentes: a interface, a classe de acesso, as bases de dados e os ficheiros do sistema. A figura 4.1 apresenta um esboço da arquitectura desenvolvida.

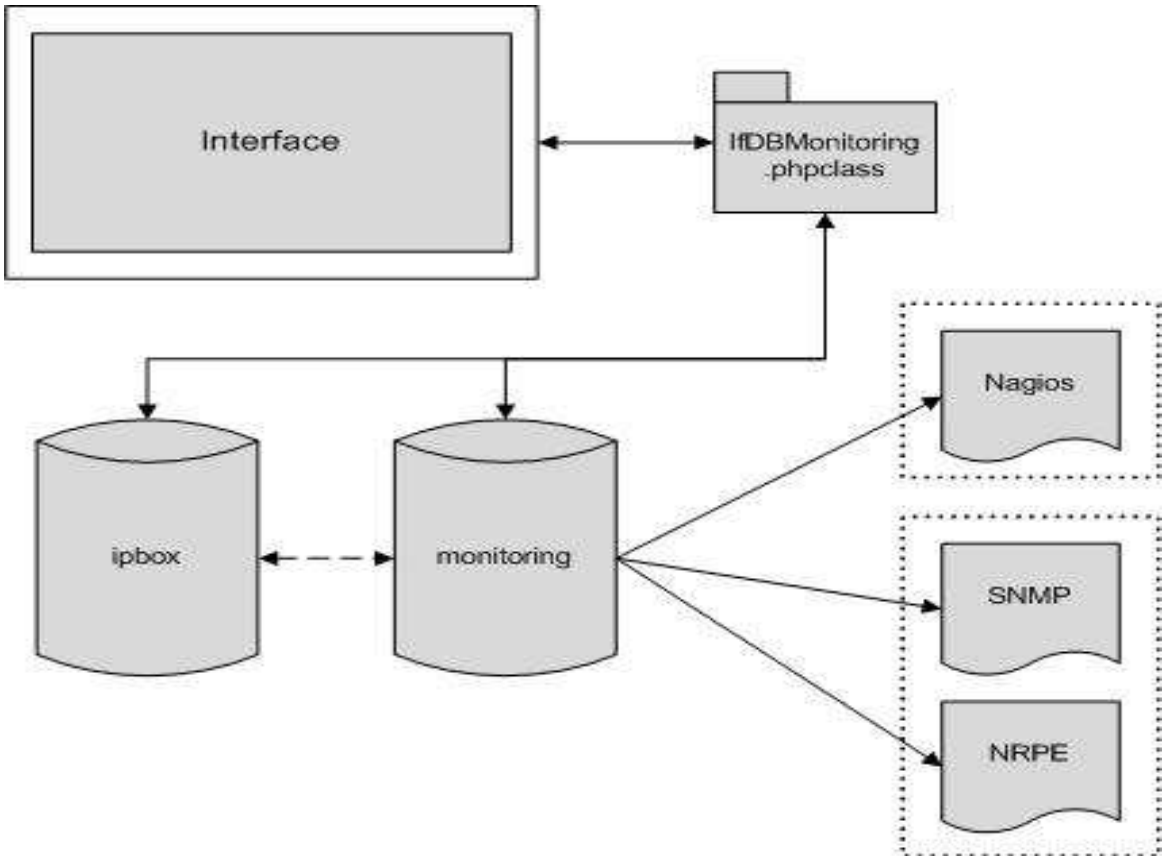

Figura 4.1 – Arquitectura do módulo de monitorização

4.3.1 - Interface de monitorização

Através da interface, o utilizador tem a possibilidade de activar/desactivar a monitorização, configurar mecanismos de alertas e ainda definir o modo de operação (cliente ou servidor), sendo que as opções disponíveis variam em função da escolha efectuada.

Caso a máquina seja configurada para funcionar como cliente, o utilizador necessita de especificar quais os endereços IP das máquinas que estão autorizadas a monitorizá-la. Para além disso, existe também a possibilidade de definir uma comunidade SNMP, limitando assim o acesso a determinadas MIBs (Management Information Base) apenas a algumas estações de monitorização.

Configurar a máquina para funcionar em modo servidor consiste em dar-lhe a capacidade de monitorizar outras estações na rede e os seus respectivos serviços. É importante referir que o facto de a IPBrick já possuir uma funcionalidade que permite adicionar novas máquinas e grupos de máquinas foi aproveitado para o desenvolvimento deste módulo, na medida em que qualquer máquina que seja adicionada fica automaticamente disponível para ser monitorizada. Dependendo do tipo de máquina adicionada, existem uma série de serviços pré-definidos que o utilizador pode escolher se quer monitorizar ou não.

Por fim, o utilizador tem ainda a possibilidade de consultar o frontend do Nagios directamente a partir da IPBrick e assim visualizar todas as configurações efectuadas.

4.3.2 - IfDBMonitoring.phpclass

A ligação entre a base de dados e a interface é feita recorrendo ao ficheiro IfDBMonitoring.phpclass, uma vez que este contém todas as "queries" que são utilizadas no acesso à base dados. Esta metodologia permite uma melhor estruturação do código desenvolvido, uma vez que todas as operações executadas sobre a base de dados estão localizadas num ficheiro específico.

4.3.3 - Base(s) de dados de monitorização

Tal como foi referido na secção 4.3.1, o facto de este módulo recorrer a funcionalidades já existentes da IPBrick, trouxe a necessidade de utilizar a base de dados ipbox pois é lá que são guardadas as informações sobre máquinas e grupos de máquinas já adicionados.

Em relação à base de dados monitoring, foi criada especificamente com o objectivo de armazenar todos os parâmetros e opções relacionados com a monitorização da rede.

4.3.4 - Ficheiros do sistema

Este processo de monitorização envolve a manipulação de três classes de ficheiros: o ficheiro do agente NRPE, o ficheiro de configuração do agente SNMP e os ficheiros de configuração do Nagios.

Relativamente aos dois primeiros, assumem particular importância quando a IPBrick é configurada para funcionar como cliente. O ficheiro do NRPE, localizado em /etc/xinetd.d/nrpe, permite especificar quais os endereços IP que estão autorizados a aceder à maquina para efectuar uma monitorização. Em relação ao ficheiro de configuração do SNMP, cuja localização é /etc/snmp/snmpd.conf, é onde são introduzidas as comunidades SNMP.

Os ficheiros de configuração do Nagios representam o local onde são definidas as máquinas e os respectivos serviços que o Nagios vai monitorizar e são manipulados quando a máquina é definida para funcionar em modo servidor.

É importante referir que para efeitos da implementação do módulo, todos estes ficheiros tiveram que ter permissões de escrita de modo a serem editados tendo em conta as configurações por parte do utilizador.

4.3.4.1 - snmpd.conf

O ficheiro /etc/snmp/snmpd.conf é executado sempre que o agente snmpd é iniciado e contém informações sobre a configuração do mesmo. Embora o conteúdo do ficheiro seja extenso e o número de opções configuráveis seja elevado, apenas algumas são relevantes para o projecto em questão.

Em primeiro lugar este ficheiro permite definir comunidades, que basicamente representam nomes que definem privilégios de acesso às MIBs. Os atributos necessários para definir uma comunidade são: nome, endereço IP que identifica a comunidade, máscara, permissões (apenas leitura, apenas escrita ou leitura e escrita) e um identificador de objecto que representa a porção da árvore MIB à qual a respectiva comunidade tem acesso. No mínimo, é apenas necessário definir o seu nome sendo que todos os outros parâmetros assumem um valor padrão (IP: 0.0.0.0; máscara: 0.0.0.0; permissões: apenas leitura e identificador de objecto: iso.3)[26].

Além disso, e embora não tendo sido implementado nada nesse sentido para este módulo de monitorização, este ficheiro também permite definir "traps" - definição de quais os hosts que são alertados quando determinado evento é gerado no agente.

4.3.4.2 - nrpe

O ficheiro do agente NRPE possui vários atributos, mas apenas um assume especial importância neste contexto. Trata-se do atributo "only_from" que permite definir os endereços IP das máquinas que estão autorizadas a aceder ao agente.

Além disso, convém referir que neste ficheiro também é possível definir a porta utilizada pelo NRPE, sendo que por defeito o seu valor é 5666 (manteve-se inalterado).

4.3.4.3 - Nagios

O Nagios instala por defeito um elevado número de ficheiros onde são definidas as configurações da ferramenta. No entanto, tendo em conta as funcionalidades que foram implementadas, a integração desta ferramenta com a IPBrick apenas exigiu a manipulação de alguns desses ficheiros[27].

Relativamente ao ficheiro commands.cfg, contém a definição de todos os comandos interpretados pelo Nagios. Uma vez que a lista de serviços a monitorizar é pré-definida para cada tipo de máquina e não existe a possibilidade de adicionar ou remover novos comandos, este ficheiro manteve-se inalterado e já preenchido com a definição de todos os comandos a utilizar.

O ficheiro templates.cfg armazena os templates dos serviços, das máquinas e dos contactos. Também neste caso, o seu conteúdo não foi alterado uma vez que os templates padrão generic-host, generic-service e generic-contact continham as directivas necessárias para caracterizar os dispositivos, os serviços e os contactos.

Em relação aos ficheiros hosts.cfg e services.cfg, foi adoptada uma metodologia ligeiramente diferente. Basicamente, foi criado um ficheiro para cada tipo de máquina existente (tipo_maquina.cfg), sendo que a definição dos hosts que pertencem a esse mesmo tipo e os serviços para cada máquina são definidos nesse ficheiro. Ex: IPBricks.cfg, linuxworkstations.cfg, servers.cfg, routers.cfg, etc.

O ficheiro contacts.cfg guarda os contactos que são definidos para serem notificados em caso de problemas.

Os ficheiros hostgroups.cfg e contactgroups.cfg, como os próprios nomes indicam, têm como objectivo criar grupos de máquinas e de contactos e definir quais os seus membros.

O anexo A fornece mais informação acerca dos ficheiros de configuração do Nagios, nomeadamente as directivas que normalmente são utilizadas e aquelas que foram necessárias para este módulo.

4.4 - Desenvolvimento

Esta secção apresenta todos os detalhes envolvidos no desenvolvimento do módulo de monitorização.

4.4.1 - Hierarquia do módulo de monitorização

A hierarquia de directorias e ficheiros utilizada obedeceu aos critérios utilizados no desenvolvimento de módulos para a IPBrick e está representada na figura 4.1.

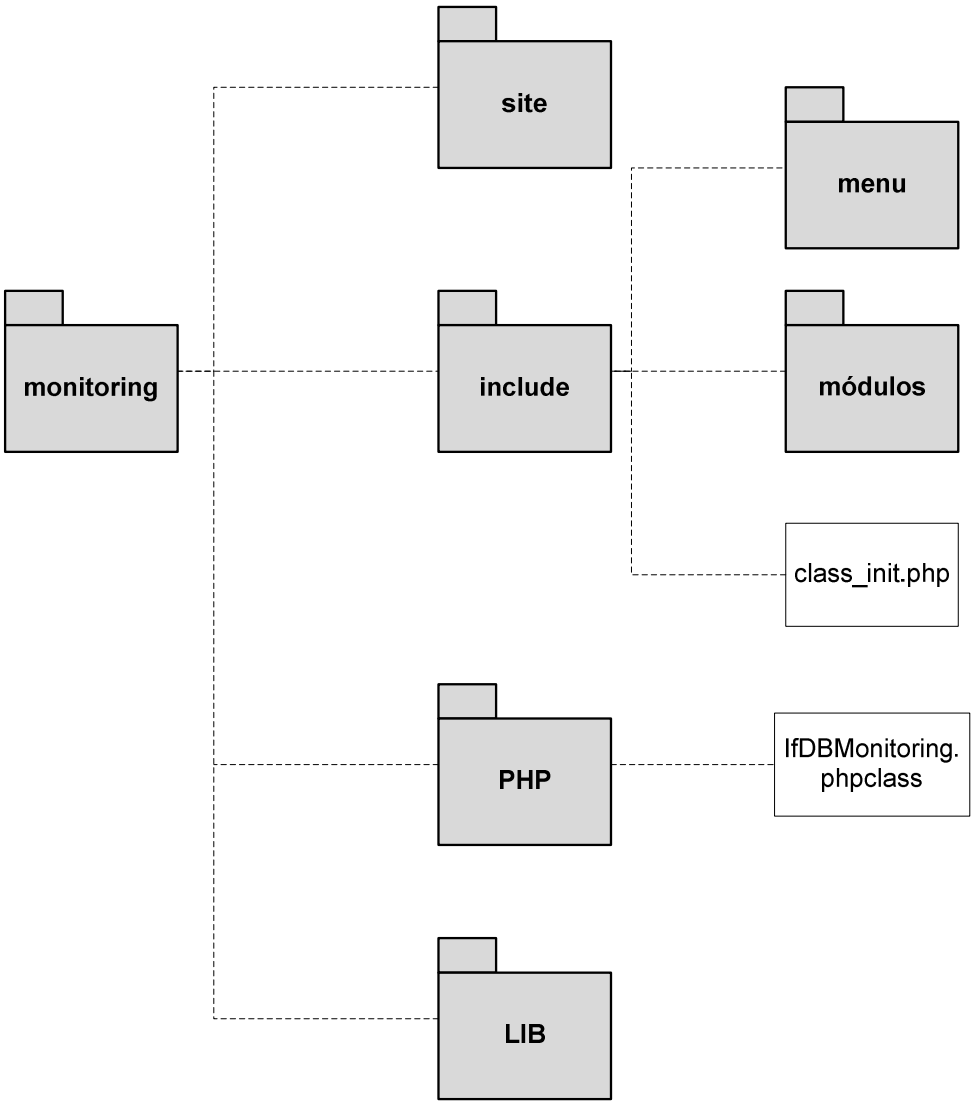

Figura 4.2 – Hierarquia das directorias que constituem o módulo

4.4.2 - Estrutura da interface

À interface da IPBrick já mencionada anteriormente, foram acrescentados um conjunto de menus e de opções que permitem ao utilizador configurar parâmetros de monitorização. Este conjunto de opções foi inserido no menu de configurações avançadas da IPBrick e denominado Monitorização.

No interior do novo submenu o utilizador tem três opções disponíveis: definições, máquinas e grupos de máquinas.

A primeira opção encaminha o utilizador para a página monitoring_settings_view.php onde é possível ligar/desligar a monitorização, definir qual o modo de operação, configurar mecanismos de alertas e visualizar o output dado pelo Nagios.

A segunda opção refere-se à página monitoring_machines_view.php e fornece uma lista das máquinas já adicionadas, podendo o utilizador para cada uma delas definir quais os serviços a monitorizar.

A terceira opção apresenta uma lista dos grupos de máquinas que pertencem ao sistema, podendo ser definidos critérios de monitorização para cada um deles. A página em questão é monitoring_machinegroups_view.php.

Cada uma das páginas contem um botão que permite aplicar as configurações e submeter as informações introduzidas para a base de dados.

4.4.3 - Funcionalidades

4.4.3.1 - Activar/desactivar monitorização

Tal como já foi referido, este módulo permite a um utilizador ligar ou desligar a monitorização para uma determinada IPBrick.

Caso a monitorização seja activada, os serviços snmpd e xinetd são inicializados e a máquina estará preparada para funcionar como cliente ou como servidor, podendo assim responder às exigências por parte do utilizador. Se a monitorização for desligada, os respectivos serviços são desligados e a máquina não só se torna inalcançável por qualquer outra estação da rede, como também fica incapacitada de efectuar qualquer tipo de monitorização.

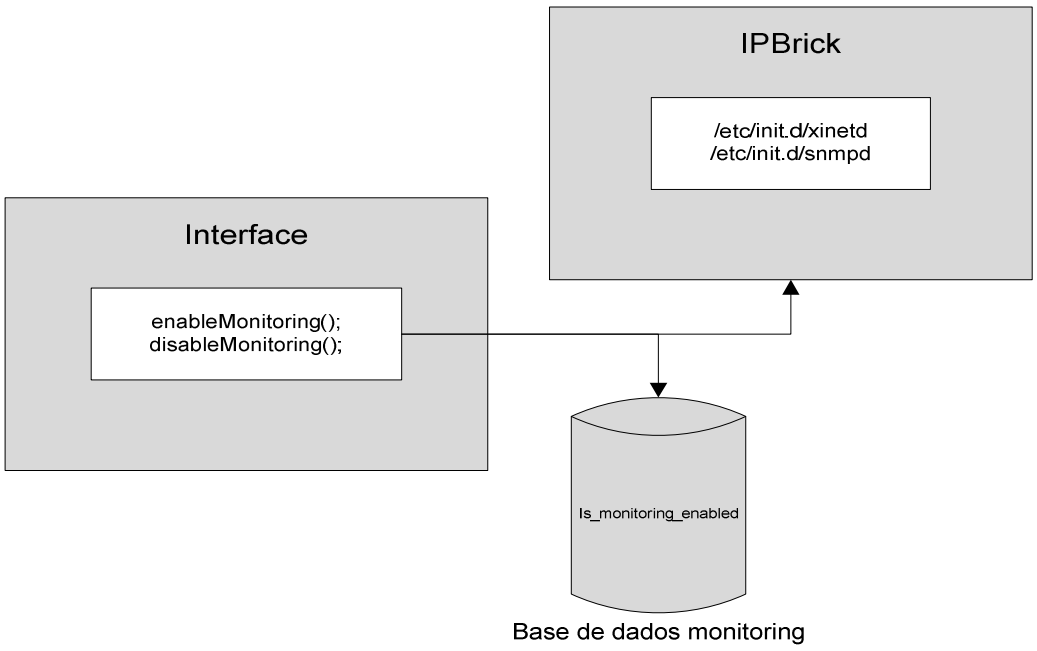

Figura 4.3 - Diagrama de actualização do estado da monitorização

Tal como é possível observar na figura anterior, existem duas funções que são executadas quando é exigido uma actualização do estado da monitorização: enableMonitoring() e disableMonitoring().

Ao serem executadas, o conteúdo da tabela is_monitoring_enabled é automaticamente alterado e a informação fica guardada na base de dados. Simultaneamente, também como é possível verificar na figura 4.3, os serviços são iniciados/terminados consoante a opção escolhida.

As imagens seguintes apresentam o conteúdo da tabela para os casos em que a monitorização está ligada e desligada, respectivamente.

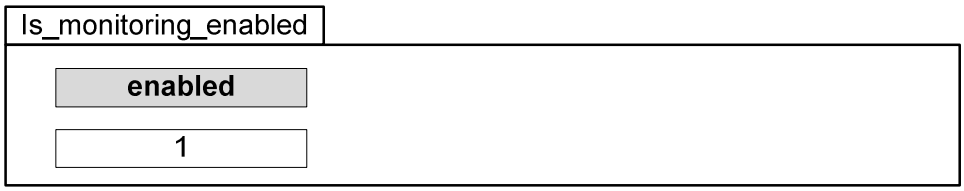

Figura 4.4 - Monitorização ligada

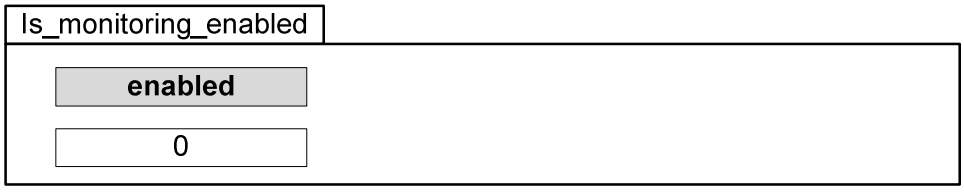

Figura 4.5 - Monitorização desligada

A tabela is_monitoring_enabled apenas contém uma coluna – enabled – uma vez que o seu objectivo é simplesmente verificar se a monitorização está ligada ou desligada. O valor da coluna pode ser 1, caso a monitorização esteja activa, ou então 0, caso contrário.

4.4.3.2 - Modo de operação

Ao activar a monitorização para uma determinada máquina, pode-se de seguida configurar o seu modo de operação.

Essa informação também irá ficar armazenada na base de dados, na tabela operation_mode.

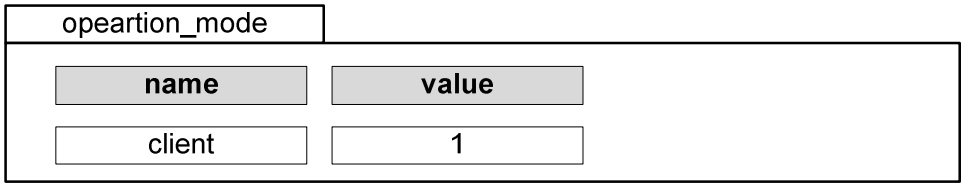

Figura 4.6 - Modo de operação cliente

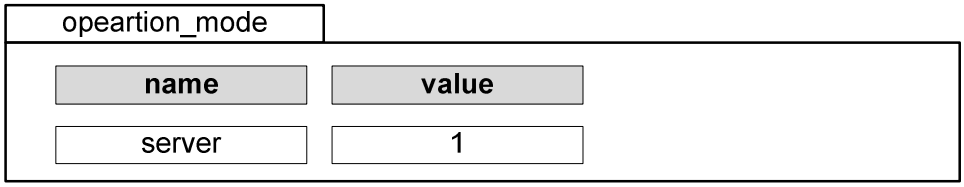

Figura 4.7 - Modo de operação servidor

Basicamente, a tabela contém duas colunas distintas: name e value. A primeira identifica o nome do modo de operação (neste caso Cliente e Servidor), enquanto a segunda identifica qual deles está activo. De notar que partiu-se do principio que uma máquina apenas pode desempenhar um modo de operação de cada vez.

Os procedimentos envolvidos neste processo são relativamente simples de explicar: uma vez que a IPBrick apenas está preparada para assumir um modo de funcionamento, sempre que um determinado modo é definido, a tabela em cima mencionada é apagada através da função clearOperationMode() e depois, através da função setOperationMode(), o novo modo é definido.

4.4.3.3 - Modo Servidor

Quando a IPBrick é configurada para operar em modo servidor, as máquinas previamente adicionadas podem começar a ser monitorizadas.

Este módulo tem a capacidade de disponibilizar as máquinas através de uma função da IPBrick que é a getSystemComputers(). Esta função acede à base de dados ipbox e coloca num vector toda a informação das máquinas, ficando este preenchido com os identificadores, nomes, endereços IP e tipos de máquinas existentes.

Por motivos de coerência e facilidade na manipulação da informação, essa informação é sincronizada com a base de dados monitoring ficando esses dados também disponíveis numa tabela da base de dados criada exclusivamente para este módulo. Este procedimento é executado utilizando a função insertMonitoredHosts(), que não faz mais do que aceder à base de dados e colocar a informação armazenada no vector na tabela monitoredHosts.

Para disponibilizar ao utilizador os grupos de máquinas já existentes, este módulo recorre à função getSystemGroupsComputers(). A partir do momento em que o identificador e o nome de todos os grupos se encontram num vector temporário, esses dados são colocados na tabela monitoredHostGroups da nova base de dados através de uma função criada para o efeito.

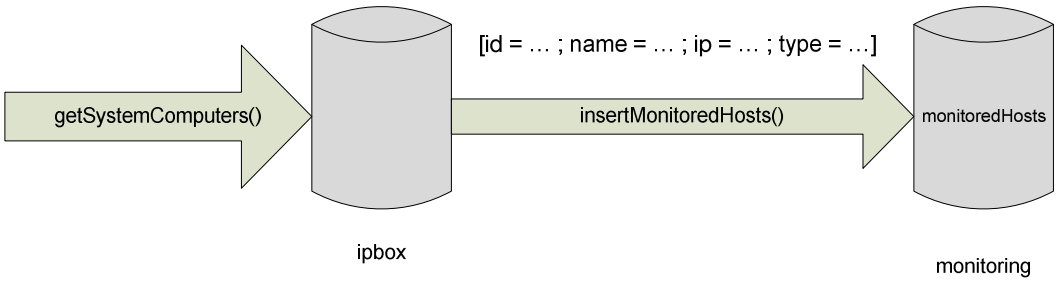

Figura 4.8 - Actualização das máquinas disponíveis para monitorização

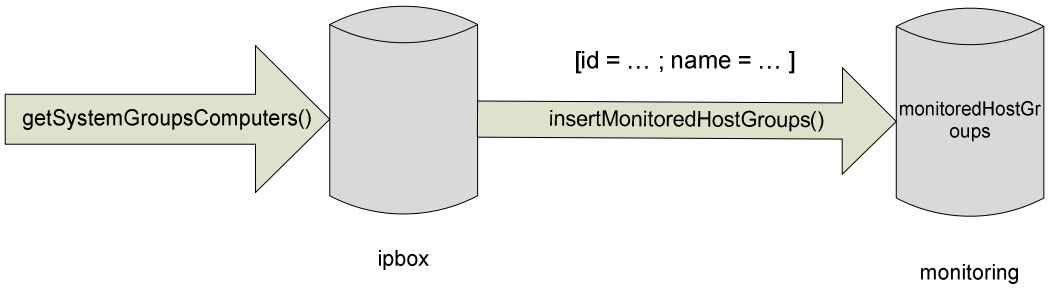

Figura 4.9 - Actualização dos grupos de máquinas disponíveis para monitorização

É importante referir que todas estas funções são executadas sempre que o utilizador decide aplicar configurações, permitindo assim haver uma constante sincronização entre as duas bases de dados e garantir que a base de dados monitoring contém sempre uma informação actualizada das máquinas e grupos de máquinas existentes.

Embora a IPBrick permita adicionar vários tipos de máquinas (estações de trabalho, estações de trabalho + softphone, estações de trabalho Linux, estações de trabalho Linux + softphone, impressoras, telefones IP, terminais Linux e terminais Windows), este módulo foi desenvolvido para suportar apenas alguns desses tipos. Entre as máquinas suportadas, destacam-se as IPBricks que funcionam como servidores e as estações de trabalho, sendo que nas primeiras a intenção é fundamentalmente monitorizar serviços enquanto nas segundas o objectivo é observar os seus recursos.

Os procedimentos para adicionar uma máquina e os respectivos objectos de análise ao sistema de monitorização do Nagios são relativamente simples. O utilizador necessita de se deslocar ao conjunto de máquinas existentes através do submenu "Máquinas" e carregar naquela que pretende monitorizar. De seguida, e tendo em conta o tipo de máquina

seleccionada, é-lhe apresentado um conjunto de parâmetros que podem ser definidos como objecto de observação.

Depois de seleccionar os serviços/recursos e de submeter as configurações, a máquina é automaticamente adicionada ao universo do Nagios, bem como os objectos que foram definidos.

Uma vez que o processo de manipulação dos ficheiros de configuração do Nagios se revelou um bocado complicado, foi adoptado um esquema específico. Sempre que algum dispositivo é introduzido, o módulo implementado efectua uma limpeza do ficheiro em questão e volta a preenche-lo com as opções definidas no momento.

Dado que o identificador da máquina seleccionada é guardado, o módulo utiliza as funções getHostNameByID e getHostIPByID para aceder à base de dados (tabela monitoredHosts) e recolher o nome e IP da máquina em causa, parâmetros essenciais para editar os ficheiros de configuração.

Relativamente à introdução da informação nos ficheiros, este procedimento é efectuado pela função addHostNagios. De seguida é apresentado um exemplo para uma estação de trabalho Linux.

```
function addHostNagios($hostname, $ip)
{
    $file = "/usr/local/nagios/etc/objects/linuxworkstation.cfg";
    $fhandle = fopen($file, "a");
    …
    fputs($fhandle, "define host {");
    fputs($fhandle, "use generic-host");
    fputs($fhandle, "host_name");
    …
    fputs($fhandle, $hostname);
    fputs($fhandle, "alias");
    fputs($fhandle, $hostname);
    …
    fputs($fhandle, "address");
    fputs($fhandle, $ip);
    …
    fputs($fhandle, "}");
}
```

Importante realçar que nem todas as linhas estão representadas na função anterior. Não foram representadas todas as mudanças de linha necessárias, apesar de serem fundamentais para que o ficheiro apresente o formato correcto de modo a ser interpretado pelo Nagios.

Caso sejam definidos serviços ou recursos para serem monitorizados, o módulo recorre à função addServiceNagios() para o efectuar. De seguida está apresentado um exemplo que adiciona um comando à mesma estação de trabalho Linux do exemplo anterior.

```
function addServiceNagios($hostname, $service, $command)
{
    $file = "/usr/local/nagios/etc/objects/linuxworkstation.cfg";
    $fhandle = fopen($file, "a");
    …
    fputs($fhandle, "define service {");
    fputs($fhandle, "use generic-service");
    fputs($fhandle, "host_name");
    …
    fputs($fhandle, $hostname);
    fputs($fhandle, "service_description");
    fputs($fhandle, $service);
    …
    fputs($fhandle, "check_command");
    fputs($fhandle, $command);
    …
    fputs($fhandle, "}");
}
```

As variáveis $service e $command que são referidas estão armazenadas na tabela monitoringServices, que já se encontra preenchida e não pode ser modificada.

Este processo de inserir máquinas e serviços para serem monitorizados pelo Nagios é suportado por uma tabela na base de dados que é monitoring_hosts_services. Esta tabela contém a associação entre dispositivos e serviços, permitindo ao módulo verificar se determinado serviço já está associado a um determinado host, para que não seja introduzida informação duplicada nos ficheiros do Nagios.

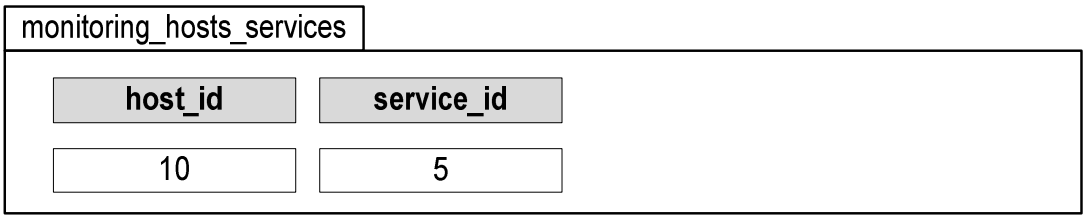

Figura 4.10 - Tabela monitoring_hosts_services

Para remover a informação dos ficheiros, o módulo utiliza a função delete_host(). Uma vez que a informação de máquinas e serviços é introduzida de forma sequencial no ficheiro, a solução passou por criar uma função que percorresse o ficheiro do Nagios e eliminasse todas as entradas entre duas definições de máquinas distintas. A função está representada em seguida.

```
function delete_host($ip)
{
    $file = "/usr/local/nagios/etc/objects/linuxworkstation.cfg";
    $file_content = file_get_contents($file);
    $aux = array();
    $aux = explode("define host", $file_content);
    unset($aux[0]);
    foreach($aux as $key => $val)
    {
            $pos = strpos($val, $ip);
            if ($pos != false) {
            $found = 1;
            unset($aux[$key]); }
    }
    if ($found == 1)
    {
    foreach($aux as $key => $val) {
    $done .= "define host" . $val; }
    file_put_contents($file, $done);
    }
}
```

4.4.3.4 - Modo Cliente

Configurar a IPBrick para funcionar como cliente consiste em activar os serviços que funcionam como suporte a este modo de operação, garantindo assim que a máquina pode ser acedida.

Ao introduzir o endereço IP de um servidor de monitorização, a ferramenta vai à tabela monitoringServers e verifica se essa máquina já existe ou não. Caso não exista, iniciam-se os procedimentos para introduzir o seu endereço no ficheiro do NRPE e na respectiva tabela. A função que executa este procedimento é a função edit_nrpe_file() e está indicada de seguida.

```
function edit_nrpe_file($ip)
{
    $file = "/etc/xinetd.d/nrpe";
    $fhandle = fopen($file, "r");
    $content = fread($fhandle, filesize($file));
    $content = str_replace("127.0.0.1", "127.0.0.1 $ip", $content);
    $fhandle = fopen($file, "w");
    fwrite($fhandle, $content);
    fclose($fhandle);
}
```

O mecanismo de funcionamento desta função consiste em encontrar o endereço IP localhost (que está sempre definido no ficheiro) e a partir daí inserir os endereços IP que são fornecidos pelo utilizador.

Para introduzir uma comunidade SNMP, o procedimento é idêntico mas a função utilizada é a edit_snmp_file(), tal como se pode verificar:

```
function edit_snmp_file($community)
{
    $file = "/etc/snmp/snmpd.conf";
    $fhandle = fopen($file, "a");
    $string = "rocommunity $community\n";
    fwrite($fhandle, $string);
    fclose($fhandle);
}
```

Adicionar uma comunidade SNMP implica editar o ficheiro snmpd.conf e colocar o nome a seguir à directiva "rocommunity". Este comando introduz uma nova comunidade que garante o acesso de leitura às MIBs.

4.4.3.5 - Alertas

O utilizador pode criar contactos que serão notificados em caso de problemas com as máquinas monitorizadas.

Este processo pode ser realizado na página monitoring_alerts.php e basicamente utiliza a função create_contact() para editar o ficheiro contacts.cfg. Simultaneamente, essa informação é introduzida no ficheiro de configuração da máquina associada de modo a que qualquer problema com ela seja redireccionado para esse mesmo utilizador.

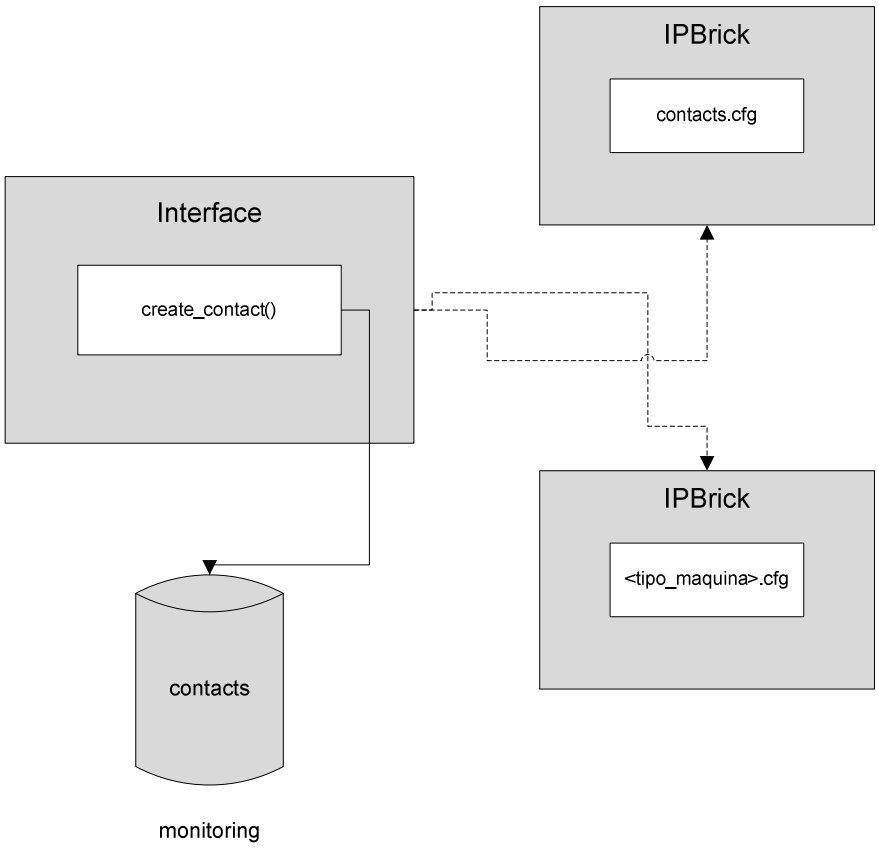

Figura 4.11 - Criação de um contacto

Neste módulo não foi implementada a funcionalidade para associar contactos a serviços, sendo que ao adicionar um alerta para uma determinada máquina, todos os seus serviços a ela associados ficam automaticamente afectados.

4.4.4 - Base de dados

Com o objectivo armazenar a informação relacionada com os aspectos de monitorização, foi criada uma nova base de dados para servir de suporte a este módulo. Esta base de dados, tal como já foi dito anteriormente, designa-se monitoring.

Embora nem todas as tabelas que foram criadas e que pertencem a esta base de dados estejam mencionadas na figura, aquelas que estão directamente relacionadas com a relação cliente/servidor são mencionadas.

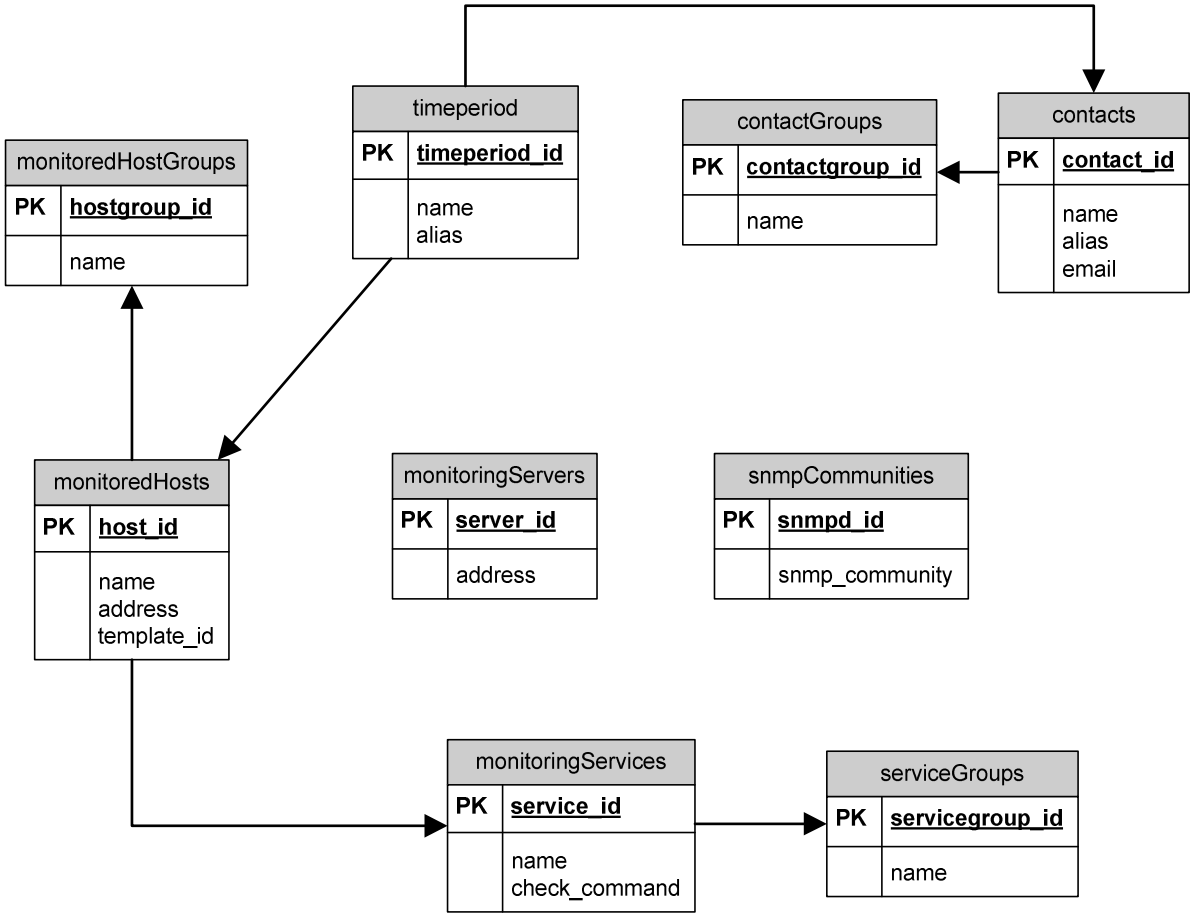

Figura 4.12 - Estrutura da base de dados

A tabela monitoredHosts, como o próprio nome indica, vai armazenar todos os hosts que irão ser monitorizados. É caracterizada por quatro colunas: identificador, nome, endereço e o id do template, que caracteriza o tipo de máquina em causa. Esta tabela está associada à tabela monitoredHostGroups, que serve para guardar todos os grupos criados. Por sua vez, esta tabela apenas contém duas colunas, uma que guarda o identificador único do grupo e outra que guarda o seu nome.

Tal como se pode verificar, as tabelas monitoringServers e snmpCommunities não têm qualquer associação às restantes. Estas duas tabelas desempenham um papel importante quando a IPBrick é configurada para operar em modo cliente, mas caracterizam-se apenas por guardar a informação dos endereços IP e das comunidades.

Uma vez que podem ser associados determinados serviços às máquinas definidas, a tabela monitoringServices armazena todos os serviços que podem ser utilizados na monitorização. Esta tabela possui três colunas: o identificador do serviço, o seu nome e a sintaxe do comando responsável por ser executado no agente NRPE.

As outras três tabelas servem para gerir os mecanismos de alertas, destacando-se a tabela contacts, que tal como já foi referida, é responsável por armazenar os endereços de email dos utilizadores definidos para receberem notificações.

4.5 - Resultados obtidos

No final do processo de implementação foram efectuados vários testes ao módulo desenvolvido e às suas funcionalidades. Este subcapítulo tem como objectivo descrever os testes que foram realizados e apresentar os resultados obtidos. É importante referir que para além da IPBrick onde o módulo foi integrado, foi disponibilizada uma segunda IPBrick de modo a que fosse possível testar todo o processo de monitorização - uma das máquinas foi configurada para desempenhar funções de servidor enquanto a outra máquina foi definida para funcionar como cliente.

4.5.1 - Funcionamento em modo cliente

Com o objectivo de verificar o funcionamento da IPBrick em modo cliente, houve a necessidade de introduzir na interface o endereço IP da máquina responsável pela monitorização. Após este procedimento ter sido efectuado, e como este processo não é unidireccional, foi preciso indicar na IPBrick de monitorização qual a máquina que iria ser observada.

Este processo exigiu a inclusão da máquina na rede, através da funcionalidade disponibilizada pela IPBrick, sendo que depois foi definida a monitorização de serviços como o SMTP, HTTP e POP3 e a monitorização do uptime através do protocolo SNMP.

Ainda na máquina de monitorização, e após terem sido aplicadas as configurações, foi possível observar a máquina remota no mapa da rede do Nagios. Este facto garante que o módulo de monitorização não apresenta qualquer tipo de problemas na manipulação e actualização dos ficheiros dos agentes NRPE e snmpd. No primeiro caso, todos os endereços IP das máquinas de monitorização são adicionados enquanto no segundo caso o ficheiro do agente snmpd fica preenchido com as comunidades previamente estabelecidas que garantem o acesso à informação.

4.5.2 - Funcionamento em modo servidor

Para verificar a operação da IPBrick em modo servidor foi aproveitado o teste realizado anteriormente. Após ser adicionada a máquina remota e serem definidos quais os serviços e recursos a observar, a consulta da interface do Nagios permite observar não só a máquina remota, como também a verificação dos serviços e recursos. O facto de estes procedimentos ocorrerem sem qualquer tipo de erro dá-nos a garantia que houve uma correcta edição e interpretação dos ficheiros de configuração do Nagios. Além disso, toda a informação presente da base de dados que é utilizada neste processo de monitorização (comandos a executar por exemplo) é correctamente transportada e interpretada pelos ficheiros do sistema.

4.5.3 - Alertas

A capacidade de gerar alertas que respondem a falhas na rede também foi testada. Ao adicionar um determinado endereço de email a partir da interface da máquina de monitorização, teoricamente criam-se as condições para o utilizador ser notificado em caso de problemas.

Ao desligar a máquina remota da rede, e uma vez que deixa de existir qualquer tipo de conectividade entre as duas IPBricks, a monitorização de serviços foi interrompida. Esta situação fez com que o módulo de monitorização, através do servidor de email da IPBrick, envia-se uma notificação para o utilizador a informá-lo do problema.

Estes procedimentos garantem que a informação é correctamente introduzida no ficheiro do Nagios responsável por armazenar os contactos utilizados nos alertas, exactamente como seria de esperar.

4.5.4 - Funcionamento geral

De um modo geral, a execução das operações que estavam disponíveis no módulo de monitorização revelaram-se bem sucedidas. Este cenário de teste com apenas duas IPBricks, embora não se enquadre com a realidade dos parques informáticos das empresas, deu uma ideia do seu funcionamento e a garantia que este módulo funciona em ambientes mais desenvolvidos.

Capítulo 5

Conclusão e trabalho futuro

É indiscutível que a evolução em termos tecnológicos tem mudado a forma de actuar da grande maioria das empresas, permitindo inclusivamente que estas começassem a monitorizar o comportamento dos seus parques informáticos de uma forma automática, mais rápida e mais detalhada. Estes procedimentos tornaram-se de tal modo importantes para o sucesso das organizações, podendo-se afirmar que em caso da sua ausência a empresa está a perder terreno face à concorrência mais directa.

Uma vez que a IPBrick não possuía quaisquer tipos de mecanismos que permitissem uma avaliação do parque informático, houve a necessidade de ser implementado um módulo que respondesse a tais exigências.

O desenvolvimento deste módulo de monitorização revelou-se um desafio bastante interessante com objectivos complicados para atingir, nomeadamente dada a enorme variedade de ferramentas de monitorização existentes no mercado.

Após um estudo detalhado da oferta existente, e tendo em conta as características e funcionalidades que eram pretendidas para o módulo, delineou-se um formato para a nova solução que fosse capaz de implementar algumas ideias e disponibilizar funcionalidades nesse sentido.

5.1 - Síntese do trabalho desenvolvido

O trabalho desenvolvido iniciou-se com um estudo das várias ferramentas de monitorização e de gestão open source existentes no mercado. O objectivo era encontrar soluções para efectuar a monitorização de todo o parque informático e obter o inventário de software e hardware das máquinas pertencentes à rede.

De seguida, e uma vez que havia a necessidade de integrar essas ferramentas na IPBrick, foi feito um estudo pormenorizado do funcionamento do sistema operativo.

Por fim, procedeu-se à implementação do módulo proposto que permitisse a monitorização de dispositivos e respectivos serviços/recursos através das ferramentas escolhidas previamente.

Embora tivesse sido estipulado no início do documento a integração de ferramentas de gestão de inventário com a IPBrick, estas funcionalidades acabaram por não ser implementadas, não só devido à falta de tempo, mas também porque a integração de ferramentas de monitorização foi considerado o objectivo principal na perspectiva da empresa onde o trabalho foi desenvolvido.

5.2 - Desenvolvimento futuro

O módulo desenvolvido para monitorizar o parque informático poderá ser melhorado futuramente através da inclusão de novas funcionalidades que não foram implementadas nesta primeira fase.

Entre essas funcionalidades destaca-se o facto do utilizador poder configurar a IPBrick para funcionar simultaneamente como cliente e como servidor. Embora este aspecto não seja crucial, acontece bastantes vezes uma determinada máquina monitorizar um subconjunto de dispositivos e estar sujeita a um controlo por parte de um servidor hierarquicamente superior. Neste contexto, esta funcionalidade será a solução ideal.

Relativamente aos mecanismos de alertas, era importante garantir que o módulo possuísse mais opções para além do correio electrónico. Por exemplo, a capacidade de enviar mensagens via telemóvel será uma mais-valia para a ferramenta. Além disso, e ainda relacionado com os alertas, poderão ser implementados mecanismos que permitam ao administrador definir quais as horas e quais os dias da semana durante os quais os utilizadores são notificados, para além dos templates que o Nagios oferece por defeito.

Uma vez que este módulo foi desenvolvido para trabalhar apenas com determinados tipos de máquinas, poderão também ser implementados processos que sejam compatíveis com a monitorização de outros dispositivos, como máquinas Windows ou até telefones IP.

Por fim, e tal como acontece em muitas ferramentas que oferecem interfaces de configuração para o Nagios, era importante implementar mecanismos que garantissem ao módulo extrair a informação de um determinado conjunto de ficheiros do Nagios e actualizar a sua base de dados tendo em conta essa informação. Esta solução iria permitir a implementação automática de uma arquitectura de rede já existente.

Referências

1. (01-06-2011). IPBrick – Plataforma de comunicações para empresas. Disponível em: http://www.ipbrick.pt.
2. (2006, 01-06-2011). iPortalMais - Soluções de Engenharia para Internet e Redes, Lda. Disponível em: http://www.iportalmais.pt.
3. A. Clemm, Ed., Network Management Fundamentals: a guide to understanding how network management technology really works. Indianapolis, IN 46240 USA: Cisco Press, 2007.
4. (04-06-2011). Nagios – The Industry Standard in IT Infrastructure Monitoring. Disponível em: http://www.nagios.org.
5. W. Kocjan, et al. (2008). Learning Nagios 3.0 a detailed tutorial to setting up, configuring, and managing this easy and effective system monitoring software.
6. J. Turnbull and Books24x7 Inc. (2006). Pro Nagios 2.0 [Text]. Available: http://ezproxy.villanova.edu/login?URL=http://www.books24x7.com/marc.asp?bookid=14663.
7. N. Enterprises. (2009-2011 04-06-2011). Nagios – Features. Disponível em: http://www.nagios.org/about/features.
8. N. Enterprises. (2009-2011 04-06-2011). Nagios – Plugins. Disponível em: http://www.nagios.org/projects/nagiosplugins.
9. Barth, Wolfgang. 2006. Nagios: System and Network Monitoring. U.S. ed. Munich San Francisco: Open Source Press;
10. E. Galstad. (1999-2007, 06-06-2011). NRPE Documentation. Disponível em: http://nagios.sourceforge.net/docs/nrpe/NRPE.pdf.
11. M. Medin. (06-06-2011). NSClient++ - Reference Manual. Disponível em: http://www.nsclient.org/nscp/wiki/internal/documentation/reference.
12. A. Vladishev, “Open Source Enterprise Monitoring with Zabbix,” presented at the Open Source Data Center Conference, Nurnberg, 2009.

13. Olups, Rihards. Zabbix 1.8 Network Monitoring. Birmingham: Packt Publishing Ltd., 2010.
14. Antal, Barzan. IT Inventory and Resource Management with OCS Inventory NG 1.02. Birmingham: Packt Publishing Ltd., 2010.
15. (2001-2011, 11-06-2011). OCS Inventory NG – Features. Disponível em: http://www.ocsinventory-ng.org/en/about/features.
16. Wiki – GLPI-Project.org. 2011 12-06-2011]; Disponível em: http://www.glpi-project.org/wiki/doku.php?id=en:welcome.
17. J.-P. Lang. (2006-2010, 15-06-2011). OCS Fusion Features Comparison. Available: http://forge.fusioninventory.org/projects/fusioninventory/wiki/Features#Operating-Systems.
18. E. S. a. J. Harrington. (2009, SCF/FEF Evaluation of Nagios and Zabbix Monitoring Systems.
19. (15-06-2011). IPBrick – Appliances. Disponível em: http://www.ipbrick.pt/index.php?oid=322.
20. (18-06-2011). Nconf – Enterprise Nagios Configurator. Disponível em: http://www.nconf.org/dokuwiki/doku.php.
21. A. Gargiulo, "Nconf – Enterprise Nagios configurator," presented at the Open Source Monitoring Conference, Nurnberg, 2009.
22. I. C. GroundWork Open Source. (2008). GroundWork Monitor 5.2.1 Bookshelf. Available: http://proddoc.groundworkopensource.com/Bookshelf_RoboHelp/Bookshelf_5-2.htm
23. (22-06-2011). Bluefish Editor: Home. Disponível em: http://bluefish.openoffice.nl/index.html
24. (22-06-2011). phpPgAdmin – Start. Available: http://phppgadmin.sourceforge.net/doku.php?id=start
25. (20-06-2011). Net-SNMP. Available: http://www.net-snmp.org/docs/readmefiles.html
26. D. Josephsen, Ed., Building a monitoring infrastructure with Nagios. Prentice Hall, 2007, First edition.

Anexo A

Ficheiros de configuração do Nagios

O funcionamento do Nagios tem por base um conjunto de ficheiros que desempenham funções específicas e que podem ser facilmente editados pelo utilizador. Este conjunto engloba os ficheiros de configuração, os plugins, os logs, os ficheiros de comandos, ficheiros temporários, etc.

No entanto, no contexto deste projecto, são os ficheiros de configuração que se apresentam como os mais importantes. Existem vários tipos de ficheiros de configuração, tais como: de dispositivos, de serviços, de contactos, de períodos de tempo, de templates, entre outros. Para todos os efeitos, apenas alguns deles tiveram a necessidade de ser editados e configurados, enquanto outros mantiveram o seu conteúdo original.

Este anexo tem como principal objectivo apresentar uma visão geral do modo como são editados os ficheiros do Nagios, nomeadamente aqueles que, de uma forma ou de outra, têm influência directa no módulo implementado.

Definição de dispositivos:

O ficheiro *hosts.cfg* tem como objectivo definir quais as máquinas de um parque informático que vão ser monitorizadas pelo Nagios. Esta solução implica que todos os dispositivos, independentemente do seu tipo, sejam definidos neste ficheiro.

No entanto, e tal como já foi referido no documento, a solução implementada seguiu uma outra metodologia na medida em que foram criados ficheiros independentes para definir cada um dos tipos de máquinas.

De seguida é apresentado um exemplo de como adicionar um dispositivo ao Nagios, neste caso uma máquina Linux.

```
define host {
    use generic-host
```

```
    host_name Linux
    alias Linux Machine
    address 192.168.100.110
    check_command check-host-alive
    max_check_attempts 10
    notification_interval 120
    notification_period 24x7
    notification_options d,u,r
}
```

Tal como é possível verificar, a definição de um determinado dispositivo exige ao utilizador parametrizar várias directivas. No entanto, no módulo implementado são utilizadas apenas algumas das opções disponíveis.

Tabela A.1 - Parâmetros utilizados na definição de uma máquina

Parâmetro	Descrição
use	Template a utilizar
host_name	Nome do dispositivo
alias	Descrição do dispositivo
address	Endereço IP do dispositivo

Definição de serviços:

Regra geral, é o ficheiro *services.cfg* que armazena todos os serviços a serem observados.

No módulo de monitorização para a IPBrick, a definição de serviços é feita no mesmo ficheiro de configuração onde se definem as máquinas a monitorizar. Este processo, embora não sendo obrigatório, apresenta-se como uma solução mais simplificada e permite ao Nagios ler a informação sequencialmente.

Também a definição de serviços exige o uso de várias directivas, mas uma vez que este módulo já tem associado um conjunto de serviços pré-definidos a cada tipo de dispositivo, apenas alguns parâmetros se revelam importantes.

A seguir está exemplificada a definição de um serviço.

```
define service {
    use generic-service
    host_name Linux
    service_description CPU Load
```

```
    is_volatile 0
    check_period 25
    max_check_attempts 10
    normal_check_interval 10
    retry_check_interval 10
    notification_interval 120
    notification_period 24x7
    notification_options w,u,c,r
    check_command check_nrpe!check_load
}
```

Tabela A.2 – Parâmetros utilizados na definição de um serviço

Parâmetro	Descrição
use	Template a utilizar
host_name	Nome do dispositivo
service_description	Descrição do serviço
check_command	Comando utilizado

Definição de contactos:

O ficheiro *contacts.cfg* guarda as informações dos utilizadores que irão ser notificados quando surgirem problemas com os hosts ou com determinados serviços. O exemplo seguinte demonstra a definição de um contacto.

```
define contact {
    use generic-contact
    contact_name aferreira
    alias André Ferreira
    service_notification_period 24x7
    host_notification_period 24x7
    service_notification_options w,u,c,r
    host_notification_options d,u,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email andreferreira@fe.up.pt
}
```

Os parâmetros utilizados para configurar um contacto no módulo implementado correspondem aos seguintes:

Tabela A.3 – Parâmetros utilizados na definição de um contacto

Parâmetro	Descrição
use	Template a utilizar
contact_name	Nome do utilizador
alias	Descrição do contacto
email	Endereço de email

Definição de grupos de dispositivos:

Os dispositivos adicionados para serem monitorizados podem ser agrupados, havendo assim a possibilidade de se efectuar monitorização a um determinado grupo, em vez de se estar a definir processos de monitorização para todos os hosts individualmente.

Este processo pode ser efectuado editando o ficheiro *hostgroups.cfg*, tal como está indicado em baixo.

```
define hostgroup {
    hostgroup_name IPBricks
    alias Grupo das IPBricks
    contact_groups adminsgrp
    members Linux
}
```

No entanto, para o módulo em questão nem todas as opções são utilizadas quando se define um grupo.

Tabela A.4 - Parâmetros utilizados na definição de um grupo de dispositivos

Parâmetro	Descrição
hostgroup_name	Nome do grupo
alias	Descrição do grupo
members	Membros do grupo

Definição de grupos de contactos:

Da mesma maneira que é possível definir grupos para reunir várias máquinas, a ferramenta Nagios também permite ao utilizador criar grupos para reunir contactos através do ficheiro *contactgroups.cfg*. As estruturas não variam muito de um caso para o outro, tal como se pode verificar.

```
define contactgroup {
    contactgroup_name adminsgrp
    alias Admins Group
    members aferreira
}
```

Para o módulo implementado na IPBrick, e tal como aconteceu com o grupo de dispositivos, apenas três directivas foram utilizadas como se pode verificar na tabela seguinte.

Tabela A.5 - Parâmetros utilizados na definição de grupos de contactos

Parâmetro	Descrição
contactgroup_name	Nome do grupo
alias	Descrição do grupo
members	Membros do grupo

Definição de períodos de tempo:

O ficheiro *timeperiods.cfg* permite definir quais os dias e quais as horas da semana que um determinado contacto recebe notificações sobre um determinado serviço ou determinado dispositivo. Este ficheiro não sofreu qualquer tipo de alteração e o seu conteúdo está indicado de seguida.

```
define timeperiod {
    timeperiod_name 24x7
    alias 24 hours a day, 7 days a week
    sunday 00:00-24:00
    monday 00:00-24:00
    tuesday 00:00-24:00
    wednesday 00:00-24:00
```

```
    thursday 00:00-24:00
    friday 00:00-24:00
    saturday 00:00-24:00
}

    define timeperiod {
        timeperiod_name workhours
        alias "Normal" working hours
        monday 08:00-18:00
        tuesday 08:00-18:00
        wednesday 08:00-18:00
        thursday 08:00-18:00
        friday 08:00-18:00
    }
```

Anexo B

Interface de monitorização

Figura A.1 - Modo de operação Cliente

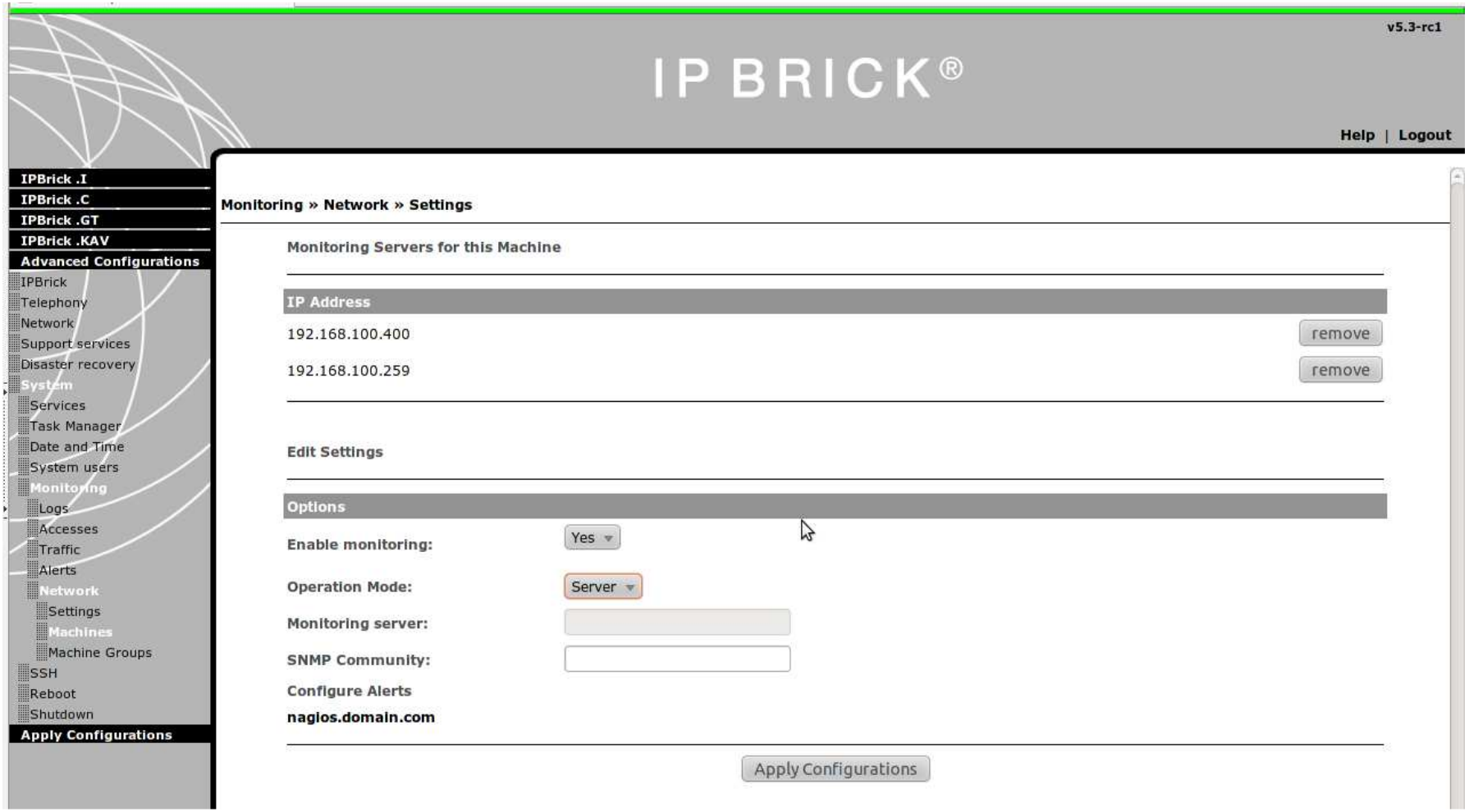

Figura A.2 - Modo de operação Servidor

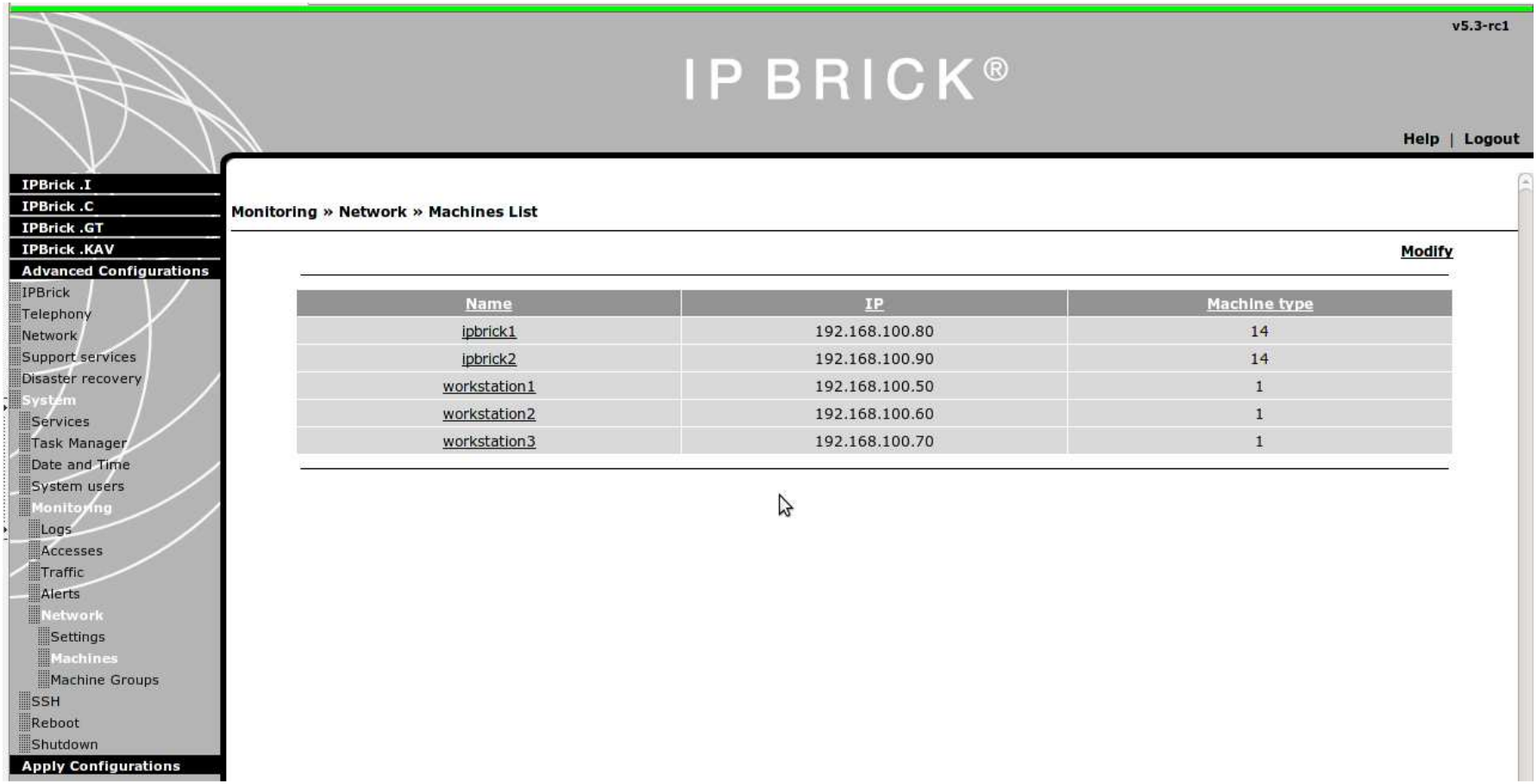

Figura A.3 - Lista de máquinas adicionadas

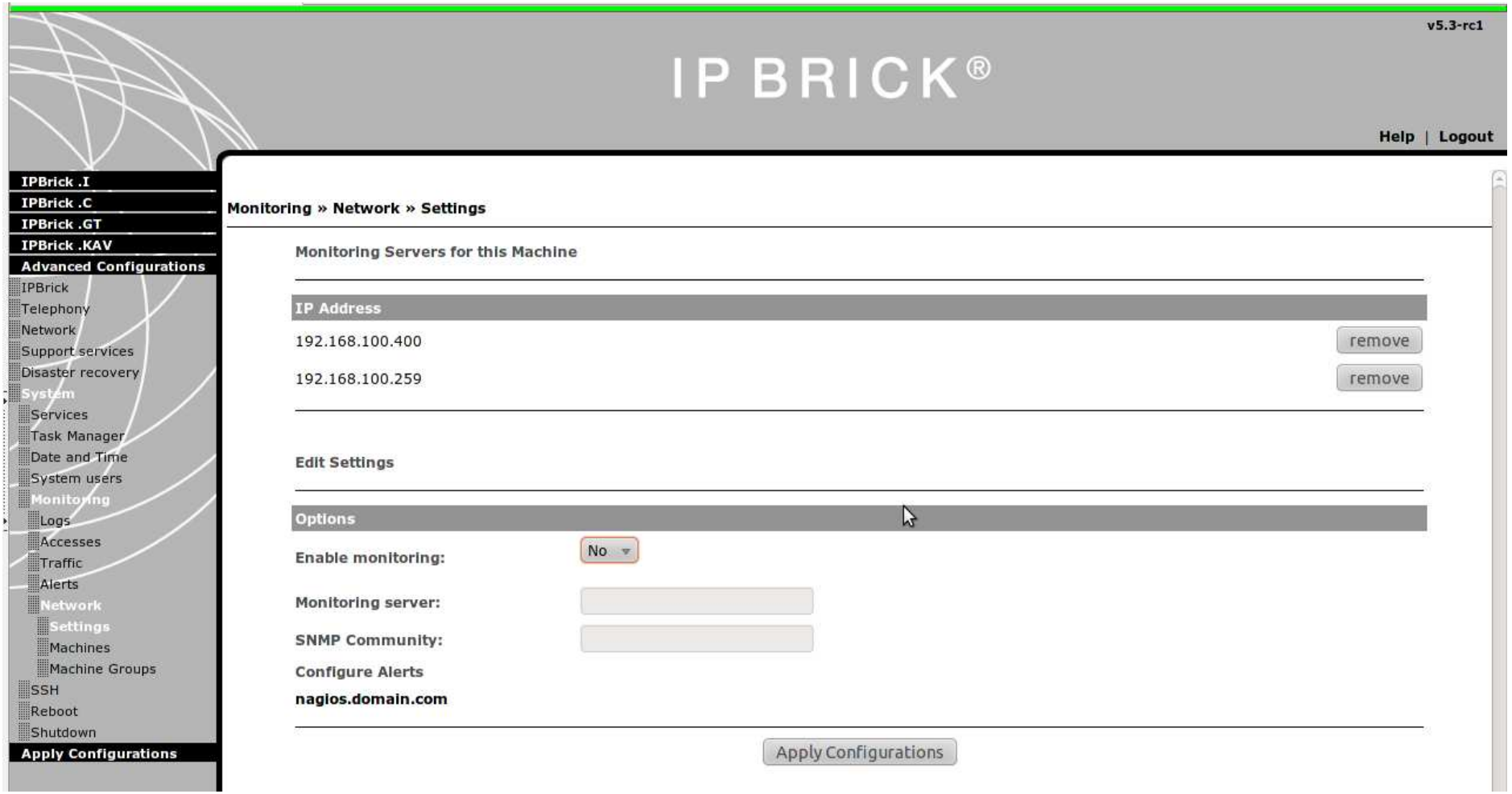

Figura A.4 - Monitorização desactivada

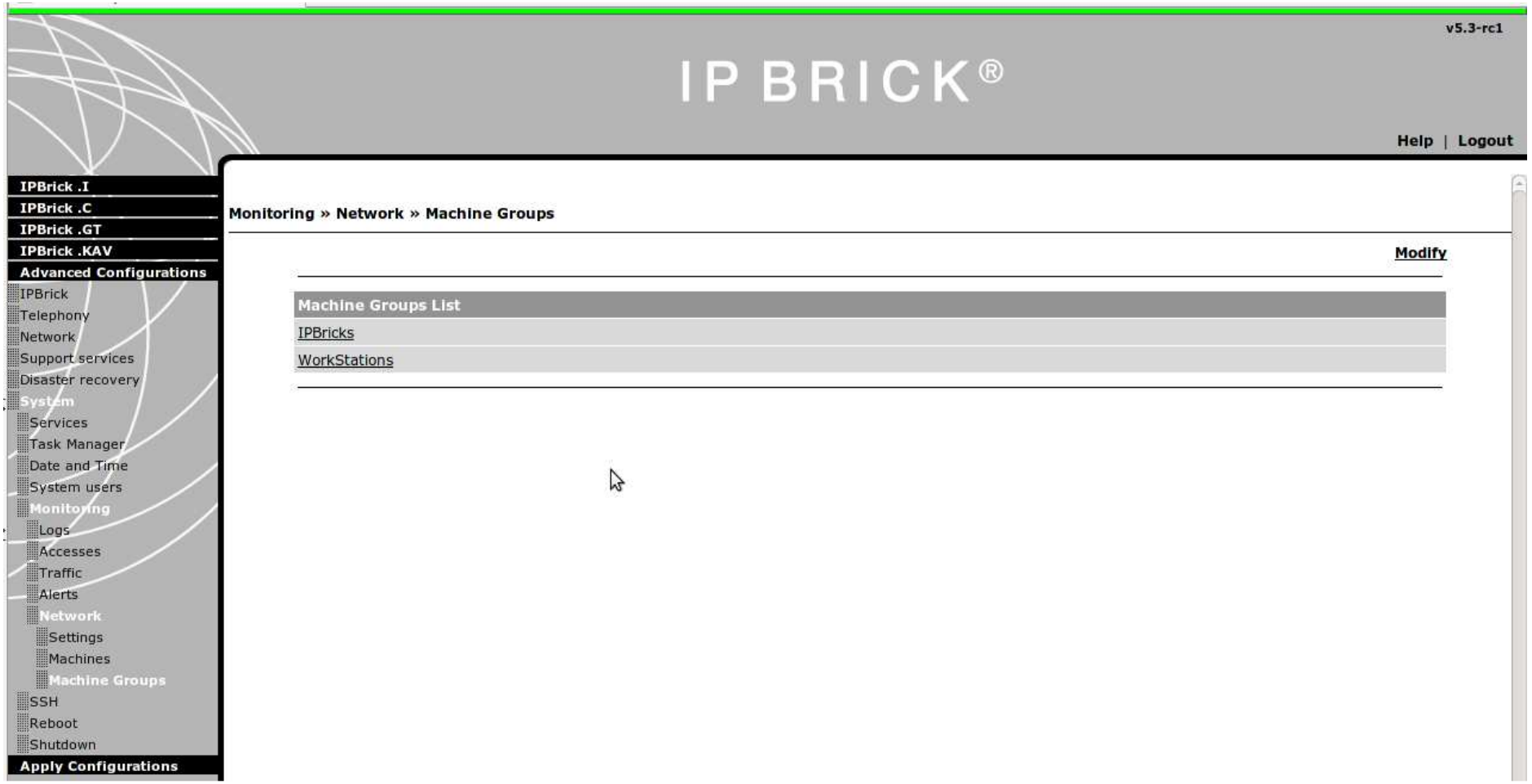

Figura A.5 - Grupos de máquinas adicionados

Figura A.6 - Serviço adicionado a uma máquina