



# Gerência de Redes de Computadores

Mauro Tapajós Santos

Liane Tarouco

Leandro Bertholdo

Francisco Marcelo Marques de Lima

Vanner Vasconcellos

**A RNP** – Rede Nacional de Ensino e Pesquisa – é qualificada como uma Organização Social (OS), sendo ligada ao Ministério da Ciência, Tecnologia e Inovação (MCTI) e responsável pelo Programa Interministerial RNP, que conta com a participação dos ministérios da Educação (MEC), da Saúde (MS) e da Cultura (MinC). Pioneira no acesso à Internet no Brasil, a RNP planeja e mantém a rede Ipê, a rede óptica nacional acadêmica de alto desempenho. Com Pontos de Presença nas 27 unidades da federação, a rede tem mais de 800 instituições conectadas. São aproximadamente 3,5 milhões de usuários usufruindo de uma infraestrutura de redes avançadas para comunicação, computação e experimentação, que contribui para a integração entre o sistema de Ciência e Tecnologia, Educação Superior, Saúde e Cultura.



Ministério da  
**Cultura**

Ministério da  
**Saúde**

Ministério da  
**Educação**

Ministério da  
**Ciência, Tecnologia  
e Inovação**



# Gerência de Redes de Computadores

Mauro Tapajós Santos  
Liane Tarouco  
Leandro Bertholdo  
Francisco Marcelo Marques de Lima  
Vanner Vasconcellos





# Gerência de Redes de Computadores

Mauro Tapajós Santos  
Liane Tarouco  
Leandro Bertholdo  
Francisco Marcelo Marques de Lima  
Vanner Vasconcellos

Rio de Janeiro  
Escola Superior de Redes  
2015

Copyright © 2014 – Rede Nacional de Ensino e Pesquisa – RNP  
Rua Lauro Müller, 116 sala 1103  
22290-906 Rio de Janeiro, RJ

Diretor Geral  
**Nelson Simões**

Diretor de Serviços e Soluções  
**José Luiz Ribeiro Filho**

### **Escola Superior de Redes**

Coordenação  
**Luiz Coelho**

Edição  
**Lincoln da Mata**

Coordenados da Área de Administração de Projetos de Redes  
**Luiz Carlos Lobato**

Equipe ESR (em ordem alfabética)  
**Adriana Pierro, Célia Maciel, Derlinéa Miranda, Edson Kowask, Elimária Barbosa, Evellyn Feitosa, Felipe Nascimento, Lourdes Soncin, Luciana Batista, Renato Duarte e Yve Abel Marcial.**

Capa, projeto visual e diagramação  
**Tecnodesign**

Versão  
**2.0.0**

Este material didático foi elaborado com fins educacionais. Solicitamos que qualquer erro encontrado ou dúvida com relação ao material ou seu uso seja enviado para a equipe de elaboração de conteúdo da Escola Superior de Redes, no e-mail [info@esr.rnp.br](mailto:info@esr.rnp.br). A Rede Nacional de Ensino e Pesquisa e os autores não assumem qualquer responsabilidade por eventuais danos ou perdas, a pessoas ou bens, originados do uso deste material.

As marcas registradas mencionadas neste material pertencem aos respectivos titulares.

Distribuição  
**Escola Superior de Redes**  
Rua Lauro Müller, 116 – sala 1103  
22290-906 Rio de Janeiro, RJ  
<http://esr.rnp.br>  
[info@esr.rnp.br](mailto:info@esr.rnp.br)

---

#### Dados Internacionais de Catalogação na Publicação (CIP)

G367      Gerência de Redes de Computadores / Mauro Tapajós Santos ... [et. al.]. – 2. ed. –  
              Rio de Janeiro: RNP/ESR, 2015.  
              320 p. : il. ; 27,5 cm.

Bibliografia: p.301-304.

1. Gerência de redes. 2. Qualidade de Serviço (telecomunicações) 3. Protocolos de comunicação. 4. Redes de computadores – diagnóstico. I. Santos, Mauro Tapajós.

CDD 005.8

# Sumário

## Escola Superior de Redes

A metodologia da ESR	xi
Sobre o curso	xii
A quem se destina	xii
Convenções utilizadas neste livro	xii
Permissões de uso	xiii
Sobre o autor	xiv

## 1. Introdução à gerência de redes

Importância do gerenciamento	1
Atuação do gerente de redes	2
Atividades comuns de gerenciamento	3
O que gerenciar?	4
Áreas funcionais de gerenciamento (modelo OSI)	5
Gerenciamento de falhas	6
Gerenciamento de desempenho	7
Gerenciamento de configuração	7
Gerenciamento de segurança	7
Gerenciamento de contabilização	8
Gerenciamento dentro do modelo OSI	8
Conceitos	9
Modelo de comunicação	9
Detecção dos sintomas do problema	11

Gerência de configuração	17
Ferramentas para a gerência de contabilização	18
Gerência de desempenho	19
Gerência de segurança	20
Modelo ITIL: Information Technology Infrastructure Library	20
Fluxo de mensagens em SNMP	23
Modelo operacional SNMP	24
RMON – Remote	25
Grupos de Objetos da RMON v1	26
Grupos de Objetos da RMON v2	26
Agentes e gerentes	27
Informação de gerenciamento	28
Conceitos de orientação a objetos	29
Structure of Management Information (SMI)	29
Abstract Sintax Notation (ASN.1)	30
Sintaxe ASN.1 de um objeto da MIB	31
Regras de codificação BER (Basic Encoding Rules – ISO 8825)	32
Representação BER (TLV)	34
Alguns identificadores ASN.1 (tags)	35
ASN.1 e BER (Exemplos)	36

## **2. Ferramentas de inspeção e monitoração de redes**

Problemas na rede	37
Processo de resolução de problemas	38
Ferramentas para inspeção e monitoração	41
Wireshark	41
TCPDUMP	58
Referências	59
Questionário para testar os conhecimentos relativos ao uso do Wireshark	60

## **3. SNMPv1, SNMPv2 e SNMPv3**

Contextualização	63
MIBs Padronizadas	65

Árvore de identificadores	66
Árvore da MIB II	67
MIB para tecnologias de transmissão	69
Novos grupos da MIB II	69
Extensões privadas à MIB	70
Structure of Management Information (SMI)	71
SNMP – Introdução	72
SMIv1 – Tipos de dados	74
SMIv1 – Definição de objetos	75
SNMPv1 – Comunicação e Operações	75
SNMPv1 – Formato da mensagem do protocolo SNMP	77
SNMPv1 – Relação MIB view e Comunidade	77
SNMPv1: Formato da PDU de resposta	78
SNMPv1: Formato do PDU para a trap	79
SNMPv1: Declaração de uma trap específica (macro ASN.1 TRAP-TYPE)	80
SNMPv1: Obtenção de valores de uma MIB	80
SNMPv1: Modificação de valores em tabelas	82
SNMPv2 – Uma nova versão	82
SNMPv2 – Evolução	83
SMIv2 – Novos tipos de dados	84
SMIv2: Textual conventions (RFC 2579)	84
Melhorias na manipulação de tabelas	85
SMIv2 – Definição de objetos	85
SNMPv2: Alterações no protocolo	85
SNMPv2 – Operações	86
SNMPv2 – Operação ‘InformRequest’ e o gerenciamento hierarquizado	87
SNMPv2 – Operação ‘getBulkRequest’	88
SNMPv2 – Mensagens	89
SNMPv2: Tipos de agentes	89
Agentes proxy	90
SNMPv3: Novo modelo de segurança	90
Arquitetura modular SNMPv3	91

Segurança SNMPv3	92
Formato da mensagem SNMPv3	93
SNMP Agente: NET-SNMP	94
NET-SNMP – Aplicações	95
NET-SNMP – Configuração	95
Suporte a SNMPv3 no pacote NET-SNMP	96
Criação de usuários SNMPv3	97
Uso das aplicações com SNMPv3	97

#### **4. Monitoramento remoto – RMON, RMON2, SMON e Host MIB**

Monitores	99
RMON – Objetivos	101
Configuração do RMON	101
Grupos da MIB RMON	102
Eventos e alarmes	103
RMON 2	103
RMON2: Grupo Protocol Directory	105
RMON2: Grupo Protocol Distribution	105
RMON2: Grupo Network Layer Host	105
RMON2: Grupo Network Layer Matrix	106
RMON2: Grupo Application Layer Host	106
RMON2: Grupo Application Layer Matrix	107
SMON: Remote Network Monitoring MIB Extensions for Switched Networks	107
SMON: Grupo smonVlanStats	108
SMON: Grupo smonPrioStats	108
SMON: Grupo portCopy	109
SMON: Grupo dataSourceCaps	109
ntop	109
Gerenciamento de hosts e sistemas	111
Gerenciamento de aplicações	116

#### **5. Aspectos e aplicações de plataformas de gerência**

Modelos de gerência	119
Network Management System (NMS)	125

Arquitetura de um sistema de gerência	<b>128</b>
Obtendo dados da atividade da rede	<b>131</b>
Descoberta da rede	<b>131</b>
Filtrando as atividades da rede baseando-se em limiares	<b>132</b>
Correlação de alarmes	<b>132</b>
Priorização de atividades	<b>134</b>
Escalabilidade	<b>134</b>
Usando uma análise FCAPS na definição de um NMS	<b>135</b>
Gerência de falhas	<b>135</b>
Gerência de Configuração	<b>137</b>
Gerência de Performance	<b>138</b>
Network Management Systems disponíveis no mercado	<b>139</b>
HP IMC – Intelligent Management Center	<b>140</b>
HP NNM: Network Node Manager	<b>141</b>
SNMPc da Catlerock	<b>142</b>
WhatsUp da IPSwitch	<b>143</b>
OPManager, da ManageEngine	<b>144</b>

## **6. Aplicações e plataformas de gerência em software livre**

Soluções OSS/FS	<b>147</b>
Framework FCAPS	<b>148</b>
Gerência de Falhas (Fault)	<b>148</b>
Gerência de configuração (Configuration)	<b>149</b>
Gerência de contabilidade (Accounting)	<b>150</b>
Gerência de desempenho (Performance)	<b>151</b>
Gerência de segurança (Security)	<b>152</b>
Ferramentas livres	<b>153</b>
Zabbix	<b>153</b>
Instalação do Zabbix	<b>154</b>
Utilizando o Zabbix	<b>155</b>
Nagios	<b>159</b>
Instalação do Nagios	<b>159</b>
Customizações do software	<b>160</b>
OpenVAS	<b>162</b>
OCS Inventory	<b>165</b>
Instalação do OCS Inventory	<b>166</b>

Spice Works	168
Instalação do SpiceWorks	168
NTOPng	172
Instalação do NTOPng	172

## **7. Tratamento de registros de ocorrências (logs) e fluxos de dados**

Importância do registro de ocorrências	175
Como registrar os LOGs	176
O servidor de LOGs (LOGHOST)	176
LOGHOST	178
O volume de informações	178
A difícil tarefa de interpretar as mensagens de LOGs	179
Definindo uma estratégia de LOGs	180
O protocolo Syslog	182
Configurando diversos clientes syslog	184
Analizando os LOGs	186
Automatizando a análise de LOGs	187
Monitoramento de fluxos	190
Introdução	190
Como é realizada a coleta dos fluxos	191
Métodos de exportação	196
Onde ativar as sondas	196
Implementações	196
Exemplos de Configurações	199
Considerações finais	201
Referências	203

## **8. Gerenciamento de performance e qualidade de serviço**

Introdução	205
<u>Demonstração</u>	206
Visão Geral Sobre QoS	207
O modelos de serviços QoS fim a fim	208
QoS na internet	209
Serviços Diferenciados	211

Condições de provisionamento de tráfego em dispositivos de borda para provedores de serviços	215
Tipos de serviços	217
Serviço melhor que best-effort	217
Provisionamento e configuração	220
Service Level Agreements	222
Mecanismos para implementar QoS	223
Policy-Based Routing	226
Gerenciamento de congestionamento	228
Mecanismos para evitar congestionamento	229
Mecanismos de policiamento e conformação	233
Qualidade de serviço na prática	234
Conclusão	236
Métricas para o gerenciamento de rede	236
Padrões para monitoração da rede	239
Fatores que influenciam o desempenho de rede	239
Atraso fim-a-fim em um sentido	241
Atraso fim-a-fim bidirecional (ida e volta)	243
Variação do atraso – Jitter	244
Perda de pacotes em um sentido	244
Vazão (largura de banda alcançável)	245
Largura de banda de contenção	246
Métricas de medições passivas	247
Atraso em um sentido (One-Way Delay – OWD)	249
Largura de banda usada	249
Fluxos	250
Disponibilidade	251
Ferramentas de diagnóstico/monitoração da rede	251
perfSONAR	261
General Framework Design (GFD)	262

## **9. Montagem em laboratório de solução de gerência**

Introdução	269
Implementando Gerenciamento	270
Avaliação de Plataformas de Gerenciamento	271

Trouble Ticket Systemas (TTS) 272

Funções e características de um Sistema TTS 273

Fan (Fully Automated Nagios) 274

Nagios 275

Objetos disponíveis no Nagios e seus relacionamentos 276

Plugins do Nagios 276

Como monitorar roteadores e switches utilizando os plugins do Nagios 277

Centreon 280

Nareto e NagVis 280

Primeiras configurações 281

Integração Nagios com OpenLdap 281

Integração Nagios com Splunk 282

Integração Nagios com o Cacti 283

Integração Nagios com Puppet 283

Integração Nagios com TTS 284

## **10. Tópicos avançados em gerenciamento**

A Internet das Coisas 287

Gerenciando a Internet do Futuro 294

Novas abordagens em gerenciamento 296

Sistemas especialistas 297

DISMAN – Distributed Management 297

Gerenciamento web-based 297

DMTF – Distributed Management Task Force 298

Gerenciamento de capacidade 299

Bibliografia 301

# **Escola Superior de Redes**

A Escola Superior de Redes (ESR) é a unidade da Rede Nacional de Ensino e Pesquisa (RNP) responsável pela disseminação do conhecimento em Tecnologias da Informação e Comunicação (TIC). A ESR nasce com a proposta de ser a formadora e disseminadora de competências em TIC para o corpo técnico-administrativo das universidades federais, escolas técnicas e unidades federais de pesquisa. Sua missão fundamental é realizar a capacitação técnica do corpo funcional das organizações usuárias da RNP, para o exercício de competências aplicáveis ao uso eficaz e eficiente das TIC.

A ESR oferece dezenas de cursos distribuídos nas áreas temáticas: Administração e Projeto de Redes, Administração de Sistemas, Segurança, Mídias de Suporte à Colaboração Digital e Governança de TI.

A ESR também participa de diversos projetos de interesse público, como a elaboração e execução de planos de capacitação para formação de multiplicadores para projetos educacionais como: formação no uso da conferência web para a Universidade Aberta do Brasil (UAB), formação do suporte técnico de laboratórios do Proinfo e criação de um conjunto de cartilhas sobre redes sem fio para o programa Um Computador por Aluno (UCA).

## **A metodologia da ESR**

A filosofia pedagógica e a metodologia que orientam os cursos da ESR são baseadas na aprendizagem como construção do conhecimento por meio da resolução de problemas típicos da realidade do profissional em formação. Os resultados obtidos nos cursos de natureza teórico-prática são otimizados, pois o instrutor, auxiliado pelo material didático, atua não apenas como expositor de conceitos e informações, mas principalmente como orientador do aluno na execução de atividades contextualizadas nas situações do cotidiano profissional.

A aprendizagem é entendida como a resposta do aluno ao desafio de situações-problema semelhantes às encontradas na prática profissional, que são superadas por meio de análise, síntese, julgamento, pensamento crítico e construção de hipóteses para a resolução do problema, em abordagem orientada ao desenvolvimento de competências.

Dessa forma, o instrutor tem participaçãoativa e dialógica como orientador do aluno para as atividades em laboratório. Até mesmo a apresentação da teoria no início da sessão de aprendizagem não é considerada uma simples exposição de conceitos e informações. O instrutor busca incentivar a participação dos alunos continuamente.

As sessões de aprendizagem onde se dão a apresentação dos conteúdos e a realização das atividades práticas têm formato presencial e essencialmente prático, utilizando técnicas de estudo dirigido individual, trabalho em equipe e práticas orientadas para o contexto de atuação do futuro especialista que se pretende formar.

As sessões de aprendizagem desenvolvem-se em três etapas, com predominância de tempo para as atividades práticas, conforme descrição a seguir:

**Primeira etapa:** apresentação da teoria e esclarecimento de dúvidas (de 60 a 90 minutos). O instrutor apresenta, de maneira sintética, os conceitos teóricos correspondentes ao tema da sessão de aprendizagem, com auxílio de slides em formato PowerPoint. O instrutor levanta questões sobre o conteúdo dos slides em vez de apenas apresentá-los, convidando a turma à reflexão e participação. Isso evita que as apresentações sejam monótonas e que o aluno se coloque em posição de passividade, o que reduziria a aprendizagem.

**Segunda etapa:** atividades práticas de aprendizagem (de 120 a 150 minutos).

Esta etapa é a essência dos cursos da ESR. A maioria das atividades dos cursos é assíncrona e realizada em duplas de alunos, que acompanham o ritmo do roteiro de atividades proposto no livro de apoio. Instrutor e monitor circulam entre as duplas para solucionar dúvidas e oferecer explicações complementares.

**Terceira etapa:** discussão das atividades realizadas (30 minutos).

O instrutor comenta cada atividade, apresentando uma das soluções possíveis para resolvê-la, devendo ater-se às aquelas que geram maior dificuldade e polêmica. Os alunos são convidados a comentar as soluções encontradas e o instrutor retoma tópicos que tenham gerado dúvidas, estimulando a participação dos alunos. O instrutor sempre estimula os alunos a encontrarem soluções alternativas às sugeridas por ele e pelos colegas e, caso existam, a comentá-las.

## Sobre o curso

A Gerência de Redes de Computadores é uma função crítica em redes de grande porte, uma vez que a complexidade de equipamentos, protocolos e aplicações torna necessária a utilização de ferramentas de software sofisticadas, capazes de diagnosticar em tempo real potenciais problemas de tráfego, falhas de equipamentos e, principalmente, redução na qualidade do serviço.

Esse curso apresenta os conceitos básicos de gerenciamento, as arquiteturas de software envolvidas, aplicações e ferramentas de diagnóstico.

## A quem se destina

A profissionais de TI envolvidos na gerência de redes e controle de qualidade, especialistas em redes de computadores e demais profissionais de TI interessados na operação e controle de redes de computadores.

## Convenções utilizadas neste livro

As seguintes convenções tipográficas são usadas neste livro:

*Itálico*

Indica nomes de arquivos e referências bibliográficas relacionadas ao longo do texto.

## Largura constante

Indica comandos e suas opções, variáveis e atributos, conteúdo de arquivos e resultado da saída de comandos. Comandos que serão digitados pelo usuário são grifados em negrito e possuem o prefixo do ambiente em uso (no Linux é normalmente # ou \$, enquanto no Windows é C:\).

### Conteúdo de slide

Indica o conteúdo dos slides referentes ao curso apresentados em sala de aula.

### Símbolo

Indica referência complementar disponível em site ou página na internet.

### Símbolo

Indica um documento como referência complementar.

### Símbolo

Indica um vídeo como referência complementar.

### Símbolo

Indica um arquivo de áudio como referência complementar.

### Símbolo

Indica um aviso ou precaução a ser considerada.

### Símbolo

Indica questionamentos que estimulam a reflexão ou apresenta conteúdo de apoio ao entendimento do tema em questão.

### Símbolo

Indica notas e informações complementares como dicas, sugestões de leitura adicional ou mesmo uma observação.

## Permissões de uso

Todos os direitos reservados à RNP.

Agradecemos sempre citar esta fonte quando incluir parte deste livro em outra obra.

Exemplo de citação: TORRES, Pedro et al. *Administração de Sistemas Linux: Redes e Segurança*.

Rio de Janeiro: Escola Superior de Redes, RNP, 2013.

## Comentários e perguntas

Para enviar comentários e perguntas sobre esta publicação:

Escola Superior de Redes RNP

Endereço: Av. Lauro Müller 116 sala 1103 – Botafogo

Rio de Janeiro – RJ – 22290-906

E-mail: [info@esr.rnp.br](mailto:info@esr.rnp.br)

## Sobre o autor

**Mauro Tapajós Santos** possui Graduação e Mestrado em Engenharia Elétrica pela Universidade de Brasília. Atualmente é servidor Analista de Planejamento e Orçamento na área de TI da Secretaria de Orçamento Federal do Ministério do Planejamento, Orçamento e Gestão. Já foi coordenador de operação na Infovia da CNI (Confederação Nacional da Indústria) pela Impsat (hoje Global Crossing). Foi analista de TI em empresas do mercado de Telecom e no Ministério Público da União e professor universitário por quase 10 anos. Tem experiência na área de TI em praticamente todas suas vertentes tais como: redes, desenvolvimento, infraestrutura e governança. Atualmente também tem conhecimentos sobre Orçamento Público.

**Liane Tarouco** possui Graduação em Licenciatura em Física pela Universidade Federal do Rio Grande do Sul (1970), mestrado em Ciências da Computação pela Universidade Federal do Rio Grande do Sul (1976) e doutorado em Engenharia Elétrica/Sistema Digitais pela Universidade de São Paulo (1990). Atualmente é professora titular da Universidade Federal do Rio Grande do Sul. Desenvolve atividade docente e de pesquisa na área de Ciência da Computação, com ênfase em Redes de Computadores e em Gerência de Rede. Atua também como pesquisadora e docente junto ao Programa de Pós-Graduação em Informática na Educação.

**Leandro Bertholdo** possui graduação em Bacharelado Em Informática pela Pontifícia Universidade Católica do Rio Grande do Sul (1993) e mestrado em Ciências da Computação pela Universidade Federal do Rio Grande do Sul (1996). Atualmente é professor da Universidade Federal do Rio Grande do Sul e coordenador técnico do POP da RNP no Rio Grande do Sul (2000). Tem experiência na área de Ciência da Computação, com ênfase em Teleinformática, atuando principalmente nos seguintes temas: redes de computadores, ipv6, gerência de rede, gerência de segurança e segurança de redes.

**Francisco Marcelo Marques de Lima** é Mestre em Engenharia Elétrica pela Universidade de Brasília (2009), Mestre em Liderança pela Universidade de Santo Amaro (2007) e pós-graduado em Segurança de Redes de Computadores pela Universidade Católica de Brasília (2003). Atualmente exerce as funções de Coordenador dos Cursos de Redes de Computadores e Segurança da Informação do IESB e Analista em TI do MPOG cedido para a CGU/PR, além de atuar como instrutor/revisor dos cursos de segurança e redes na RNP e instrutor dos cursos de Planejamento Estratégico (PDTI) e Gestão de Contratos de TI (GCTI) na ENAP. Possui mais de 15 anos de experiência na área de Ciência da Computação com ênfase em Segurança da Informação, Redes e Construção de Software tendo exercido funções como: Coordenador Geral de TI do INCRA (DAS 4).

**Vanner Vasconcellos** trabalha no Ponto de Presença no Estado do Pará (PoP-PA) da Rede Nacional de Ensino e Pesquisa (RNP) como Coordenador Técnico e também é professor do Curso de Especialização em Suporte a Rede de Computadores e Internet da Universidade Federal do Pará (UFPA). É graduado em Tecnologia de Processamento de Dados pelo Centro de Universitário do Pará (CESUPA), especialista em Redes de Computadores pela UFPA. Com mais de 20 anos de experiência em TI, atua na RNP há 17 anos na área de gerência de infraestrutura de redes e datacenter, tendo participado de projetos pioneiros de conectividade e Internet no Pará, Amapá e Maranhão.

# 1

## Introdução à gerência de redes

objetivos

Conhecer a forma de estruturação; Entender os objetos gerenciáveis; Compreender a SMI – Estrutura da Informação de Gerenciamento; Entender como funciona o Management Information Base (MIB); Conhecer ASN.1 e BER.

conceitos

Necessidades de gerenciamento; Áreas de gerenciamento; Arquiteturas de gerenciamento.

### Importância do gerenciamento

- Crescimento vertiginoso das LANs e WANs.
- Mais aplicações e usuários gerando complexidade: equipamentos, hardware e software heterogêneos; empresas dependentes das redes para elevar eficiência e lucros.
- Necessidade de gerenciamento visando a coordenação (controle de atividades e monitoração do uso) de recursos materiais (modems, roteadores, enlaces físicos etc.) e lógicos (protocolos, configurações etc.), fisicamente distribuídos na rede, assegurando, na medida do possível, confiabilidade e performance aceitáveis e segurança das informações.

O contínuo crescimento em número e diversidade dos componentes das redes de computadores têm tornado a atividade de gerenciamento de rede cada vez mais complexa. Isso se agrava quando estão envolvidos muitos fornecedores. O impacto de uma falha na rede produz uma perda na receita ou impossibilidade de prestação de serviços que causa transtornos financeiros, institucionais e, por vezes, com consequências bastante sérias e implicações até mesmo de cunho legal.

O isolamento e o teste dos problemas das redes tornaram-se mais difíceis devido à diversidade dos níveis do pessoal envolvido e variação nas formas de monitoração usadas pelos fornecedores dos equipamentos.

Uma pesquisa realizada por Blum (2008) apontou as barreiras para o aprimoramento dos centros de gerência e operação de rede (Network Operation Center – NOC). A tabela seguinte indica o resultado de uma pesquisa realizada por Blum em relação às principais barreiras para as metas de qualidade no gerenciamento de rede.

A tabela seguinte indica o obstáculo percebido e a percentagem de instituições que participaram da pesquisa e indicaram aquele fator. Muitas instituições indicaram mais de um obstáculo e por isso a soma dos percentuais é maior do que 100%.

Barreira ao aprimoramento Capacidade do NOC	%
Falta de equipe especializada	56
Justificar custo/benefício para a administração	47
Custo dos produtos muito alto	45
Falta de produtos ou ferramentas	40
Dificuldade de implementar produtos ou ferramentas	40
Treinamento disponível inadequado	37
Produtos e ferramentas disponíveis não atendem requisitos	33
Falta de métricas	33
Falta de Acordos de Nível e Serviço (SLA – Service Level Agreement)	30
Rotatividade excessiva do pessoal	28

**Tabela 1.1**  
Pesquisa realizada  
por Blum (2008).

Segundo essa pesquisa, os principais serviços oferecidos pelo centro de gerenciamento de rede e de TI incluíam a administração de servidores, redes locais, redes virtuais, segurança, equipamentos dos clientes, conexões de longa distância, aplicações, VOIP e até serviços convencionais de telefonia. Gerenciamento da rede local era o mais frequente, seguido das VPNs e dos servidores. VOIP foi o serviço que estava crescendo mais intensamente.

A preocupação com a qualificação da equipe para cuidar de todas essas questões era um dos pontos destacados e isso deriva do fato de que novas tecnologias surgem constantemente requerendo um contínuo esforço da equipe para implementar e administrar os novos serviços e tecnologia que são cada vez mais complexas conforme salientado por Farrel (2009).



Em vista desses fatores, o desafio de manter custos sob controle, treinar novamente a equipe, reter a equipe e recrutar novos profissionais é constante.

## Atuação do gerente de redes

A atividade de gerenciamento de rede é usualmente exercida por um grupo de pessoas que inclui, no mínimo, um operador de rede e um administrador de rede.

Normalmente, o operador controlador da rede realiza a contínua monitoração da rede a partir de uma estação onde são exibidos os dados sobre a situação da rede e de seus componentes.



**Figura 1.1**  
Exemplo do gerenciamento de rede.

- **Coleta de dados:** processo automático (ou não) de monitoração dos recursos gerenciados;
- **Diagnóstico:** tratamento e análise realizados a partir dos dados coletados (determinação da causa);
- **Ação:** reação (ou não) ao problema.

Gerenciamento significa ter o controle e poder agir em função de informações coletadas que mostram situações determinadas. Por exemplo: um link de dados pode apresentar atraso; uma ação possível seria rotear novamente o tráfego para outro link.

A gerência de redes de computadores é uma das áreas mais complexas quando falamos em gerência de TI, pois os contratos de prestação de serviços em TI estão cada vez mais criteriosos, envolvendo compromissos com garantias de qualidade. Portanto, é preciso poder identificar todos os ativos de rede, monitorá-los, coletando informações relevantes que serão exibidas em gráficos e poderão ser analisadas na medida em que vão sendo recebidas e com vistas a determinar anomalias de comportamento e acionar o atendimento, quando necessário. Esses procedimentos visam facilitar a identificação e solução dos problemas de maneira rápida e eficaz.

## Atividades comuns de gerenciamento

- Monitoramento.
- Controle.

Para o gerenciamento de redes de computadores, uma alternativa é usar a própria infraestrutura existente para alcançar os elementos ou pontos definidos da rede, na busca por informações e no disparo de ações sobre esses equipamentos (testes ou comandos de reconfiguração). Outra opção seria montar uma rede paralela à rede existente, que tivesse intersecções nos pontos de interesse.

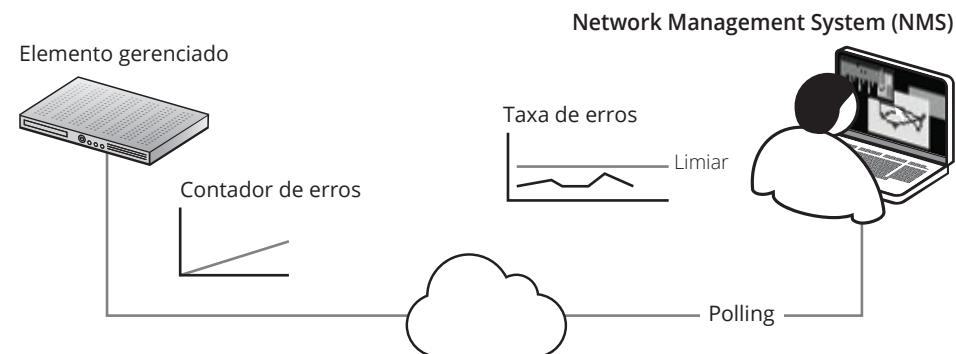
A situação atual das redes é que os equipamentos nelas usados têm condições para prover uma grande quantidade de informações sobre seu próprio funcionamento, bem como de outros dispositivos não gerenciáveis e que estão também conectados (volume de tráfego, erros etc.).

Adicionalmente, os componentes de software também têm sido estruturados de modo a usar a mesma estratégia, e são atualmente desenvolvidos para proporcionar grande quantidade de informações sobre seu próprio funcionamento (volume de transações atendidas, desempenho etc.). Em função disso, a quantidade e diversidade de informações sobre o funcionamento das redes, passíveis de obtenção da rede, de seus componentes e serviços, cresce continuamente. Essa situação tende a ocasionar mais dificuldades na determinação de problemas.

Por isso, além da necessidade de monitorar a rede, existe a necessidade de reconhecer, manipular e tratar toda essa informação obtida de forma a permitir diagnósticos mais precisos das causas dos problemas e que levem a soluções mais rapidamente. A atividade de gerenciamento da rede envolve uma gama de atividades mutuamente interdependentes, tais como:

- Registrar a ocorrência de eventos;
- Estabelecer critérios para o disparo de alarmes;
- Detectar e diagnosticar a ocorrência de falhas;
- Conhecer e controlar alterações nos equipamentos;
- Acompanhar o desempenho da rede e dos serviços de rede;
- Garantir a segurança;
- Contabilizar recursos;
- Verificar Acordo de Nível de Serviço (ANS) estipulado no contrato.

A figura 1.2 ilustra uma situação típica nesse contexto. Os elementos gerenciados (roteadores, switches, servidores etc.) continuamente registram dados sobre seu próprio funcionamento.



**Figura 1.2**  
Visão geral do gerenciamento de rede.

Esses dados podem ser enviados sob demanda (polling) ou, periodicamente, para uma estação de gerência onde um sistema de gerenciamento de rede (NMS – Network Management System) coleta, soma, compara com limiares pré-estabelecidos informações sobre indicadores de desempenho, tais como quantidade de erros por unidade de tempo, volume de tráfego etc.).

Os dados tratados são apresentados em gráficos para a equipe de gerência de rede, mas podem também gerar mensagens de alerta, em caso de problemas. Essas mensagens podem ser encaminhadas de diversas formas (e-mail, torpedos etc.) para os operadores humanos responsáveis por fazer o diagnóstico e restaurar os padrões de operação da rede a valores aceitáveis.

## O que gerenciar?

- Equipamentos de rede.
- Aplicações e serviços de rede.
- Banco de dados.
- Dispositivos de armazenamento.

Aplicações de gerenciamento são aplicações que tratam de dados derivados do funcionamento da rede e de seus serviços.

No âmbito de redes, várias são as possibilidades de equipamentos ou itens passíveis de monitoramento e controle. Em um nível mais baixo, há a figura do elemento de rede (equipamento). Já em nível mais elevado há a consideração das redes como um todo (como serviços). Praticamente qualquer coisa na rede pode ser gerenciada para permitir a exata noção da realidade da rede em questão.

A forma como as informações de gerenciamento da rede fluem é definida por padrões de gerência de rede que especificam a forma de representar os dados sobre a rede e a forma de comunicar esses dados para a aplicação de gerenciamento.

Hoje em dia, todos os equipamentos que compõem uma rede de comunicação de dados, mesmo que sejam de fabricantes diferentes, possuem compatibilidade com os protocolos de gerenciamento de redes de computadores padronizados, possibilitando assim a interoperabilidade entre redes híbridas.

## Áreas funcionais de gerenciamento (modelo OSI)

O modelo Open System Interconnection (OSI), definido pela International Standard Organization (ISO), foi um primeiro esforço para produzir uma forma padronizada de interoperação entre sistemas. Na arquitetura de rede preconizada pela ISO, as funções necessárias para a interconexão entre sistemas foram organizadas sete camadas, tal como ilustrado na figura 1.3.

Cada camada seria responsável por uma parte da tarefa. A camada inferior, denominada de Camada do Nível Físico, cuida do envio e recebimento dos dados binários (bits) através do interface físico. Essa camada agrupa os bits recebidos em quadros (ou frames), passando para a camada de enlace, que vai analisar os dados que compõem a parte do cabeçalho e as demais informações contidas no quadro. Respostas podem ser geradas diretamente pela camada de enlace quando necessário (confirmações de recebimento correto ou pedidos de retransmissão dos dados, por exemplo).



**Figura 1.3**  
Modelo OSI.

Se o quadro contém dados que devem ser tratados pelas camadas superiores, esses são passados para a camada imediatamente superior. Esse procedimento continua até os dados chegarem à camada de aplicação, quando são então passados para a aplicação apropriada.

A função de comunicação das informações de gerenciamento é proposta para ser implementada na camada de aplicação. Assim, o protocolo de gerenciamento é um protocolo entre aplicações.

Na arquitetura usada na internet, houve uma simplificação desse modelo em camadas, que passou a contemplar apenas as camadas física, de enlace, de rede, de transporte e de aplicação. Mas a função de gerenciamento também é suposta ser realizada na camada de aplicação.

Nas duas arquiteturas, a função de gerenciamento de redes foi organizada em cinco categorias, algumas vezes referidas pela sigla FCAPS (Fault, Configuration, Accounting, Performance, Security). Assim, o gerenciamento de redes costuma ser estruturado nestas cinco áreas:

- Gerenciamento de falhas;
- Gerenciamento de desempenho;
- Gerenciamento de configuração;
- Gerenciamento de segurança;
- Gerenciamento de contabilização.



Esas áreas, definidas pelo modelo OSI de gerenciamento, mas também usada no modelo internet de gerenciamento, são uma referência às atividades e focos possíveis dentro de uma solução de gerenciamento.

Assim, pode existir mais de uma equipe encarregada de gerenciar diferentes áreas. Por exemplo, uma equipe pode estar encarregada de gerenciar configuração e problemas, enquanto outra pode estar encarregada da segurança da rede e concentra sua coleta de informações nesse sentido.

Itens como número de acessos, tentativas de login e serviços de rede acessados são informações úteis caso o objetivo seja o gerenciamento de segurança. Outros itens, como número de quedas de um link ou taxa de erros em uma LAN, não são tão importantes para a gerência de segurança como seriam para o gerenciamento de desempenho.

## Gerenciamento de falhas

O gerenciamento de falhas é o conjunto de facilidades que habilita a detecção, isolamento e correção de condições anormais de operação do ambiente. Tais condições podem ser persistentes ou transitórias, e os problemas são manifestados como erros. O gerenciamento de falhas demanda facilidades para: manter e examinar logs de erros, receber e registrar notificação de erros e atuar em função destas, rastrear problemas, executar sequências de testes de diagnóstico e corrigir problemas com vistas a assegurar a operação contínua da rede. As principais funções dessa categoria de gerenciamento da rede são:

- Detecta, isola e registra o problema;
- Registra as ocorrências;
- Executa testes de diagnóstico;
- Realiza a investigação do ocorrido;
- Comportamento proativo (preferível) ou reativo.



O gerenciamento de falhas envolve usualmente as seguintes etapas:

- Detecção dos sintomas do problema;
- Isolamento do problema;
- Reparo do problema automaticamente (se possível) ou manualmente;
- Teste do reparo em todos os subsistemas importantes;
- Registro da detecção e do modo como o problema foi resolvido.



## Gerenciamento de desempenho

A preocupação com o desempenho da rede deve ser constante, pois à medida que o padrão de uso se altera, mudam as demandas sobre os canais de comunicação, sobre os elementos ativos da rede (roteadores, switches) e sobre os equipamentos envolvidos na prestação dos serviços. Quando um usuário percebe uma lentidão na resposta a suas demandas, não tem como saber qual é o recurso responsável pelo baixo desempenho; ele apenas diz que a rede está lenta e leva essa reclamação até a central de atendimento. Mas a equipe de gerenciamento de rede pode e deve ter condições para não apenas identificar a causa de gargalos, mas também antecipar tais situações a partir da observação sistemática da tendência de crescimento da demanda.

Naturalmente, esse acompanhamento necessita de ferramentas que registrem os dados relativos à demanda e sejam capazes de apresentar o resultado da sua análise em forma de médias, gráficos etc. Nesse sentido, a gerência de desempenho tem um conjunto de incumbências que envolve:

- Controlar o “comportamento” dos recursos de rede;
- Avaliar as atividades de comunicação oferecidas na rede;
- Monitorar a operação diária da rede;
- Localizar pontos críticos no sistema (gargalos ou pontos de falha que inibiriam parte substancial do uso da rede);
- Registrar dados de operação;
- Auxiliar funções análise da demanda de planejamento de capacidade.



## Gerenciamento de configuração

Gerenciamento de configuração é o conjunto de facilidades que exerce o controle sobre o que está na rede: identifica, coleta e provê informações sobre o status dos recursos da rede de forma que torne possível a qualquer instante conhecer o que está operacional, em uso ou simplesmente disponível para utilização em caso de emergência. Essa categoria de funções têm as seguintes características:

- Considera a rede como um sistema dinâmico em permanente transformação;
- Provê manutenção da estrutura física e lógica da rede;
- Acompanha o ciclo de vida do componente e sua configuração;
- Identifica componentes em um nível apropriado e suas relações (topologia);
- Monitora cada componente, documentando as trocas que devem manter os requisitos básicos estabelecidos;
- Mantém registros dos status de cada componente (topologia e dispositivos);
- Executa alterações na configuração do sistema, visando isolar falhas, aliviar situações críticas ou atender as necessidades dos usuários.



## Gerenciamento de segurança

O gerenciamento de segurança relaciona-se com os aspectos de acompanhamento e controle do uso da rede para assegurar seu uso apropriado. Isso implica no uso de mecanismos de autenticação, controlar autorizações e permissões de acesso associadas, gerenciar chaves de segurança e uso apropriado da rede. Essa categoria de funções é a que:

- ▣ Cuida dos mecanismos e procedimentos de proteção;
- ▣ Cria, apaga e controla os serviços e mecanismos de segurança;
- ▣ Trata da distribuição da informação relacionada com segurança e seus eventos;
- ▣ Mantém registros de eventos relativos à segurança;
- ▣ Suporta e garante a política de segurança adotada.



### Gerenciamento de contabilização

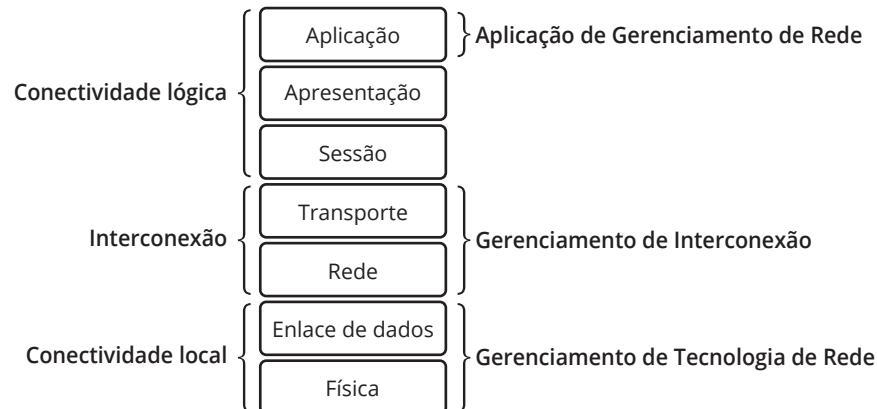
O gerenciamento de contabilização é o conjunto de facilidades que permite o controle e eventualmente a tarifação pelo uso dos recursos. Isso implica em informar os usuários sobre os custos incorridos ou recursos consumidos, estabelecer limites para o uso dos recursos, combinar os custos quando são usados recursos variados, tal como por exemplo meios de comunicação. Nesse sentido, essa categoria de funções de gerenciamento de rede envolve atribuições tais como:

- ▣ Controla recursos;
- ▣ Permite que tarifas sejam aplicadas aos recursos de rede (discos compartilhados, banda, arquivamento remoto, serviços de telecomunicações, e-mails etc.);
- ▣ Viabiliza a identificação de custos para a rede e seus recursos;
- ▣ Mantém limites de consumo;
- ▣ Efetua a melhor distribuição de recursos e alimenta trabalhos de planejamento.



## Gerenciamento dentro do modelo OSI

Uma aplicação de gerenciamento vai atingir os elementos de rede e suas entidades na maioria das vezes através da própria rede. Isso cria uma preocupação com o quanto a solução de gerenciamento pesará sobre a própria rede que se quer gerenciar.



**Figura 1.4**  
Gerenciamento  
dentro do  
Modelo OSI.

A aplicação de gerenciamento poderá visualizar níveis de conectividade e trabalhar em qualquer um deles:

- ▣ Nível de conectividade local (tecnologia de rede);
- ▣ Nível de interconexão;
- ▣ Nível de conectividade lógica (aplicação).

## Conceitos

- Informação de gerência.
- Protocolos de gerência.
- Arquitetura de gerenciamento:
  - Fornece a estrutura geral do sistema de gerência.
  - Descreve componentes dentro do sistema e suas funções.
  - Mostra os relacionamentos entre os componentes.

Uma aplicação de gerência nada mais é do que uma aplicação que trata de dados. Esses dados deverão obedecer a determinada estrutura para que seja possível criar soluções de gerenciamento que atuem sobre os mais diversos ambientes de rede.

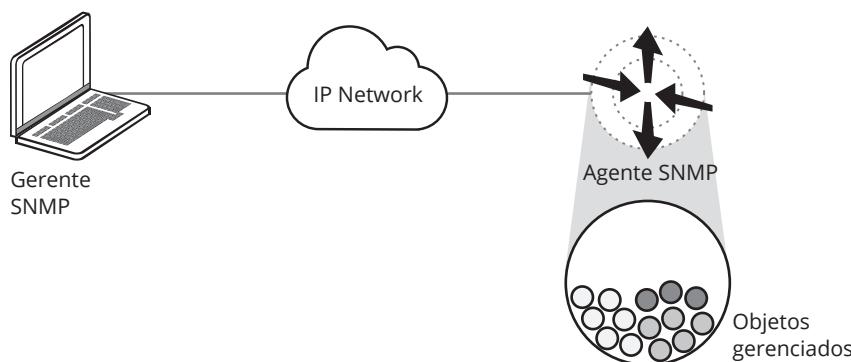
Um protocolo de gerência é normalmente um protocolo de nível de aplicação (possui especificidade inerente ao gerenciamento de rede). Ele visa atender as demandas de comunicação entre as entidades definidas na arquitetura de gerenciamento.

As duas arquiteturas clássicas de gerenciamento de rede são:

- Arquitetura OSI de gerenciamento, derivada dos padrões de sistemas abertos Open Systems Interconnection (OSI), que é mais antiga e foi prevista para ser usada em um contexto de rede onde os sistemas interoperam usando os protocolos padronizados para cada uma das camadas do modelo OSI. Nesse caso, o protocolo de gerenciamento de rede usado é o Common Management Information Protocol (CMIP).
- Arquitetura baseada no modelo Simple Network Management Protocol (SNMP), derivada dos padrões internet/TCP-IP, que é atualmente a mais disseminada. Embora tenha sido derivada da arquitetura de gerenciamento OSI, foi simplificada, como destaca o próprio nome do protocolo.

## Modelo de comunicação

- Preocupa-se em definir como os dados de gerenciamento são comunicados entre os processos do gerente e do agente.
- Modelo usado no protocolo SNMP envolve o uso das entidades gerente e agente.



**Figura 1.5**  
Modelo de  
comunicação.

Nesse modelo, a comunicação é estabelecida entre o gerente, onde é executado o Network Management System (NMS), que é a aplicação de gerenciamento de rede executada em uma estação de gerência, e um agente que é um módulo sendo executado no dispositivo de rede (roteador, switch, computador de mesa, servidor etc.). A responsabilidade do agente

é manter e atualizar dados sobre o funcionamento do dispositivo. Esses dados são armazenados sob a forma de objetos gerenciados e consistem em representações de valores de contadores, status, limiares etc. Os objetos gerenciados são estruturados em uma Management Information Base (MIB).

Um agente pode conter um grande número de objetos gerenciados e é através deles que o NMS vai poder descobrir o que está acontecendo em cada dispositivo gerenciado na rede. Em alguns casos, os agentes também podem incluir dados sobre outros dispositivos, a partir da observação direta ou indireta de seu comportamento e transmissões.

## Modelo funcional

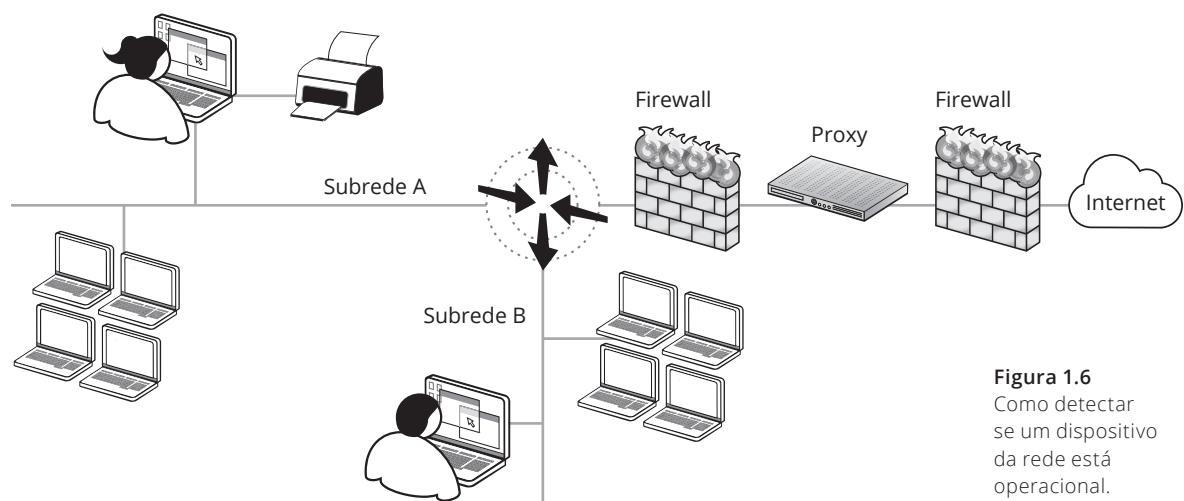
O modelo funcional de gerenciamento de rede é definido em função das cinco áreas de interesse, propostas pela ISO, modelo FCAPS (Fault, Configuration, Accounting, Performance, Security), que indicam a direção do esforço dos que desenvolvem e empregam soluções para gerenciar redes e sistemas. Assim, os produtos e serviços de gerenciamento de rede são organizados com base nesse modelo funcional, e as aplicações de gerenciamento de redes também costumam ser organizadas com base nessas cinco áreas funcionais:

- Falhas;
- Configuração;
- Contabilização;
- Desempenho;
- Segurança.



Para apoiar essas áreas são utilizadas diversas ferramentas de gerenciamento. Algumas são mais básicas, acionadas mediante o uso de uma simples interface de linha de comando. Outras são mais complexas, sendo invocadas através do uso de programas com interfaces gráficas e que por vezes envolvem um trabalho prévio de configuração.

Como um exemplo, temos o caso da figura 1.6. Supondo que o usuário da subrede B não consiga acessar a internet, quais providências para diagnóstico inicial da situação poderiam ser desencadeadas?



**Figura 1.6**  
Como detectar  
se um dispositivo  
da rede está  
operacional.

Buscando solucionar esse problema usando como base as cinco áreas funcionais, vários procedimentos poderiam ser desencadeados e envolver o uso de ferramentas de apoio para diagnosticar e solucionar o problema.

A primeira categoria, o gerenciamento de falhas ou problemas, poderia ser executada de acordo com as cinco etapas do gerenciamento de falhas anteriormente apresentadas.

## Detecção dos sintomas do problema

Nessa fase, seria necessário investigar se a percepção do usuário de que não conseguia acessar algum serviço remota era derivada de um problema remoto ou local e nesse segundo caso se o problema era sentido também nas demais máquinas e subredes. Umas das abordagens mais comuns utilizadas tanto por técnicos da rede como pelos próprios usuários envolve o uso de comandos básicos dos sistemas, como *ping* e *traceroute*.

Assim, inicialmente, poderia ser invocado o comando *ping*, através da janela de comando da estação de trabalho do usuário na subrede B, para um primeiro teste. Ping é um utilitário que usa o protocolo ICMP para testar a conectividade entre equipamentos. Seu funcionamento consiste no envio de pacotes para o equipamento de destino e na “escuta” das respostas. Se o equipamento de destino estiver ativo, uma “resposta” é devolvida ao computador solicitante na qual é indicado o tempo consumido tal como no exemplo seguinte.

Exemplo de uso do ping para testar conectividade com um roteador em uma rede interna, que poderia ser o roteador mostrado na figura 1.5 e que interliga as subredes A e B com a conexão com a internet, que passa pelo firewall. Esse roteador poderia estar usando um endereçamento IP privativo e o comando *ping* poderia ser direcionado inicialmente para esse endereço.

```
C:\>ping 192.168.1.1  
Disparando 192.168.1.1 com 32 bytes de dados:  
Resposta de 192.168.1.1: bytes=32 tempo=3ms TTL=64  
Resposta de 192.168.1.1: bytes=32 tempo=1ms TTL=64  
Resposta de 192.168.1.1: bytes=32 tempo=2ms TTL=64  
Resposta de 192.168.1.1: bytes=32 tempo=3ms TTL=64  
Estatísticas do Ping para 192.168.1.1:  
Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
```

Aproximar um número redondo de vezes em milissegundos:

Mínimo = 1ms, Máximo = 3ms, Média = 2ms

Nesse exemplo, as mensagens ICMP enviadas usaram o tamanho padrão de 32 bytes. O TTL de 64 indica que esse era o número máximo de nós atravessados na rede pelo pacote para tentar chegar ao endereço IP 192.168.1.1. Isso evita que um pacote gerado pelo ping fique circulando na rede indefinidamente em caso de problema de roteamento.

Se esse procedimento não tivesse conseguido obter resposta do equipamento ao qual o ping foi dirigido, a mensagem recebida seria algo tal como:

### Saiba mais

O tempo, em milissegundos, no exemplo, mostra que o roteador estava muito próximo do computador que enviou o ping, e a resposta não tardou muito.

```
C:\>ping 198.168.2.1

Disparando 198.168.2.1 com 32 bytes de dados:

Esgotado o tempo limite do pedido.

Estatísticas do Ping para 198.168.2.1:

Pacotes: Enviados = 4, Recebidos = 0, Perdidos = 4 (100% de perda),
```

Nesse caso, poderíamos concluir que o equipamento 198.168.2.1 não existe na rede ou está inoperante. Essa dúvida poderia ser resolvida com um outro comando básico que indicaria o caminho (conjunto de roteadores) até o equipamento destino. Esse teste poderia ser realizado com o comando *traceroute*, que é uma ferramenta de diagnóstico que rastreia a rota de um pacote através de uma rede de computadores que utiliza o protocolo IP. Seu funcionamento está baseado no uso do campo *time-to-live (TTL)* do pacote IPv4, destinado a limitar o tempo de vida dele. Esse valor é decrementado a cada vez que o pacote é encaminhado por um roteador. Ao atingir o valor zero, o pacote é descartado e o originador é alertado por uma mensagem ICMP TIME\_EXCEEDED.

Através da manipulação do campo TTL (que recebe valores que vão sendo incrementados de uma unidade a cada passo), uma série de datagramas UDP dirigidos ao destino final vão sendo gerados com diferentes e crescentes valores de TTL. Assim, é possível receber essa mensagem de TIME\_EXCEEDED de cada um dos roteadores no caminho do pacote e calcular o tempo que foi usado para chegar até cada um dos roteadores intermediários. Existem implementações desse comando para praticamente todos os Sistemas Operacionais, incluindo aqueles usados em roteadores. A maneira de invocar o comando pode variar ligeiramente (tracert em ambiente Windows, traceroute em outros ambientes). Um exemplo de resultado possível de ser obtido com o comando *traceroute* é apresentado a seguir:

```
C:\>tracert www.rnp.br

Rastreando a rota para www.rnp.br [200.130.35.4] com no máximo 30 saltos:

 1  1714 ms      1 ms      2 ms  192.168.1.1
 2  11 ms       16 ms      8 ms  10.79.4.1
 3  12 ms       11 ms     11 ms c915c002.virtua.com.br [201.21.192.2]
 4  11 ms       10 ms     11 ms as1916.rs.ptt.br [200.219.143.3]
 5  19 ms       16 ms     17 ms rs-sc-10g-oi.bkb.rnp.br [200.143.252.58]
 6  28 ms       30 ms     35 ms sc-sp-10g-oi.bkb.rnp.br [200.143.252.65]
 7  71 ms       48 ms     37 ms sp-rj-10g-oi.bkb.rnp.br [200.143.252.70]
 8  56 ms       62 ms     57 ms rj-df-10g-oi.bkb.rnp.br [200.143.252.78]
 9  58 ms       56 ms     88 ms landf-mxdf-10g-int.bkb.rnp.br
[200.143.255.170]
10  63 ms      55 ms     55 ms rt.pop-df.rnp.br [200.130.101.94]
```

```
11    57 ms    58 ms    57 ms  kerberos.na-df.rnp.br [200.130.35.4]
```

Rastreamento concluído.

Essas duas ferramentas poderiam ser usadas não apenas da máquina onde está o usuário que experimenta o problema, mas também de outras máquinas na mesma subrede e em outras subredes. Isso permitiria conhecer a extensão do problema relatado observando a resposta, por ausência de resposta em cada caso. No exemplo anterior, é possível constatar que o acesso desde a máquina onde o tracert foi disparado até o servidor www.rnp.br está operacional. O passo seguinte seria isolar o problema relatado pelo usuário.

Um aspecto a ressaltar sobre o uso dessas ferramentas básicas é que elas geram tráfego na rede, e isso precisa ser minimizado, sob pena de prejudicar o funcionamento da rede com excesso de tráfego de teste. Outro ponto a atentar é que esse tráfego pode ser filtrado por algum roteador ou firewall ao longo da rota, e a ausência de uma resposta pode ser causada por esse processo de filtragem.

### Saiba mais

Observe que o retardo informado é impreciso, pois depende do tráfego encontrado pelos pacotes usados nessas ferramentas ao longo de sua rota, bem como da carga dos roteadores envolvidos no seu encaminhamento e ambos variam ao longo do tempo.

### Isolamento do problema

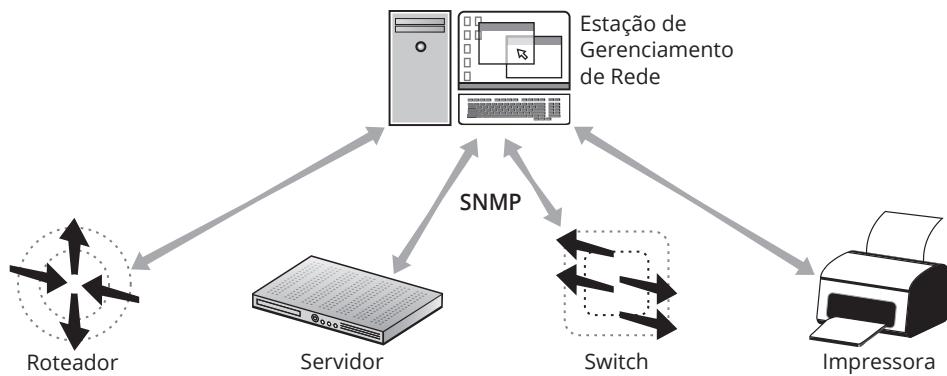
O isolamento do problema demanda alguma forma de inspeção ou teste dos equipamentos envolvidos, desde aquele onde o problema foi constatado, até o destino que seu usuário pretende acessar. Dependendo dos resultados dos testes com ping e traceroute, seria possível verificar se o equipamento do usuário era o único a sentir o problema ou se apenas tinha sido o primeiro a constatar uma falha mais ampla, que inibiria o acesso de todos os equipamentos de uma ou mais subredes ou até mesmo de todas as máquinas da intranet. A causa poderia ser uma falha na conexão de saída daquela rede com a internet ou uma indisponibilidade do servidor remoto. O comando *ping* mostraria simplesmente se o servidor remoto estava acessível. O traceroute mostraria se algum problema e impedimento ou excessivo retardo estaria ocorrendo na rota, a partir de algum ponto. Os tempos de resposta recebidos de cada passo do rastreamento com o traceroute permitiram averiguar se a rede tem comportamento normal, degradado ou sem conexão a partir de algum ponto da rota.

Após isolar a causa do problema identificando o(a) equipamento(s) ou componente(s) de software com comportamento incorreto, o trabalho passaria para a fase de solução do problema.

### Reparo do problema automaticamente (se possível) ou manualmente

O reparo do problema pode às vezes ser conseguido de forma automática (ou quase), com uma reinicialização do equipamento o que pode dirimir algum estado inconsistente do software sendo executado naquela máquina.

Mas em outros casos há que passar ao diagnóstico das causas do mau funcionamento, e isso implica em inspecionar mais detalhadamente aquele equipamento para obter informações sobre os fatores que poderiam afetar seu comportamento. Essas informações podem, em parte, ser obtidas mediante inspeção dos objetos gerenciados e o uso do protocolo SNMP para efetuar o polling (consulta aos dados do agente SNMP) daquela máquina permite obter informações sobre o sistema usado, portas de comunicação, tráfego, erros recebidos, tabelas de roteamento etc. Isso envolve o uso de algum software que pode ser no mínimo um MIB Browser para obter os dados da MIB do agente SNMP no equipamento sendo analisado. Uma plataforma de gerência de rede mais completa, tal como Tivoli ou HP Open View, entre outras, com banco de dados com a definição de todos os equipamentos da rede, sua topologia e características proporciona condições para inspecionar remotamente os componentes da rede, tal como ilustrado na figura 1.7.



**Figura 1.7**  
SNMP para inspecionar equipamentos da rede.

O resultado da inspeção dos equipamentos mediante o uso de SNMP poderia retornar informações tais como:

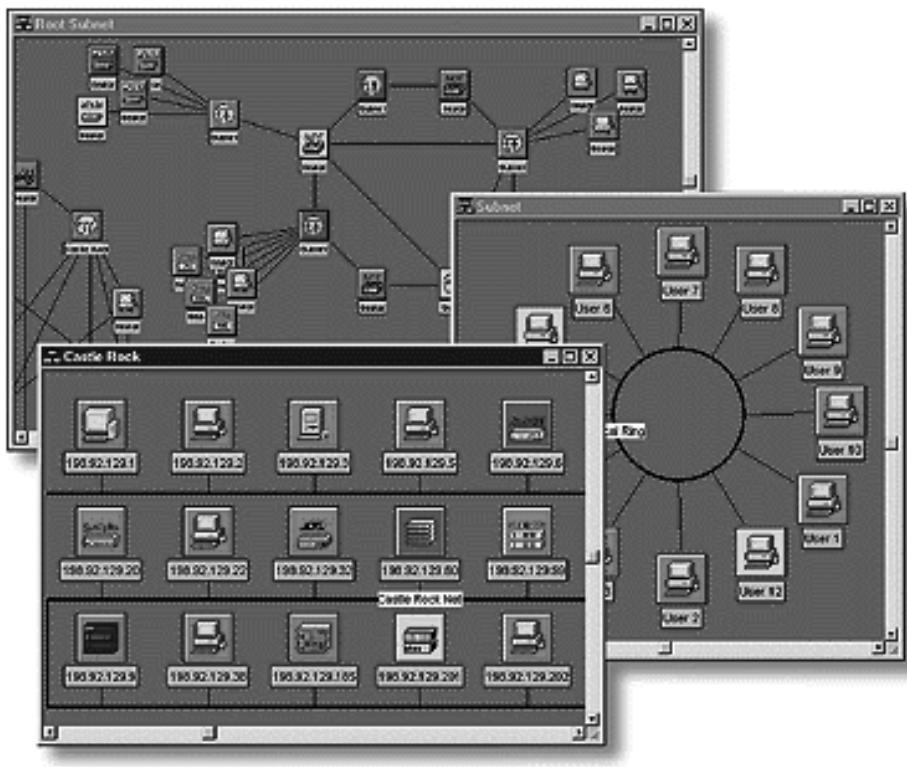
- Versão do Sistema Operacional e do software de rede, serviços providos pelo equipamento;
- Localização do equipamento e pessoa ou setor responsável por este;
- Quantidade e tipo de interfaces de rede bem como sua situação operacional e tráfego recebido e enviado (com taxas de erros);
- Roteador default, rotas;
- Conexões estabelecidas com aquele equipamento e portas (TCP ou UDP) em uso.

O reparo do equipamento com problemas pode ter de ser feito manualmente envolvendo substituição ou alteração do software, ajuste de parâmetros de configuração ou mesmo substituição de hardware (interfaces ou mesmo o equipamento como um todo).

#### Teste do reparo em todos os subsistemas importantes

Quando o processo de reparo é suposto estar finalizado, testes de aceitação precisam ser feitos para avaliar se todos os subsistemas envolvidos e dependentes daquele que experimentou problemas de funcionamento encontram-se operacionais em condições normais. Os comandos de *ping* e *traceroute* podem ser usados para verificar a conectividade. No caso de uso de plataformas de gerenciamento, diagramas da rede com cores associadas ao status dos dispositivos (verde para funcionamento normal, amarelo com erros intermitentes e vermelho para dispositivo com problema, por exemplo) já passam a mostrar a situação dos dispositivos da rede.





**Figura 1.8**  
Dados de status dos dispositivos na estação de gerenciamento da rede.

Todavia, essas indicações podem ser incompletas, evidenciando apenas que existe conectividade a nível de rede. Testes realizados a nível de aplicação também necessitam ser realizados para verificar se os serviços envolvidos foram reiniciados apropriadamente. Em algumas situações, os serviços executados em um servidor não retornam automaticamente após uma reinicialização como era de esperar. Em outras situações, o comportamento anômalo de uma estação pode ter ocasionado uma alteração nas regras do firewall, que passou a inibir alguns tipos de acessos daquela estação.

#### Registro da detecção e do modo como o problema foi resolvido

Problemas tendem a se repetir na rede e a utilização de um “Sistema de Registro de problemas” auxilia a equipe de gerência de rede no diagnóstico do problema com a possibilidade de apresentação de problemas anteriores ocorridos com características similares, mas também agilizam o processo de controle da rede, porque permitem uma comunicação mais direta com os responsáveis pela gerência da rede. O “Registro de Problema” deve prover um histórico completo do problema, de forma que qualquer operador possa tomar alguma iniciativa sem que para isso tenha de consultar outro operador que pode ter atuado naquele caso em um turno anterior.

O sistema de registro de problemas deve permitir um melhor escalonamento de problemas atribuindo prioridades aos mesmos. Os supervisores e operadores poderão tomar decisões acerca da necessidade ou não de mais pessoal ser alocado a um determinado problema, e a prioridade dos registros pode mudar de acordo com a hora do dia ou em resposta a alarmes de tempo. Um “timeout” pode ser associado para cada registro de problema. Caso o problema não seja resolvido em tempo, automaticamente é acionado um alarme.

O sistema de registro de problemas fornece mecanismos para a obtenção de estatísticas, tais como “Tempo médio entre falhas” (Mean Time Between Failures – MTBF) e “Tempo médio de conserto” (Mean Time To Repair – MTTR).

A estrutura de um registro de problema consiste de três partes: cabeçalho, atualizações e dados da resolução. O cabeçalho é responsável pelas informações de abertura do problema e inclui informações como:

- Hora e data do início do problema;
- Operador que está abrindo o registro;
- Severidade do problema;
- Uma linha descrevendo o problema (para uso em relatórios);
- Máquina envolvida;
- Rede envolvida;
- Endereço da máquina envolvida;
- Endereço da máquina destino;
- Próxima ação;
- Hora e data para alarme;
- Para quem esse registro deveria ser enviado;
- Responsável pelo registro;

As informações de atualização representam as ações e diagnósticos realizados ao longo do ciclo de vida do problema.

Os dados de resolução representam as informações que resumem o problema para futuras análises estatísticas e, também, para uso em problemas similares contendo informações tais como: hora e data da resolução, descrição da solução do problema, componentes afetados, quem verificou o problema depois que foi resolvido e quem foi consultado para auxílio na resolução do problema.

O estabelecimento de uma baseline (base de referência) com dados sobre a rede e sua operação típica auxilia a reconhecer e identificar comportamento anômalos além de fornecer a base para modificar as configurações com confiabilidade, pois em caso de erro haverá uma configuração conhecida e que funcionou em algum momento anterior à mudança a para a qual se pode retornar.

Esse conjunto de informações, derivados do processo de resolução de problemas (troubleshooting) na rede, não apenas proporciona um registro sistemático das mudanças na rede, tal como preconizado pelas boas práticas de gerência de configuração, mas pode servir como ajuda no diagnóstico de problemas futuros. Isso é especialmente importante dada a alta rotatividade na equipe de gerência de rede que é problema importante. Organizar essas informações de modo que possam ser consultadas, manualmente ou através de um sistema automatizado (sistema especialista), acelera o processo de diagnóstico, pois os problemas na rede tendem a repetir-se. Aumentar o nível de automação no gerenciamento de rede oferece diversas vantagens, conforme salientado por Comer (2007):

- Redução no tempo requerido para realizar a tarefa de diagnosticar e consertar os problemas na rede;
- Redução na probabilidade de erro de julgamento sobre as causas e formas de correção dos problemas;
- Garantir o uso de estratégias e política de uso consistentes na rede como um todo, evitando soluções específicas definidas localmente que possam não atender ao acordo de qualidade de serviço da organização.



## Gerência de configuração

A configuração dos dispositivos da rede controla o comportamento da rede. A gerência de configuração é o processo de descobrir e configurar os dispositivos críticos que asseguram o funcionamento do backbone e de todos os demais equipamentos ligados. Nesse sentido, a organização dessa função de gerenciamento deve permitir rápido acesso às informações de configuração, que devem ser mantidas continuamente atualizadas. Isso implica em um conjunto de fases ou etapas para o estabelecimento e continuidade do gerenciamento de configuração:

### Reunião de informações de configuração (automática ou manual)

O estabelecimento de uma baseline (base de referência) com dados sobre a rede e sua operação típica auxilia a reconhecer e identificar comportamentos anômalos, além de fornecer a base para modificar as configurações com confiabilidade, pois em caso de erro haverá uma configuração conhecida e que funcionou em algum momento anterior à mudança a para a qual se pode retornar.

### Armazenamento das informações de configuração e atualização dinâmica

O registro automático ou não das alterações na rede permite conhecer e poder emitir relatórios atualizados sobre todos os componentes e serviços da rede. Quando ocorrem alterações na topologia ou nos dispositivos da rede mudanças na configuração precisam ser desencadeadas sob pena de tornar parte da rede inoperante. Em tais casos conhecer as relações de dependência (que dispositivos dependem ou estão interligados, física ou logicamente) uns dos outros é essencial. Como exemplo simples, pode-se citar o caso de um serviço de DNS desativado por problema de hardware ou software e que, mesmo substituído por outro servidor na rede, pode ocasionar impossibilidade de operação de uma parte da rede até que as reconfigurações dos equipamentos que utilizavam aquele servidor como “default” sejam reconfiguradas.

Diversas ferramentas podem ser utilizadas para definir ou alterar a configuração dos dispositivos da rede. A forma mais básica e usual consiste em acessar diretamente o dispositivo mediante o uso de Telnet/SSH no roteador ou na máquina/servidor para fazer login e depois utilizar operações em modo de linha de comando através do Command Line Interface (CLI) daquele equipamento. Esse procedimento é bastante manual e trabalhoso.

Em alguns casos, um sistema centralizado de registro e atualização de configurações pode ser mantido. As alterações são realizadas nas versões ali armazenadas e depois transferidas para os dispositivos da rede mediante o uso de algum protocolo de transferência de arquivos.

### Fornecimento de relatórios periódicos de configuração e de inventário

Mecanismos de automação do gerenciamento da configuração podem propiciar grande redução no esforço e tempo consumido para manter atualizado o registro da rede e seus componentes, que são muitos e diversificados, tal como comentado por Comer (2007), que cita uma relação de elementos de rede a controlar: switches (básicas e com VLAN), pontos de acesso para redes sem fio, sistemas de modem (ADSL, TV a cabo), interface para redes de longa distância, redes metropolitanas (conexões ópticas e via rádio), roteadores, firewall, servidor DNS, servidor DHCP, servidor web, sistemas de平衡amento de carga HTTP.

Assim, a função de gerenciamento de configuração tem como características relevantes os seguintes aspectos:



- ▣ Considera a rede como um sistema dinâmico em permanente transformação;
- ▣ Provê e suporta na manutenção da estrutura física e lógica da rede;
- ▣ Acompanha o ciclo de vida do componente e sua configuração;
- ▣ Identifica componentes em um nível apropriado e suas relações (topologia);
- ▣ Monitora cada componente, documentando as trocas que devem manter os requisitos básicos estabelecidos;
- ▣ Mantém registros dos status de cada componente (topologia e dispositivos);
- ▣ Apoia a execução de alterações na configuração do sistema, visando isolar falhas, aliviar situações críticas ou atender a necessidades dos usuários.



## Ferramentas para a gerência de contabilização

A gerência de contabilização é o processo usado para medir parâmetros de utilização da rede para que a atividade dos usuários individuais ou grupos (departamentos, setores, unidades) possa ser regulada de forma adequada para efeitos de contabilidade ou compensação. É semelhante à gestão de desempenho, e o primeiro passo para a gestão de contabilização apropriado é medir a utilização de todos os recursos de rede importantes.

A gerência de contabilização provê informações para o replanejamento e redimensionamento da rede (gerenciamento de desempenho). Mediante o contínuo acompanhamento e registro dos tipos de uso da rede, é possível identificar, por exemplo, usos não alinhados com a missão crítica da instituição (uso supérfluo), mas que estejam consumindo recursos críticos da rede ou dos demais recursos de TI. Diversas ferramentas têm sido usadas para a gerência de contabilização e algumas são relacionadas a seguir:

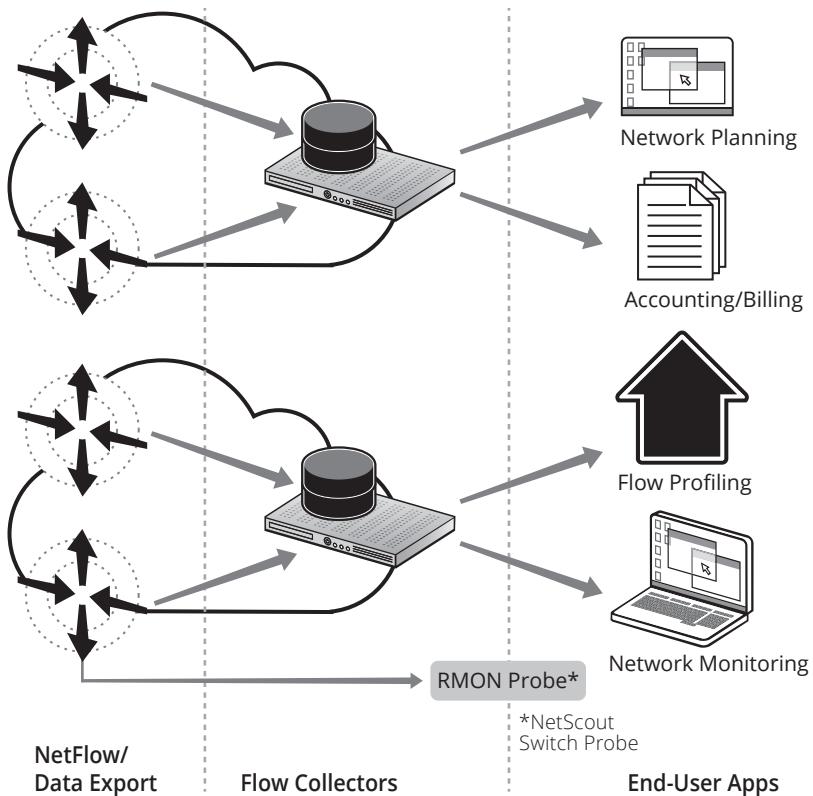
- ▣ Cisco NetFlow (Cisco 2007);
- ▣ NBAR: Network Based Application Recognition (CISCO 2008).



Entre os benefícios derivados da gerência de contabilização, podemos citar:

- ▣ Controle dos recursos;
- ▣ Permitir que tarifas sejam aplicadas aos recursos de rede (discos compartilhados, banda, arquivamento remoto, serviços de telecomunicações, e-mails etc.);
- ▣ Viabilizar a identificação de custos para a rede e seus recursos;
- ▣ Manter limites de consumo;
- ▣ Efetuar a melhor distribuição de recursos e alimentar trabalhos de planejamento.





**Figura 1.9**  
Um cenário de gerenciamento de rede com diversos componentes.

## Gerência de desempenho

A gerência de desempenho envolve manter desempenho (performance) da rede em níveis aceitáveis através de medições e gerência de diversas variáveis ligadas ao desempenho. Exemplos variáveis indicadores de performance incluem: throughput, tempos de resposta, utilização do enlace, atrasos (delay) etc.

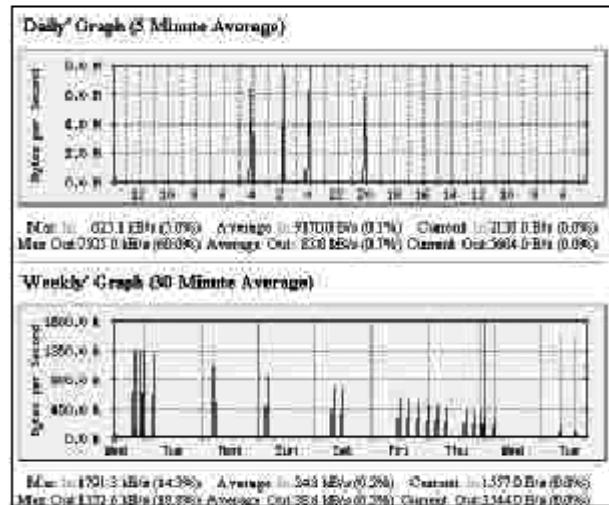
A função da gerência de desempenho visa:

- ▣ Controlar o “comportamento” dos recursos de rede;
- ▣ Avaliar as atividades de comunicação oferecidas na rede;
- ▣ Monitorar a operação diária da rede;
- ▣ Localizar pontos críticos no sistema;
- ▣ Registrar dados de operação;
- ▣ Auxiliar funções de planejamento e análise.

A gerência de desempenho pode ter um caráter reativo ou proativo:

- ▣ **Reativo:** quando o desempenho se torna inaceitável (o valor medido está a seguir do limite: threshold), o dispositivo gerenciado reage enviando um alerta para o SGR.
- ▣ **Proativo:** na gerência de desempenho proativa, uma simulação é usada para projetar o modo como o crescimento da rede afetará as métricas de desempenho. Esses simuladores alertam os administradores no sentido de impedir os problemas, antes que afetem os usuários da rede. Na gerência de desempenho proativa, uma simulação é usada para projetar o modo como o crescimento da rede afetará as métricas de desempenho. Esses simuladores alertam os administradores no sentido de impedir os problemas, antes que afetem os usuários da rede.

## MRTG MULTI ROUTER TRAFFIC GRAPHER



**Figura 1.10**  
Exemplo de aplicativo que exibe dados da rede via gráficos dinamicamente atualizados.

## Gerência de segurança

A gerência de segurança tem como meta controlar o acesso a recursos da rede, prevenir sabotagem (intencional ou não) e acesso não autorizado a informações sensíveis. Ela auxilia gerentes na criação de um ambiente de rede seguro.

Nesse sentido, envolve o uso de ferramentas como Intrusion Detection System (IDS), firewall, antivírus, entre outros. Dentre as funções dessa categoria funcional, podemos ressaltar:

- Cuidar dos mecanismos e procedimentos de proteção;
- Criar, apagar e controlar os serviços e mecanismos de segurança;
- Tratar da distribuição da informação relacionada com segurança e seus eventos;
- Manter registros de eventos relativos à segurança;
- Suportar e garantir a política de segurança adotada.

## Modelo ITIL: Information Technology Infrastructure Library

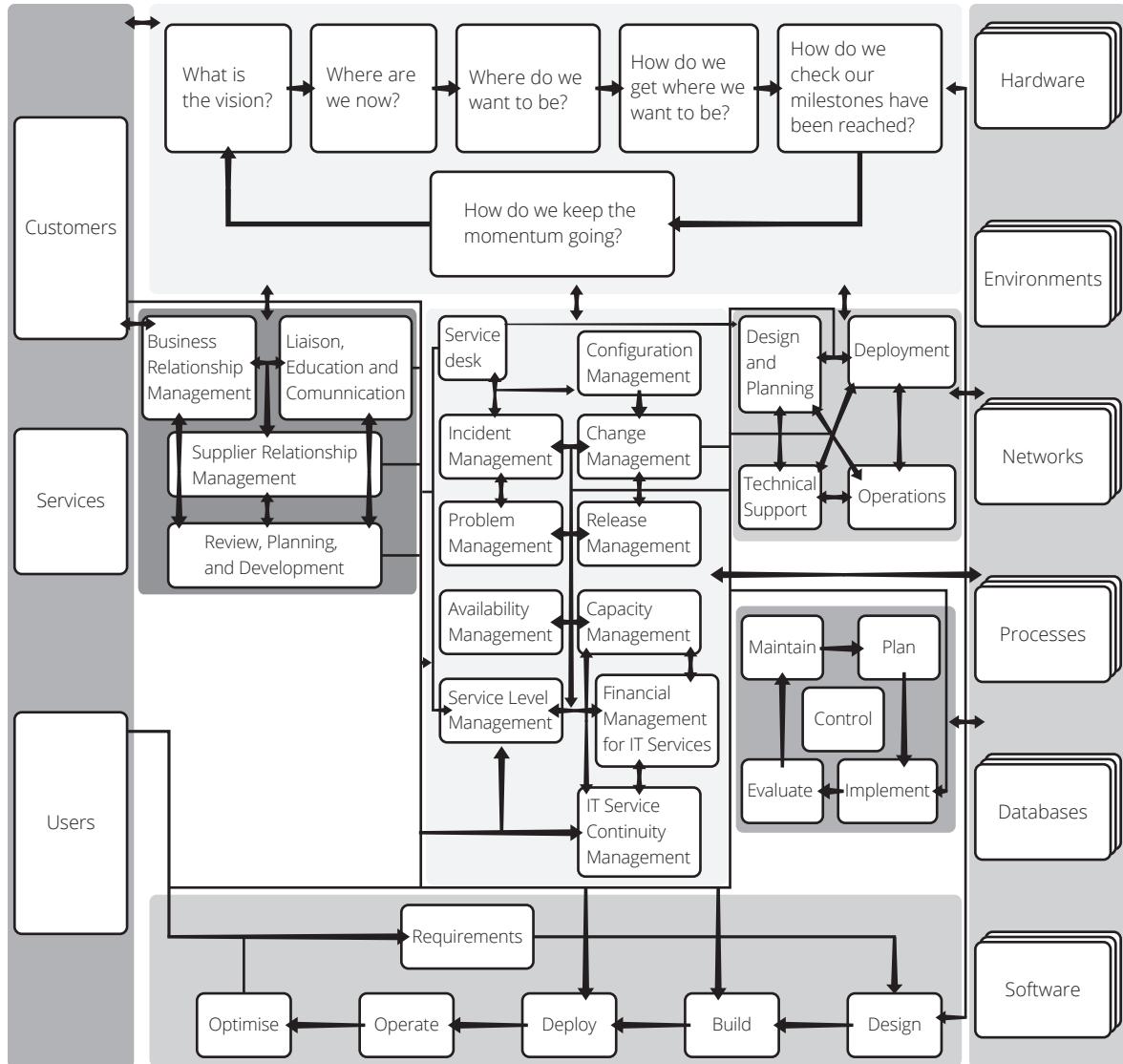
Embora a maioria das instituições tenha iniciado a organização do gerenciamento de rede com base no modelo FCAPS (ISO/OSI), mesmo no uso da arquitetura de gerenciamento para a internet, um crescente número de instituições está começando a utilizar o modelo **ITIL**. Esse modelo funcional mais abrangente começou a receber atenção e tem sido adotado como base para a organização, não apenas do gerenciamento de rede, mas de toda a infraestrutura de TI das organizações.

A ITIL busca promover a gestão com foco no cliente e na qualidade dos serviços de tecnologia da informação (TI). A ITIL endereça estruturas de processos para a gestão de uma organização de TI apresentando um conjunto abrangente de processos e procedimentos gerenciais.

Embora o modelo funcional ITIL também utilize as áreas funcionais definidos pela ISO contempla um entrelaçamento entre elas juntamente com outros serviços, tal como ilustrado na figura seguinte.

### ITIL

Information Technology Infrastructure Library é uma biblioteca de boas práticas (do inglês best practices) desenvolvida no final dos anos 80 pela Central Computer and Telecommunications Agency (CCTA) e atualmente sob custódia da Office for Government Commerce (OGC), da Inglaterra.



Os principais processos e funções do ITIL para suportar o gerenciamento de serviços de TI são:

**Figura 1.11**  
Modelo funcional  
ITIL.

- ▣ **Central de Serviços (Service Desk):** a Central de Serviços é um serviço básico do Gerenciamento de Serviços de Tecnologia de Informação, tal como definido pela Information Technology Infrastructure Library (ITIL). Ele se destina a fornecer um ponto único de contato para atender as necessidades de comunicação dos usuários e funcionários, mas também tem como meta satisfazer o cliente e os objetivos de TI do provedor. “Usuário” refere-se ao usuário real do serviço, enquanto o “cliente” refere-se à entidade que está pagando pelo serviço. Muitas organizações têm implementado um ponto único de contato para o atendimento do usuário, usando alguns tipos de central de serviços que envolvem diferenciados níveis de habilidade e possibilidade de resolução para chamadas de serviço, tais como:
  - ▣ Call center.
  - ▣ Contact center.
  - ▣ Help desk: é um termo da língua inglesa que designa o serviço de apoio a usuários para suporte e resolução de problemas técnicos, informática, telefonia e tecnologias de informação.

- ▣ **Gestão de Incidentes:** reduzir o tempo de indisponibilidade (downtime) dos serviços, oferecendo às organizações a capacidade de primeiro detectar os incidentes e depois selecionar o suporte correto para resolvê-los o mais rápido possível retornando o serviço ao estado normal, definido como uma operação de serviços dentro dos limites do contrato de nível de serviço (SLA). Implica em:
  - ▣ Detecção, atendimento e registro;
  - ▣ Gerenciamento das solicitações de serviço;
  - ▣ Classificação e suporte inicial;
  - ▣ Investigação e diagnóstico;
  - ▣ Solução e recuperação.
  - ▣ Fechamento dos problemas.
- ▣ **Gestão de Problemas:** minimizar o impacto no negócio dos incidentes e problemas causados pelos erros na infraestrutura de TI e prevenir incidentes recorrentes desses mesmos erros. Envolvem:
  - ▣ Registro e classificação de problemas;
  - ▣ Investigação e diagnóstico de problemas;
  - ▣ Controle de erro.
  - ▣ Fechamento dos problemas.
- ▣ **Gestão de Mudança:** minimizar o impacto da mudança requerida para resolução do incidente ou problema, mantendo a qualidade dos serviços, bem como melhorar a operacionalização da infraestrutura;
- ▣ **Gestão de Liberação:** prevenir a indisponibilidade do serviço, garantindo que as instalações de versões de hardware e software estejam seguras, autorizadas e devidamente testadas;
- ▣ **Gestão de Configuração:** identificar e controlar os ativos de TI e itens de configuração existentes na organização, estabelecendo o relacionamento destes aos serviços prestados. Envolve:
  - ▣ Estabelecer itens de configuração;
  - ▣ Acessar itens de configuração;
  - ▣ Alterar/revisar itens de configuração.
- ▣ **Gerenciamento do Nível de Serviço:** garantir o acordo de nível de serviço (SLAs) previamente estabelecido entre o fornecedor e o cliente;
- ▣ **Gerenciamento financeiro para o serviço de TI:** demonstrar ao cliente o custo real dos serviços prestados e gerenciá-los de forma profissional;
- ▣ **Gerenciamento de disponibilidade:** garantir a disponibilidade e confiabilidade dos recursos de TI, a fim de assegurar a satisfação do cliente e a reputação do negócio;
- ▣ **Gerenciamento de capacidade:** assegurar que a capacidade da infraestrutura de TI está adequada às demandas do negócio conforme a necessidade e no tempo esperado, observando sempre o gerenciamento do custo envolvido;
- ▣ **Gerenciamento da continuidade do serviço de TI:** atender todo o processo de gerenciamento da continuidade do negócio, assegurando que os recursos técnicos e sistemas de TI sejam recuperados quando requeridos, no tempo desejado.



## Fluxo de mensagens em SNMP

- Mensagens trocadas no protocolo SNMP poderiam usar TCP ou UDP como protocolo de transporte, mas o UDP foi o escolhido, por questões de performance. UDP é sem conexão e mais “leve” que o TCP.
- O gerente geralmente implementa timeout e “retransmissão” se não obtiver resposta.

Os objetos gerenciados são nomeados com base em uma árvore de nomeação padronizada. Essa árvore de nomeação determina uma sequência de números que identificam cada objeto gerenciado. A figura seguinte mostra como essa árvore de nomeação é estruturada. Um objeto gerenciado integrante da MIB 2 terá uma identificação que vai iniciar pela seguinte sequência de identificadores: 1.3.6.1.2.1

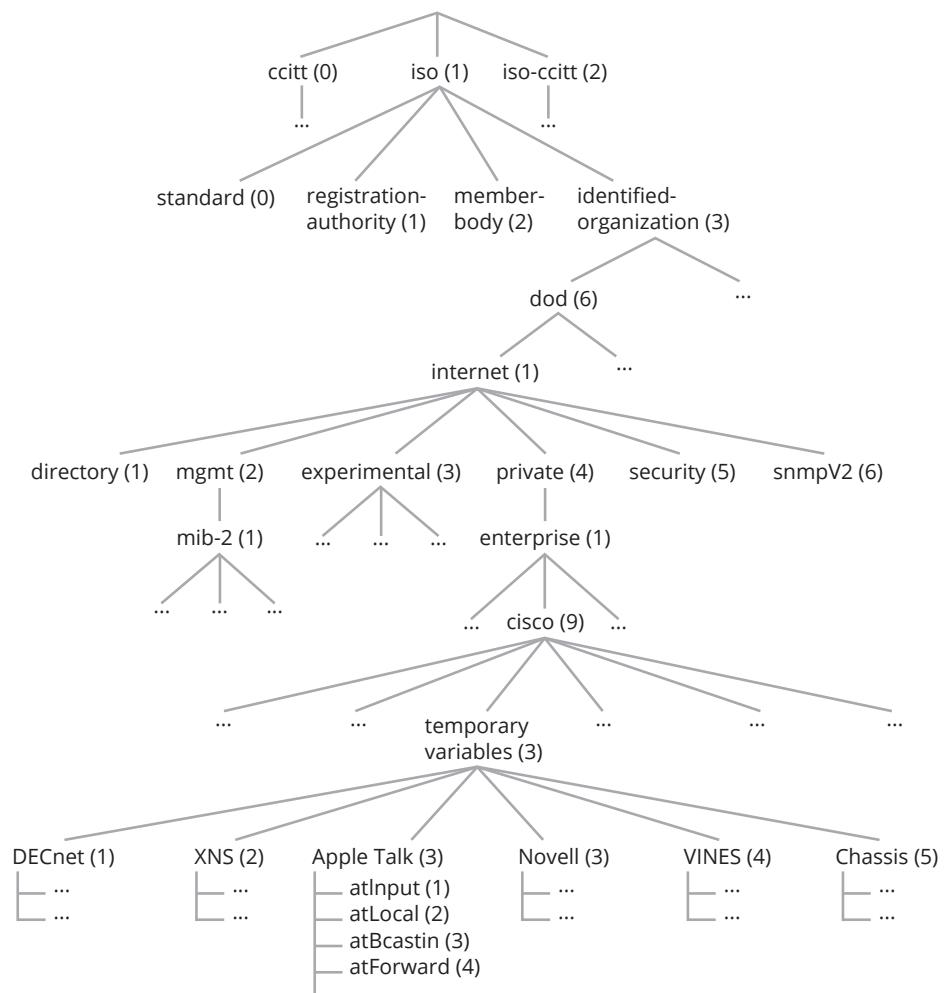
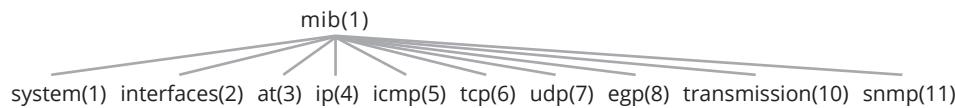


Figura 1.12  
MIB OID.

A continuidade dos identificadores dos objetos gerenciados indicará se eles pertencem a algum dos grupos de objetos nomeados na figura 1.13.

## MIB-II

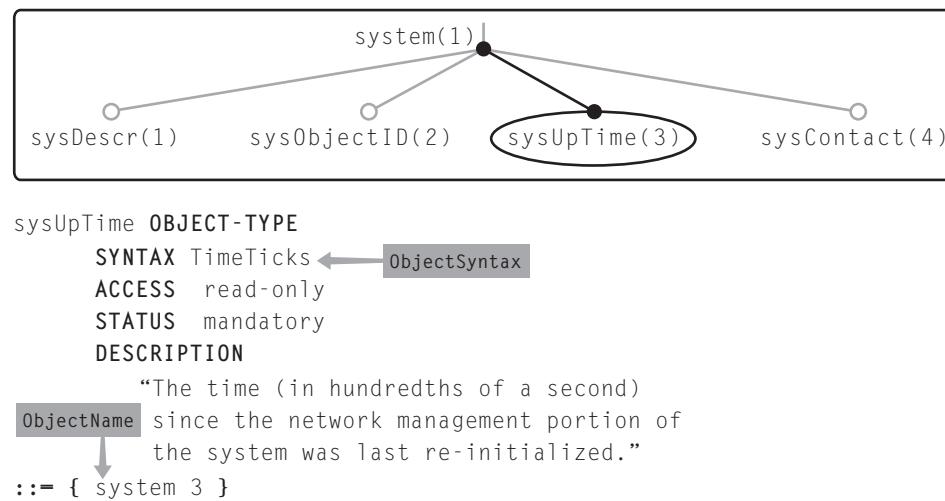
### The MIB sub-tree



**Note:** There is an object cmot(9) under the mib but it has become almost superfluous and for all intents and purposes is not one of the SNMP manageable groups within mib.

**Figura 1.13**  
MIB II.

A especificação de um objeto gerenciado incluirá, além de sua identificação, também suas características, ou seja, se ele é um número inteiro ou um conjunto de caracteres, se permite apenas leitura ou pode ser atualizado pelo gerente SNMP etc. A figura 1.14 ilustra a definição de um objeto gerenciado.



**Figura 1.14**  
Definição de objetos no SNMP.

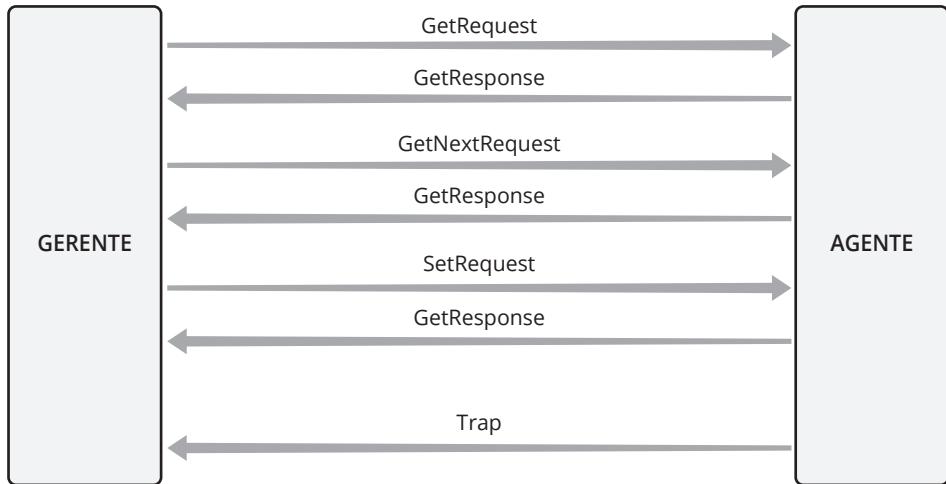
## Modelo operacional SNMP

O protocolo SNMP trabalha utilizando o seguinte conjunto de operações:

- **GetRequest:** requisição para capturar o valor de uma variável;
- **SetRequest:** requisição de mudança de um valor de uma variável feita pelo gerente;
- **GetNextRequest:** requisição para capturar o valor da próxima variável;
- **Response:** mensagem de resposta do agente para as solicitações de Get e Set;
- **Trap:** notificação do agente ao gerente, comunicando o acontecimento de um evento predeterminado.



Essas mensagens são encapsuladas pelo protocolo UDP e transportadas em datagramas IP. Através do seu uso, a estação de gerenciamento da rede pede informações para o agente SNMP, altera valores de objetos gerenciados no agente e obtém respostas a esses comandos. Adicionalmente, o agente SNMP pode gerar mensagens não solicitadas ou notificações enviadas para o gerente SNMP. A figura 1.15 ilustra essas interações.



**Figura 1.15**  
Definição de  
objetos no SNMP.

O protocolo SNMP foi inicialmente definido pelo RFC 1067 em 1988, mas sofreu uma evolução e a versão 2 foi publicada em 1993, no RFC 1442, e logo recebeu ampla aceitação. Nessa segunda versão, mecanismos de segurança foram adicionados, mas em caráter opcional, e essa é a versão mais disseminada. Em 1995, foi publicada a versão, definida no RFC 1861, que incorpora mecanismos de segurança com autenticação forte e criptografia.

Versão	Nível	Autenticação	Criptografia	O que acontece?
SNMPv1	noAuthNoPriv			Usa o texto <i>community</i> para autenticar
SNMPv2c	noAuthNoPriv			Usa o texto <i>community</i> para autenticar
SNMPv3	noAuthNoPriv	Username		Usa o <i>username community</i> para autenticar
	authNoPriv	MD5ou SHA		Usa os algoritmos de autenticação - HMAC-MD5 e HMAC-SHA
	authPriv	MD5ou SHA	DES	Usa criptografia 56-bit DES

**Tabela 1.2**  
Protocolo SNMP –  
versões.

Tendo em vista a necessidade de monitorar e obter informações sobre todos os dispositivos da rede, mesmo aqueles que não tenham um agente SNMP ou o que o agente esteja desativado, foi proposta em 1991 uma ampliação na função do agente SNMP com vistões à sua atuação de modo análogo ao de um analisador de rede. A expansão da função foi proposta mediante o uso de uma ampliação da MIB (RFC 1271), com novos conjuntos de objetos gerenciados para armazenar as informações derivadas da operação do agente em modo de operação com captura de todo o tráfego por ele percebido na rede.

Suas principais características são:

- MIB específica para monitoração de redes;
- Provê uma maneira efetiva e eficiente de monitorar comportamento de segmentos de rede;
- Objetivos (segundo a RFC 1271) envolvem:
  - Operação off-line;
  - Monitoramento proativo;
  - Notificação de eventos significativos somente à estação de gerência;
  - O monitor deve suportar gerentes múltiplos;

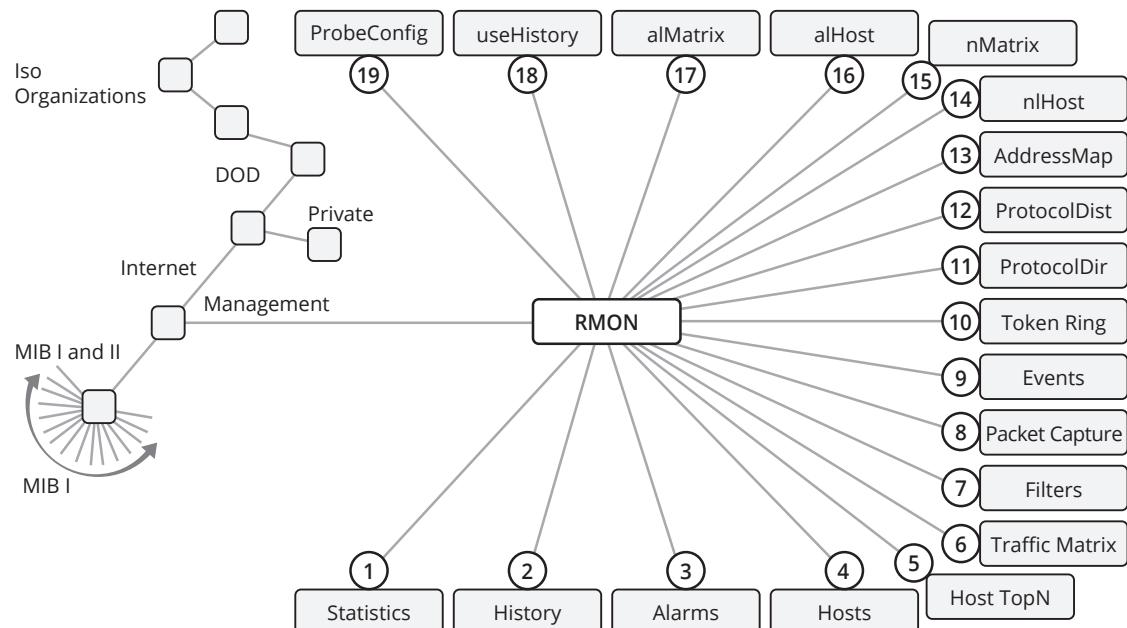
## Grupos de Objetos da RMON v1

- **Statistics:** estatísticas de Rede em tempo real;
- **History:** histórico de estatísticas pré-selecionadas;
- **Alarm:** Traps que serão enviadas caso as estatísticas excedam um determinado valor;
- **Hosts:** estatísticas de rede de uma máquina específica;
- **Hosts top N:** Registro de "N" conexões ativas num determinado período;
- **Matrix:** Matrix com o tráfego entre máquinas;
- **Filter:** Filtro para capturar somente o dado interessado;
- **Capture:** coleção de dados filtrados pelo grupo "filter";
- **Event:** envia alertas, Traps SNMP;
- **Token Ring:** específico para enlaces Token Ring;

## Grupos de Objetos da RMON v2

- **Protocol Directory:** lista de protocolos que podem ser monitorados;
- **Protocol Distribution:** estatística de tráfego por protocolo;
- **Address Map:** mapeamento de IP para MAC ADDRESS;
- **Network-Layer Host:** estatística de tráfego de camada 3 separada por máquina;
- **Network-Layer Matrix:** tráfego de camada 3 com o registro de origem e destino;
- **Application-Layer Host:** estatística de tráfego por protocolo e máquina;
- **Application-Layer Matrix:** estatística de tráfego por protocolo, máquina, origem e destino;
- **User History:** amostras periódicas de uma variável específica;
- **Probe Configuration:** utilizado para configurações remotas;
- **RMON Conformance:** requisitos para funcionar com padrão RMON v2.

Figura 1.16  
Classes de objetos  
da RMON MIB.



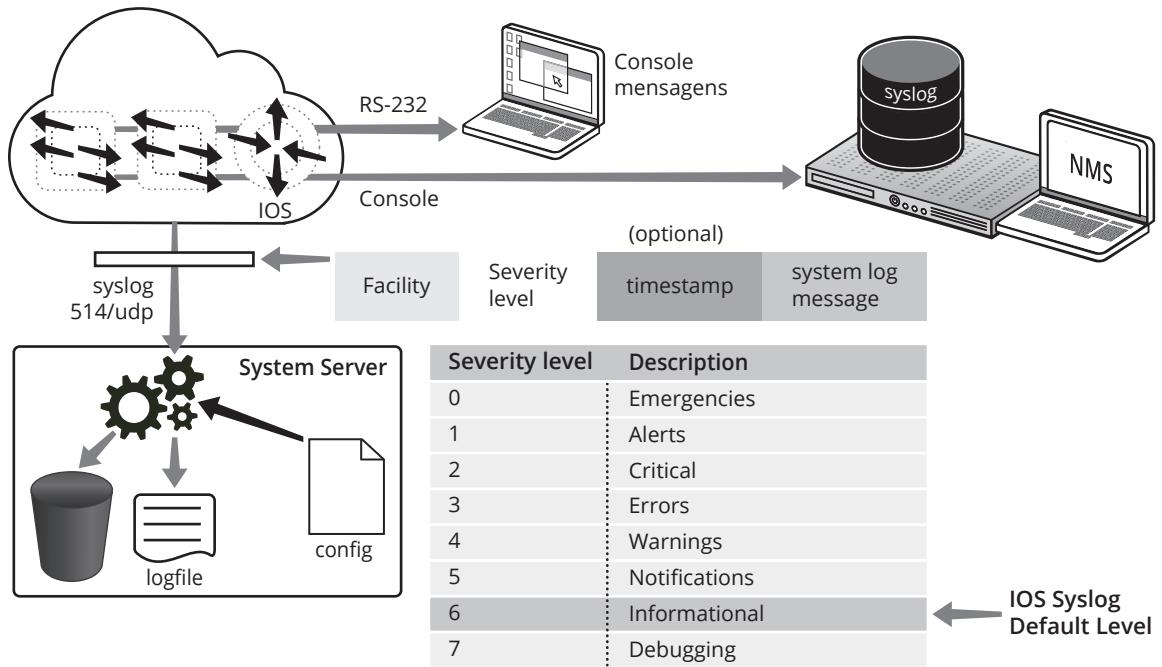


Figura 1.17  
Syslog – log de eventos.

Um agente RMON pode utilizar o serviço syslog para o envio de alertas e notificações. Syslog é um padrão criado pela IETF para a transmissão de mensagens de log em redes IP. O protocolo syslog é tipicamente usado no gerenciamento de computadores e na auditoria de segurança de sistemas.

O protocolo syslog é bastante simples: o remetente envia uma mensagem de texto de acordo com a RFC 5424 para o destinatário (também chamado "syslogd", "serviço syslog" ou "servidor syslog"). Essas mensagens podem ser enviadas tanto por UDP quanto por TCP e seu conteúdo pode ser puro ou utilizando SSL.

## Agentes e gerentes

Entidades com papéis definidos:

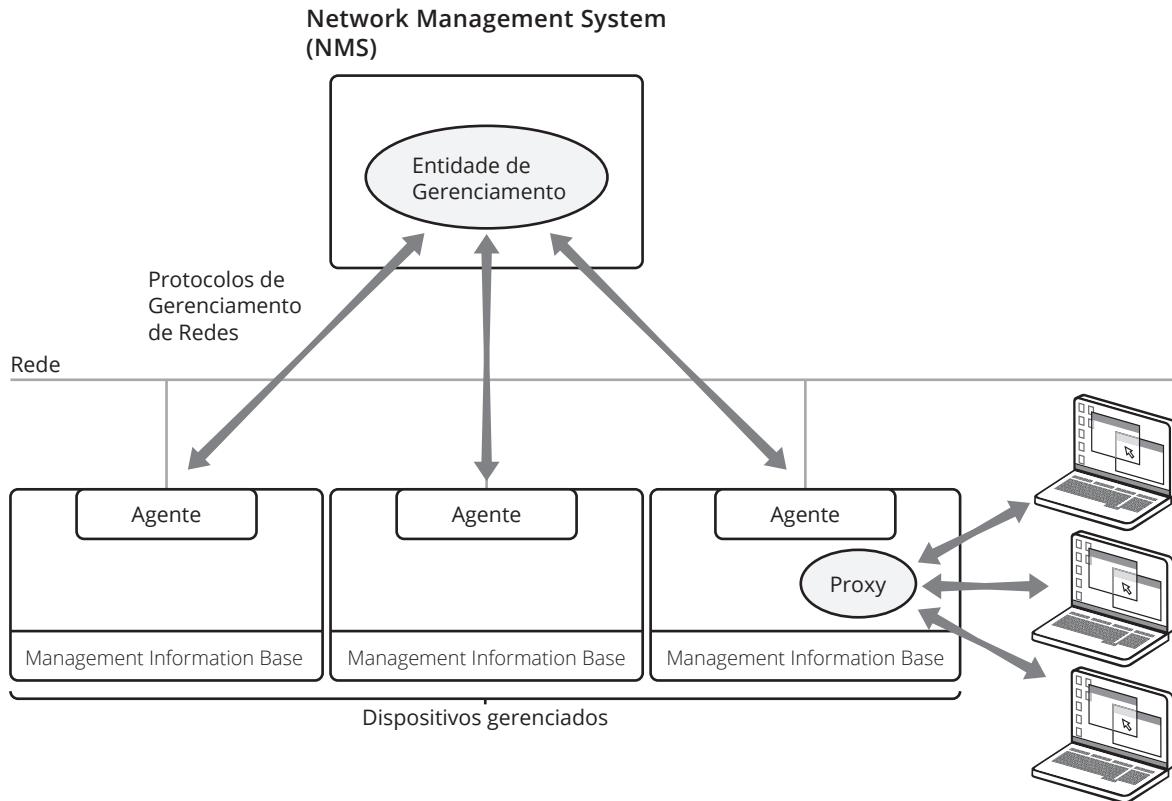
- Gerentes.
- Agentes.

A arquitetura de gerenciamento de redes usada tanto no modelo OSI quanto no modelo SNMP é baseada na existência das entidades gerente e agente:

- **Gerente:** entidade responsável em coletar as informações e disparar ações. Atua sobre os agentes. Agrega inteligência ao sistema, podendo realizar tarefas mais complexas, como gerar relatórios para o usuário;
- **Agente:** entidade localizada perto do item (ou internamente a ele), gerenciando (como um equipamento) esse mesmo item e respondendo às solicitações do gerente via rede. Permite o acesso às informações de gerenciamento localizadas localmente, mantendo-as como um reflexo da realidade do equipamento.

Os protocolos de gerenciamento permitem ao gerente disparar comandos para o agente através da rede, como qualquer outra aplicação de rede. Por isso, os protocolos de gerenciamento normalmente são protocolos de camada de aplicação, específicos para a atividade de gerenciamento e implementados em aplicações de gerenciamento.

Um agente pode funcionar como um proxy representando equipamentos que não são capazes de prover informações sobre si mesmo e sobre seu funcionamento, tal como preconizado na arquitetura de gerenciamento daquela rede. Um agente proxy interage com dispositivos da rede usando qualquer forma possível e mantém atualizados na MIB as informações relativas os dispositivos representados.



**Figura 1.18**  
NMS e agentes.

## Informação de gerenciamento

As informações usadas para fins de gerenciamento da rede são estruturadas de acordo com padrões que determinam sua:

- Organização;
- Nomenclatura;
- Regras e procedimentos.

Apesar dessa padronização, ainda restam muitas decisões que os responsáveis pelo gerenciamento da rede precisam tomar, como:

- Qual o intervalo de tempo para coletar as informações?
- Que informação se deseja coletar?

Essas decisões precisam ser avaliadas, tendo em vista evitar uma sobrecarga de tráfego na rede de processamento nos equipamentos envolvidos. A decisão sobre quanta informação é interessante e/ou necessária para o gerenciamento dependerá do equipamento e contexto de rede onde ele estiver. Mesmo assim, muitas das informações de gerenciamento relacionadas podem ser assumidas. Por exemplo: uma informação sempre útil de um equipamento como um roteador é a sua tabela de roteamento ou a quantidade de pacotes enviados, recebidos e descartados, por erro ou impossibilidade de reencaminhamento.

É necessária uma estrutura de informação de gerenciamento para gerenciar a massa de informações associadas às complexas redes multifabricantes que existem;

- ▣ Uma nomenclatura deve ser definida para descrever essas informações;
- ▣ Devem ser definidos procedimentos de acesso e alteração das informações;
- ▣ O tratamento e a modelagem de informação de gerenciamento normalmente abordam uma estratégia baseada em objetos. Por isso, é importante recordar esses conceitos;
- ▣ A abstração de objetos permite que sejam capazes de modelar aspectos reais da situação ou comportamento dos dispositivos da rede. Assim, os objetos representarão uma realidade ou aspecto concreto tal como a quantidade de portas em um roteador, a quantidade de pacotes recebidos em uma porta, os identificadores (número IP) de origem e destino de uma conexão TCP etc.

## Conceitos de orientação a objetos

- ▣ Classes.
- ▣ Encapsulamento.
- ▣ Operações (métodos).
- ▣ Eventos.
- ▣ Relacionamentos entre os objetos, associação, herança etc.

Uma instância individual de um objeto gerenciado é uma variável da MIB.

Templates (modelos) são usados para a definição dos objetos. As implementações devem seguir as informações descritas nos templates para uniformizar a compreensão e a manipulação da informação. Exemplo: se um fabricante armazena e trata o endereço IP de um equipamento como do tipo string e outro como do tipo “inteiro long”, as aplicações que vão coletar e tratar essas informações poderiam estar trabalhando com tipos de dados incompatíveis.

## Structure of Management Information (SMI)

- ▣ Como organizar os dados de gerenciamento de rede?
- ▣ De que forma podemos identificar um objeto?
- ▣ Como definir e descrever um objeto?
- ▣ De que forma podemos codificar (serializar) um objeto?
- ▣ Como as variáveis decorrentes dessas ações devem ser representadas?

Em um nível mais alto, a Structure of Management Information (SMI) é usada na criação de um módulo de informação (um pacote de informações relevantes usadas pelo gerenciamento com SNMP). Em resumo, a SMI é usada para a criação de módulos de MIB.

Por exemplo, a SMI para arquitetura TCP/IP (SNMP) define:

- ▣ O modo como são definidos os módulos de MIB;
- ▣ O subconjunto da linguagem de descrição de dados, no padrão ASN.1 (explicado adiante), usado com SNMP;
- ▣ O modo como todas as construções ASN.1 deverão ser serializadas para envio pela rede;
- ▣ O formato dos identificadores de objetos na árvore MIB para gerenciamento internet;
- ▣ A descrição dos objetos gerenciados SNMP;

- A codificação desses objetos para transmissão na rede, feita através das regras básicas de codificação (BER, Basic Encoding Rules).
- A SMI é particular para uma determinada arquitetura de gerenciamento de rede.

## Abstract Sintax Notation (ASN.1)

Atributos principais para cada objeto:

- Nome do objeto (identificador).
- Sintaxe do objeto.
- Codificação do objeto.



A descrição da MIB formaliza sua estrutura, cada objeto que ela contém e suas características. É necessária uma definição textual da MIB, de forma a padronizá-la e permitir que ela seja implementada por outros fabricantes e desenvolvedores.

As variáveis nas MIBs são descritas por um subconjunto de facilidades de uma linguagem de descrição de dados chamada Abstract Sintax Notation (ASN.1) que é um padrão ISO (8824). A ISO definiu essa notação, derivada da recomendação ITU X.409, que permite definir tipos de dados simples e complexos, bem como os valores que tais tipos podem assumir. Essa notação, denominada Notação para Sintaxe Abstrata Um (ASN.1 Abstract Sintax Notation One). A notação não indica o valor dos dados, apenas sua forma. Adicionalmente existem algoritmos, denominados Regras Básicas de Codificação (Basic Encoding Rules) que determinam o valor dos octetos representando tais valores tal como serão transmitidos com os protocolos de transferência de dados.

ASN.1 utiliza alguns tipos primitivos (elementos de dados) com os quais podem ser compostas estruturas mais complexas. Os tipos primitivos ou básicos são:

- BOOLEAN;
- INTEGER;
- BITSTRING;
- OCTETSYSTRING;
- NULL.

Estruturas complexas são definidas agregando-se tais tipos primitivos de algumas formas previstas na ASN.1. As principais formas de estruturação de tipos compostos ou “construídos” são:

- **SEQUENCE**: lista ordenada de tipos;
- **SEQUENCE OF**: iteração ilimitada de um único tipo;
- **SET**: lista não ordenada de tipos estruturação;
- **SET OF**: interação ilimitada de um único tipo (a ordem não é importante);
- **CHOICE**: um campo que consiste de um valor dentre os tipos listados.

ASN.1 define uma notação para especificação de valores e para definição de tipos. A definição de um tipo consiste em uma coleção de campos que, no mais baixo nível, consiste em um identificador, uma possível etiqueta (rótulo ou “tag”), uma referência e uma possível indicação de que aquele campo é opcional (pode ser omitido).

No menor nível, os campos da definição são combinados usando mecanismos de estruturação que começam com o nome do tipo da estrutura e então, em geral, uma lista de campos separados por vírgulas e envolvidos em chaves “{}”. É válido usar aninhamento de estruturas.

Assim, um tipo será definido, segundo a ASN.1, usando uma das sequências de tipo válidas, do tipo Builtin ou Defined:

```
Type ::= BuiltinType|DefinedType  
BuiltinType ::= BooleanType|  
              IntegerType|  
              BitStringType|  
              OctetStringType|  
              NullType|  
              SequenceType|  
              SequenceOfType|  
              SetOfType|  
              SetOfType|  
              ChoiceType|  
              TaggedType|  
              AnyType|  
              CharacterSetType|  
              UsefulType|
```

O DefinedType especifica sequências externas usadas para referir definições de tipos e valores.

## Sintaxe ASN.1 de um objeto da MIB

Os objetos de uma MIB são especificados usando ASN.1, bem como o próprio protocolo SNMP, que também é especificado com ASN.1. A figura seguinte ilustra a definição do objeto sysUpTime nessa forma de notação.

```
sysUpTime OBJECT-TYPE  
    SYNTAX  TimeTicks  
    ACCESS  read-only  
    STATUS  mandatory  
    DESCRIPTION  
        "The time (in hundredths of a second) since the network management  
        portion of the system was last re-initialized."  
        ::= { system 3 }
```

**Figura 1.19**  
Definição de um  
objeto gerenciado  
em ASN.1.

Essa definição é tanto legível pelas pessoas como por programas e isso permite que novos objetos gerenciados sejam incorporados a uma MIB e as aplicações de gerenciamento de rede podem ler essa especificação e compilar a definição dos novos objetos, incorporando-os ao acervo de objetos gerenciados que serão consultados periodicamente (processo de polling) pelo gerente, usando o protocolo SNMP e cujos valores serão retornados pelo agente SNMP.

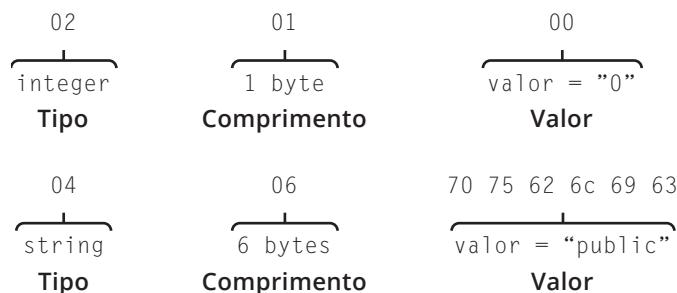
## Regras de codificação BER (Basic Encoding Rules – ISO 8825)

As “Basic Encoding Rules” proveem um algoritmo que especifica como um valor de qualquer estrutura (tipo) definida usando ASN.1 deve ser codificada para transmissão. O uso do algoritmo no sentido contrário permite que qualquer receptor que tinha conhecimento da definição do tipo ASN.1 possa decodificar os bits que chegam em um valor daquele tipo.

A codificação de um tipo de dados ASN.1 (usando as Basic Encoding Rules) permite que um receptor, sem conhecimento da definição de tipo, reconheça o inicio e o fim das construções (SEQUENCE, SET, etc.) e os octetos representando os tipos básicos de dados (BOOLEAN, INTEGER). No uso mais simples da notação, e também possível determinar, a partir da codificação, as construções efetivamente usadas e os tipos de dados básicos.).

Cada valor de dados a ser codificado terá uma representação consistindo de:

- **T:** tipo;
- **C:** comprimento: quando indefinido, termina com dois octetos nulos;
- **V:** valor – pode ser um valor (número ou cadeia de caracteres) ou uma série de outros componentes TCV (tipos complexos ou construído).



**Figura 1.20**  
Codificação em  
uma informação  
de um objeto  
gerenciado.

Para se entender melhor essa codificação, vamos analisar o primeiro octeto (oito bits) da sequência de bits codificados, que especifica o tipo de identificador (Type), como exibido na figura 1.21.

8	7	6	5	4	3	2	1
Class		P/C	Number				

**Figura 1.21**  
Type na ASN.1

Os bits 7 e 8 identificam a classe, de acordo com a tabela 1.3.

Class	bit 8	bit 7
Universal	0	0
Application	0	1
Context-specific	1	0
Private	1	1

**Tabela 1.3**  
Classes em ASN.1

- Se a classe for “Universal”, ou seja, se os bits 8 e 7 contiverem o valor “0”, o tipo do valor do dado é ASN.1 nativo;
- Caso a classe seja “Application” com o valor do bit 8 igual a “0” e o valor do bit 7 igual a “1”, isso significa que o valor é para uma aplicação específica, como por exemplo o serviço de diretório definido pela série de padrão X.500 da ITU-T;



- Se a classe for do tipo “Context-specific”, ou seja, o bit 8 com valor “1” e o bit 7 com valor “0”, significa que o tipo de dado é uma estrutura específica e TAGs são utilizadas para fazer a distinção entre esses tipos;
- Se a classe for do tipo “Private”, significa que o dado é específico de um fabricante.

Já o bit 6, também conhecido como primitivo/construído: P/C – se estiver com o valor “0” significa que é um valor primitivo, como por exemplo, um valor INTEIRO. Caso o bit primitivo/construído – P/C – esteja com o valor “1” significa que o dado é do tipo construído e os bits de 1 à 5 serão o campo TAG utilizado para identificar esse tipo de dado.

A tabela da figura 1.23 descreve o campo TAG e seus valores correspondentes, ou seja, se os bits de 1 a 5 tiverem o valor “0”, significa que é o fim do dado. Caso o bit 5 contenha o valor “1” e os bits de 1 a 4 tiverem o valor “0001”, significa que o dado é do tipo BOOLEAN, assim como temos a seguir.

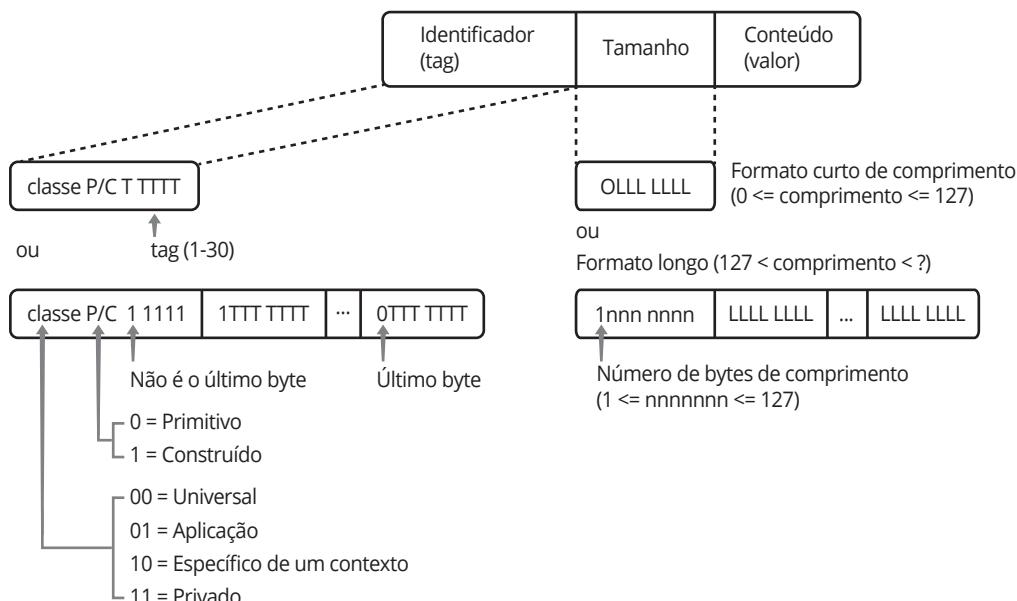
Nome	P/C	Número (decimal)	Número hexadecimal)	Número binário
EOC (End-of-Content)	P	0	0	00000
BOOLEAN	P	1	1	00001
INTEGER	P	2	2	00010
BIT STRING	P/C	3	3	00011
OCTET STRING	P/C	4	4	00100
NULL	P	5	5	00101
OBJECT IDENTIFIER	P	6	6	00110
Object Descriptor	P	7	7	00111
EXTERNAL	C	8	8	01000
REAL (float)	P	9	9	01001
ENUMERATED	P	10	A	01010
EMBEDDED	C	11	B	01011
UTF8String	P/C	12	C	01100
RELATIVE-OID	P	13	D	01101
SEQUENCE and SEQUENCE OF	C	16	10	10000
SET and SET OF	C	17	11	10001
NumericString	P/C	18	12	10010
PrintableString	P/C	19	13	10011
T61String	P/C	20	14	10100
VideotexString	P/C	21	15	10101
IA5String	P/C	22	16	10110
UTCTime	P/C	23	17	10111
GeneralizedTime	P/C	24	18	11000

Nome	P/C	Número (decimal)	Número hexadecimal)	Número binário
GraphicString	P/C	25	19	11001
VisibleString	P/C	26	1A	11010
GeneralString	P/C	27	1B	11011
UniversalString	P/C	28	1C	11100
CHARACTER STRING	P/C	29	1D	11101
BMPString P/C 30 1E	P/C	30	1E	11110

**Tabela 1.4**  
Tags usados em ASN1.

## Representação BER (TLV)

Codificação de comprimento definido  
(tipos primitivos ou construídos)



Para não limitar os números de ramos na árvore, foi criada uma maneira de representar números realmente grandes. Se o número for menor que 127, ele é codificado num único byte. Se for maior, é codificado em vários bytes, de acordo com a figura acima.

**Figura 1.22**  
BER Basic Encoding Rules.

Um tipo de dado muito comum em gerenciamento é o OBJECT IDENTIFIER. Sua codificação não é óbvia: os primeiros 2 dígitos, x.y, devem ser codificados num único número através da fórmula  $40x+y$  em decimal, e depois para hexadecimal.

No caso de 1.3 (x=1, y=3) => 43 em decimal resulta em 2B em hexadecimal.

## Alguns identificadores ASN.1 (tags)

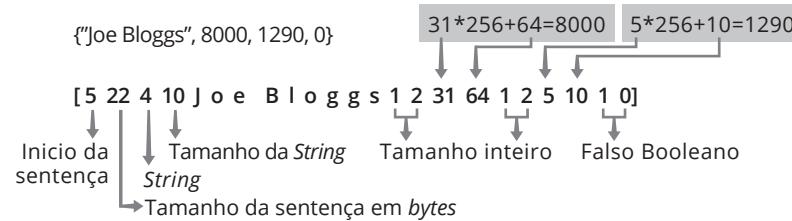
INTEGER	02
BIT STRING	03
OCTET STRING	04
NULL	05
OBJECT IDENTIFIER	06
SEQUENCE	30
IpAddress	40
Counter	41
Gauge	42
TimeTicks	43
Opaque	44
Counter64	46
UInteger32	47
Get-Request-PDU	A0
GetNextRequest-PDU	A1
GetResponse-PDU (response-PDU)	A2
SetRequest-PDU	A3
Trap-PDU	A4
GetBulkRequest-PDU	A5
InformRequest-PDU	A6
SNMPv2-Trap-PDU	A7

- ▣ Exemplos de tipos primitivos: INTEGER, ENUMERATED, OCTET STRING, OBJECT IDENTIFIER, NULL, BOOLEAN, BIT STRING, REAL, “Counter”, “Gauge”, “TimeTicks”, “IpAddress”, “Network Address”, “Integer32”, “Counter32”, “Gauge32”, “Counter64”, “UInteger32” etc.
- ▣ IpAddress (OCTET STRING de tamanho 4);
- ▣ Counter (unsigned 32-bit integer);
- ▣ Gauge (unsigned 32-bit integer);
- ▣ Timeticks (unsigned 32-bit integer);
- ▣ Opaque (tipos não usados em SNMPv1);
- ▣ Outros: “DateAndTime”, “DisplayString”, “MacAddress”, “PhysAddress”, “TimeInterval”, “TimeStamp”, “TruthValue”, “VariablePointer” — todos são “textual conventions” usados como tipos de dados;
- ▣ Exemplos de tipos construídos (a partir de tipos primitivos): SEQUENCE, SEQUENCE OF, SET, SET OF, CHOICE etc.



## ASN.1 e BER (Exemplos)

```
Struct employee {           employee :: SEQUENCE {  
    char     name[32];      Name      OCTET String, -- 32 characters  
    int      salary;        Salary     INTEGER;  
    int      entryDate;    entryDate INTEGER;  
    int      sex;          Sex       BOOLEAN;  
};
```



**Figura 1.23**  
Exemplo de codificação usando BER.

Veja no exemplo como as regras BER determinam uma única serialização para o conjunto de dados a ser transmitido.

Mesmo a troca de dados entre aplicações em plataformas diferentes não corre o risco de ter incoerências na compreensão das informações.



# 2

## Ferramentas de inspeção e monitoração de redes

objetivos

Descrever o processo de resolução de problemas, aprender as características das ferramentas de inspeção e monitoração de redes.

conceitos

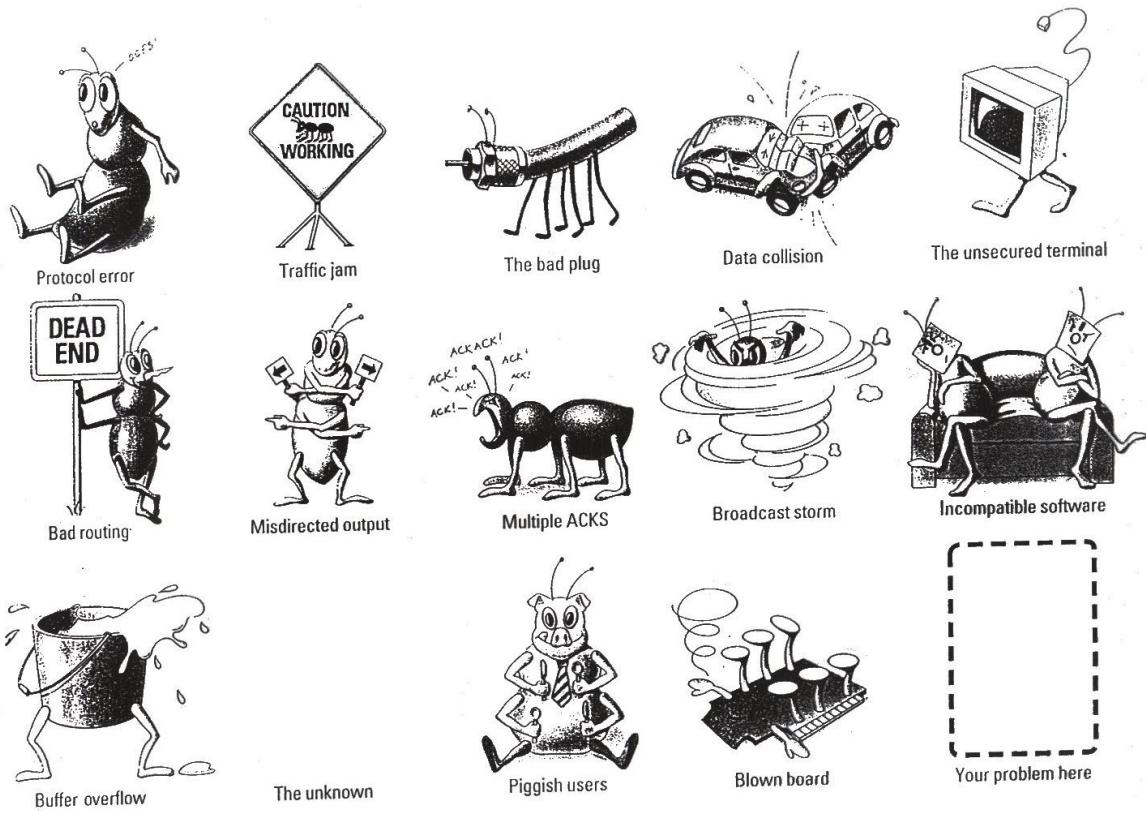
Problemas na rede, Processo de resolução de problemas, Ferramentas para inspeção e monitoração, Programa Wireshark, Programa TCPDUMP.

Inúmeros problemas podem ocorrer em uma rede. A capacidade de detectar e diagnosticar depende das ferramentas disponíveis e do conhecimento de quem as utiliza. Esse conhecimento vai além de conhecer as funções da ferramenta. É preciso que exista conhecimento suficiente para interpretar as informações que cada tipo de ferramenta é capaz de fornecer. Conforme salientado por Sanders (2007), muitas coisas podem causar problemas na rede, desde uma infecção por spyware simples até um complexo erro de configuração do roteador, e é impossível resolver todos os problemas imediatamente. O melhor que se pode fazer é estar totalmente preparado com o conhecimento e as ferramentas para reagir a esses e outros tipos de problemas.

### Problemas na rede

Os problemas que ocorrem na rede são caracterizados por sintomas que precisam ser percebidos e analisados para um diagnóstico das causas. O desenho na figura 2.1 mostra um cartum com um conjunto de possíveis causas de um comportamento errático em um segmento de rede local:





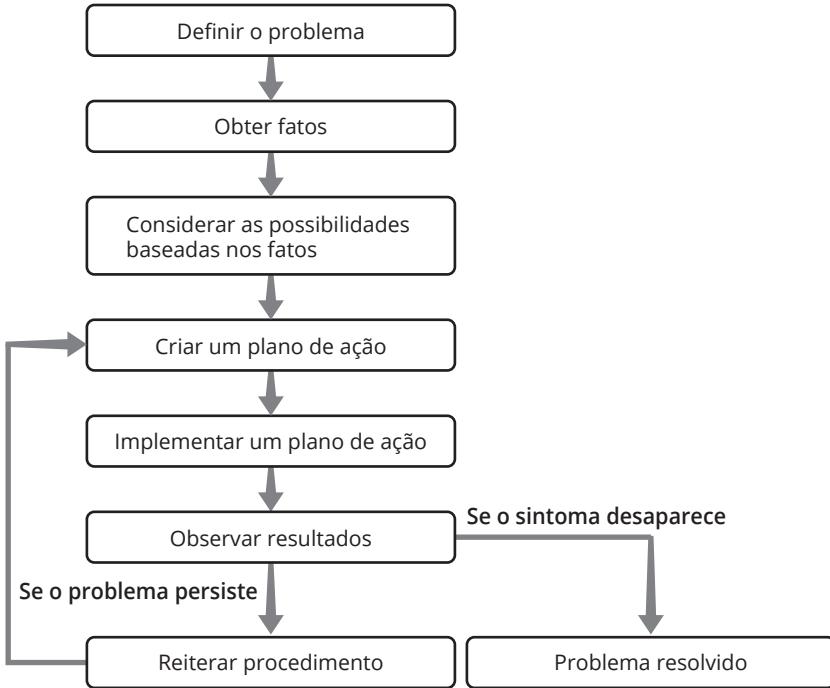
**Figura 2.1**  
Causas de mau  
funcionamento de  
uma rede local.

## Processo de resolução de problemas

Os problemas, como se pode verificar nessa figura, podem ser ocasionados por inúmeras causas, tais como mau funcionamento de hardware, software ou mesmo usuário gerando tráfego excessivo. Realizar o diagnóstico de um problema de mau funcionamento em rede requer uma abordagem sistemática. Uma proposta de abordagem para diagnóstico de problemas de rede foi sugerida em um guia de diagnóstico da CISCO (2008) e emprega três grandes etapas:

- Definir o problema, compondo um quadro descritivo em que todos os sintomas estão relacionados;
- Identificar os possíveis problemas que possam ocasionar aqueles sintomas;
- Buscar evidências sobre a ocorrência de cada um desses possíveis problemas, iniciando pelos mais usuais.

Esse processo de resolver problema é detalhado no diagrama da figura 2.2.



**Figura 2.2**  
Processo de resolução de problemas.

No primeiro passo, para cada sintoma percebido, as causas associadas são relacionadas. Muitas vezes o registro do problema surge com informações incompletas ou vagas. Por exemplo, um usuário pode reclamar que o servidor está lento. Com essa informação, não se pode saber se a lentidão ocorre apenas para um determinado servidor, para uma determinada aplicação ou se também há lentidão no acesso a outros servidores. Ao efetuar o registro do problema, é importante formular uma série de perguntas com o objetivo de conseguir uma definição o mais completa possível para o problema. Em algumas instituições, são preparados roteiros de perguntas, encadeadas de forma a conduzir a uma caracterização do problema em determinada área de possíveis causas.

**!** Algumas situações não são passíveis de serem percebidas pelo usuário que relata o problema, e demandarão inspeção de dados de configuração pela equipe de suporte de rede e de sistemas.

No exemplo citado, se um sintoma for resposta muito lenta de um servidor, uma lista de causas poderia incluir desde host mal configurado, sobrecarregado, serviço configurado de forma inadequada, problemas de roteamento ou filtragem de pacotes no meio do caminho e até interface de rede com defeito no lado do cliente. Esses últimos motivos poderiam ocasionar perda intermitente de pacotes, levando à necessidade de retransmissão por ocorrência de timeout no protocolo de transporte ou mesmo na aplicação.

Assim, para avançar no processo de diagnóstico, o passo seguinte, conforme ilustrado na figura 2.2, consiste em obter fatos e isso implica em buscar informações mais detalhadas sobre os sintomas e equipamentos e sistemas envolvidos. Essas informações podem ser conseguidas mediante o uso de:

- Uso de comandos de inspeção, como *ping* e *traceroute*, além de outros para testar conectividade ao longo da rota;

- Analise do tráfego na rede tanto no segmento ao qual o servidor está ligado quanto no lado do cliente. Essa análise demanda a captura de tráfego, que pode ser feita no próprio servidor, usando comandos como *traceroute*, ou com a ajuda de analisadores de rede que capturam todo o tráfego e possibilitam diferentes modos de agregação e filtragem, permitindo exibir pacotes por endereço originador, destinatário, protocolo etc., bem como compor relatórios totalizando o tráfego por diferentes critérios, ou ainda mostrar sequências de pacotes intercambiados por determinados pares de endereço com o uso de protocolos específicos, entre outras técnicas;
- Análise do registro de logs do servidor para complementar a inspeção de tráfego total em todos os acessos ao servidor.

Essa informação mais rica e detalhada permitirá conhecer a situação da rede, a partir do tráfego capturado e analisado. Isso possibilitará estreitar o escopo da busca, eliminando classes de problemas. No caso da resposta lenta do servidor no exemplo anterior, a presença de tráfego intenso em algum trecho da rede no caminho entre o usuário e o servidor poderia ser constatada e seria uma possível causa. Se o problema relatado fosse total falta de acesso ao servidor, a detecção de algum tráfego na rede, proveniente do endereço do servidor, permitiria eliminar causas intermediárias, tal como roteamento incorreto ou enlace intermediário inoperante.

Depois de obter as informações que permitem conhecer a situação, o passo seguinte consiste em criar um plano de ação para tratar os problemas potenciais que poderiam ocasionar os sintomas percebidos. Não é tarefa fácil identificar as causas possíveis. Isso demanda conhecimento técnico que permita analisar se os sintomas percebidos e os dados obtidos da monitoração são os que um sistema operando de condições normais deveria exibir. Mas também demanda conhecimento heurístico, derivado das experiências anteriores de diagnóstico de correção de falhas na rede. Possíveis fontes de apoio e referências nessa fase são:

- **Manuais:** explicam o funcionamento esperado e indicam possíveis causas para problemas comuns;
- **Tabelas de troubleshooting:** associam sintomas a causas prováveis;
- **Experiência anterior:** aponta causas de sintomas similares em situações anteriores;
- **Sistemas especialistas:** correlacionam causas e sintomas tipicamente relacionados, e até podem sugerir ações corretivas possíveis.

O plano de ação, baseado no conjunto de possibilidades identificadas, busca eliminar uma ou mais possíveis causas para os sintomas, mediante observação e análise da rede, com o tráfego normal e com tráfego de teste de cada uma das hipóteses a investigar. O plano de ação deve limitar-se a investigar uma variável de cada vez, pois com muitas alterações sendo realizadas concomitantemente, pode ocorrer um retorno da rede ao estado operacional sem que seja possível perceber qual ou quais foram as medidas que solucionaram o problema. Assim, o recomendável é implementar a alteração/correção planejada e então observar, testar e avaliar o resultado. A observação/teste envolve o uso de comandos de inspeção de configuração, de status de equipamentos e monitoração da rede.

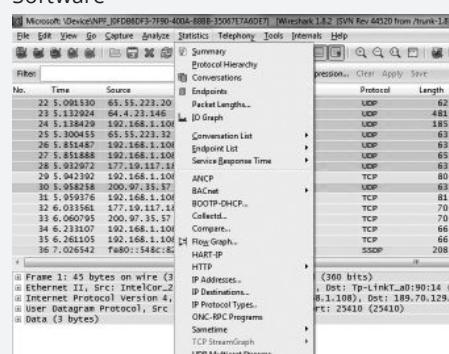
Se o problema desapareceu, então a situação está resolvida e resta apenas fechar o registro de ocorrência registrando a causa efetivamente diagnosticada e a forma de resolver o problema. Isso ampliará a base de conhecimento sobre problemas de rede e suas causas, facilitando investigações de futuros problemas tanto pela mesma equipe como por outras equipes que vierem a substituir os que atuaram na investigação.

Se o problema não pode ser resolvido com a estratégia seguida, o plano de ação deve ser refinado com base nos resultado e um novo plano de ação deve ser definido e implementado, reiteradamente, até que o problema seja solucionado.

## Ferramentas para inspeção e monitoração

Diversas ferramentas para inspeção e monitoração da rede podem ser utilizadas:

- Analisadores a nível físico, como cable scanners, que auxiliam a avaliar as condições de transferência de dados da rede e permitem verificar se existe alguma atenuação ou mesmo rompimento do meio de transmissão (metálico ou ótico);
- Analisadores de protocolos capazes de capturar, registrar, analisar e reconhecer determinados padrões de troca de mensagens dos protocolos mais comuns nas redes, tanto nas camadas de enlace, de rede, de transporte como de aplicação.

Analisadores do meio físico	Analisadores de rede e protocolos
<p><b>Cable Scanner:</b></p> <ul style="list-style-type: none"> <li>■ <b>Time Domain Reflectometer (TDR):</b> Um TDR opera transmitindo um sinal no cabo. Interrupções e outros problemas provocam uma reflexão do sinal com características específicas. O TDR mede quanto tempo leva para o sinal refletido retornar, e com isso calcula a distância até uma avaria no cabo. TDR também pode ser usado para medir o comprimento de um cabo;</li> <li>■ <b>Optical Time Domain Reflectometer (OTDR):</b> pode medir o comprimento da fibra, localizar rompimentos, atenuação anormal etc.</li> </ul> 	<p><b>Hardware e software</b></p>  <p><b>Software</b></p> 

**Figura 2.3**  
Ferramentas de análise de rede.

Existem diversas soluções comerciais em todas as categorias de analisadores, mas um produto tem atraído a atenção e interesse da comunidade de rede pelo fato de ser oferecido como software livre, distribuído com uma licença General Public License (GNU) versão 2. Trata-se do Wireshark, um software para monitoração de rede e análise de protocolo.

### Wireshark

O Wireshark é um programa que captura e analisa o tráfego de rede, permitindo diversas formas de organização ou apresentação do que foi capturado. As funcionalidades do Wireshark são parecidas com o tcpdump (abordado posteriormente), mas com interface gráfica, mais informação e com ampla variedade de filtros que podem tanto limitar a captura como ajustar o que é apresentado, para facilitar a análise.

O desenvolvimento desse software começou em 1998, com o projeto que inicialmente foi denominado Ethereal. Após a migração do principal autor, Gerald Combs, para outra empresa, o desenvolvimento prosseguiu, com novo nome para o software. Atualmente conta com a colaboração de uma imensa comunidade de desenvolvedores que tanto contribuem para o desenvolvimento em si (em modalidade de software livre), como agregam ferramentas comerciais adicionais (Wireshark 2012).

O Wireshark executa na maioria das plataformas, incluindo Windows, OS X, Linux e UNIX, e sua marca registrada pertence à Wireshark Foundation.

Alguns produtos comerciais, desenvolvidos a partir do Wireshark ou que o incluem, estão também disponíveis no mercado. A CACE Technologies oferece um adaptador AirPcap, que possibilita captura de tráfego 802.11 a/b/g/n e sua inspeção com o Wireshark. Adicionalmente, essa empresa vende um produto denominado CACE Pilot, para avaliar tendência do tráfego a longo prazo. A Cisco embutiu o Wireshark na linha de switch Nexus 7000.

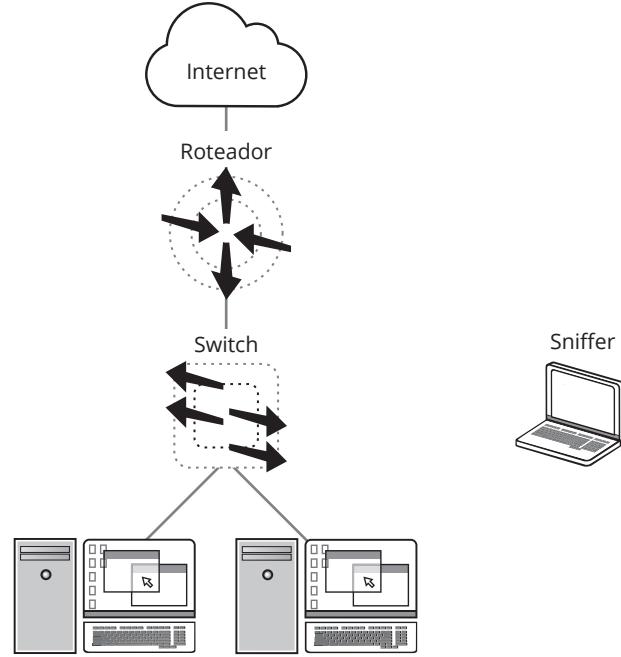
### Conhecendo o Wireshark

O funcionamento do Wireshark, analogamente a qualquer outro analisador de rede, envolve três passos: coletar, converter e analisar.

- Coletar

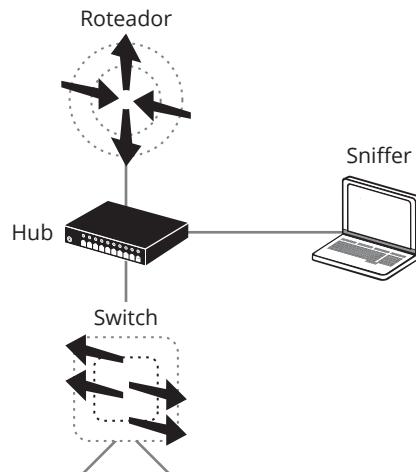
Nesse passo, o equipamento onde o software é executado opera com sua interface em modo promíscuo (modo em que captura todo o tráfego percebido na rede, mesmo que o endereço do destinatário não seja o dessa máquina). Assim, todo o tráfego sendo transmitido naquele segmento de rede é percebido e coletado pelo software. Esse processo é também referido como “sniffer”, e o equipamento com o software de captura também costuma ser designado como “sniffer”.

Nesta etapa, é preciso escolher apropriadamente o local na rede onde o sniffer vai ser colocado. Supondo uma topologia simples de rede, tal como a ilustrada na figura 2.4, existiriam várias alternativas para conectar o sniffer, dependendo dos dados que se deseja coletar.



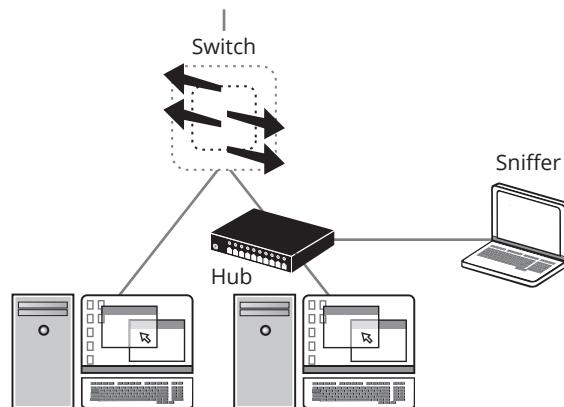
**Figura 2.4**  
Locais onde colocar o sniffer para capturar o tráfego da rede.

Se o objetivo fosse capturar e analisar o tráfego que flui entre o roteador de borda da rede e a internet, uma solução seria incluir um hub entre o roteador e o switch (tal como ilustrado na figura 2.5) interligando nesse hub o sniffer, que poderia então capturar todos os pacotes, fluindo entre eles.



**Figura 2.5**  
Hub para integrar o sniffer no caminho de uma interligação a investigar.

Se o sniffer fosse interligado em uma das portas do switch, a segmentação de tráfego que naturalmente é feita pelo switch impediria o sniffer de capturar e inspecionar o tráfego fluindo entre um dos equipamentos, mostrados na figura 2.4, e o roteador. Para conseguir capturar e inspecionar esse tráfego, poderia ser repetida a mesma estratégia de incluir um hub no caminho, tal como ilustrado na figura 2.6.



**Figura 2.6**  
Hub para derivar um ponto de captura em uma porta do switch.

Alguns modelos de switch permitem duplicar o tráfego de um ou mais portas disponibilizando uma cópia do tráfego em alguma outra porta. Isso costuma ser designado como “port mirroring” ou espelhamento de porta. Nesse caso, não seria necessário o uso do hub para permitir ao sniffer capturar o tráfego da porta desejada. Bastaria configurar o switch para fazer o espelhamento do tráfego e conectar o sniffer na porta designada para receber a cópia do tráfego.

Esse mecanismo de capturar tráfego também pode existir em alguns equipamentos que contenham um agente RMON, o que permitirá coletar dados sobre o tráfego e mesmo capturar seletivamente pacotes, enviado o resultado da captura e análise do tráfego para uma estação gerenciadora da rede, mediante o uso do protocolo SNMP. Maior detalhamento do agente RMON será apresentado em capítulo subsequente.

■ Converter

Depois de capturar o tráfego, o sniffer atua convertendo os dados binários capturados, de modo que sejam mais facilmente interpretáveis.

Os protocolos usados são reconhecidos e os dados contidos em cada segmento capturado são identificados e apresentados com os respectivos cabeçalhos, tal como ilustrado na figura 2.7.

No.	Time	Source	Destination	Protocol	Length	community
13	1.640103	tigre.local	www.rnp.br	ICMP	74	
Frame 13: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)						
Ethernet II, Src: IntelCor_25:f1:4e (00:24:d6:25:f1:4e), Dst: Tp-LinkT_a0:90:14 (f8:d1:11:a0:90:14)						
Internet Protocol Version 4, Src: tigre.local (192.168.1.103), Dst: www.rnp.br (200.130.35.4)						
Version: 4						
Header length: 20 bytes						
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))						
Total Length: 60						
Identification: 0x32c4 (12996)						
Flags: 0x00						
Fragment offset: 0						
Time to live: 128						
Protocol: ICMP (1)						
Header checksum: 0x5a67 [correct]						
Source: tigre.local (192.168.1.103)						
Destination: www.rnp.br (200.130.35.4)						
[Source GeoIP: Unknown]						
[Destination GeoIP: unknown]						
Internet Control Message Protocol						
Type: 8 (Echo (ping) request)						
Code: 0						
Checksum: 0x4d56 [correct]						
Identifier (BE): 1 (0x0001)						
Identifier (LE): 256 (0x0100)						
Sequence number (BE): 5 (0x0005)						
Sequence number (LE): 1280 (0x0500)						
[Response In: 14]						
Data (32 bytes)						
Data: 6162636465666768696a6b6c6d6e6f707172737475767761...						
[Length: 32]						

O nível de detalhamento desejado pode variar, e é possível selecionar interativamente quais os campos de cada cabeçalho devem ser apresentados. O pacote exibido na figura 2.7 ilustra uma interação do protocolo ICMP que ocorre quando um teste de ping é comandado. Nesse caso, podemos observar que a máquina cujo endereço IP é 192.168.1.103 (endereço privativo) envia um pacote ICMP do tipo Echo request para a máquina 200.130.35.4, que no momento do teste foi o endereço <http://www.rnp.br>. A tradução de número IP com o nome e domínio do endereço envolvido no teste pode ser feita pelo software Wireshark porque ele permite configurar se a resolução de nomes deve ou não ser feita e, em caso positivo, ele consulta um servidor de DNS para obter o nome associado ao número IP que apareceu no cabeçalho do pacote capturado.

■ Analisar os dados capturados é o terceiro e mais importante passo do processo.

O software trabalha sobre os dados capturados, já decodificados, e permite inspecionar os cabeçalhos e demais informações, tendo em vista os protocolos presentes em cada camada.

**Figura 2.7**  
Decodificação de protocolos feita pelo Wireshark.

Camada	Protocolos
Aplicação	HTTP, SMTP, FTP, SNMP
Transporte	TCP, UDP
Rede	IP, ICMP, ARP, RIP, OSPF
Enlace	Ethernet

**Figura 2.8**  
Protocolos usuais na internet e camadas em que atuam.

A análise pode ser feita considerando um único pacote capturado ou sequências de pacotes. O tráfego pode ser classificado em categorias:

- **Broadcast:** tráfego que é dirigido a todos os dispositivos na rede;
- **Multicast:** tráfego que é dirigido a um endereço destinatário que representa um grupo. Equipamentos integrantes do grupo devem ter previamente dirigido um pedido de ingresso no grupo multicast. Essa informação fica registrada em roteadores que podem receber um fluxo de pacotes e replicar esse fluxo por vários caminhos onde existam receptores registrados;
- **Unicast:** tráfego de um computador particular para outro computador particular.

### Instalando o Wireshark

O software Wireshark pode ser encontrado no site <http://www.wireshark.org> e, seguindo a opção de download, poderá ser encontrada a última versão estável e também versões mais recentes, mas que ainda estão em desenvolvimento. O software poderá ser obtido em versão preparada para ser instalada em diversos ambientes, ou mesmo em código-fonte, tal como referido por Chapell (2010). Existem versões para diversos Sistemas Operacionais: Windows, Apple Mac OS X, Debian GNU /Linux, rPath Linux, Sun Solaris/i386, Sun Solaris/ Sparc e Ubuntu. As versões de instaladores que podem ser obtidas no site do Wireshark incluem:

- Windows Installer (64-bit);
- Windows Installer (32-bit);
- Windows U3 (32-bit);
- Windows PortableApps (32-bit);
- OS X 10.6 and later Intel 64-bit .dmg;
- OS X 10.5 and later Intel 32-bit .dmg;
- OS X 10.5 and later PPC 32-bit .dmg;
- Source Code.

O software Wireshark também pode ser instalado em máquina virtual. A instalação é feita a partir da execução do aplicativo baixado do site do Wireshark. No caso de um ambiente Windows (64 bits), por exemplo, o aplicativo Wireshark-win64-1.10.8.exe efetua a instalação do software que pode começar a ser imediatamente utilizado. Quando o Wireshark é instalado, o acesso à interface de rede (cabeados ou sem fio) é intermediado por rotinas de acesso, tais como: WinCap (ambiente Windows), AirCap (interface para capturar tráfego 802.11) ou libcap (ambiente \*NIX ).

Adicionalmente, o Wireshark pode ser usado para analisar arquivos com tráfego capturado por outros analisadores de protocolos que usam variados formatos, como:

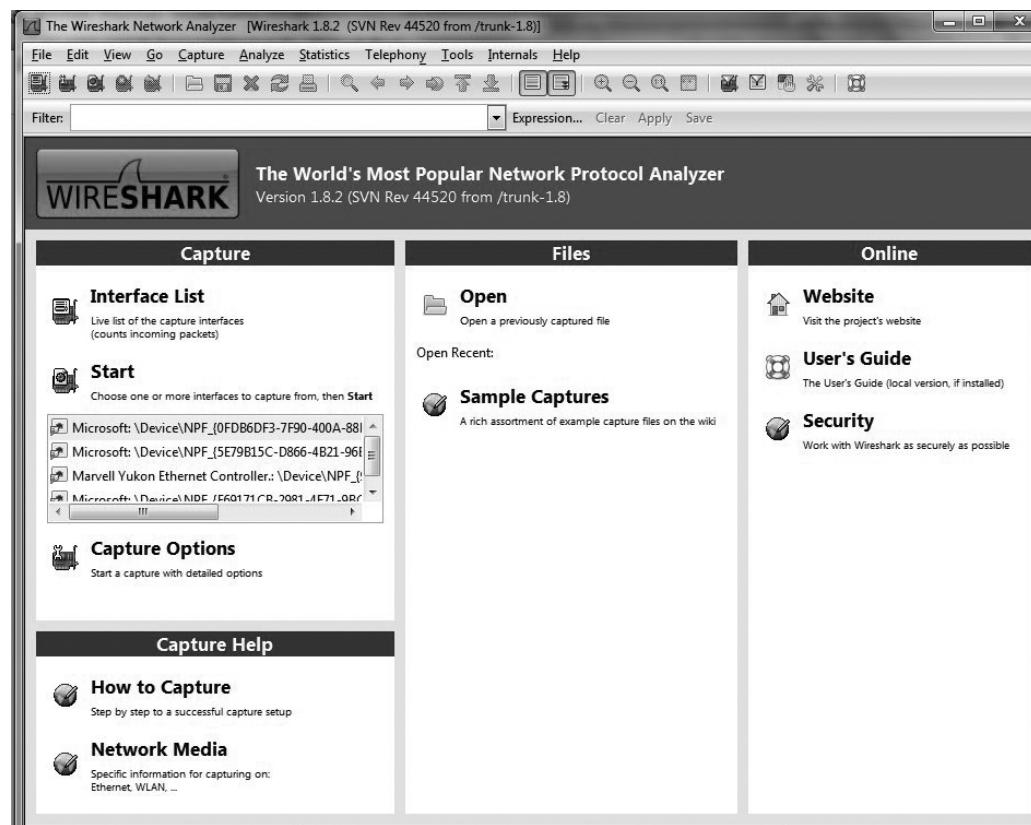
- libpcap: captures from Wireshark/TShark/dumpcap, tcpdump, and various other tools using libpcap's/tcpdump's capture format;
- pcap-ng: "next-generation" successor to libpcap format;
- Microsoft Network Monitor captures;
- AIX's iptrace captures;
- Cinco Networks NetXRay captures;
- Network Associates Windows-based Sniffer captures;
- Network General/Network Associates DOS-based Sniffer (compressed or uncompressed) captures;



- ▣ WAN/LAN analyzer captures;
- ▣ Network Instruments Observer version 9 captures;
- ▣ files from HP-UX's nettle;
- ▣ the output from VMS's TCPIPtrace/TCPtrace/UCX\$TRACE utilities;
- ▣ Linux Bluez Bluetooth stack hcidump -w traces;
- ▣ Juniper Netscreen snoop files;
- ▣ Symbian OS btsnoop files;
- ▣ Apple PacketLogger files;
- ▣ Files from Aethra Telecommunications' PC108 software for their test instruments.

## Utilizando o Wireshark

Quando o Wireshark é executado, a tela inicial exibida na figura 2.9 é apresentada.



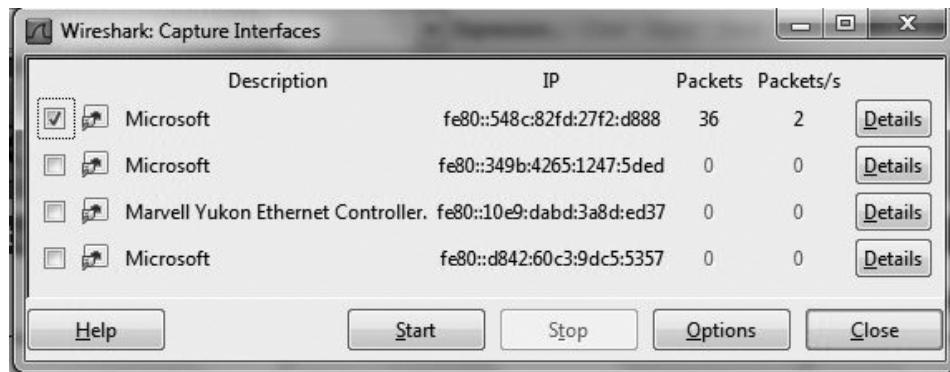
Essa página de abertura tem quatro seções:

- ▣ **Área de captura (Capture)**: configuração do processo de captura;
- ▣ **Área de arquivos (Files)**: permite selecionar e abrir arquivos com capturas previamente realizadas, para análise no Wireshark;
- ▣ **Área online (Online)**: permite acessar material de ajuda que está online na rede;
- ▣ **Área de ajuda para a captura (Capture Help)**: contém orientações e instruções para realizar a captura.

**Figura 2.9**  
Tela inicial do Wireshark.

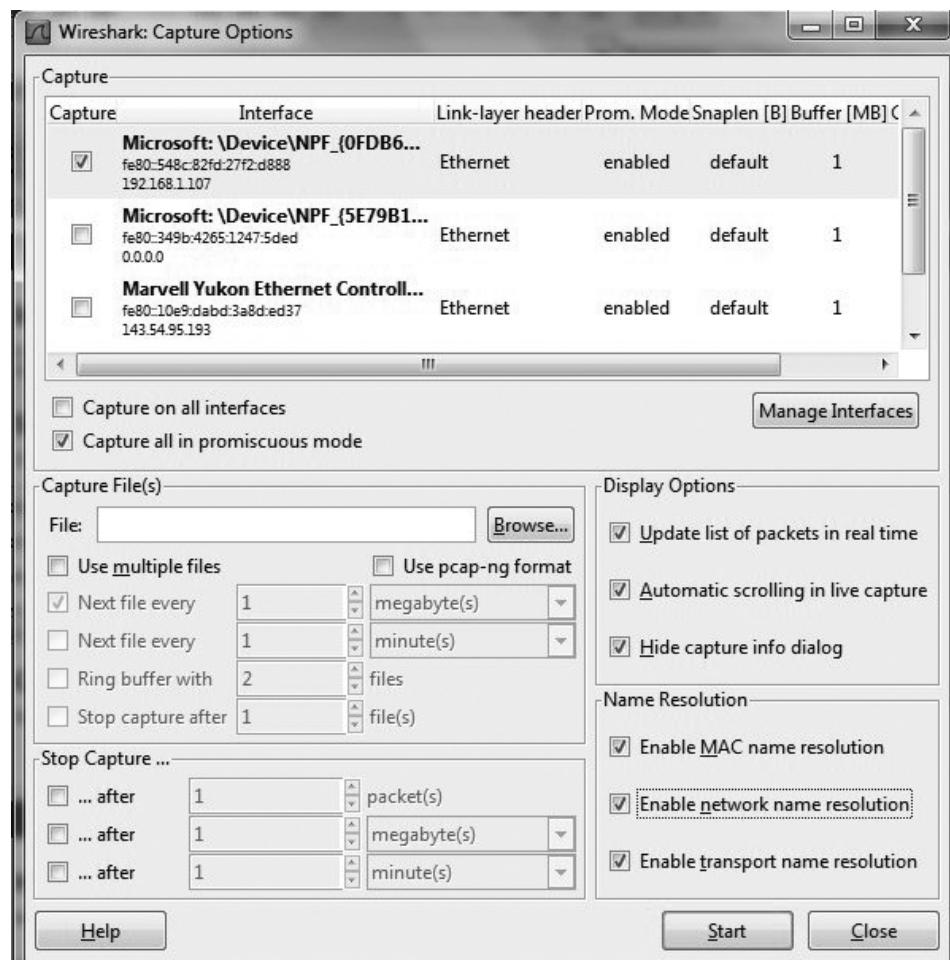
A área de captura apresenta uma lista de interfaces ativas reconhecidas pelo Wireshark.

Clicando em "Interface List", a janela tal como mostrada na figura 2.10, é exibida. Para iniciar a captura por alguma das interfaces listadas, basta selecioná-lo e depois clicar no botão "Start".



**Figura 2.10**  
Interfaces ativas reconhecidas pelo Wireshark.

O botão “Options” permite configurar aspectos específicos da captura, tal como ilustrado na figura 2.11.

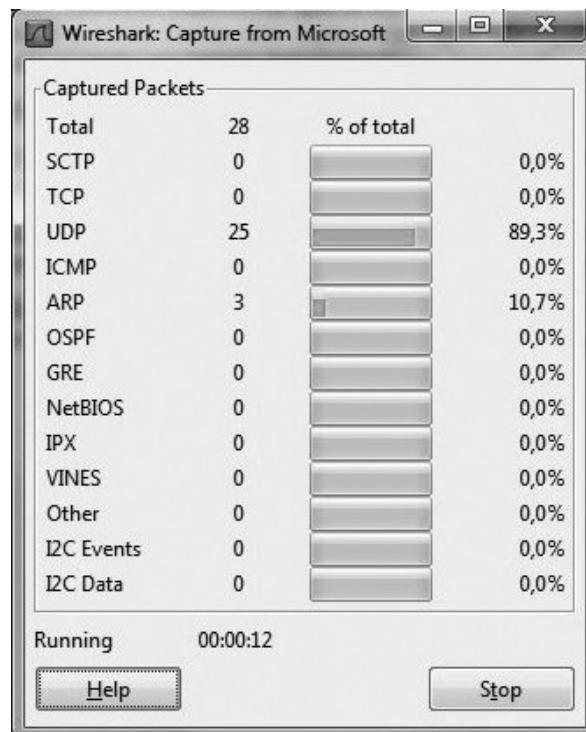


**Figura 2.11**  
Configurando opções de captura no Wireshark.

Se houver dúvidas sobre o que escolher, uma sugestão é deixar as marcações de opções no modo como são encontradas (modo default). Mas também é possível ir ajustando as opções para que o processo de captura e o resultado melhor atendam a necessidade de quem está investigando um problema na rede. As opções apresentadas são explicadas a seguir:

- ▣ **Capture on all interfaces:** melhor deixar desmarcado, pois o normal é capturar tráfego apenas em uma interface, a menos que seja necessário combinar a captura de dois segmentos diferentes da rede;

- ▣ **Capture all in promiscuous mode:** se o interesse na captura for inspecionar o tráfego de outras máquinas usando algumas das configurações mostradas nas figuras 2.5 e 2.6 ou mesmo port mirroring, então deve ser marcada a opção captura em modo promíscuo. Mas se o interesse da captura for analisar o tráfego enviado e recebido pela própria máquina na qual o Wireshark está sendo executado, então essa caixa de opção pode ficar desmarcada;
- ▣ **Capture File(s):** indicando um diretório e nome de arquivo, os dados capturados serão armazenados naquele lugar, além se serem exibidos na janela de exibição. Múltiplos arquivos podem ser usados e reusados de forma circular ou sequencial, com o próximo arquivo começando a ser utilizado em função de tempo decorrido (minutos) ou área ocupada (megabytes);
- ▣ **Stop Capture:** permite especificar que a captura deva ser encerrada após uma determinada quantidade de pacotes terem sido capturados, após uma quantidade de megabytes terem sido coletados ou após um tempo determinado ter decorrido;
- ▣ **Display Options:**
  - ▣ **Update list of packets in real time:** promove a exibição dos pacotes que estão sendo capturados;
  - ▣ **Automatic scrolling in live capture:** define se haverá rolamento de tela na medida em que os pacotes estão sendo capturados e exibidos;
  - ▣ **Hide capture info dialog:** ativa uma pequena janela que mostra um resumo do que está sendo capturado, tal como ilustrado na figura 2.12.

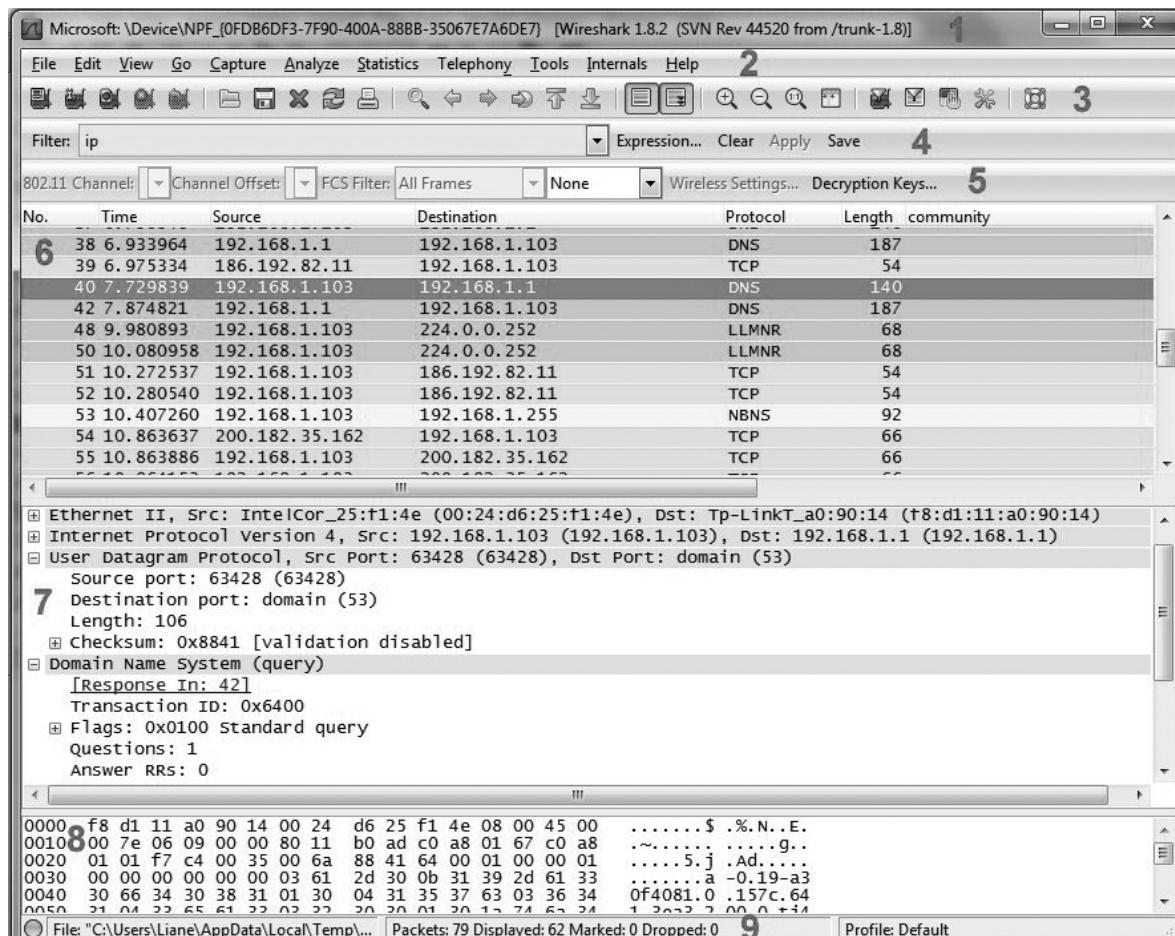


**Figura 2.12**  
Janela de informações sobre a captura.

- ▣ **Name Resolution:**
  - ▣ **Enable MAC name resolution:** se desmarcado, exibe apenas os caracteres hexadecimais do endereço MAC. Se essa opção for marcada, coloca informação sobre o fabricante da interface, quando possível;

- **Enable network name resolution:** se desmarcado, exibe apenas o endereço IP. Se essa opção é marcada, é consultado o serviço de DNS para exibir o nome host em vez do endereço IP. Essa opção pode ocasionar perda de captura de pacotes, pois requer que o Wireshark consuma processamento e tempo para realizar a resolução de nomes;
- **Enable transport name resolution:** se essa opção é marcada, será exibida a aplicação associada com a porta utilizada pela camada de transporte.

O botão de “Start” permite iniciar a captura de pacotes com as opções configuradas na janela de opções. Depois que um arquivo com pacotes capturados previamente é aberto ou uma nova captura é iniciada, a tela principal do Wireshark é exibida, tal como ilustrado na figura 2.13.



**Figura 2.13** As áreas assinaladas nessa tela são:  
Tela principal do Wireshark.

1. Título;
2. Menu (texto);
3. Barra de ferramentas principal;
4. Barra de ferramentas de filtragem;
5. Barra de ferramentas de rede sem fio;
6. Painel com a lista de pacotes;
7. Painel com os detalhes dos pacotes;
8. Painel com os bytes dos pacotes;
9. Barra de status.

## Título

O título pode ser configurado usando a opção de editar o layout em "Edit"/ "Preferences"/ "Layout".

## Menu

O menu localizado logo a seguir do título apresenta todas as funcionalidades de trabalho com o Wireshark, onde podemos destacar como principais:

- **File:** permite abrir ou salvar um arquivo de captura para análise posterior, além de exportar determinados pacotes;
- **Edit:** nesse submenu, é possível marcar pacotes para posterior análise, sinalizar um novo pacote como sendo o início da captura ou alterar as preferências de visualização da ferramenta;
- **View:** permite alterar as opções de visualização da ferramenta, com destaque para a opção **Coloring rules**, que possibilita trocar as cores de visualização dos pacotes segundo as regras determinadas nessa opção;
- **Go:** opções de navegação entre os pacotes. Observe as teclas de atalho informadas nesse submenu, para facilitar navegação;
- **Capture:** refere-se às opções de capturas. Essas funcionalidades estão representadas pelos primeiros ícones da barra de ferramentas principal;
- **Analyze:**
  - **Display Filters...**: permite a realização de filtros específicos, abordado posteriormente na barra de ferramenta de filtragem;
  - **Enabled Protocols:** seleção de quais protocolos serão apresentados para visualização no Wireshark;
  - **Follow TCPStream:** apresenta o conteúdo dos pacotes que fazem parte do mesmo fluxo (conversa) do pacote selecionado. Por exemplo, se o pacote selecionado é referente a uma comunicação HTTP, nessa opção será possível visualizar o conteúdo do arquivo HTML na íntegra.
- **Statistics:** visualização de estatísticas, abordado posteriormente;
- **Telephony:** permite a realização de filtros sobre protocolos específicos de telefonia ou VoIP.
- **Tools:**
  - **Firewall ACLs rules:** auxilia na criação de regras de firewall, gerando automaticamente regras nas principais plataformas existentes, como CISCO, IPTables, Windows netsh e outras;
  - **LUA:** API de programação para o Wireshark.

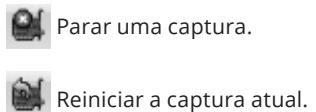
## Barra de ferramentas principal

A barra de ferramentas principal apresenta as principais ferramentas de trabalho. Nos cinco primeiros ícones estão as opções referentes ao controle de capturas de pacotes:

 Visualizar a lista de interfaces para realizar uma nova captura, conforme a figura 2.10.

 Alterar as opções de captura, conforme a figura 2.11.

 Iniciar uma nova captura.



### Barra de ferramentas de filtragem

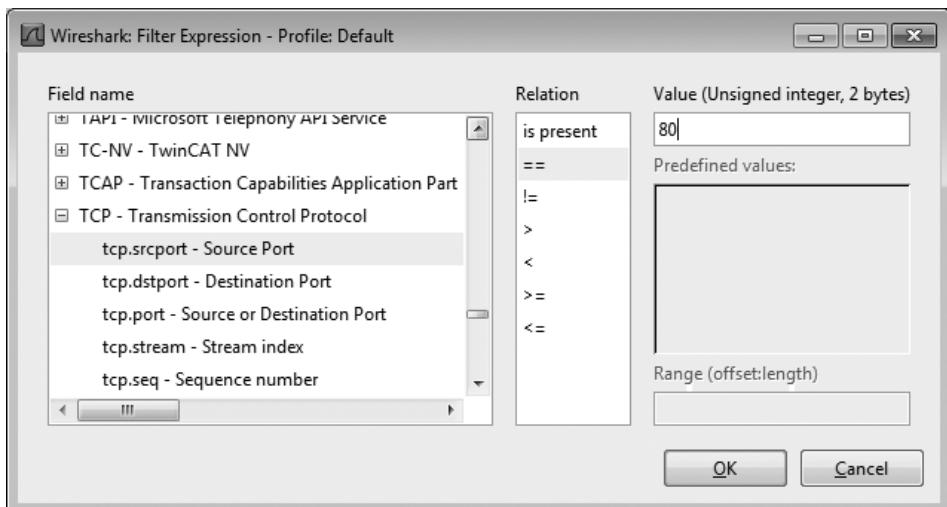
Após a realização de uma captura, normalmente precisamos realizar filtros de visualização para selecionar apenas as informações pertinentes ao nosso objetivo. Esse filtro pode ser realizado por qualquer protocolo ou informação capturada. Como demonstrado na figura 2.14, os filtros podem ser realizados diretamente na barra de ferramentas.

**Figura 2.14**  
Barra de ferramentas de filtragem.



Contudo, para facilitar a criação dos filtros, o Wireshark disponibiliza uma interface de geração de filtros, que pode ser acessada pela opção “Expression...”. Nessa opção, conforme a figura 2.15, é apresentada uma relação de todos os protocolos conhecidos pelo Wireshark e o conteúdo de seus cabeçalhos.

**Figura 2.15**  
geração de filtros  
Wireshark.



Para mais informações e exemplos sobre filtros de visualização no Wireshark, acesse <http://wiki.wireshark.org/DisplayFilters>.

No exemplo da figura 2.15, está sendo realizado um filtro por todos os pacotes onde a porta TCP de origem(tcp.srcport) seja igual a 80.

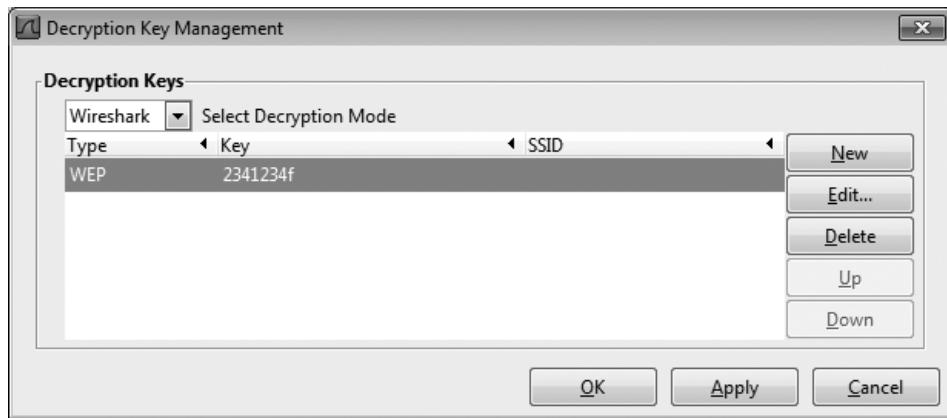
**Figura 2.16**  
Filtro composto/  
Filtro com erro  
de sintaxe.

Na barra de ferramenta de filtragem, também é possível realizar filtros compostos através dos operadores E (&&) e OU ( || ), como no exemplo da figura 2.16. Caso ocorra algum erro de sintaxe no filtro, o Wireshark altera a cor da pesquisa.



### Barra de ferramentas rede sem fio

Nessa barra é possível cadastrar a chave de descriptografia da rede wireless a qual você quer monitorar, usando para isso a opção “Description Keys...” (figura 2.17). Isso é necessário quando o WinPcap (software que realiza a captura dos pacotes no Wireshark) não possui suporte ao adaptador de rede wireless utilizado.



**Figura 2.17**  
Cadastro de chaves descriptográficas.

Vale ressaltar que na rede wireless é possível capturar apenas os pacotes encaminhados para a estação onde está executando o sniffer. Para capturar todos os pacotes em modo promíscuo, é necessário adquirir a interface wireless tal como o AirPcap.



**Figura 2.18**  
Aircap.

### Painel com a lista de pacotes

Nesse painel, são mostrados todos os pacotes em ordem de captura. Observe que a coluna Time apresenta o tempo em segundos do pacote capturado em relação ao primeiro pacote. Source e Destination apresentam o endereço de origem e destino do pacote, podendo ser um endereço MAC ou endereço IP. Protocol apresenta o protocolo de mais alto nível que existe no pacote, Length apresenta o tamanho do pacote e Info é uma interpretação do Wireshark de informações relevantes que o pacote contenha ou represente.

### Painel com detalhes do pacote

Nesse painel, é possível visualizar todas as informações contidas nos cabeçalhos de todas as camadas na qual o pacote pertence. Na figura 2.18 temos o exemplo de um pacote de requisição HTTP, onde através das áreas grifadas é possível observar o endereço MAC de origem e destino, o endereço IP de origem e destino, o conteúdo da camada de transporte referente a um conteúdo TCP e informações da camada de aplicação do protocolo HTTP, incluindo o nome do host acessado e o comando enviado, que nesse exemplo foi um GET.

No.	Time	Source	Destination	Protocol	Length	Info
111	6.11815600	10.10.1.13	143.54.1.20	HTTP	403	GET / HTTP/1.1
122	6.29536200	10.10.1.13	143.54.1.20	HTTP	401	GET /book.gif HTTP/1.1
Frame 111: 403 bytes on wire (3224 bits), 403 bytes captured (3224 bits) on interface 0						
Ethernet II, Src: Pegatron_b5:06:9c (70:71:bc:b5:06:9c), Dst: ThomsonT_8b:2f:70 (08:76:ff:8b:2f:70)						
<b>Destination: ThomsonT_8b:2f:70 [08:76:ff:8b:2f:70]</b>						
<b>Source: Pegatron_b5:06:9c [70:71:bc:b5:06:9c]</b>						
Type: IP (0x0800)						
Internet Protocol Version 4, Src: 10.10.1.13 (10.10.1.13), Dst: 143.54.1.20 (143.54.1.20)						
Version: 4						
Header length: 20 bytes						
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))						
Total Length: 389						
Identification: 0x533a (21306)						
Flags: 0x02 (Don't Fragment)						
Fragment offset: 0						
Time to live: 128						
Protocol: TCP (6)						
Header checksum: 0x0ad8 [correct]						
Source: 10.10.1.13 [10.10.1.13]						
Destination: 143.54.1.20 [143.54.1.20]						
[Source GeoIP: Unknown]						
[Destination GeoIP: Unknown]						
Transmission Control Protocol, Src Port: 54994 (54994), Dst Port: http (80), Seq: 1, Ack: 1, Len: 349						
Source port: 54994 [54994]						
Destination port: http (80)						
[Stream index: 3]						
Sequence number: 1 (relative sequence number)						
[Next sequence number: 350 (relative sequence number)]						
Acknowledgment number: 1 (relative ack number)						
Header Length: 20 bytes						
Flags: 0x018 (PSH, ACK)						
Window size value: 65392						
[calculated window size: 65392]						
[window size scaling factor: -2 (no window scaling used)]						
Checksum: 0x0ac [validation disabled]						
[SEQ/ACK analysis]						
Hypertext Transfer Protocol						
<b>GET / HTTP/1.1\r\n</b>						
Host: penta.ufrgs.br\r\n						
Connection: keep-alive\r\n						
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n						
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/29.0.1547.66						
Accept-Encoding: gzip,deflate,sdch\r\n						
Accept-Language: pt-BR,pt;q=0.8,en-US;q=0.6,en;q=0.4\r\n						

**Figura 2.19 Painel de bytes do pacote**

Detalhamento  
do pacote.

Nesse último painel da figura 2.13 estão as informações exatamente como foram capturadas pelo WinPcap. Observe que o pacote está sendo demonstrado em dois formatos, à esquerda em hexadecimal e à direita em ASCII. Esse painel é importante para sabermos a posição em que as informações se encontram dentro do pacote para, se necessário, realizar pesquisas sobre elas.

### Barra de status

A barra de status apresenta informações de pacotes capturados, mostrados, marcados e, principalmente, de pacotes perdidos. Isso nos auxilia a descobrir se realmente estamos capturando todos os pacotes recebidos na interface de rede ou necessitamos alterar alguma opção do Wireshark para que isso ocorra.

### Filtros de captura no Wireshark

Conforme abordado no tópico sobre a barra de filtragem do Wireshark, as opções dessa barra são filtros de visualização. Isso significa que os pacotes são capturados em sua totalidade e depois filtrados.

Contudo, dependendo do fluxo de informações que passam pela estação de captura, muitos pacotes podem ser perdidos pela grande quantidade de pacotes recebidos. Também pode ocorrer que o objetivo da captura de pacotes seja a busca de algo específico. Nesses casos, podemos realizar um filtro de captura, o qual será realizado pela biblioteca WinPcap no momento do recebimento dos pacotes na interface de rede, desonerando os recursos de hardware e software para o armazenamento de todos os pacotes capturadas.

Esse filtro deverá ser realizado antes do início da captura, diretamente na tela inicial do Wireshark na opção “Capture Options” ou através do menu “Capture”//“Options” (figura 2.20).

Dentro das opções de captura, selecione a interface na qual será realizada a captura (1) e clique duas vezes. Com isso, a janela de configuração da interface será aberta e a opção de filtros de captura (2) estará disponível.

A escolha de filtros de captura apropriados é importante para limitar a quantidade de pacotes capturados, pois em redes com alto tráfego a quantidade de pacotes é elevada e o fluxo de tráfego contém muitos pacotes que podem não interessar ao processo de diagnósticos em curso. Adicionalmente, quando a quantidade de pacotes capturada é muito grande, o funcionamento do próprio software fica degradado (mais lento).

Outro aspecto a considerar é que pode ocorrer perda de pacotes pela incapacidade do equipamento de captura. Mesmo que uma interface de rede de um equipamento, tal como um notebook, possa ter capacidade de operar em velocidade alta (1 Giga, por exemplo), isso não significa que o equipamento vai conseguir capturar, decodificar e armazenar um fluxo de bits por segundo próximo dessa velocidade. Equipamentos dedicados à captura de tráfego têm capacidade de captura mais elevada.

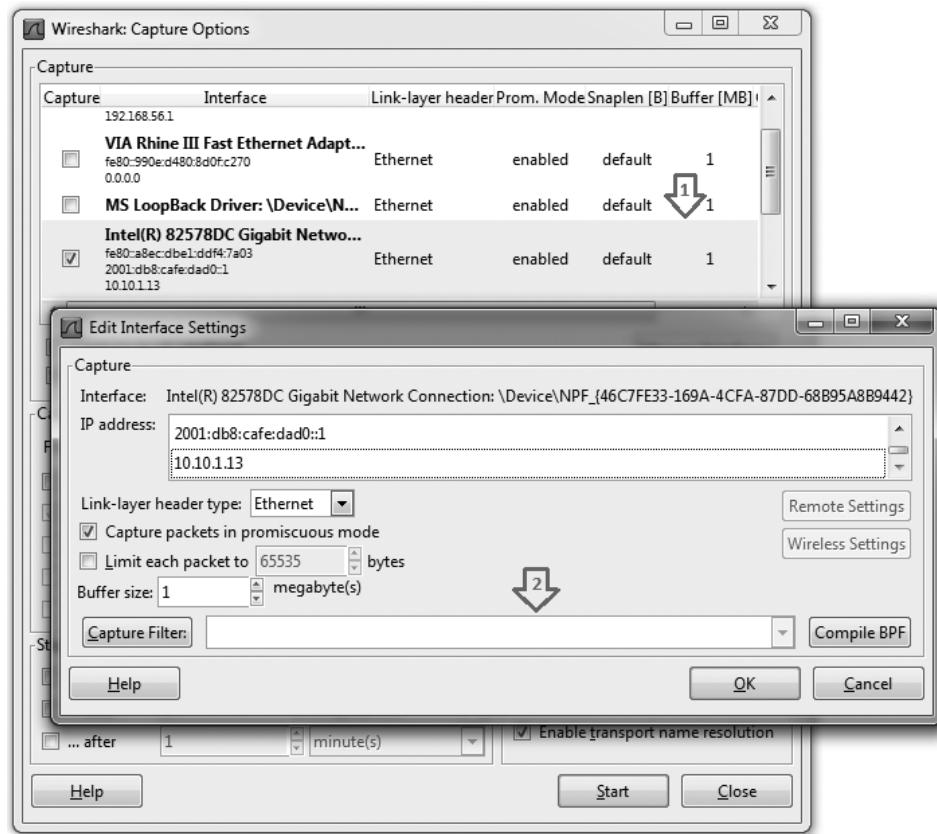


Figura 2.20  
Filtros de captura.

No Wireshark, a sintaxe dos filtros de captura é diferente da sintaxe dos filtros de visualização, pois, como já mencionado, os filtros de captura são realizados pela WinPcap, que utiliza a mesma sintaxe do aplicativo tcpdump, apresentado posteriormente nesse capítulo. Para melhor entendimento, a explicação dos filtros de capturas mais usuais serão apresentados em exemplos:

Filtro	Sintaxe do filtro
Capturar o tráfego de pacotes recebidos e enviados do host 10.10.1.13	host 10.1.1.13
Capturar os pacotes originados da rede 10.1.1.0/24	src net 10.1.1.0/24
Capturar os pacotes que possuem a rede 200.132.0.0/16 como destino e encaminhados para a porta 80	dst net 200.132.0.0/16 and port 80
Capturar todos os pacotes TCP que a porta não seja 80 e 443	tcp and not ( port 80 or port 443)

**Tabela 2.1**

Exemplos de filtros de captura do Wireshark.

### Estatísticas no Wireshark

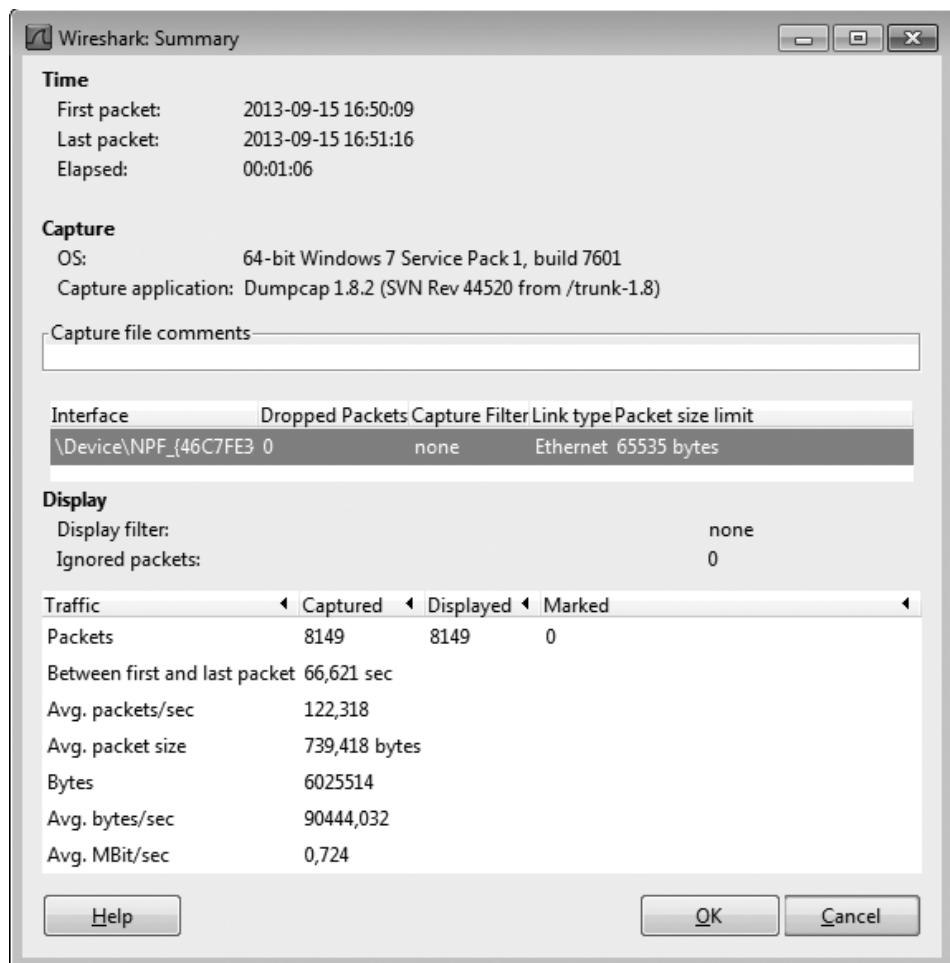
Além de capturar pacotes, o Wireshark é uma poderosa ferramenta estatística, que nos auxilia na análise da rede. Através das estatísticas conseguimos responder perguntas como:

- Quais são os protocolos mais utilizados na minha rede?
- Qual é o tamanho médio de pacotes da rede?
- Quantos % de broadcast há na rede?



Outros exemplos de filtros podem ser acessados em  
<http://wiki.wireshark.org/CaptureFilters>

Através do menu “Statistics”/“Summary”, conforme a figura 2.21, podemos verificar o tempo de captura realizada, a quantidade de pacotes capturados, qual o tamanho em bytes da captura e a média de bytes e pacotes por segundo.



**Figura 2.21**  
Sumarização da captura realizada.

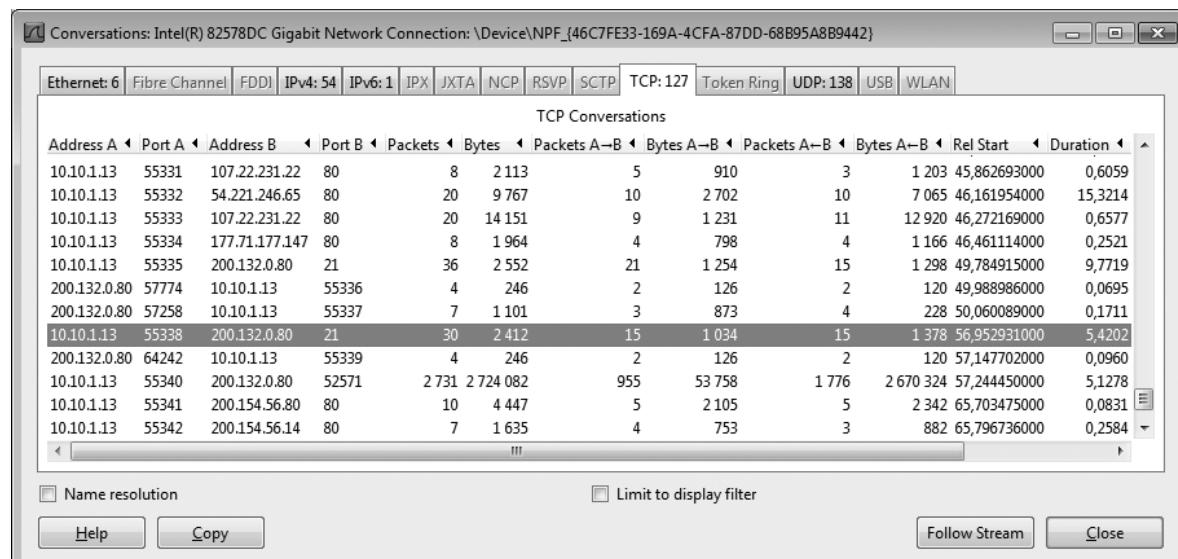
No menu "Statistics"/"Protocol Hierarchy", obtemos a proporção de pacotes por protocolos.

Como no exemplo da figura 2.22, que mostra pacotes IPv6 e pacotes IPv4, sendo em sua maioria TCP e 21,77% transmissão de dados por FTP.

Protocol	% Packets	Packets	% Bytes	Bytes	Mbit/s	End Packets	End Bytes	End Mbit/s
Frame	100,00 %	8149	100,00 %	6025514	0,724	0	0	0,000
Ethernet	100,00 %	8149	100,00 %	6025514	0,724	0	0	0,000
Internet Protocol Version 4	99,75 %	8129	99,83 %	6015509	0,722	0	0	0,000
Transmission Control Protocol	93,89 %	7651	98,68 %	5945953	0,714	4493	2325421	0,279
Secure Sockets Layer	7,82 %	637	8,10 %	488243	0,059	637	488243	0,059
Hypertext Transfer Protocol	8,57 %	698	7,59 %	457421	0,055	362	200651	0,024
File Transfer Protocol (FTP)	0,56 %	46	0,06 %	3818	0,000	46	3818	0,000
FTP Data	21,77 %	1774	44,83 %	2670885	0,321	1774	2670885	0,321
Data	0,04 %	3	0,00 %	165	0,000	3	165	0,000
User Datagram Protocol	4,43 %	361	1,01 %	60918	0,007	0	0	0,000
Hypertext Transfer Protocol	0,86 %	70	0,45 %	27377	0,003	70	27377	0,003
Routing Information Protocol	0,04 %	3	0,00 %	198	0,000	3	198	0,000
Domain Name Service	3,51 %	286	0,55 %	32997	0,004	286	32997	0,004
Dropbox LAN sync Discovery Protocol	0,02 %	2	0,01 %	346	0,000	2	346	0,000
Internet Control Message Protocol	1,44 %	117	0,14 %	8638	0,001	117	8638	0,001
Internet Protocol Version 6	0,22 %	18	0,16 %	9903	0,001	0	0	0,000
User Datagram Protocol	0,22 %	18	0,16 %	9903	0,001	0	0	0,000
Hypertext Transfer Protocol	0,22 %	18	0,16 %	9903	0,001	18	9903	0,001
Address Resolution Protocol	0,02 %	2	0,00 %	102	0,000	2	102	0,000

Através do menu "Statistics"/"Conversations", visualizamos os fluxos de dados captados, a figura 2.23 apresenta 6 fluxos Ethernet, os quais contém 54 fluxos IPv4 e 1 fluxo IPv6, divididos em 127 fluxo TCP e 138 UDP. Nos fluxos TCP o Wireshark apresenta a quantidade de bytes, pacotes e a duração de cada fluxo separado por endereço e porta.

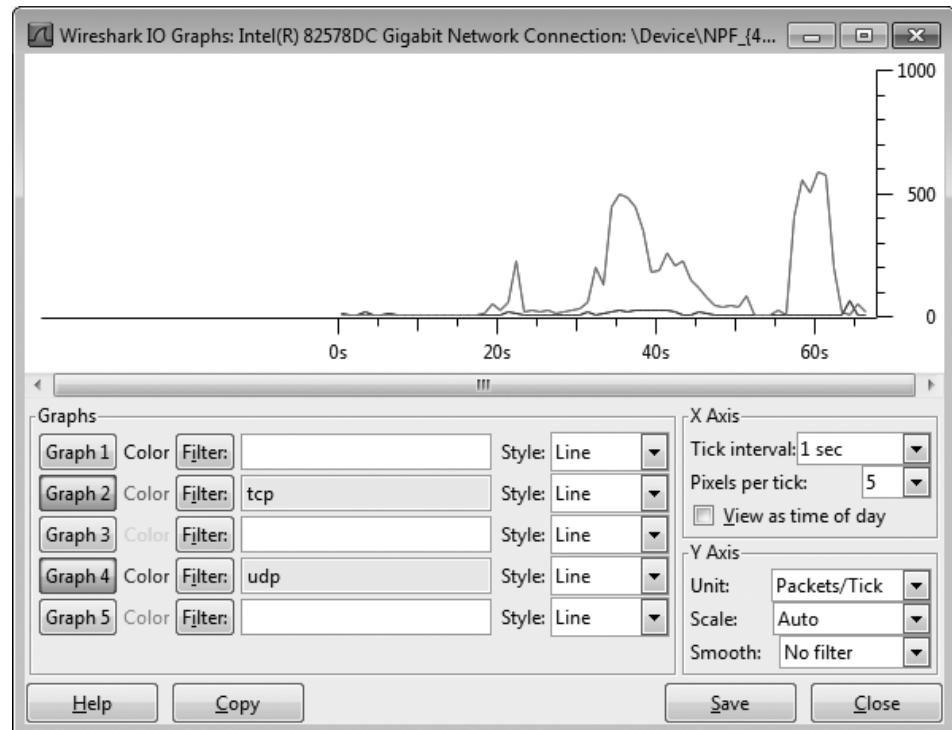
Figura 2.22  
Hierarquia de protocolos.



Gráficos de bytes ou pacotes por período também estão disponíveis no Wireshark através do menu "Statistics"/"IO Graph". No exemplo da figura 2.24, foi gerado um gráfico de total de pacotes por segundo dos protocolos TCP e UDP. Essa funcionalidade permite cinco diferentes filtros no mesmo gráfico.

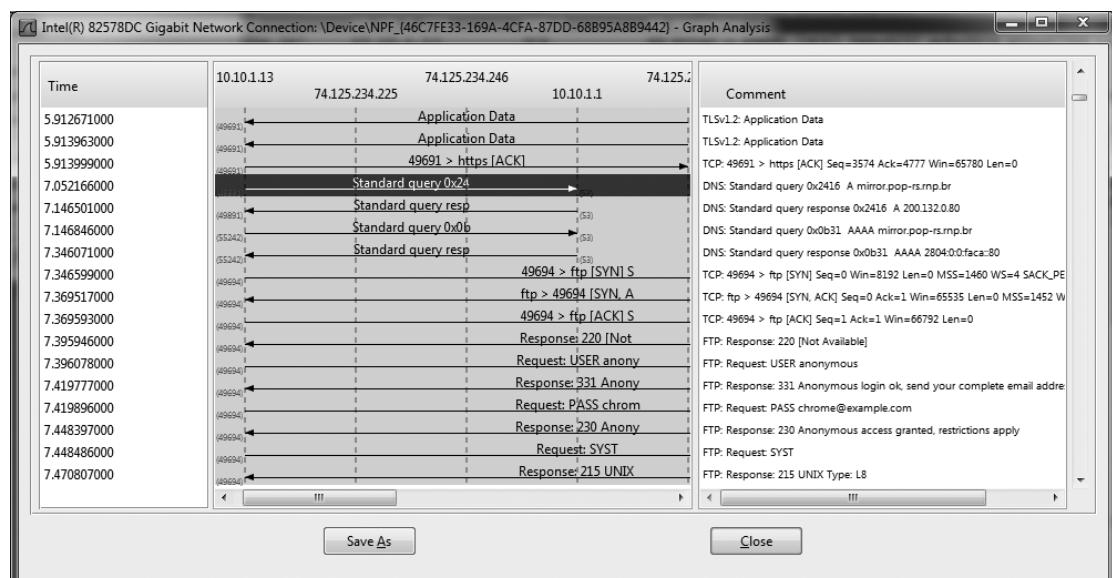
Figura 2.23  
Fluxos de dados.

No endereço [http://www.wireshark.org/docs/wsug\\_html\\_chunked/](http://www.wireshark.org/docs/wsug_html_chunked/) você encontra o manual completo do Wireshark, com todas as suas funcionalidades.



**Figura 2.24**  
Geração de gráficos do Wireshark.

Um segundo tipo de gráfico está disponível na ferramenta. Ele demonstra qual é o fluxo dos pacotes entre origem/destino de uma comunicação. Conforme a figura 2.25, onde encontra o resultado do gráfico gerado pela opção “Statistics”/“Flow Graph”, podemos observar que em um determinado momento o IP 10.10.1.3 realizou uma consulta DNS para o destino 10.10.1.1.



**Figura 2.25**  
O Fluxo da comunicação.

No Wireshark existem mais de 20 tipos de estatísticas, das quais foram abordadas as mais usuais. No endereço [http://www.wireshark.org/docs/wsug\\_html\\_chunked/](http://www.wireshark.org/docs/wsug_html_chunked/) você encontra o manual completo do wireshark com todas as suas funcionalidades.

## TCPDUMP

O tcđump é outra ferramenta de monitoração que está disponível para a maioria dos Sistemas Operacionais derivados do Unix, incluindo MacOS. Nesse aplicativo, que utiliza a LibPcap, as capturas de pacotes são realizadas sem interface gráfica. Sendo assim, os resultados das capturas poderão ser visualizadas no terminal do Sistema Operacional, salvas em formato texto ou no formato LibPcap. Utilizando o formato LibPcap, é possível visualizar a captura no Wireshark.

Formato simplificado:

```
#tcpdump [opções] [filtro]
```

[opções]

- **-i**: determina a interface. Dica: execute o comando *ifconfig* para saber as interfaces de rede existentes;
- **-s**: maior tamanho de pacote a ser capturado;
- **-c**: número de pacotes a serem capturados;
- **-n**: evita a resolução de nomes para os endereços e portas;
- **-v**: modo detalhado, inclui mais informações do pacote. Teste também as opções -vv e -vvv para ter mais detalhes;
- **-w <arquivo>**: nome do arquivo que armazenará a captura no formato LibPcap;
- **-r <arquivo>**: utilize essa opção para visualizar os pacotes de uma captura realizada anteriormente com a opção -r.

[filtro]

Os filtros do TCPDUMP são iguais aos filtros de captura do Wireshark, permitindo filtros por endereço IP, portas, redes de origem destino ou qualquer campo existente no cabeçalho do protocolo a ser capturado. Todas as possibilidades de filtros podem ser acessadas no link <http://www.manpagez.com/man/7/pcap-filter/>

Exemplos:

Capturando todo o tráfego da interface eth2.

```
#tcpdump -i eth2
```

Capturando 500 pacotes com até 1500 bytes e gravando em arquivo.

```
#tcpdump -i eth2 -s 1500 -c 500 -w pacotesCapturados.pcap
```

Monitorando os pacotes recebidos de uma máquina específica.

```
#tcpdump -i eth2 src host 143.54.1.26
```



## Referências

- CHAPPELL, Laura. Wireshark Network Analysis: The Official Wireshark Certified Network Analyst Study Guide. 2010. Protocol Analisys Institute.
- CHAPPEL, Laura. Wireshark 101 – Essential Skills for Network Analisys. Protocol Analysis Institute, Inc. 2013
- CISCO. Cisco BTS 10200 Softswitch Troubleshooting Guide, Release 4.4 2008: Cisco Systems, Inc.
- COMBS, Gerald. Wireshark DisplayFilters.  
Disponível em: <http://wiki.wireshark.org/DisplayFilters>
- HORN, Mike. Wireshark CaptureFilters.  
Disponível em: <http://wiki.wireshark.org/CaptureFilters>
- JACOBSON, Van; LERES, Craig; MCCANNE, Steven. TCPDUMP man pages: Lawrence Berkeley National Laboratory, University of California, Berkeley, CA.  
Disponível em <http://www.manpagez.com/man/1/tcpdump/>
- LAMPING ,Ulf; SHARPE, Richard; WARNICKE, Ed. Wireshark User's Guide.  
Disponível em [http://www.wireshark.org/docs/wsug\\_html\\_chunked/](http://www.wireshark.org/docs/wsug_html_chunked/)
- SANDERS, Ghris. Practical Packet Analysis – Using Wireshark to solve real world problems. 2007 No Starch Press Inc. San Francisco.



## Questionário para testar os conhecimentos relativos ao uso do Wireshark

1. Qual funcionalidade fica disponível somente com o modo de operação promíscuo?

- Permitir a um adaptador WLAN capturar pacotes independente do valor do SSID.
- Possibilitar que uma interface capture pacotes ARP request/result.
- Possibilita a uma interface capturar pacotes que são enviados para qualquer endereço MAC.
- Possibilitar que uma interface capture pacotes endereçados para endereços broadcast e multicast.

2. Qual afirmativa sobre o tráfego TCP ilustrado na figura está correta?



- O cliente HTTP solicitou um arquivo gráfico.
- O servidor HTTP recusou a tentativa de conexão TCP do cliente.
- O servidor HTTP redirecionou o pedido do cliente para outro servidor.
- O cliente HTTP enviou um pedido HTTP GET para o servidor HTTP.

3. Que tipo de tráfego pode ser visto quando você conecta Wireshark diretamente a um switch sem configurar port spanning ou port mirroring?

- Broadcast.
- Ruído e interferência.
- Consultas DNS de todos os hosts.
- Quadros que contém erros de CRC.

4. Quais dos filtros a seguir podem ser usados como filtro de captura ou como filtro de apresentação (display)?

- dns.
- udp.
- dhcp.
- broadcast.

5. Qual afirmativa sobre as definições na tela de opções de captura mostrada está correta?



- Wireshark vai resolver endereços IP em nomes de host.
- Wireshark vai rolar a tela para mostrar o pacote mais recente.
- Wireshark vai tentar resolver valores OUI para todos os endereços MAC.
- Wireshark automaticamente vai parar a captura depois que dois arquivos forem salvos.

6. Qual o tipo de pacote pode ser transmitido pelo Wireshark quando você habilita a resolução de nomes?

- DHCP requests.
- UDP multicasts.
- ping broadcasts.
- inverse DNS queries.

7. Qual entre as afirmativas seguintes relativas à janela de Opções de Captura (Capture Options) mostrada está correta?



- Wireshark resolve endereços IP em nomes de host.
- Wireshark rola a tela para exibir os pacotes mais recentemente capturados.
- Wireshark tenta resolver valores OUI para todos os endereços MAC.
- Wireshark encerra a captura automaticamente depois que dois arquivos tiverem sido salvos.

8. Que filtro de display pode ser usado para exibir todo o tráfego DHCP?

- bootp
- dhcp
- tcp-port == 68
- ip.addr ==[address\_of\_dhcp\_server]

9. Qual afirmativa entre as seguintes é verdadeira?

- Pacotes marcados são marcados apenas temporariamente.
- Pacotes marcados podem ser usados para gerar filtros de exibição.
- Pacotes marcados podem ser criados usando regras de colorir pacotes.
- Pacotes marcados são automaticamente salvos em um arquivo temporário.

10. Qual tipo de tráfego pode ser visto quando se conecta equipamento com Wireshark diretamente em um switch sem configurar port spanning ou port mirroring?

- Tráfego broadcast.
- Ruído e interferência.
- Consultas DNS de todos os hosts.
- Quadros que contenham erros

# 3

## SNMPv1, SNMPv2 e SNMPv3

objetivos

Conhecer MIB; Aprender sobre Linguagem SMI; Entender os Protocolos SNMP;  
Saber sobre os tipos de agentes SNMP.

conceitos

MIIB; Linguagem SMI; Protocolo SNMPv1; Protocolo SNMPv2; Protocolo SNMPv3;  
Tipos de agentes SNMP.

### Contextualização

Motivação:

- Necessidade de controlar equipamentos heterogêneos.
- Interligação em redes TCP/IP não possui protocolo único no nível de enlace.
- Os equipamentos podem estar localizados em pontos arbitrários da interligação de redes.

As atuais redes de computadores apresentam os mais diversos equipamentos: computadores com diferentes Sistemas Operacionais (Windows, Linux e OSX), equipamentos de comunicação de dados (switches, roteadores e access points para redes sem fio), impressoras conectadas em computadores ou ligadas diretamente nas redes, tablets, telefones celulares etc. Esses equipamentos estão interconectados em redes heterogêneas, isto é, com diferentes padrões físicos e de enlace: redes com fio, redes sem fio, redes de longa distância e redes locais.

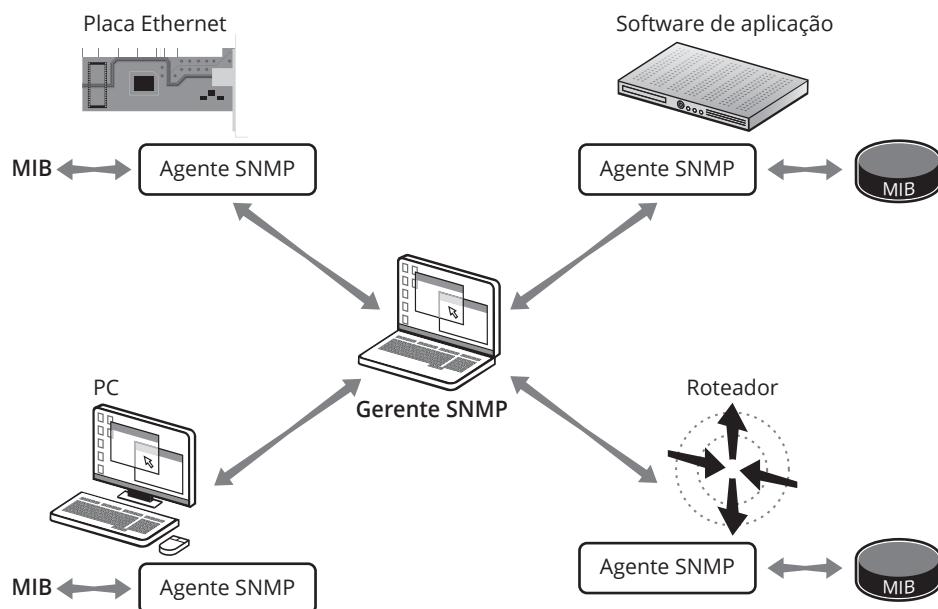
Portanto, para realizar a gerência da rede é necessário obter informações de uma variedade considerável de equipamentos, dispostos em redes distintas. Para que essa diversidade não se constitua em um problema, há necessidade de padrões que definam:

- Um modelo de gerência: gerente ou agente;
- As informações que podem ser obtidas: Management Information Base (MIB);
- As ações que podem ser realizadas: MIB;
- A forma como as operações podem ser requisitadas: protocolo Simple Network Management Protocol (SNMP);
- A maneira como novas informações podem ser incorporadas ao modelo: Structure of Management Information (SMI).

A figura 3.1 representa de forma esquemática o modelo de gerência utilizado nas redes TCP/IP. Os equipamentos podem executar um dentre dois papéis: gerente ou agente. O gerente é

a entidade: geralmente software executando em um computador: que coleta informações a respeito dos dispositivos que se deseja gerenciar. Além de coletar informações a serem apresentadas aos responsáveis técnicos pela rede, também envia comandos aos equipamentos gerenciados. O agente é a entidade instalada nos diversos nós da rede que se pretende gerenciar e que coleta informações no hardware e no Sistema Operacional para repassá-las ao gerente. Dessa forma, uma impressora ou um computador que deve ser gerenciado remotamente terá uma implementação de um agente que responde às mensagens enviadas pelo gerente. É usual também utilizar os termos gerente e agente para indicar o dispositivo propriamente dito que executa uma ou outra função.

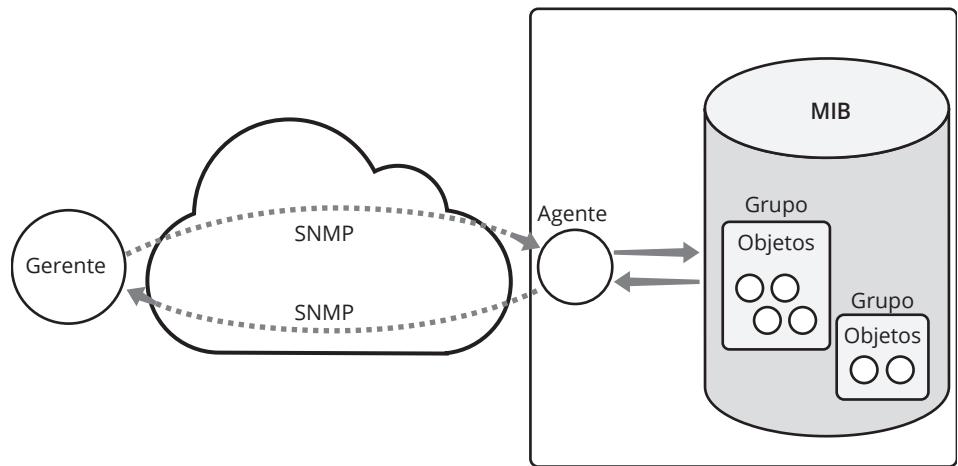
O agente SNMP também pode enviar à estação gerente mensagens disparadas em função da ocorrência de eventos pré-determinados. Essas mensagens são chamadas de traps e costumam ser usadas para gerar alertas de forma assíncrona, ou seja, sem depender de solicitação da estação gerente.



**Figura 3.1**  
Esquematização do modelo de gerência TCP/IP.

A comunicação entre gerentes e agentes ocorre através de um protocolo chamado Simple Network Management Protocol (SNMP). Já as informações de gerenciamento são chamadas de objetos gerenciados. O gerente, quando envia uma mensagem para o agente, solicita o envio de algum objeto gerenciado ou mesmo de um conjunto deles. Por exemplo, o número de erros de uma interface de rede é considerado um objeto gerenciado. Se o gerente pretende realizar uma configuração remota no agente, atuará sobre um objeto gerenciado. Por exemplo, o status de uma interface (ligada ou desligada) é um objeto gerenciado que poderia ser utilizado para ligar ou desligá-la.

O conjunto de objetos gerenciados conhecidos por um agente, isto é, implementado por ele, recebe a denominação de Management Information Base (MIB). Apesar do nome, a MIB não indica como os objetos gerenciados estão armazenados, mas define a forma como eles devem ser disponibilizados aos gerentes. A figura 3.2 mostra como ocorre uma consulta de um gerente a um agente.



**Figura 3.2**  
Solicitação de objeto gerenciável pelo gerente ao agente.

Para que gerentes e agentes possam comunicar-se, existem MIBs padronizadas. A especificação dos objetos gerenciáveis de uma MIB, seus significados e operações permitidas é feita com uma linguagem de definição de dados denominada Structure of Management Information (SMI), que é um subconjunto de uma linguagem mais abrangente denominada ASN.1 (Abstract Syntax Notation One).

## MIBs Padronizadas

MIB I:

- Definida pela RFC 1156.
- Versão inicial da MIB para gerenciamento de redes baseadas em TCP/IP.

MIB II:

- Definida pela RFC 1213.
- Superconjunto da MIB I.
- Estende a MIB I, incluindo novos objetos e módulos. Também tornou alguns objetos obsoletos.
- Acrescenta suporte a outros protocolos de interligação.

É importante destacar que por MIB entende-se a coleção dos objetos gerenciados. Em tese, qualquer um pode definir uma MIB, mas a existência de MIBs padronizadas permite que qualquer equipamento ligado à internet seja administrado pelos mais diferentes softwares de gerência de redes. Se cada fabricante implementasse em seus dispositivos um conjunto diferente de objetos gerenciados, seria muito difícil desenvolver softwares gerenciadores de rede genéricos.

A MIB I e a MIB II são padrões estipulados através de Request For Comments (RFCs). Essas RFCs definem os objetos gerenciados que deve ser implementados por qualquer agente. A MIB I é definida na FRC 1156, de maio de 1990. Ela apresenta oito grupos de objetos, listados na tabela 3.1.

Nome do grupo	Função dos objetos do grupo
System	Informações gerais do sistema.
Interfaces	Informações sobre as interfaces de rede.
Address Translation	Relação endereço físico: endereço de rede (IP).
IP	Informações do protocolo IP (pacotes recebidos e enviados, TTL, erros, tabela de endereços, tabela de roteamento etc.)
ICMP	Estatísticas ICMP.
TCP	Informações do protocolo TCP e tabela de conexões.
UDP	Informações do protocolo UDP.
EGP	Informações do protocolo EGP.

**Tabela 3.1**  
Grupos de objetos da MIB I (RFC 1156).

A MIB II é definida na RFC 1213 de março de 1991. Existem RFCs que tratam de grupos específicos, tais como:

- ▣ RFC 4293, de abril de 2006- grupo IP;
- ▣ RFC 4022, de março de 2005: grupo TCP;
- ▣ RFC 4113, de junho de 2005: grupo UDP;

Em relação à MIB I, as mudanças mais significativas foram:

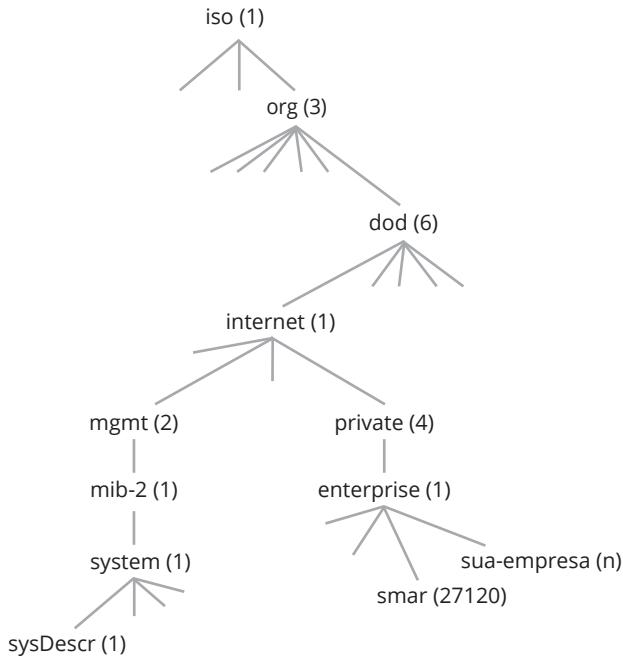
- ▣ O objeto atTable (tabela que relaciona endereço físico com endereço de rede) foi marcado como deprecated, significando que seria removido em versões futuras;
- ▣ Novos objetos foram acrescentados no grupo System;
- ▣ Ajustes foram realizados nos grupos interfaces, IP, TCP, UDP e EGP. O grupo EGP, na realidade, é raramente utilizado;
- ▣ Adição de um novo grupo chamado “Transmission”, para identificar meios de transmissão;
- ▣ Adição de um grupo chamado “SNMP”, para guardar informações do protocolo de gerência de redes.

É possível interpretar os padrões MIB I e MIB II como definições de classes, sendo que os objetos armazenados em cada equipamento, no banco de dados MIB, são instâncias da classe MIB I ou MIB II.

## Árvore de identificadores

Considerando que as operações de gerência ocorrem a partir de objetos gerenciados definidos em MIBs que estão implementadas em agentes, é necessário que esses objetos possam ser identificados de forma unívoca. O identificador de um objeto é chamado de Object Identifier (OID) e é formado com base em uma árvore de identificadores de objetos. A figura 3.3 corresponde a uma parte da árvore de identificadores utilizada. Ela apresenta todo o caminho do nó raiz (iso) até um objeto gerenciado, cujo nome é sysDescr. Esse objeto contém a descrição do agente.





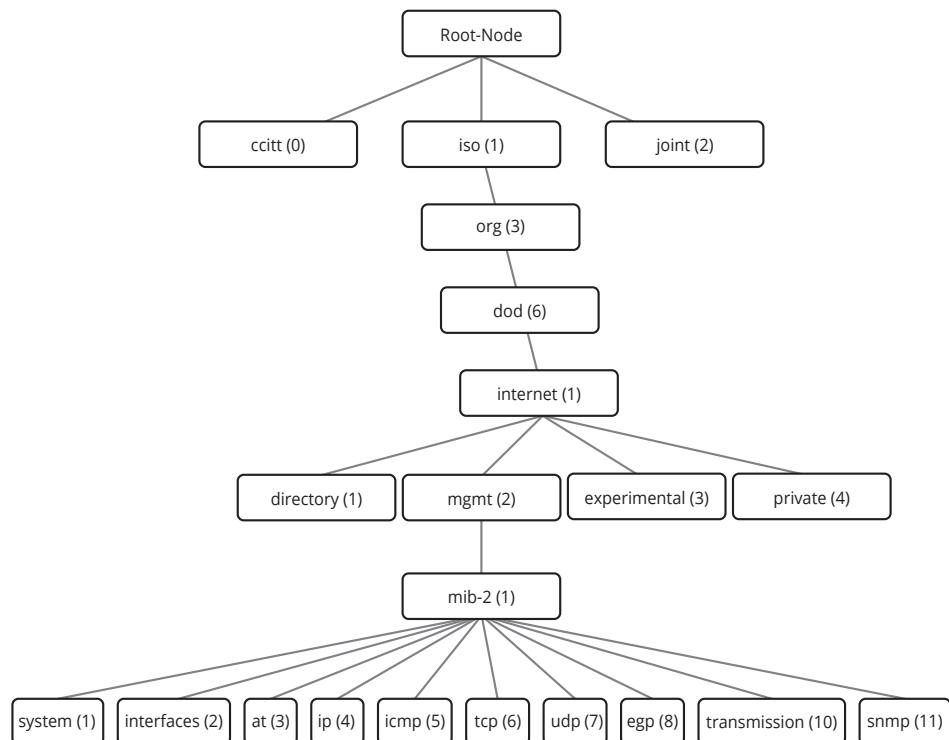
**Figura 3.3**  
Parte da árvore de OIDs (identificadores de objetos).

A identificação do objeto gerenciado dá-se pela concatenação dos números ou dos nomes da árvore, separados por pontos ("."). Essa concatenação corresponde ao caminho que vai da raiz até o objeto em questão. Dessa forma, o objeto gerenciado sysDescr tem como identificador único as seguintes representações:

- 1.3.6.1.2.1.1.1
- iso.org.dod.internet.mgmt.mib-2.system.sysDescr

## Árvore da MIB II

É importante conhecer melhor os grupos de objetos gerenciados presentes na definição da MIB-II. A figura 3.4 apresenta a árvore com esses grupos.



**Figura 3.4**  
Árvore com os grupos da MIB-II.

- ▣ **system**: define a lista de objetos pertencentes à operação do sistema em geral, tais como identificação, tipo de hardware utilizado, Sistema Operacional e softwares de rede.
- Os seguintes objetos fazem parte desse grupo:
  - ▣ **sysDescr**: descrição do dispositivo;
  - ▣ **sysObjectID**: identificação do software que implementa o agente SNMP;
  - ▣ **sysUpTime**: tempo de execução do agente SNMP;
  - ▣ **sysContact**: dados do responsável pelo dispositivo;
  - ▣ **sysName**: nome do dispositivo;
  - ▣ **sysLocation**: localização física do dispositivo;
  - ▣ **sysServices**: camadas do modelo OSI implementadas pelos dispositivo.
- ▣ **interfaces**: apresenta dois objetos principais: o número de interfaces de rede do dispositivo (*ifNumber*) e uma tabela com dados sobre cada interface (*ifTable*). As informações da tabela são semelhantes àquelas apresentadas pelo comando *show interface* nos roteadores Cisco e pelo comando *ifconfig* em sistemas Unix like: *ifDescr*, *ifType*, *ifMtu*, *ifSpeed*, *ifPhysAddress*, *ifAdminStatus*, *ifOperStatus*, *ifLastChange*, *ifInOctets*, *ifInUcastPkts*, *ifInNUcastPkts*, *ifInDiscards*, *ifEnErrors*, *ifInUnknownProtos*, *ifOutOctets*, *ifOutUcastPkts*, *ifOutNUcastPkts*, *ifOutDiscards*, *ifOutQlen* e *ifSpecific*;
- ▣ **at**: armazena a tabela de mapeamento entre endereços físicos de rede e endereços de interligação de redes (IP). Esse grupo possui o status deprecated, ou seja, não está mais em uso, estando presente somente por motivos de compatibilidade;
- ▣ **ip**: define aspectos relacionados ao protocolo IP. Alguns objetos são escalares e outros são do tipo tabela. Como objetos escalares, podem ser citados *ipForwarding*, *ipDefaultTTL*, *ipInReceives*, *ipOutReceives*, entre outros. Já as tabelas são:
  - ▣ **ipAddrTable**: tabela com endereço, máscara, broadcast e tamanho máximo do datagrama remontado para cada interface;
  - ▣ **ipRouteTable**: tabela com rotas para diferentes destinos;
  - ▣ **ipNetToMediaTable**: tabela de mapeamento entre endereços IP e de enlace.
- ▣ **icmp**: define contadores para cada uma das mensagens do protocolo ICMP. Alguns exemplos são: *icmplnMsgs*, *icmplnErrors* e *icmplnDestUnreachs*;
- ▣ **tcp**: define as informações relacionadas ao protocolo TCP. Nesse grupo, há objetos escalares que indicam parâmetros do protocolo, tais como o timeout máximo de retransmissão (*tcpRtoMax*) ou número máximo de conexões (*tcpMaxConn*). Também há uma tabela denominada *tcpConnTable* que mantém informações sobre conexões TCP com os seguintes campos: *tcpConnState*, *tcpConnLocalAddress*, *tcpConnLocalPort*, *tcpConnRemAddress*, *tcpConnRemPort*. É importante observar que as informações relacionadas diretamente com conexões específicas estão disponíveis apenas enquanto a conexão estiver ativa. Dessa forma, quando as conexões são encerradas, seus dados são descartados, restando apenas as informações contabilizadas nos contadores;
- ▣ **udp**: esse grupo é semelhante ao tcp. Armazena as informações relacionadas ao protocolo UDP. Foram definidos quatro objetos escalares (*udplnDatagrams*, *udpNoPorts*, *udplnErrors* e *udplnOutDatagrams*) e uma tabela (*udpTable*), que relaciona as portas que aceitam datagramas para cada endereço local;



- ▣ **egp:** Define as informações referentes ao protocolo Exterior Gateway Protocol (EGP), como a tabela de roteamento, a quantidade de mensagens recebidas, permitindo distinguir aqueles que resultaram em erros ou que contêm erros, a quantidade de mensagens enviadas, entre outros. Apesar de ser um protocolo obsoleto, pois foi substituído pelo Border Gateway Protocol (BGP), as informações sobre esse protocolo ainda encontram-se disponíveis na MIB II;
- ▣ **transmission:** corresponde a um espaço para definição de novas MIBs, especificando os diversos meios de comunicação utilizados. É importante observar que esse grupo foi inserido somente na MIB II, não estando presente na MIB I;
- ▣ **snmp:** assim como o grupo transmission, o grupo snmp também foi inserido na MIB II. Ele define objetos para o gerenciamento do próprio agente SNMP. Como exemplos podem ser citados os seguintes objetos:
  - ▣ **snmpInPkts:** contador de pacotes SNMP recebidos;
  - ▣ **snmpinBadVersions:** contador de pacotes recebidos com a versão errada do protocolo.

## MIB para tecnologias de transmissão

- ▣ Definida no grupo transmission, com OID 10.
- ▣ Usada para definir o valor da coluna ifType nos objetos ifTable de cada interface.
- ▣ Exemplos:
  - ▣ Ethernet RFC 1643
  - ▣ Ds1 RFC 1406
  - ▣ DS3 RFC 1407
  - ▣ PPP RFC 1471
  - ▣ Frame-Relay RFC 1315 e 1604
  - ▣ Sonet-SDH RFC 1595
  - ▣ aal5

O grupo transmission define objetos para a caracterização do meio de transmissão utilizado, considerando, também, a camada de acesso ao meio (enlace).

Cada RFC (Request for Comment) define um padrão de comunicação. Dessa forma, são definidos, em alguns casos, vários objetos MIB para tratar diferentes aspectos da tecnologia de enlace e camada física. Como exemplo, considere a RFC 1643, que trata do padrão Ethernet. Nessa RFC são definidos os objetos ethernet-csmacd (6), iso88023-csmacd (7) e slarLan (11). Da mesma forma para a RFC 1471, também são definidos vários objetos sob o grupo transmission, utilizando diferentes OID.

Os dados usados nesse grupo são replicados no atributo ifType de cada linha do objeto ifTable, que armazena as informações das interfaces do equipamento. Dessa forma, é possível definir o tipo de tecnologia empregada em cada interface do equipamento monitorado.

## Novos grupos da MIB II

A partir da definição da MIB II, novos grupos foram criados:

- ▣ **appleTalk** (13): RFC 1742
- ▣ **ospf** (191): RFC 1850
- ▣ **bgp** (15): RFC 1657

- rmon (16): RFC 1724
- dns (32): RFC 1611
- rip2 (23): RFC 1724
- Application (27): RFC 1565



À medida que novas necessidades surgem, novos grupos de objetos são adicionados à MIB II. Esses grupos são definidos em RFCs próprias. Assim, a quantidade de objetos do grupo mib-2 é ampliada à medida que essas RFC são publicadas.

É interessante observar que o objeto application, definido pela RFC 1565 com OID 27, define padrões para aplicações de rede, como Mail Transfer Agent (MTAs), facilitando assim o monitoramento, já que as informações tornam-se padronizadas.

Se, por um lado, os objetos gerenciados definidos diretamente na RFC 1213 são implementados em praticamente todos os dispositivos que permitem a gerência SNMP, o mesmo não ocorre para as extensões.

## Extensões privadas à MIB

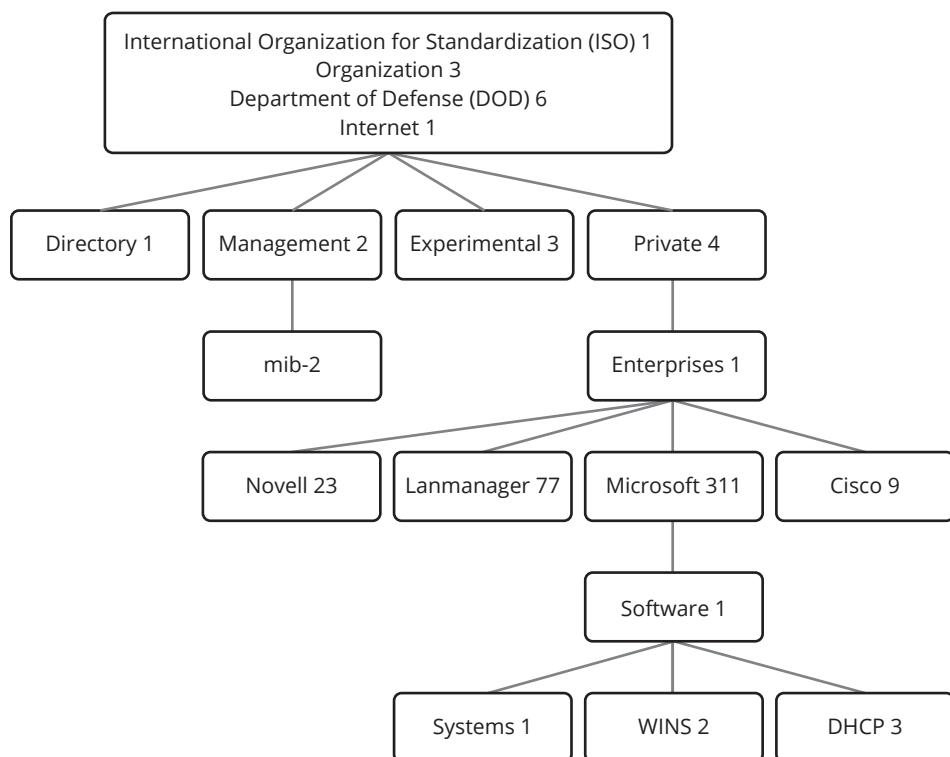
Sob a subárvore iso.organization.dod.internet.private (1.3.6.1.4) são alocados os identificadores para instituições privadas, como a Microsoft, Cisco e Oracle, que estão vinculados ao grupo enterprises (1.3.6.1.4.1). Dessa forma, os diversos fabricantes de equipamentos podem definir novos objetos gerenciados sob sua responsabilidade, específicos para os seus produtos. Como exemplo, considere o formato dos arquivos do Microsoft Word, que se encontra definido em (1.3.6.1.4.1.311.1.2.840.113556.4.2).

Inicialmente, novos objetos são criados no ramo experimental (1.3.6.1.3), sendo transferidos posteriormente pelo fabricante para o ramo private.

A figura 3.5 apresenta parte da árvore com nós privados.



A lista completa de identificadores de organizações pode ser obtida em: <http://www.iana.org/assignments/enterprise-numbers>.



**Figura 3.5**  
Árvore de OIDs com nós privados.



Um exemplo prático é a MIB definida pela CISCO para suas VLANS (CISCO Private VLAN MIB). Os objetos constantes nessa MIB permitem a gerência das redes locais virtuais nos equipamentos da CISCO. Embora, inicialmente, possa-se pensar que apenas gerentes da CISCO fariam uso dessa MIB, por ser um padrão publicado na mesma linguagem de todas as MIBs (SMI), é possível desenvolver gerentes que atuem sobre os objetos ali definidos ou mesmo compilar a MIB em gerenciadores de redes genéricos que permitem a adição de novas MIBs.

## Structure of Management Information (SMI)

- Linguagem usada para definir as informações de gerenciamento.
- Assegura que a sintaxe e a semântica dos dados sejam bem definidos e isentos de ambiguidades.
- Subconjunto da notação ASN.1 (Abstract Syntax Notation One).
- Versões:
  - SMIv1.
    - Utilizada com SNMPv1.
    - Definida e alterada nas RFCs 1155, 1212 e 1215.
  - SMIv2.
    - Utilizada em SNMPv2 e SNMPv3.
    - Definida e alterada nas RFCs 1442, 1902 e 2578.



A Structure of Management Information (SMI) é uma linguagem que permite a definição de objetos gerenciados. Com essa linguagem é possível definir e descrever os objetos da MIB e agrupá-los em subconjuntos. Essa tarefa somente é possível porque a SMI define tipos básicos de dados, que serão empregados na definição dos atributos dos objetos.

Uma vez que a SMI é um subconjunto da ASN.1, é possível ampliá-la para atender demandas futuras. Atualmente existem duas versões: a primeira, denominada SMIv1 (RFCs 1155, 1212 e 1215), é empregada com o protocolo SNMPv1. A segunda versão, denominada SMIv2 (RFC 2578), é utilizada com as versões dois e três do protocolo SNMP.

Como existem diferentes versões de SMI para diferentes versões do protocolo SNMP, estas serão estudadas em conjunto com as respectivas versões do protocolo em que são empregadas. Nesta sessão, será apresentado um exemplo básico para compreensão de qualquer uma das versões.

A definição de um objeto gerenciado apresenta cinco partes:

- Nome do objeto e seu identificador (OID);
- Sintaxe do objeto que deve corresponder a um tipo primitivo;
- Definição textual do objeto;
- A autorização de acesso do objeto (read-only, read-write, write-only ou not-accessible);
- A obrigatoriedade de implementação do objeto (mandatory, optional ou obsoleto).

Essa organização é perceptível na definição do objeto sysDescr, apresentada a seguir.

```
sysDescr OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..255))
    ACCESS  read-only
```

```

STATUS mandatory

DESCRIPTION

    "A textual description of the entity. This value
    should include the full name and version
    identification of the system's hardware type,
    software operating-system, and networking
    software. It is mandatory that this only contain
    printable ASCII characters."

 ::= { system 1 }

```

A definição começa com o nome do objeto seguido pela palavra-chave “OBJECT-TYPE”. O identificador está localizado no fim da definição, antecedido pelo objeto hierarquicamente superior na árvore de OIDs (system). A seção SYNTAX indica que é limitado a um tipo básico chamado DisplayString, que pode ter 0 a 255 caracteres. A seção ACCESS limita o objeto à leitura, isto é, ele não pode ser atualizado pelo gerente remoto. A seguir, a seção STATUS sinaliza a obrigatoriedade de implementação e a seção DESCRIPTION explica a semântica do objeto gerenciável.

## SNMP – Introdução

Características:

- ▣ Aberto.
- ▣ Padrão de mercado: TCP/IP.
- ▣ Simples.
  - ▣ Especifica poucas operações.
  - ▣ Baseado no paradigma de busca e armazenamento (fetch-store paradigm).
- ▣ Implementado na camada de aplicação.
- ▣ Utiliza o protocolo User Datagram Protocol (UDP) na camada de transporte.

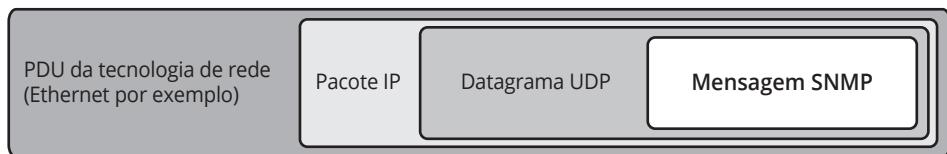


O Simple Network Management Protocol (SNMP) é o padrão mais popular para protocolo de gerência de rede. É um padrão aberto adotado por vários fabricantes e operadoras, definindo o funcionamento da arquitetura de gerenciamento de redes TCP/IP.

É um protocolo simples de ser implementado em todo tipo de equipamento e flexível o bastante para aceitar modificações posteriores, uma vez que possui poucas operações, além de utilizar o paradigma fetch-store. Nesse paradigma, o gerente busca (fetch) as informações do agente para descobrir o status deste último. Já para atuar sobre o agente, o gerente altera valores de variáveis (store) do agente. Sempre que uma variável de um agente é alterada, uma ação será executada.

O protocolo SNMP é um protocolo de nível de aplicação na arquitetura de rede TCP/IP. Por utilizar como transporte o protocolo UDP, ele é responsável pela detecção de mensagens não recebidas. A figura 3.6 mostra o encapsulamento do protocolo.

**Figura 3.6**  
Encapsulamento do protocolo SNMP.



O SNMP faz uso da técnica de timeout para recuperar eventuais datagramas perdidos. Para evitar retransmissões infinitas, é utilizado um limite de tentativas, cujo valor é configurado pelo administrador do gerente.

Essa solução implica menor sobrecarga aos links e equipamentos, uma vez que a adoção do protocolo TCP exigiria o estabelecimento e o encerramento de conexões (three-way handshake), bem como pacotes de confirmação para as mensagens enviadas (acknowledgement). Isso aumentaria consideravelmente a quantidade de pacotes trafegados pelo link. Além disso, o estabelecimento de conexões amplia o consumo de memória nos equipamentos de interligação, que normalmente é restrita.

#### Versões

- SNMPv1:
  - Padrão histórico IETF;
  - RFC 1157 de 1990;
  - Segurança baseada em comunidades;
  - Troca de informação baseada em textos simples.
- SNMPv2:
  - RFC Principais 3416, 3417 e 3418;
  - Amplia as operações definidas na versão 1;
  - Unificação dos PDU (Protocol Data Unit);
  - Possui variantes.
- SNMPv3:
  - RFC 3410 à 3418 e 2576;
  - Avanço em segurança e privacidade.

Existem três versões principais do protocolo. O SNMPv1 é definido pela RFC 1157 de 1990. Usa SMIv1, apresenta as operações Get, GetNext, Set e Trap. O controle de acesso é baseado no conceito de comunidade e não é seguro.

O SNMPv2 é definido pelas RFCs 3416, 3417 e 3418 e introduziu três modificações importantes:

- Nos tipos de dados e nos objetos gerenciados;
- No modelo de segurança, que acabou por mostrar-se inviável;
- Nas operações disponibilizadas pelo protocolo.

O SNMPv3 é definido nas RFCs 3410, 3418 e 2576. A grande alteração trazida por esse protocolo é o modelo de segurança, que permite uma implementação muito mais robusta do que o esquema baseado em comunidades.

## SMIV1 – Tipos de dados



- Integer.
- Octet String.
- Counter.
- Object Identifier.
- NULL.
- Sequence.
- Sequence of.
- IpAddress.
- NetworkAddress.
- Gauge.
- TimeTicks.
- Opaque.

O padrão SMIV1 define doze tipos de dados, utilizados para a definição de objetos:

- **Integer**: número inteiro de 32 bits, normalmente utilizado para representar estados de um equipamento. Independente da arquitetura do equipamento utilizado (32 ou 64 bits), esse tipo de dados é sempre 32 bits;
- **OctetString**: define uma string de um ou mais octetos (bytes). Também utilizado para representar endereços Media Access Control (MAC);
- **Counter**: número inteiro de 32 bits, variando entre 0 e 232: 1. Ao atingir o valor máximo, as variáveis desse tipo voltam a armazenar o valor inicial (zero), uma vez que esse tipo de dados pode ser apenas incrementado e de forma unitária;
- **ObjectIdentifier**: string composta por números decimais separados por pontos. Sua finalidade é representar os OID da árvore de identificação da ISSO;
- **NULL**: valor nulo;
- **SEQUENCE**: define uma lista que contém zero ou mais tipos;
- **SEQUENCE OF**: define um objeto que é formado por uma sequência de um determinado tipo. A diferença entre o tipo SEQUENCE e SEQUENCE OF é que o primeiro é composto por um número determinado de tipos variados, já o segundo é composto por uma lista variável de um determinado tipo. Por exemplo, uma linha de uma tabela é um SEQUENCE; uma tabela é um SEQUENCE OF de uma linha;
- **IpAddress**: representa um endereço IPv4. É importante observar que as versões SMIV1 e SMIV2 não permitem representar nativamente endereços IPv6 de 128 bits;
- **NetworkAddress**: representa endereços de redes baseadas em IPv4;
- **Gauge**: similar ao tipo Counter, esse tipo de dados constitui um contador do tipo inteiro, variando entre 0 e 232: 1. No entanto, dados do tipo Gauge podem ser incrementados e decrementados, diferentemente do tipo Counter;
- **TimeTicks**: número inteiro de 32 bits utilizado para representar o tempo. Sua representação é feita em milésimos de segundos;
- **Opaque**: utilizado para permitir que qualquer tipo de dados seja armazenado em forma de um Octet String.



## SMIv1 – Definição de objetos

O texto a seguir corresponde à definição do objeto sysUpTime.

```
sysUpTime OBJECT-TYPE
    SYNTAX Time-Ticks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The time (in hundredths of a second)since the network
        management portion of the system was last re-initialized."
    ::= { system 3 }
```

Os objetos e seus atributos são definidos para representar informações diretamente relacionadas como equipamento. Sua definição se dá através da macro ASN.1 OBJECT-TYPE, contendo as seguintes informações:

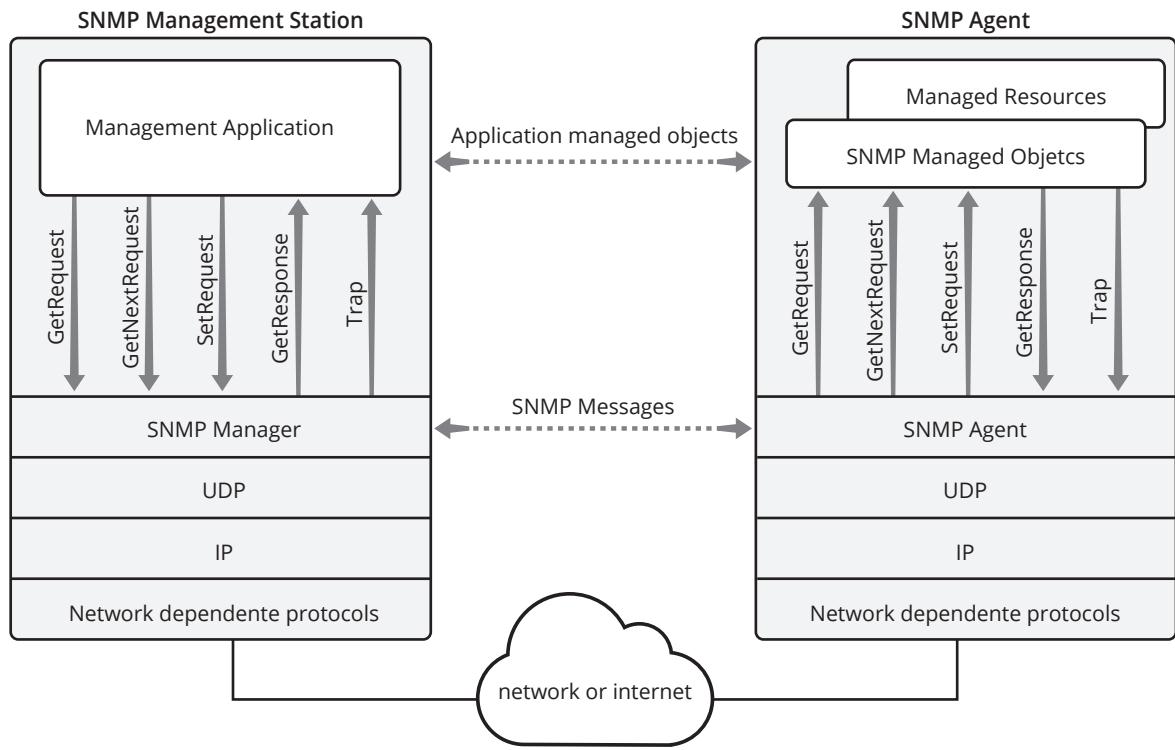
- **SYNTAX:** define o tipo de dados a ser utilizado;
- **ACCESS:** dá a definição do tipo de acesso permitido. Os valores possíveis são read-only, read-write, write-only e not-accessible. O valor de not-accessible pode não fazer sentido à primeira vista, mas pode ser usado, por exemplo, para evitar que se solicite uma tabela inteira em uma operação de busca (Get). Isso acontece porque a declaração de uma tabela, na verdade, é a declaração de uma sequência de linhas. Essa sequência tem um OID como qualquer outro item de dado ASN.1. No entanto, o protocolo não permite que a tabela inteira seja lida em uma única operação. Assim, declara-se essa sequência como not-accessible para evitar pedidos diretos a ela, sem evitar que os objetos que a compõem sejam buscados.
- **STATUS:** define o status do objeto, assumindo os valores mandatory, optional e obsolete;
- **DESCRIPTION:** descrição livre sobre o tipo de informação a ser armazenada.

Finalmente, ao fim das definições, deve-se indicar a qual nó pai o novo objeto será subordinado e qual o seu OID. No exemplo apresentado, o objeto sysUptime encontra-se na sub-árvore system com o OID 3.

## SNMPv1 – Comunicação e Operações

O protocolo SNMPv1 não necessita de um serviço de transporte confiável, utilizando, geralmente, o protocolo UDP. Como é possível perceber na figura 3.7, o protocolo SNMP, independente da versão, utiliza a arquitetura cliente-servidor (pedidos-respostas) nas etapas de solicitação de informações.

 Interessante observar que o gerente tem o papel de cliente e o agente tem o papel de servidor, já que o gerente faz solicitações para o cliente. A exceção são os Traps.

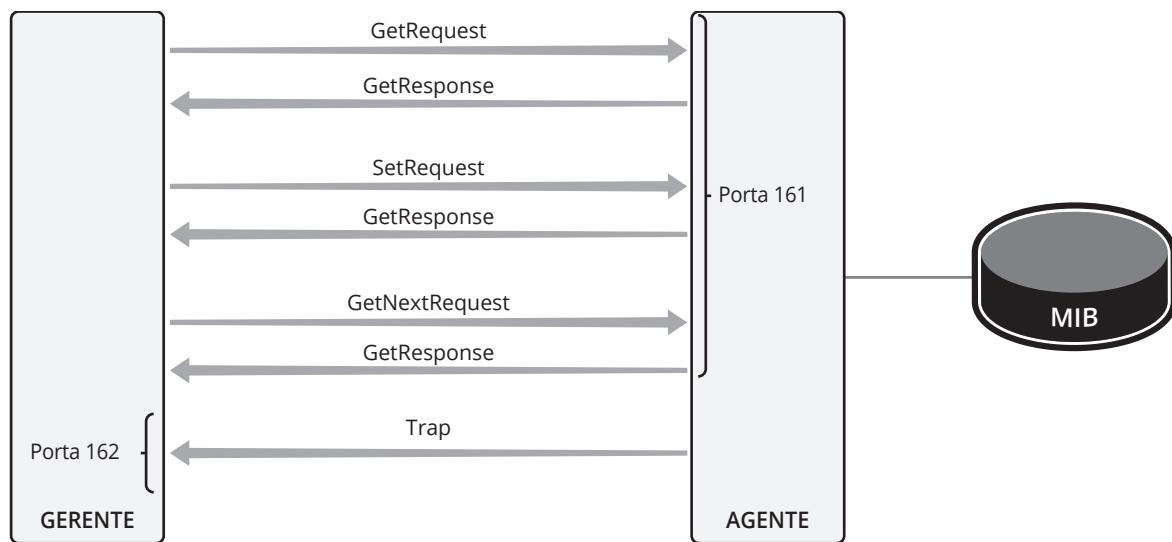


O SNMPv1 possui quatro operações e cinco mensagens. As quatro operações são:

- **get**: utilizada para recuperar um objeto específico;
  - **get-next**: usada para recuperar o próximo objeto, a fim de percorrer a MIB de um agente;
  - **set**: utilizada para modificar ou criar um objeto;
  - **trap**: usada para que o agente notifique o gerente sobre a ocorrência de um evento.
- Essas quatro operações são executadas a partir das cinco mensagens disponíveis no protocolo SNMPv1:
- **GetRequest**: mensagem enviada pelo gerente ao agente, solicitando um objeto específico;
  - **GetNextRequest**: mensagem enviada pelo gerente ao agente solicitando o objeto imediatamente posterior ao referenciado no pedido;
  - **SetRequest**: mensagem enviada pelo gerente ao agente, solicitando a modificação do valor de uma variável ou a criação de uma linha em uma tabela;
  - **GetResponse**: mensagem do agente ao gerente, relativa a qualquer uma das três mensagens citadas anteriormente nessa lista;
  - **Trap**: mensagem assíncrona enviada pelos agentes ao gerente (não apresenta resposta).

**Figura 3.7**  
Arquitetura cliente-servidor do protocolo SNMP.

A figura 3.8 apresenta as mensagens trocadas entre agentes e gerentes.



**Figura 3.8**  
Mensagens do protocolo SNMPv1.

A mensagem do protocolo SNMPv1 é composta pelos seguintes campos:

- **Versão:** versão do protocolo (0 para a versão 1);
- **Comunidade:** espécie de senha usada para recuperar ou alterar objetos da MIB;
- **PDU:** especificação do tipo de mensagem.

A figura 3.9 é apresenta de forma esquemática o formato da mensagem.

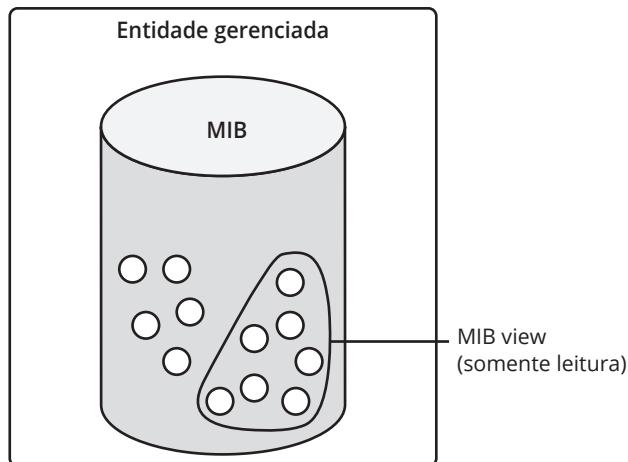
**Figura 3.9**  
Esquema de  
uma mensagem  
SNMPv1.

Versão	Comunidade	PDU (Protocol Data Unit)
Número inteiro indicando a versão SNMP (0 para SNMPv1)	Este campo carrega o nome de comunidade que o originador da mensagem está usando	Dados efetivos de gerência a serem analisados e processados

O protocolo SNMPv1 suporta um esquema muito simples de autenticação. Uma comunidade é um nome que especifica o nível de acesso às facilidades oferecidas pelos agentes. A ideia é prevenir que gerentes desautorizados acessem a MIB de um determinado agente, restringindo, assim, a informação vista por determinado gerente. Esse fraco esquema de autenticação inibe o uso da operação de Set pela inerente falta de segurança.

## SNMPv1 – Relação MIB view e Comunidade

Uma MIB view é um subconjunto de objetos de uma MIB de um determinado dispositivo, relacionados por grau de acesso. Assim, o controle de acesso SNMP define que uma determinada comunidade está restrita às variáveis de uma determinada “MIB view”. A figura 3.10 mostra que uma MIB view corresponde a um subconjunto de objetos gerenciados de um agente.



**Figura 3.10**  
MIB view.

## SNMPv1: Formato da PDU de resposta

Valores de status do erro:

- 0: noError (não houve erro).
- 1: tooBig (a resposta é muito grande para a implementação).
- 2: noSuchName (uma das variáveis não consta na MIB do agente requisitado).
- 3: badValue (erro no pedido de "Set").
- 4: readOnly (uma mensagem "Set" tenta alterar uma variável "read-only").
- 5: genError (erro desconhecido).

Os campos de um PDU SNMPv1 de resposta (GetResponse), conforme a figura 3.11, são:

- **Tipo de PDU:** especifica a operação SNMPv1 entre as cinco apresentadas:
  - GetRequest-PDU = 0
  - GetNextRequest-PDU = 1
  - GetResponse-PDU = 2
  - SetRequest-PDU = 3
  - Trap-PDU = 4
- **Request ID:** número sequencial para associar pedidos com respostas;
- **Error Status:** indicação se foi possível realizar a operação solicitada e eventual código de erro;
- **Error Index:** em caso de erro, identifica o objeto no qual houve o problema;
- **Lista de objetos retornados:** identifica objetos e valores.

Tipo PDU	ID	Status do erro	Índice do erro	Objeto 1 Valor 1	Objeto 2 Valor 2	Objeto x Valor x
Objetos Vinculados						

**Figura 3.11**  
Formato da PDU de um GetResponse.

Em uma mensagem do tipo GetResponse, o campo *error status* indica o que ocorreu e o campo *error index* aponta para o primeiro objeto que apresentou falha. Os seguintes códigos de falhar são usados:

- **noError (0):** não houve falha;
- **tooBig (1):** o agente não conseguiu enviar a resposta por ocupar muito espaço;

- ▣ **noSuchName (2)**: o objeto não existe para a comunidade especificada;
- ▣ **badValue (3)**: houve uma tentativa de alterar um objeto para um valor inconsistente;
- ▣ **readOnly (4)**: houve uma tentativa de alterar um objeto que para a comunidade especificada não pode ser sobreescrito;
- ▣ **genErr (5)**: qualquer outro erro.

## SNMPv1: Formato do PDU para a trap

Campo Tipo genérico da trap:

- ▣ 0 – coldStart: reinicialização do dispositivo; configuração do agente não é alterada.
- ▣ 1 – warmStart: reinicialização do dispositivo com possível alteração da configuração do agente.
- ▣ 2 – linkDown: falha em um dos links do agente.
- ▣ 3 – linkUp: volta de um link do agente.
- ▣ 4 – authenticationFailure: falha na autenticação de uma mensagem recebida.
- ▣ 5 – egpNeighborLoss: perda da comunicação com um vizinho EGP.
- ▣ 6 – enterpriseSpecific: traps específicas.

A mensagem do tipo Trap reporta ao gerente eventos, problemas e condições anormais. No entanto, com o aumento da concentração de equipamentos presentes nas interligações de redes, não é raro que vários Traps estejam reportando o mesmo evento. Dessa forma, é necessário que se realize a correlação de alarmes, a fim de agrupar tais Traps.

Da mesma forma, em redes com muitos dispositivos, fica inviável realizar pollings periódicos. Assim, devem-se adotar estratégias de monitoramento, onde os pollings são realizados com menor frequência, priorizando os casos em que traps foram recebidas.

Como é possível observar na figura 3.12, os campos que compõem as mensagens de Trap no SNMPv1 são:

- ▣ **Tipo de PDU**: especifica a operação SNMPv1 entre as cinco apresentadas (0x04);
- ▣ **Enterprise**: ObjID que indica o tipo de dispositivo que enviou a mensagem;
- ▣ **Endereço do agente**: endereço IP do objeto que gerou a mensagem;
- ▣ **Tipo genérico de Trap**: identifica o evento que gerou a mensagem;
- ▣ **Código específico da trap**: identifica o evento que gerou a mensagem se não corresponder a um evento genérico, e sim a algo específico criado pelo fabricante do equipamento;
- ▣ **Timestamp**: tempo passado entre o momento em que foi gerada a trap e a última reinicialização (sysUpTime);
- ▣ **Lista de pares objeto-valor**: objetos reportados pela trap.

**Figura 3.12**  
PDU de uma mensagem do tipo Trap.

Tipo de PDU	Enterprise	Endereço do Agente	Tipo Genérico da trap	Código da trap	Time Stamp	Objeto 1 Valor 1	Objeto 2 Valor 2	Objeto x Valor x
Objetos Vinculados								

Os seguintes tipos genéricos de Traps estão definidos para o padrão:

- ▣ **coldStart (0)**: dispositivo reiniciou;
- ▣ **warmStart (1)**: agente reiniciou;

- **linkDown (2)**: interface de rede mudou de ativa para desligada;
- **linkUp (3)**: interface de rede mudou de desligada para ativa;
- **authenticationFailure (4)**: foi recebida mensagem SNMP com erro de autenticação;
- **egpNeighborLoss (5)**: houve perda de vizinhança no protocolo EGP;
- **nenhum (6)**: há um evento específico do fabricante, que deve ser consultado no campo posterior.

## SNMPv1: Declaração de uma trap específica (macro ASN.1 TRAP-TYPE)

```

frDLCIStatusChange TRAP-TYPE
ENTERPRISE frame-relay
VARIABLES {frCircuitIndex, frCircuitDlc, frCircuitState}
DESCRIPTION
"This trap indicates that the indicated Virtual Circuit has changed
state. It has either been created or invalidated, or has toggled
between the active and inactive states."
::=1

```

Traps específicas são definidas pelo valor do “enterprise ObjID” e o número “specific trap”. Existe uma macro (TRAP-TYPE) para definição de traps.

## SNMPv1: Obtenção de valores de uma MIB

Objetos com valor único (instância):

- **1.3.6.1.2.1.1.1.0**

Objetos com valores múltiplos (múltiplas instâncias)

- **1.3.6.1.2.12.2.1.2.1**
- **1.3.6.1.2.12.2.1.2.2**
- **1.3.6.1.2.12.2.1.2.3**

Para fins de recuperação do valor de um objeto gerenciado, pode-se considerar que existem dois tipos de objetos: aqueles que apresentam uma única instância e aqueles que apresentam múltiplas instâncias. O primeiro corresponde aos objetos escalares, como sysDescr. O segundo tipo corresponde a objetos posicionados dentro de tabelas, como ifSpeed, que apresentará valor para cada interface de rede do equipamento. Essa distinção é importante porque um objeto gerenciado, no contexto do protocolo SNMP, é identificado OID acrescido de um identificador de instância. Dependendo do tipo de objeto, o sufixo acrescentado ao OID para identificação da instância será diferente.

Toda instância de objeto escalar será identificada pelo OID seguido do algarismo 0 (zero).

Assim, para recuperar uma instância de sysDescr, o gerente deverá enviar ao agente a seguinte mensagem:

- GetRequest 1.3.6.1.2.1.1.1.0 ou seja, está solicitando o valor do objeto gerenciado iso.org.dod.internet.mgmt.mib-2.system.sysDescr.0.

Toda instância de um objeto de uma tabela será identificado pelo OID, seguido pelo valor do campo de índice. Quando um OID é um valor escalar, o índice tem valor zero. Quando o objeto gerenciável é integrante de uma tabela, o valor do índice está relacionado com a posição dele na tabela. Por exemplo, pela definição a seguir, percebe-se que a tabela ifTable é formada por uma sequência de objetos do tipo ifEntry. O objeto ifEntry, por sua vez, corresponde uma linha da tabela e define que o índice utilizado para identificar cada linha é o objeto ifIndex.

```

ifTable OBJECT-TYPE
    SYNTAX  SEQUENCE OF IfEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A list of interface entries. The number of
         entries is given by the value of ifNumber."
    ::= { interfaces 2 }

ifEntry OBJECT-TYPE
    SYNTAX  IfEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "An interface entry containing objects at the
         subnetwork layer and below for a particular
         interface."
    INDEX   { ifIndex }
    ::= { ifTable 1 }

```

Para identificar a velocidade da segunda interface do dispositivo, utiliza-se o OID seguido pelo algarismo 2. Pode-se considerar que o gerente envia a seguinte mensagem para o agente:

- GetRequest 1.3.6.1.2.1.2.2.1.5.2 ou seja deseja obter o valor do objeto gerenciável iso.org.dod.internet.mgmt.mib-2.interfaces.ifTable.ifEntry.ifSpeed.2

Nem sempre os identificadores de instâncias são números sequenciais (1, 2, 3). O padrão permite que seja usado qualquer valor, como por exemplo utilizar números IP em uma tabela de rotas. Na tabela ipAddressTable, por exemplo, o índice corresponde ao endereço IP.

Também é possível recuperar objetos gerenciados com a operação GetNextRequest, que busca o próximo objeto da MIB. Deve-se considerar que a MIB é uma árvore percorrida da esquerda para a direta. Assim, se a operação GetNext do OID iso.org.dod.internet.mgmt.mib-2.system for executada, será retornado o valor do objeto mib2.system.sysDescr.0. Ao executar a operação GetNext...system.sysDescr.0, será retornado o valor do objeto system.sysObjectID.0. Portanto, utilizando-se a identificação do objeto retornado pela operação anterior é possível percorrer toda a MIB.

Ressalte-se que uma tabela será percorrida com sucessivos GetNext, coluna por coluna. A operação GetNext...interfaces.ifTable.ifEntry.ifSpeed.1 retornará a velocidade da próxima interface (...interface.ifTable.ifEntry.ifSpeed.2) e não o endereço físico da interface (...interface.ifTable.ifEntry.ifPhysAddress.1).

Tanto a operação Get quanto GetNext podem solicitar mais de um objeto gerenciado. No entanto, se houver problema em algum deles, nenhum será retornado.

## SNMPv1: Modificação de valores em tabelas

- Tabelas são conjuntos de objetos com valores múltiplos.
- Uso do objeto do tipo “EntryStatus”.



A modificação do valor de um objeto escalar é trivial. O protocolo possui a operação SetRequest e pode-se imaginar que basta enviar, nessa operação, a identificação do objeto que se quer alterar e o valor correspondente. A modificação de um objeto de uma tabela segue o mesmo esquema. A diferença está na identificação da instância.

No entanto, duas outras operações podem ser feitas em uma tabela: criação e exclusão de uma linha. Para manipular as linhas das tabelas com o protocolo SNMP, independente da versão, é usado um objeto de status, cujo tipo é “EntryStatus”. Esse objeto possui dois valores possíveis: “valid” ou “invalid”.

Ao se criar uma linha, uma operação de “SetRequest” conterá valores para os objetos que compõem a tabela. Caso não se tenha todos os valores necessários, é possível que valores default sejam usados para inicializar os demais. A linha é criada realmente quando o valor do objeto de status for alterado para “valid”.

Para excluir uma linha, altera-se o valor do objeto de status para “invalid”.

## SNMPv2 – Uma nova versão

- 1992: o IETF anunciou a chamada por propostas para uma nova versão.
- 1992: A primeira resposta veio nas RFCs de 1351 a 1353 e foi chamada de “SNMP seguro” (incompatível com a versão 1).
- 1993: A versão 2 foi oficialmente apresentada e hoje é chamada de SNMPv2 “clássica” ou SNMPv2p (baseada em parties).
- 1996: A versão SNMPv2 definitiva é chamada de SNMPv2c (baseada em comunidades).



A partir das limitações do protocolo SNMPv1, uma segunda versão foi desenvolvida. Dentre as limitações destacam-se:

- Não é adequado para coletar grandes volumes de dados, porque precisa indicar a identificação de cada objeto gerencial a ser solicitado ao agente, gerando bastante tráfego de overhead, que pode crescer muito no caso de redes grande porte prejudicando a performance da rede;
- Possui autenticação baseada unicamente em nomes de comunidades, que podem ser facilmente obtidos a partir de uma inspeção de tráfego usando um sniffer, pois as mensagens do protocolo SNMPv1 não são criptografadas. Essa facilidade de obter indevidamente o nome da comunidade e então poder usar essa informação para alterar de forma não autorizada valores de configuração dos agentes SNMP inviabiliza a sua adoção para controle (Set). Assim, o SNMP é atualmente mais utilizado para fins de obtenção de dados dos agentes (monitoramento);



- Não permite comunicação entre gerentes;
- Funcionalidade limitada em relação ao CMIP/CMIS (protocolo/serviço de gerenciamento da arquitetura OSI);
- A característica atômica das mensagens, ou seja, o fato de uma mensagem obter total sucesso ou total fracasso nas operações (não há a possibilidade de sucesso parcial em um GetRequest que tenha, por exemplo, solicitado um conjunto de valores de objetos gerenciáveis);
- Impossibilidade de configuração remota dos agentes.

## SNMPv2 – Evolução

SNMPv2p – Party-based SNMPv2.

- Não obteve sucesso.
- Contexto de operação baseado em parties.
- Dificuldade para descoberta automática.

SNMPv2c – Community-based SNMPv2.

- Contexto de operação baseado em comunidades.
- Manteve algumas características do SNMPv2p.
  - Facilidade para declaração de objetos.
  - Melhor desempenho na troca de mensagens.
  - Melhor tratamento de erros.

A abrangente proposta SNMPv2 “clássica” (SNMPv2p) baseada em parties apresentou um modelo de comunicação entre gerente e agente que utilizava uma autenticação para limitar o acesso a um conjunto apenas de objetos gerenciados. Em 1995 foi feita mais uma revisão do protocolo em resposta à baixa aceitação da versão SNMPv2p. Tal revisão se deu principalmente no que diz respeito ao contexto baseado em parties, à configuração dos agentes (para permitir a autenticação), à dificuldade de descoberta automática da rede e de implementação do modelo administrativo e de segurança. O único consenso então obtido foi aceitar as novidades no protocolo, porém permanecendo o contexto de operação sob a antiga forma de nomes de comunidades como elemento de autenticação (fraca).

Posteriormente, foi apresentada mais uma versão de SNMP, agora chamada de SNMPv2c (baseada em comunidades). Nesta, o modelo administrativo apresentado na versão “clássica” e baseado em parties foi completamente descartado.

Apesar do fracasso, pode-se considerar que há aspectos positivos apresentados na versão SNMPv2p, como a criação de extensões da linguagem (que facilitam a declaração de novos objetos) e a melhoria da performance do protocolo na troca de informações com um melhor tratamento de erros. Uma útil experiência prática foi obtida nas implementações e testes com SNMPv2p.

Em resumo, SNMPv2c assimilou somente as novas mensagens, correções e SMIv2, esperando ainda melhorias em:

- Segurança (o grande problema);
- Configuração remota;
- Infraestrutura administrativa.

## SMIv2 – Novos tipos de dados

Tipo	Descrição
INTERGER	Inteiros no intervalo de $-2^{31}$ a $2^{31} - 1$
UInterger32	Inteiros no intervalo de 0 a $2^{32} - 1$
Counter32	Inteiro não-negativo a ser adicionado ao módulo $2^{32}$
Counter64	Inteiro não-negativo a ser adicionado ao módulo $2^{64}$
Gauge32	Inteiro não-negativo a ser adicionado ou diminuído, mas que não deve exceder o valor máximo. O valor máximo não pode ser maior que $2^{32} - 1$
Time Ticks	Inteiro não-negativo que representa a hora (módulo de $2^{32}$ ) em centenas de segundos
Octet String	String de textos binários; pode ser limitada a 255 octetos
OIPAddress	Endereço IP de 32 bits
Opaque	Campo arbitrário de bits
Bit String	Lista de nomes dos bits
Object Identifier	Nome do objeto ou outro elemento padronizado. O valor é uma sequência de até 128 números inteiros não-negativos.

**Tabela 3.2**  
Novos tipos de dados.

A SMIv2, definida pelas RFCs 2578 e 2579, é um superconjunto da SMIv1. Alguns novos tipos foram criados e houve uma melhora na definição dos tipos antigos. É importante observar que a SMIv2 é uma evolução totalmente compatível com a SMIv1, com exceção do novo tipo de dado Counter64 (contador com 64 bits).

## SMIv2: Textual conventions (RFC 2579)

- DisplayString: OCTET STRING (SIZE (0...255))
- PhysAddress – OCTET STRING
- MacAddress: OCTET STRING (SIZE (6))
- TruthValue – INTEGER { true(1), false(2) }
- TestAndIncr: INTEGER (0...2147483647)
- VariablePointer – OBJECT IDENTIFIER
- RowPointer: OBJECT IDENTIFIER
- RowStatus – INTEGER { active(1), notInService(2), notReady(3), createAndGo(4), createAndWait(5), destroy(6) }
- TimeStamp – TimeTicks
- DateAndTime: OCTET STRING (SIZE 8|11)



## Melhorias na manipulação de tabelas

Valores de “RowStatus”:

- ▣ 1 – active: linha operacional.
- ▣ 2 – notInService: linha desabilitada.
- ▣ 3 – notReady: linha ainda não completa.
- ▣ 4 – createAndGo: criar a linha e disponibilizá-la.
- ▣ 5 – createAndWait: criar a linha, mas esperar por outros valores.
- ▣ 6 – destroy: deletar todos os objetos da linha.

E exemplo de convenção de texto muito útil é a que permite uma melhor manipulação de tabelas. A nova convenção de texto “RowStatus” substituiu a antiga coluna “EntryStatus”. O tratamento das operações com linhas de tabelas ficou mais claro. Por exemplo, às vezes não é possível preencher uma linha da tabela em um único set, e por isso deve-se mandar “esperar” pelo resto dos dados (“CreateAndWait”).

## SMIv2 – Definição de objetos

A definição de objetos gerenciados a partir da SMIv2 sofreu algumas modificações. A cláusula MAX-ACCESS substituiu a anteriormente utilizada (ACCESS da SMIv1). Assim, é possível definir o nível máximo de acesso, através dos valores:

- ▣ not-accessible;
- ▣ accessible-for-notify;
- ▣ read-only;
- ▣ read-write;
- ▣ read-create.

Também foram incorporados novos valores para a cláusula STATUS:

- ▣ current;
- ▣ obsolet;
- ▣ deprecated.

## SNMPv2: Alterações no protocolo

- ▣ Novas opções de pilhas de transporte.
- ▣ Remoção de perda de resposta em transações atômicas do tipo getRequest/getNextRequest.
- ▣ Operação setRequest mais segura (Execução em duas fases).
- ▣ “response” é o novo nome da operação “getResponse”.
- ▣ Novas mensagens SNMP:
  - ▣ getBulkRequest
  - ▣ informRequest
  - ▣ Report (Não implementado em SNMPv2)
  - ▣ Notification

Algumas alterações foram realizadas no SNMPv2, destacando-se:

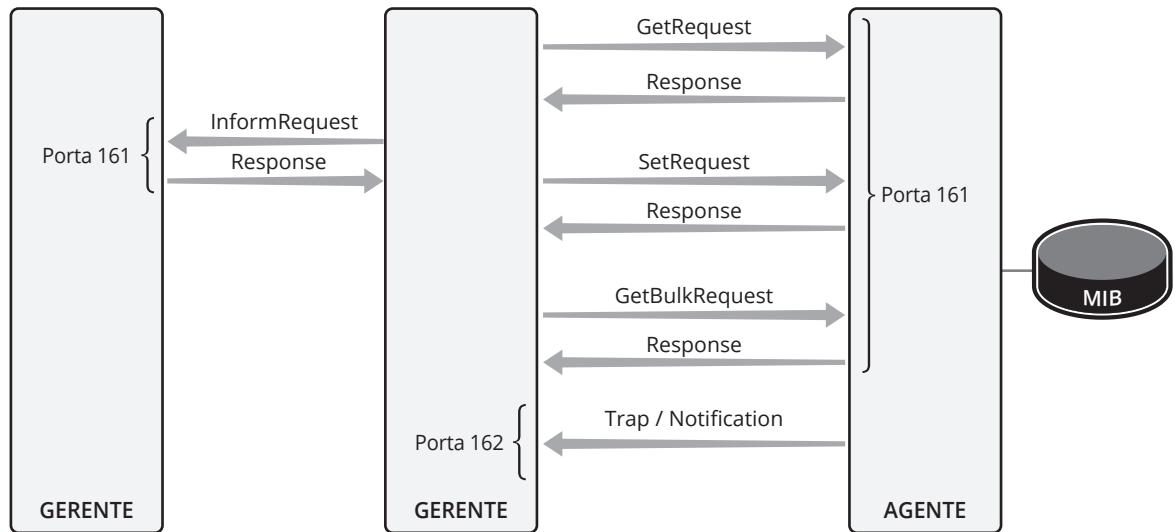
- O protocolo de transporte utilizado na primeira versão do protocolo SNMP é o UDP. A RFC 1906, depois substituída pela 3417, definiu novos protocolos de transporte para serem utilizados: SNMP over OSI, SNMP over DDP (Appletalk) ou SNMP over IPX (Novel). Mais tarde foram definidas outras formas de transporte para o SNMP (RFC 4490 refere SNMP sobre UDP, TCP, SSH e TLS);
- As operações GetRequest e GetNextRequest no SNMPv1 podem solicitar mais de um objeto gerenciado de cada vez. No entanto, se um identificador apresentar problemas, será gerada uma resposta de erro. Ou todos os objetos são retornados, ou nenhum deles. No SNMPv2, isso não ocorre. Se houver um erro do tipo noSuchObject, noSuchInstance e endOfMibView para um objeto gerenciado pedido, os outros serão retornados;
- Foi criado um novo tipo de Trap, conhecido pelo nome Inform, que apresenta a característica de ser respondido pelo gerente. A intenção é diminuir a possibilidade de perda da notificação;
- A fim de permitir a rápida recuperação de uma porção de uma tabela, foi criada a operação GetBulkRequest. Como parâmetro dessa operação, são passados objetos gerenciados, um contador que indica quantos objetos são escalares e, portanto, não devem ser pesquisados como uma tabela, e outro contador que indica quantas instâncias dos outros objetos devem ser buscadas no máximo;
- A operação setRequest passou a ser executada em duas fases. Na primeira, as variáveis são testadas, e somente na segunda são alteradas, com vistas a manter a atomicidade das alterações solicitadas (tudo ou nada).

## SNMPv2 – Operações

Com a não aceitação do modelo de segurança proposto em SNMPv2p, a mensagem SNMPv2c possui os mesmos delimitadores e cabeçalhos da versão SNMPv1, com exceção do campo “versão”, que agora tem valor 1 indicando a versão 2, além da manutenção do mesmo esquema de comunidades da versão 1.

A figura 3.13 mostra o mesmo cabeçalho e as diferenças nas operações: comunicação entre gerentes, a adição da mensagem InformRequest, a mudança de nome da mensagem GetResponse para Response e a adição da mensagem GetBulkRequest.





Versão	Comunidade	PDU (Protocol Data Unit)
--------	------------	--------------------------

**Figura 3.13**  
Mensagens do protocolo SNMPv2.

Os valores do campo tipo de PDU no protocolo SNMP v2 são:

```

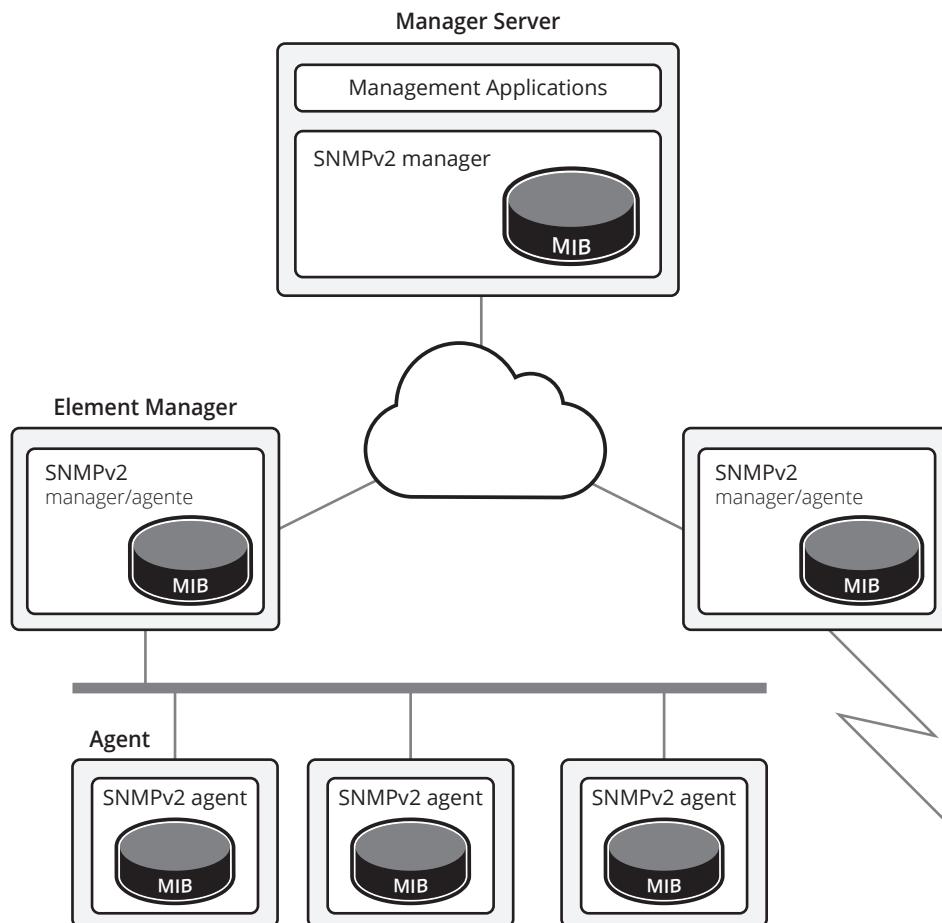
GetRequest-PDU = [0]
GetNextRequest-PDU = [1]
Response-PDU = [2]
SetRequest-PDU = [3]
-- [4] que era o antigo Trap foi obsoletado
GetBulkRequest-PDU = [5]
InformRequest-PDU = [6]
SNMPv2-Trap-PDU = [7]
Report -PDU = [8]

```

## SNMPv2 – Operação ‘InformRequest’ e o gerenciamento hierarquizado

Como descrito anteriormente, há uma nova mensagem semelhante ao Trap no SNMPv2. O InformRequest é um Trap com confirmação. Essa mensagem é muito interessante para a criação de uma hierarquia de gerentes, conforme a figura 3.14.

Um gerente envia um InformRequest a outro em resposta a eventos complexos, como a ultrapassagem de um limite máximo de erros ou muitas falhas em operações. Essa nova possibilidade permite a hierarquização da estrutura gerencial do SNMPv2.



**Figura 3.14**  
Hierarquia de gerentes.

## SNMPv2 – Operação ‘getBulkRequest’

No SNMPv1, quando eram desejadas grandes quantidades de dados de uma MIB, a limitação do tamanho das respostas era função da implementação no agente distante. Se era pedido um volume muito grande de informações para um único Get, a resposta era vazia com o campo *error status* com valor “*tooBig*”. Se era pedido um volume muito pequeno de informações para um único Get, a coleta de dados não era eficiente.

A nova operação GetBulkRequest otimiza a recuperação de um volume considerável de variáveis, pedindo eficientemente o máximo de dados que um agente pode enviar por meio de uma mensagem “response”. O pedido realizado em uma mensagem GetBulkRequest pode ser tanto de variáveis individuais ou de linhas de uma tabela.

A mensagem GetBulkRequest pode ser constituída de vários objetos gerenciáveis, assim como a mensagem Get ou GetNext. No entanto, em uma mensagem GetBulkRequest é possível indicar o número de repetições que se deseja realizar. Esse número de repetições corresponde a repetidas operações *getNext* sem necessidade de novas mensagens.

A figura 3.15 apresenta um exemplo de GetBulkRequest. Os dois primeiros objetos são escalares e lidos como uma instância única. Os outros objetos fazem parte de uma tabela e apresentam várias instâncias. No exemplo, são buscadas seis instâncias de cada objeto.

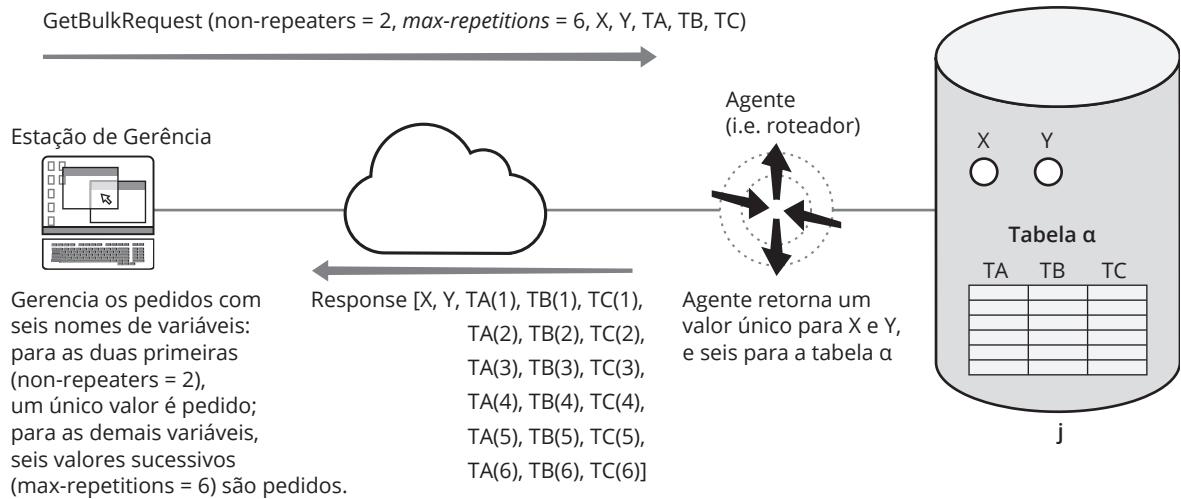


Figura 3.15

Exemplo de consulta usando GetBulkRequest.

## SNMPv2 – Mensagens

Como é possível observar na figura 3.16, na versão dois do protocolo SNMP houve uma unificação de formatos de mensagens, evitando assim o overhead ao processar diferentes PDU. Dessa forma, ao enviar uma mensagem do tipo Get, envia-se os campos de status e índice de erro com valor 0. Já nas mensagens getbulkRequest, tais campos são utilizados para informar os itens:

- **non-repeaters:** especifica o número máximo de objetos escalares a ser enviado no “response”;
- **max-repetitions:** define o número máximo de vezes que deve se seguir pelas variáveis de várias instâncias.

Tipo PDU	Request-ID	0	0	Variable-bindings
----------	------------	---	---	-------------------

(a) GetRequest-PDU, GetNextRequest-PDU, SetRequest-PDU, SNMPv2-Trap-PDU, InformRequest-PDU

Tipo PDU	Request-ID	Status Erro	Índice Erro	Variable-bindings
----------	------------	-------------	-------------	-------------------

(b) Response-PDU

Tipo PDU	Request-ID	Non-repeaters	Max-repeaters	Variable-bindings
----------	------------	---------------	---------------	-------------------

(c) GetBulkRequest-PDU

Nome 1	Valor 1	Nome 2	Valor 2	...	Nome n	Valor n
--------	---------	--------	---------	-----	--------	---------

(d) Variable-bindings

Figura 3.16  
Formato das mensagens do SNMPv2.

## SNMPv2: Tipos de agentes

Agentes extensíveis.

- Arquitetura aberta.
- Design modular.
- Permite adaptações para novos requisitos.



Agentes monolíticos.

- Não são extensíveis.
- Otimizados para determinadas plataformas de hardware e SO.
- Melhor desempenho.

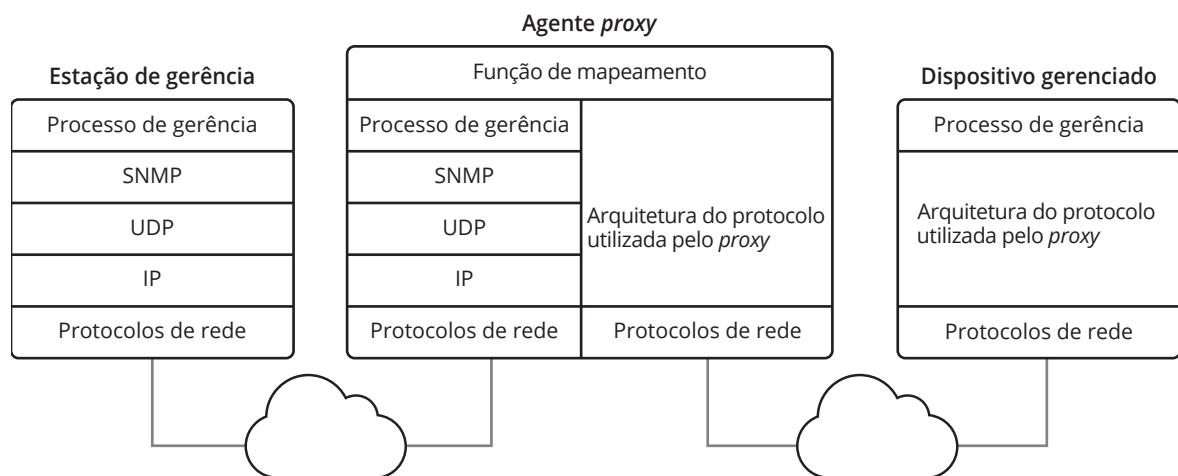


Agentes SNMP basicamente podem ser construídos de duas formas:

- **Agentes extensíveis:** são desenvolvidos com arquitetura aberta, com design modular permitindo adaptações para novos requisitos de dados de gerenciamento e extensões operacionais;
- **Agentes monolíticos:** não são extensíveis. São construídos com otimizações para determinadas plataformas de hardware ou SO, visando melhor desempenho.

## Agentes proxy

Um agente proxy é uma entidade que repassa uma mensagem para outro agente. Isso pode ser necessário para acessar um agente que não fala SNMP ou para acessar um agente que não apresenta suporte de rede TCP/IP. A figura 3.17 apresenta uma agente proxy interconectando um gerente a um dispositivo gerenciado que não está na rede TCP/IP.



Além das duas situações apresentadas, um agente proxy pode ser utilizado para aumentar a segurança de um determinado ambiente de rede.

**Figura 3.17**  
Agente proxy.

## SNMPv3: Novo modelo de segurança

User-based security model.

- autenticação.
- Privacidade.

View-based access control.

- tipos de usuários.
- partes da MIB.



Para mais informações:  
<http://www.snmplink.org/>

Entre outras modificações, uma das mais significativas realizadas na versão 3 do protocolo SNMP está a implementação de um modelo de segurança mais seguro. Cada mensagem SNMPv3 apresenta parâmetros de segurança. São usadas chaves para garantir a autenticação e a privacidade na comunicação entre gerentes e agentes.

A fim de determinar que parte da MIB pode ser vista por determinado usuário, é utilizado o conceito de MIB view. Esse controle é feito através de várias tabelas:

- vacmContextTable;
- vacmSecurityToGroupTable;
- vacmAccessTable;
- vacmViewTreeFamilyTable.

Para cada acesso, é necessário que o agente verifique se o usuário tem direito de fazê-lo.

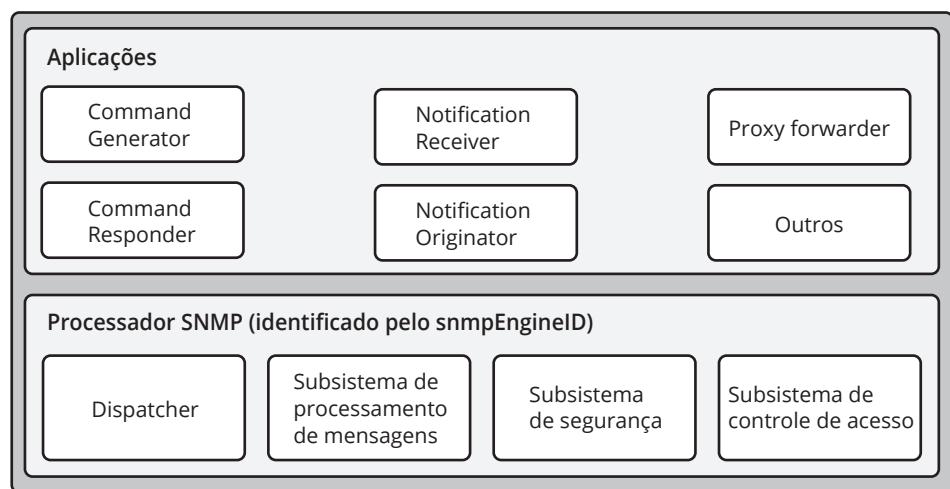
Para tal, o usuário é mapeado para uma MIB view que consiste dos objetos que podem ser acessados. Haverá diferentes visões de acordo com os diferentes modelos de segurança (com criptografia, com autenticação etc.) e de acordo com a operação que está sendo realizada.

Com a implementação do SNMPv3, é possível utilizar o protocolo para atuar sobre agentes, já que ele apresenta esquema de segurança suficientemente robusto.

## Arquitetura modular SNMPv3

A arquitetura SNMPv3 (figura 3.18) é composta por módulos que interagem provendo serviços uns aos outros, através de primitivas.

Figura 3.18  
Arquitetura  
SNMPv3.



Os conceitos principais da arquitetura SNMPv3 são:

- **Entidades SNMP:** corresponde ao que se chamava de agente e gerente SNMP, composto por duas partes: aplicações e engine;
- **SNMP engines:** o engine é composto por quatro partes:
  - **Dispatcher:** responsável por enviar e receber mensagens;
  - **Subsistema de processamento de mensagem:** responsável por preparar a mensagem para enviar ou extrair os dados, conforme os protocolos disponíveis;
  - **Subsistema de segurança:** responsável pelo controle de autenticação e privacidade (criptografia) das mensagens;
  - **Subsistema de controle de acesso:** responsável por determinar se o acesso deve ser liberado.
- **Aplicações SNMPv3:** existem cinco tipos de aplicações internas:
  - **Geradores de comando:** geram comando para coletar dados ou alterar valores;

- **Respondedores de comando:** proveem acesso aos dados;
- **Originadores de notificação:** iniciam uma mensagem do tipo trap ou inform;
- **Recebedores de notificação:** recebem as mensagens do tipo trap ou inform;
- **Encaminhadores de proxy:** encaminham mensagem entre entidades SNMP.

Um gerente ou agente, agora entidades SNMP, executarão suas funções através da interação dos elementos citados.

## Segurança SNMPv3

RFC 3414 – The User-Based Security Model for Version 3 of the Simple Network Management Protocol (SNMPv3).

- Descrição de ameaças, mecanismos de segurança, algoritmos de segurança e tipos de dados.

RFC 3415 – View-Based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP).

- Descrição do mecanismo de controle de acesso às informações de gerenciamento.

A segurança do protocolo SNMPv3 é baseada em dois documentos fundamentais:

- **RFC 3414 – The User-Based Security Model for Version 3 of the Simple Network Management Protocol (SNMPv3):** descreve ameaças, mecanismos, algoritmos, serviços de segurança e tipos de dados usados com o objetivo de prover segurança ao protocolo;
- **RFC 3415 – View-Based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP):** descreve como é definido o controle de acesso às informações de gerenciamento.

O modelo proposto de segurança baseado em usuário é flexível, de forma que no futuro novos algoritmos podem ser usados sem alterações no modelo. Seu único protocolo de privacidade proposto é baseado no algoritmo DES.

Ao enviar uma mensagem, pode-se optar por uma das seguintes opções (campo “msgFlags”):

- Sem autenticação, sem privacidade;
- Com autenticação, sem privacidade;
- Com autenticação, com privacidade.

Nesse modelo, autentica-se um usuário chamado de “principal”.

O modelo de segurança é responsável pelo processamento da segurança da mensagem e pela implementação de mecanismos para sua garantia. Um único modelo de segurança foi proposto até o momento para SNMPv3: o modelo User-Based Security Model (USM), embora nada impeça a oferta de outros modelos de segurança no futuro. As ameaças consideradas pelo modelo de segurança baseado em usuário concentram-se no trajeto da informação pela rede, levando em consideração as características do tráfego SNMP (fragmentado em muitas mensagens).

O modelo de segurança baseado em usuário oferece os seguintes serviços:

- Autenticação: protocolos HMAC-MD5-96 e HMAC-SHA-96;
- Privacidade: protocolo DES em modo CBC;
- Proteção contra atrasos e reenvios de mensagens através de mecanismos de temporização.



Ameaça	Tratada?	Mecanismo
Replay de mensagens	Sim	<i>timestamp</i>
Mascaramento	Sim	HMAC (MD5 ou SHA)
Integridade	Sim	HMAC (MD5 ou SHA)
Privacidade	Sim	DES
Negação de serviço	Não	
Análise de tráfego	Não	

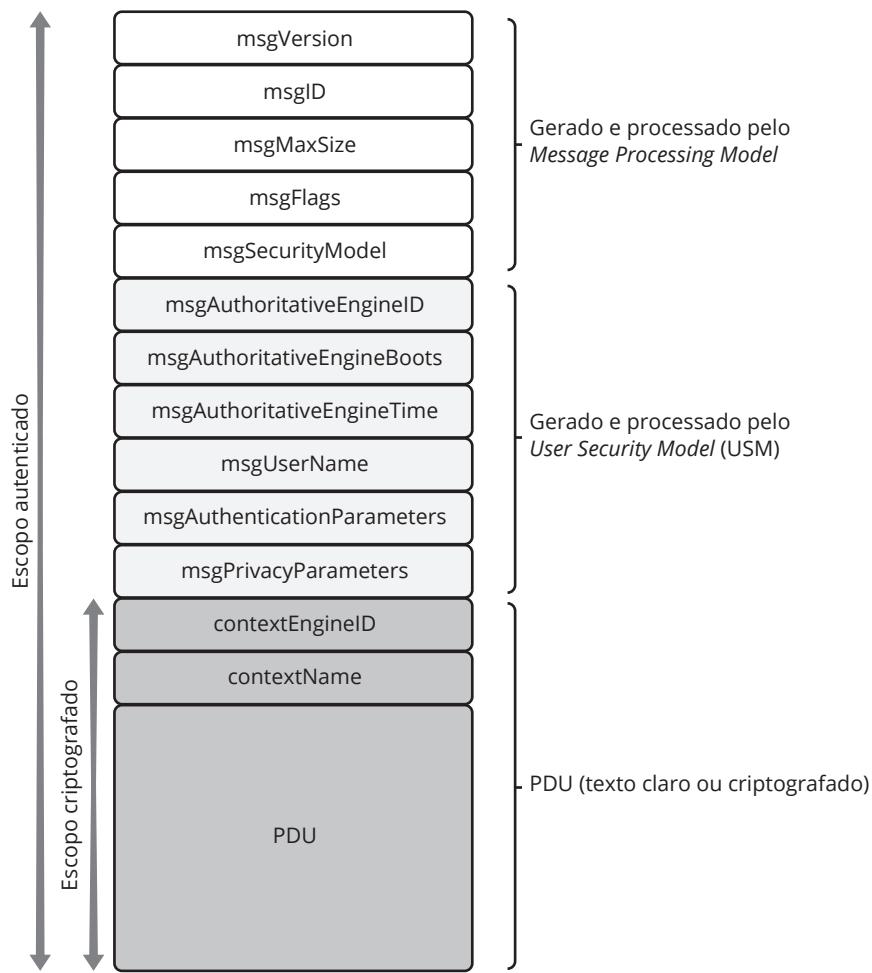
**Tabela 3.3**  
Tabela de ameaças.

## Formato da mensagem SNMPv3

O protocolo SNMPv3 apresenta um novo formato de mensagem, necessário pela sua maior complexidade. A figura 3.19 apresenta os campos descritos a seguir:

- **msgVersion**: especifica a versão do protocolo em uso (3 – SNMPv3);
- **msgID**: identificador usado para dar coordenadas requests e responses;
- **msgMaxSize**: informa o maior tamanho de mensagem suportado;
- **msgFlags**: string de bits que identificam a existência de report, autenticação e encriptação na mensagem;
- **msgSecurityModel**: identifica o modelo de segurança em uso. Valores possíveis até o momento: SNMPv1, SNMPv2c e USM;
- **msgSecurityParameters**: string com parâmetros a serem processados pelo Security Subsystem (depende do modelo de segurança);
- **contextEngineID**: identifica uma engine que está sob determinado contexto. Define a aplicação a qual o PDU está relacionado;
- **contextName**: identificador do contexto usado pelo PDU. É passado como parâmetro para o dispatcher e o View-based Access Control Model (VACM);
- **PDU**: é o PDU SNMPv2c.





**Figura 3.19**  
Formato da  
mensagem do  
protocolo SNMPv3.

## SNMP Agente: NET-SNMP

- Agente SNMP: snmpd
- Arquivos de configuração:
  - /etc/snmp/snmp.conf
  - /etc/snmp/snmpd.conf

O pacote NET\_SNMP (<http://net-snmp.sourceforge.net/>) é o pacote SNMP padrão no mundo Linux, possuindo também versões para Windows. Ele é composto de um agente, várias aplicações em linha de comando e uma biblioteca SNMP. Ele pode ser encontrado também nos repositórios dos Sistemas Operacionais Linux, sendo necessário instalar os pacotes snmp e snmpd.

Seu agente, o “snmpd”, pode ser configurado para qualquer versão SNMP (1, 2 e 3) sobre IPv4 ou IPv6. Além disso, o agente suporta várias MIBs, sendo extensível (SMUX e AgentX).

Os arquivos de configuração do pacote estão no diretório “/etc/snmp/”. Para o agente, os arquivos relevantes são:

- *snmp.conf*: configurações padrão para as aplicações;
- *snmpd.conf*: configurações específicas para o agente;
- O arquivo *snmpd.conf* possui inúmeras opções. Para mais informações, veja a man page relacionada através do comando *man snmpd.conf*.



## NET-SNMP – Aplicações

```
Snmpget          snmpgetnext  
snmpwalk        snmphtable  
snmpdelta  
snmpdf  
snmpnetstat     snmpstatus  
snmpset  
snmptranslate  
tkmib  
snmptrap  
snmptrapd
```

Entre as aplicações disponíveis no pacote, as mais utilizadas são:

- ▣ **snmpget, snmpgetnext:** obtém informação de dispositivos SNMP com requests;
- ▣ **snmpwalk:** obtém toda a MIB implementada no agente através de múltiplos requests;
- ▣ **snmphtable:** obtém uma tabela SNMP completa;
- ▣ **snmpdelta:** monitora as alterações nas variáveis de uma MIB;
- ▣ **snmpdf:** obtém informações de espaço em disco na máquina remota;
- ▣ **snmpnetstat:** obtém informações de rede (status e configuração);
- ▣ **snmpstatus:** obtém informações de estado do dispositivo remoto;
- ▣ **snmpset:** altera a configuração de dispositivos SNMP;
- ▣ **snmptranslate:** mostra o conteúdo de MIBs em formato numérico ou textual;
- ▣ **tkmib:** um “MIB browser” gráfico desenvolvido em Tk/Perl;
- ▣ **snmptrap:** envia traps SNMP;
- ▣ **snmptrapd:** recebe notificações SNMP e dá destino a elas (log repassa para outro gerente SNMP ou para outra aplicação). Possui um arquivo de configuração próprio: /etc/snmp/snmptrapd.conf.

## NET-SNMP – Configuração

- ▣ Utilitário para configuração: snmpconf
- ▣ Utilitário para testes: snmpitest



Os arquivos de configuração do pacote estão no diretório “/etc/snmp/”.

Cada um deles possui sintaxes com várias opções. O comando *snmpconf* auxilia na configuração desses arquivos. Esse comando, inclusive, pode comentar um arquivo de configuração existente, visando torná-lo mais legível.

O comando *snmpitest* inicia um diálogo interativo com um agente SNMP. Através desse é possível enviar vários comandos SNMP e obter respostas.



Para que o agente instalado funcione e possa ser testado é necessário:

- ▣ Instalar o pacote `snmp-mibs-downloader`, que baixa todas as MIBs padrão. Sem isso, o acesso através de nomes da árvore de OIDs fica prejudicado;
- ▣ Comentar a linha “`mibs:`”, presente no arquivo `/etc/snmp/snmp.conf`;
- ▣ Atualizar as MIBs com o comando `download-mibs`;
- ▣ Ajustar no arquivo `snmpd.conf` para que computadores podem realizar consultas (`agentAddress udp:161,udp6:[::1]:161`);
- ▣ Permitir que a comunidade pública leia os objetos a partir de qualquer equipamento (`rcommunity esr-rnp default`).

## Supporte a SNMPv3 no pacote NET-SNMP

Duas configurações impactam a troca de mensagens:

- ▣ Configurações de usuários USM.
- ▣ Configurações de controle de acesso VACM.

Comandos relacionados:

- ▣ `snmpconf`
- ▣ `snmpusm`
- ▣ `snmpvacm`



Como foi visto nesta sessão de aprendizagem, o SNMPv3 apresenta várias novidades, principalmente no que diz respeito à segurança na troca de mensagens SNMP.

O agente NET-SNMP oferece suporte à SNMPv3 através de sua configuração. O arquivo de configuração do agente (`/etc/snmpd.conf`) deve ser adequado para uso com SNMPv3.

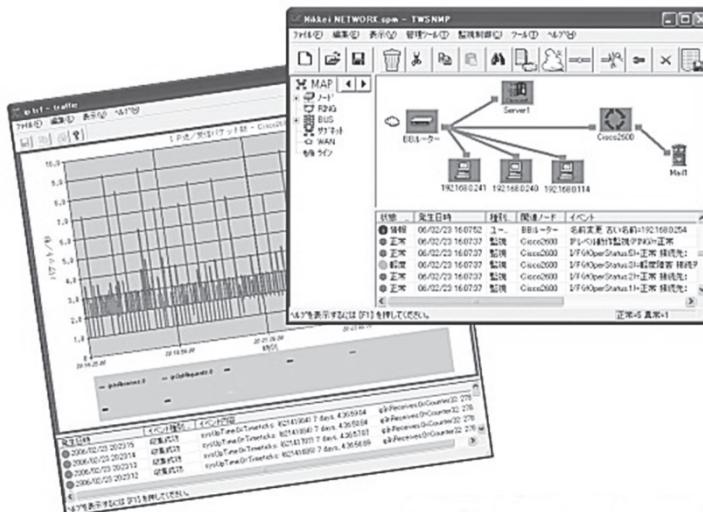
Mais uma vez, o script “`snmpconf`” pode auxiliar nessa configuração; mas lembre-se de que nem todas as opções possíveis são configuradas por ele.

O comando `snmpusm` é usado para a criação de usuários do modelo de segurança USM.

O comando `snmpvacm` é usado para a criação de entradas no controle de acesso VACM.

Para mais informações, use as man pages:

- ▣ `man snmpconf`
- ▣ `man snmp_config`
- ▣ `man snmpusm`
- ▣ `man snmpvacm`



**Figura 3.20**  
Telas do NET-SNMP.

## Criação de usuários SNMPv3

No arquivo `/var/lib/snmp/snmpd.conf`:

- `createUser [-e ENGINEID] username (MD5|SHA) authpassphrase [DES|AES] [privpassphrase]`
- `rouser USER [noauth|auth|priv [OID]]`

Para se criar um usuário de nome “initial”, acrescenta-se a linha “`createUser initial MD5 password DES`” no arquivo no arquivo `/var/lib/snmp/snmpd.conf`.

Essa linha informará o agente das passwords (de autenticação e privacidade) que serão usadas pelo usuário nesse agente. Na carga do agente, ele lê essa linha, gera as chaves localizadas correspondentes e apaga a linha.

O motivo é manter gravadas em disco somente as chaves localizadas, e não mais a password original do usuário.

## Uso das aplicações com SNMPv3

Opções SNMPv3 para as aplicações em linha de comando:

- Na última coluna estão palavras-chave para o arquivo `snmp.conf` (para manter as configurações como default).

Parameter	Command Line Flag	snmp.conf token
securityName	<code>-u NAME</code>	<code>defSecurityname NAME</code>
authProtocol	<code>-a (MD5   SHA)</code>	<code>defAuthType (MD5 SHA)</code>
privProtocol	<code>-x (AES   DES)</code>	<code>defPrivType DES</code>
authkey	<code>-A PASSPHRASE</code>	<code>defAuthPassphrase PASSPHRASE</code>
privKey	<code>-X PASSPHRASE</code>	<code>defPrivPassphrase PASSPHRASE</code>
securityLevel	<code>-l (noAuthNo Priv authNoPriv authPriv)</code>	<code>defSecuritylevel (noAuthNo Priv authNoPriv authPriv)</code>
context	<code>-n CONTEXTNAME</code>	<code>defContext CONTEXTNAME</code>

- ▣ Cada usuário possui um nome (securityName), uma opção de autenticação (authProtocol), uma opção de privacidade (privProtocol), além das chaves criptográficas correspondentes (authKey e privKey);
- ▣ A autenticação é feita “assinando” a mensagem com a chave do usuário em questão, usando um dos protocolos possíveis (HMAC-MD5 ou HMAC-SHA). As chaves são geradas a partir de uma senha de, no mínimo, 8 caracteres;
- ▣ A encriptação é feita codificando a parte de dados da mensagem com um dos protocolos possíveis (AES ou DES);
- ▣ O “securityLevel” de uma mensagem indica se ela terá autenticação e/ou privacidade;
- ▣ Todas essas informações são passadas para as aplicações em linha de comando. Valores default podem ser armazenados em arquivo *snmp.conf*.

Exemplo de “getRequest” SNMPv3 autenticado:

```
snmpgetnext
-v 3 (versão=SNMPv3)
-n "" (contexto)
-u MD5User (securityName)
-a MD5 (protocolo de autenticação)
-A "Frase" (passphrase de autenticação)
-l authNoPriv (securityLevel)
test.net-snmp.org (destino)
sysUpTime (objeto pedido)
```

#### **Comando**

```
snmpgetnext -v 3 -n "" -u MD5User -a MD5 -A "Frase" -l authNoPriv
test.net-snmp.org sysUpTime
```

#### **Resposta**

```
system.sysUpTime.0 = Timeticks: (83491735) 9 days, 15:55:17.35
```



# 4

## Monitoramento remoto – RMON, RMON2, SMON e Host MIB

objetivos

Conhecer os monitores de redes, aprender as características da MIB RMON e RMON2 e suas extensões SMON. Aprender as características do monitor de tráfego ntop. Entender o gerenciamento de hosts e sistemas.

conceitos

Monitores, RMON – Objetivos e Configuração, Grupos da MIB RMON, Eventos e alarmes, RMON 2, SMON, Monitor ntop, Gerenciamento de hosts e sistemas.

### Monitores

Monitores são equipamentos e/ou softwares usados para observar e controlar uma determinada LAN ou conjunto de dispositivos. São também chamados de “probes” e possuem algum tipo de inteligência, além de armazenar dados coletados.

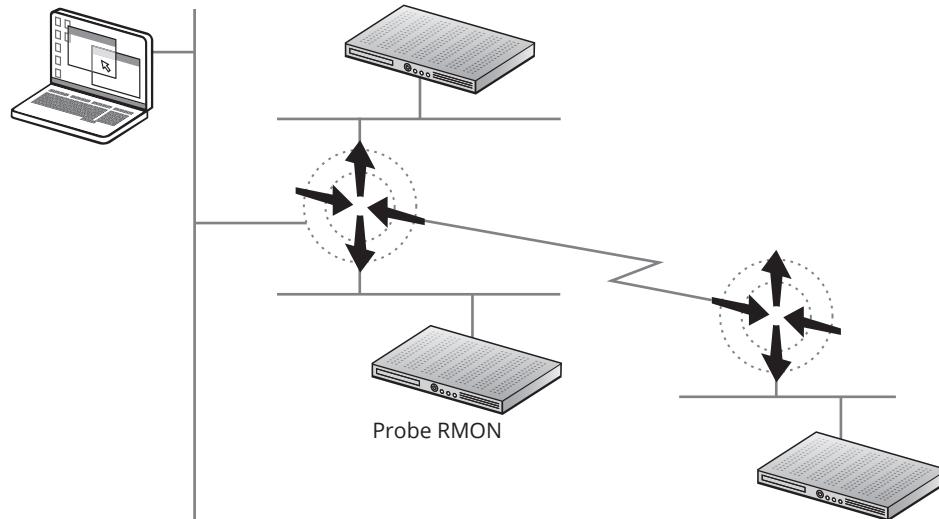
Monitores são úteis na medida em que conhecem o comportamento normal da rede e alertam um gerente, com uma mensagem de trap, sobre a ocorrência de uma anormalidade. Para tal, precisam conhecer limites de erros e perfis de tráfego.

Uma característica importante de um monitor é a sua independência em relação ao gerente. Mesmo que ocorra uma falha no gerente, os monitores continuam a coletar dados e a armazená-los. Podem ter múltiplas interfaces de rede, permitindo o monitoramento de várias redes locais distintas ou do tráfego entre elas.

Monitores sabem o que ocorre na rede e possuem inteligência para reagir a certas situações, sendo extremamente úteis. A figura 4.1 mostra o desenho de uma rede com um monitor, chamado Probe RMON, e um gerente.

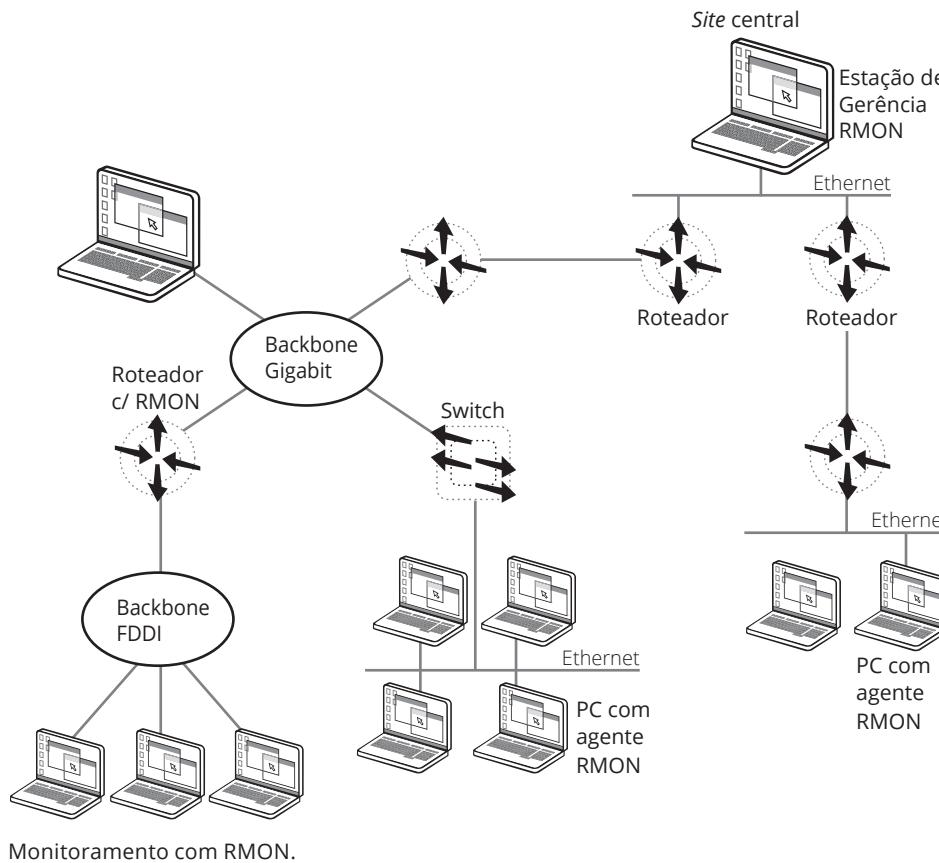


**Estação de Gerência RMON**



**Figura 4.1**  
Rede local com  
um monitor  
(Probe RMON).

A figura 4.2 apresenta uma rede mais complexa em que a função de monitoramento está espalhada por diversos dispositivos. É possível visualizar dois computadores pessoais e um roteador funcionando como monitores.



**Figura 4.2**  
Rede com três  
monitores.



## RMON – Objetivos

- 1. Operação independente da estação gerenciadora;
- 2. Monitoramento proativo (depende de recursos e da rede);
- 3. Detecção de problemas – monitoramento preemptivo;
- 4. Dados de valor adicionado (especialização);
- 5. Suporte a vários gerentes.

RMON significa Remote Network Monitoring e corresponde a uma MIB definida nas seguintes RFCs:

- **RFC 2819:** RMON1;
- **RFC 4502:** RMON2.

As MIBs I e II oferecem somente informações de dispositivos individuais. RMON oferece informações que dizem respeito à rede de forma distribuída, como contadores de pacotes e erros em uma determinada LAN. O objetivo dessa MIB é permitir o monitoramento da rede e o posterior tratamento por um gerente.

Através dos objetos gerenciados o monitor RMON é configurado, armazena informações e envia traps aos gerentes. A gerência ocorre através do protocolo SNMP, permitindo que diferentes gerentes sejam utilizados. Os dados de configuração e monitoração são armazenados em tabelas definidas na MIB RMON. A MIB também permite o compartilhamento do monitor entre várias estações gerentes.

Os grupos de objetos RMON ficam abaixo do ramo da MIB-II.

## Configuração do RMON

- Diz respeito a como o monitor fará a coleta dos dados.
- Usa duas tabelas: tabela de controle (configuração das coletas) e tabela de dados (resultados).
- Dois tipos de dados da tabela de controle definem o acesso aos dados:
  - OwnerString: é uma string que identifica o dono da entrada na tabela.
  - EntryStatus: é um valor inteiro que indica o estado atual da entrada na tabela.  
Valores possíveis:
    - valid (1), createRequest (2), underCreation (3), invalid (4).

A configuração do RMON basicamente diz respeito ao que se deseja monitorar e aos limites a serem testados. Para cada estatística, existem parâmetros definidos pelo gerente (exemplo: intervalo de medição).

Normalmente, existem duas tabelas: uma de controle (“control table”) e uma de resultados do monitoramento (“data table”). A partir do momento em que a tabela de controle é configurada, a monitoração vai gerar informações que serão dispostas na tabela de dados.

As tabelas de controle apresentam dois campos que evitam problemas na coleta de dados:

- **OwnerString:** diferentes gerentes podem solicitar determinada monitoração, por isso, há necessidade de um campo que identifique quem fez o pedido; cada entrada nessa tabela possui um identificador que será usado depois para reconhecer o dado coletado na tabela de resultados;

- **EntryStatus:** para encerrar uma monitoração, deve-se alterar o status da linha correspondente para invalid, a fim de não perder os dados; para criar uma monitoração, o status deve ter o valor createRequest; a monitoração inicial quando o status for igual a valid.

Alguns problemas que podem surgir com a utilização de um mesmo monitor por vários gerentes:

- Muitos pedidos concorrentes de vários gerentes podem sobrecarregar a capacidade de um monitor;
- Uma única estação gerente pode alocar recursos do monitor e mantê-los por muito tempo;
- Uma estação gerente pode alocar recursos e sofrer um problema, impedindo a liberação desses recursos para outros gerentes.

## Grupos da MIB RMON

- Grupos de estatísticas de tráfego e erro:
  - statistics (1)
  - history (2)
  - host (4)
  - hostTopN (5)
- Matriz de tráfego entre sistemas:
  - matrix (6)
- Grupos de filtragem e captura de tráfego:
  - filter (7)
  - packet capture (8)
- Grupos de alarmes e eventos:
  - alarm (3)
  - event (9)



A RMON 1 permite o monitoramento do tráfego no nível de enlace. Ela é composta por nove grupos de objetos gerenciados principais:

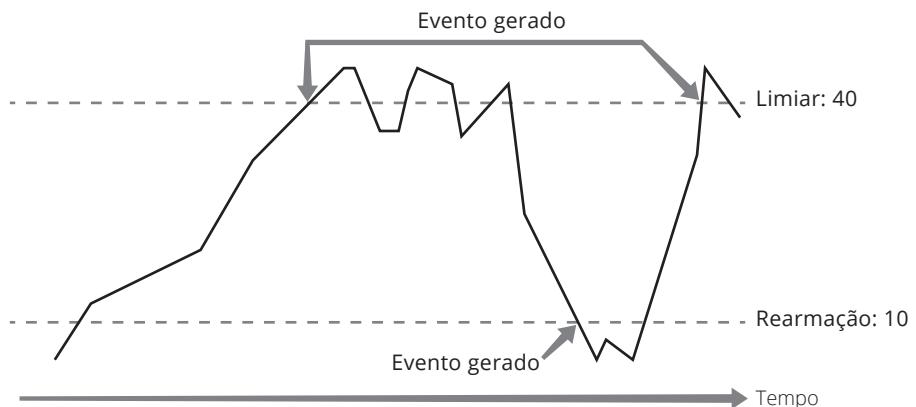
- **statistics (1):** contém estatísticas de tráfego (octetos, colisões, erros etc.) para cada interface e rede;
- **history (2):** controla a amostragem estatística periódica dos diversos tipos de rede e armazena as amostras para posterior recuperação;
- **alarm (3):** o grupo contém variáveis que devem ser vigiadas, e os eventos que serão disparados no caso dessas variáveis ultrapassarem certos limites. Esses limites definem os indicativos de problemas e de normalidade. Sua implementação exige o grupo "event";
- **host (4):** contém estatística para cada host na rede;
- **hostTopN (5):** controla e armazena relatórios com os hosts que apresentarem números mais relevantes (tops) para diferentes estatísticas. Os hosts são ordenados e apresentados em uma tabela para fins de geração de relatórios. Requer a implementação do grupo host;
- **matrix (6):** armazena estatísticas para conversas entre pares de hosts. Aqui há o reconhecimento das origens e destinos dos pacotes que trafegam na rede. Uma entrada é criada para cada nova informação de comunicação entre dois endereços obtida de pacotes recebidos. Útil para detecção de intrusos;

- ▣ **filter (7)**: define critérios de filtragem dos pacotes;
- ▣ **capture (8)**: permite o armazenamento dos resultados da filtragem feita. Requer a implementação do grupo filter;
- ▣ **event (9)**: define cada evento, seu tipo e a última ocorrência. Eventos normalmente são gerados pelo atingimento de um limite previamente estabelecido no grupo "alarm" ou como resultado de um filtro definido grupo "filter". Pode definir ações, como notificar um gerente via trap ou atualizar um arquivo de log, ou ainda acionar uma captura de tráfego.

## Eventos e alarmes

Alarmes podem ser configurados para dispararem em resposta a eventos observados pelo monitor. Por exemplo, o monitoramento do número de erros em um segmento de LAN pode ser parametrizado com um limite superior ("rising threshold") e um inferior ("falling threshold").

Eventos podem ser configurados para que sejam disparados em função dos parâmetros acima. Por exemplo, o envio de uma trap que alerta para o alto número de erros na LAN e de outra trap informando o retorno a um estado de normalidade (figura 4.3).



**Figura 4.3**  
Eventos e alarmes.

## RMON 2

Diferença para a RMON 1:

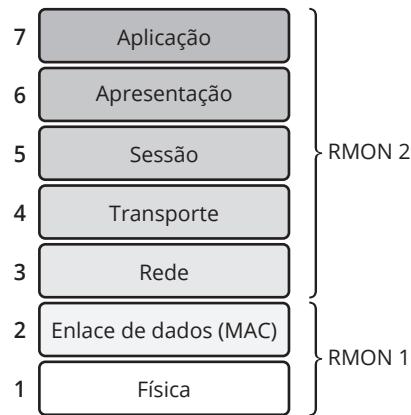
- ▣ Camada de rede.
- ▣ Camada de aplicação.

Grupos:

- ▣ Protocol Directory (11).
- ▣ Protocol Distribution (12).
- ▣ Address Map (13).
- ▣ Network Layer Host (14).
- ▣ Network Layer Matrix (15).
- ▣ Application Layer Host (16).
- ▣ Application Layer Matrix (17).
- ▣ User History Collection (18).
- ▣ Probe Configuration (19).



Como visto, RMON 1 captura e analisa quadros (nível de enlace). Já a MIB RMON2 permite a análise das camadas superiores da arquitetura TCP/IP (figura 4.4).



**Figura 4.4**  
Camadas monitoradas pela RMON1 e RMON2.

Com isso, os fenômenos de tráfego podem ser melhor compreendidos (fatos como aplicações mais “pesadas”, servidores mais acessados, protocolos mais exigidos, etc.). Qualquer camada acima de rede é chamada de camada de aplicação, o que não significa que seja da camada 7.

Com a RMON2 é possível analisar questões pertinentes aos protocolos mais utilizados em uma rede local, por exemplo. Torna-se viável identificar as aplicações que causam momentos de pico no uso da rede e até mesmo tomar decisões quanto à disposição de equipamentos em switches e VLANs.

Os seguintes grupos de objetos gerenciados são definidos na MIB RMON 2 (RFC 4502):

- **Protocol Directory (protocolDir)**: informações sobre os diversos protocolos que o monitor pode analisar;
- **Protocol Distribution (protocolDist)**: dados do tráfego apresentado por protocolo;
- **Address Map (addressMap)**: dados do mapeamento de endereços MAC em endereços de rede e portas;
- **Network Layer Host (nlHost)**: estatísticas do protocolo de rede por host;
- **Network Layer Matrix (nlMatrix)**: estatísticas do protocolo de rede por pares de hosts;
- **Application Layer Host (alHost)**: estatísticas dos protocolos de aplicação hosts;
- **Application Layer Matrix (alMatrix)**: estatísticas dos protocolos de aplicação por pares de hosts;
- **User History Collection (usrHistory)**: dados específicos de usuários;
- **Probe Configuration (probeConfig)**: parâmetros operacionais de configuração do monitor RMON (configurações de interações com interfaces seriais, aspectos de download de informações, destinos de traps etc.).

A seguir apresentamos alguns grupos para melhor compreensão do funcionamento da RMON2.



## RMON2: Grupo Protocol Directory

Função do grupo.

- ▣ Configuração geral.

Tabelas do grupo.

- ▣ protocolDirTable.

O grupo Protocol Directory é a base para configuração da RMON2. Possui uma tabela (protocolDirTable), na qual são definidos os protocolos monitorados. A MIB considera que podem ser monitorados protocolos de rede e de aplicação. Deve-se considerar que no contexto da RMON2 aplicação é qualquer protocolo acima da camada de rede. Isso inclui os protocolos de transporte.

A configuração dessa tabela deve ser realizada antes do agente iniciar. Cada linha dessa tabela estabelece uma monitoração associada a um dono. Dessa forma, é possível configurar que determinado gerente pretende monitorar vários protocolos de rede e aplicação.

## RMON2: Grupo Protocol Distribution

Função do grupo.

- ▣ Estatísticas gerais.

Tabelas do grupo.

- ▣ protocolDistControlTable.
- ▣ protocolDistStatTable.

O grupo Protocol Distribution apresenta uma tabela de controle e outra de dados.

Na tabela de controle (protocolDistControlTable), configura-se as interfaces em que se deseja monitorar os protocolos definidos no grupo anterior. Nela é armazenada, além da interface, o número de quadros não analisados, o momento em que a monitoração foi ativada e o gerente que configurou a entrada.

A tabela de dados (protocolDistStatTable) armazena a quantidade de quadros e o número de octetos de cada protocolo monitorado.

## RMON2: Grupo Network Layer Host

Função do grupo.

- ▣ Estatísticas do protocolo de rede por host.

Tabelas do grupo.

- ▣ hlHostControlTable.
- ▣ nlHostTable.

O grupo Network Layer Host também apresenta duas tabelas. A tabela de controle (hlHostControlTable) permite determinar as interfaces a serem monitoradas (hlHostControlDataSource). A monitoração é associada a um gerente e apresenta um tamanho máximo.

Na tabela de dados (nlHostTable) serão armazenadas estatísticas do protocolo de rede (normalmente IP) para cada endereço: pacotes recebidos sem erro, pacotes enviados sem erro, octetos recebidos, octetos enviados e pacotes não unicast enviados. A leitura periódica dessa tabela permite montar relatórios mostrando o tráfego no nível de rede por host.



## RMON2: Grupo Network Layer Matrix



Função do grupo:

- Estatísticas do protocolo de rede por pares de hosts (conversas).

Tabelas do grupo:

- hlMatrixControlTable
- nlMatrixSDTable
- nlMatrixDSTable
- nlMatrixTopNControlTable
- nlMatrixTopNTable

O grupo Network Layer Matrix permite analisar o tráfego, na camada de rede, entre duas estações. Na tabela de configuração hlMatrixControlTable, o gerente deve determinar a interface a ser monitorada e um determinado limite de armazenamento de pares. Essa tabela também serve para determinar a monitoração de pares de estações no nível de aplicação. Nela, não são descritas as conversas a serem monitoradas, mas que se deseja gerar estatísticas para as mensagens trocadas entre quaisquer pares.

Os resultados da monitoração disparada em hlMatrixControlTable são colocados em duas tabelas. Ambas as tabelas são indexadas pelo protocolo de rede monitorado e pelos endereços. A tabela nlMatrixSDTable tem o endereço origem com precedência na indexação sobre o endereço destino. A tabela nlMatrixDSTable é semelhante à primeira, mas o índice corresponde ao endereço destino antecede o endereço origem.

Também é possível gerar estatísticas sobre os pares que apresentam maior tráfego. Para tal, utiliza-se a tabela nlMatrixTopNControlTable para configurar a monitoração e a tabela nlMatrixTopNTable para recuperar os resultados.

## RMON2: Grupo Application Layer Host



Função do grupo.

- Estatísticas dos protocolos de aplicação hosts.

Tabelas do grupo:

- alHostTable.
- hlHostControlTable (outro grupo mas controla).

O grupo Application Layer Host apresenta uma tabela denominada alHostTable. Essa tabela apresenta informações sobre o tráfego dos protocolos aqui denominados de aplicação (acima da camada de rede) para cada endereço. A configuração de qual interface deve ser monitorada é feita na tabela hlHostControlTable, do grupo Network Host Layer. Para cada protocolo e cada máquina, é possível verificar o número de pacotes de entrada e saída e o número de octetos de entrada e saída. Pode-se, por exemplo, determinar o tráfego HTTP de um determinado servidor.



## RMON2: Grupo Application Layer Matrix



Função do grupo.

- ▣ Estatísticas do protocolos de aplicação por pares de hosts (conversas).

Tabelas do grupo:

- ▣ alMatrixSDTable
- ▣ alMatrixDSTable
- ▣ alMatrixTopNControlTable
- ▣ alMatrixTopNTable
- ▣ hlMatrixControlTable (outro grupo mas controla)

O grupo Application Layer Matrix permite coletar estatísticas de pares de máquinas que utilizem determinado protocolo de aplicação. Ele é semelhante ao grupo Network Layer Matrix, com a diferença de coletar dados das camadas mais altas da arquitetura de redes. Inclusive, a monitoração é controlada na tabela hlMatrixControlTable, do grupo Network Layer Matrix.

Os resultados são dispostos nas tabelas alMatrixSDTable e alMatrixDSTable. Também é possível gerar uma lista dos pares com maior tráfego na tabela alMatrixTopNTable, cujo controle é realizada em hlMatrixControlTable.

## SMON: Remote Network Monitoring MIB Extensions for Switched Networks



1. Dificuldades de uma rede com switches.
2. Extensão da RMON.
  - ▣ RCF 2613.
  - ▣ Grupo smonVlanStats.
  - ▣ Grupo smonPrioStats.
  - ▣ Grupo portCopy.
  - ▣ Grupo dataSourceCaps.

A utilização da MIB RMON é baseada na capacidade de os monitores capturarem o tráfego de uma rede. Tal técnica data da época em que as redes locais eram construídas com hubs e não switches. A utilização de hubs fazia com que mensagens encaminhadas para determinado equipamento fossem replicadas para todas as portas. A natureza broadcast de uma rede baseada em hubs tornava a monitoração do tráfego uma tarefa trivial, exigindo apenas que o monitor tivesse capacidade de processar e armazenar o que estava sendo transmitido.

A utilização de switches, ao mesmo tempo em que proporcionou melhor desempenho e maior segurança para as redes Ethernet, introduziu uma dificuldade de monitoração do tráfego. Com switches, os dados são transmitidos apenas para as portas nas quais o destino está conectado. Dessa forma, a simples colocação de uma estação em um switch não permite a captura dos pacotes que são transmitidos na rede local. Até mesmo quadros de broadcast são restritos a algumas portas na medida em que se utilizam VLANs.



A fim de superar essa dificuldade e permitir o uso de monitores de tráfego em redes com switches, foi criada uma extensão da RMON, denominada SMON. Nessa MIB são definidos quatro grupos responsáveis pela solução de monitoração:

- smonVlanStats
- smonPrioStats
- dataSource
- portCopy

## SMON: Grupo smonVlanStats

1. Função: monitorar pelo identificador da VLAN.
2. Tabelas do grupo.
3. Exemplos de dados monitorados.



O grupo smonVlanStat é responsável pela configuração e monitoração das tabelas da VLANs. A monitoração é baseada no identificador da VLAN, conforme definido no padrão 802.1Q. Através desse grupo, é possível obter informações de alto nível de uso total de uma VLAN e tráfego não unicast.

A tabela smonVlanStatsControlTable permite a configuração da monitoração. Já a tabela smonVlanIdStatsTable contém as estatísticas coletadas. Entre os dados coletados, podem ser citados:

- Número total de pacotes de determinada VLAN;
- Número total de octetos de determinada VLAN;
- Número total de pacotes não unicast de determinada VLAN;
- Número total de octetos não unicast de determinada VLAN;
- Momento da última alteração da estatística.

## SMON: Grupo smonPrioStats



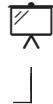
1. Função: monitorar pelo TCI.
2. Tabelas do grupo.
3. Exemplo de dados monitorados.

O grupo smonPrioStat permite a monitoração a partir da definição codificada no campo Tag Control Information (TCI) das VLANs. Os dados relativos a determinada TCI são agrupados em estatísticas. A tabela smonPrioStatsControlTable permite a configuração da monitoração e a tabela smonPrioStatTable contém os dados monitorados. Podem-se citar os seguintes exemplos de contadores:

- Número total de pacotes contados no nível de prioridade selecionado;
- Número de octetos contados no nível de prioridade selecionado.



## SMON: Grupo portCopy



1. Função: copiar quadros para outra porta.
2. Tabela do grupo.

O grupo portCopy permite copiar todos os quadros de determinada origem para uma porta do switch. É possível realizar cópias um-para-um, um-para-muitas, muitas-para-uma ou muitas-para-muitas. Ao realizar essa configuração, deve-se estar atento para a possibilidade de perdas se a porta destino não tiver capacidade de suportar o tráfego gerado. Incentiva-se que os dispositivos de rede suportem pelo menos esse grupo da MIB SMON a fim de que seja possível configurar o espelhamento e algum monitor que implemente RMON possa fazer a captura dos dados.

A tabela portCopyTable gerencia o espelhamento. Cada linha da tabela define um relacionamento entre origem e destino. Os campos *portCopySource*, *portCopyDest*, *portCopyDestDropEvents*, *portCopyDirection* e *portCopyStatus* formam a tabela e definem, respectivamente, a porta que terá os pacotes redirecionados, a porta para aonde os pacotes serão redirecionados, o número de vezes que pacotes não foram redirecionados por falta de recursos, que tipo de dado será redirecionado (recebido, transmitido ou ambos) e o estado operacional do redirecionamento. Deve-se observar que os campos *portCopyDestDropEvents* e *portCopyStatus* não são preenchidos na configuração. São informações geradas pelo monitor SMON.

## SMON: Grupo dataSourceCaps



1. Função: prover informações sobre as portas para monitoramento.
2. Tabela principal.

O grupo dataSourceCaps descreve fontes de dados e capacidade de redirecionamento de portas. Esses dados podem ser utilizados pelo sistema de gerência da rede para descobrir os atributos de um agente. A tabela é preenchida pelo agente SMON.

A tabela dataSourceCapsTable armazena a identificação do objeto (fonte de dados), as capacidades associadas a ele (monitoração de erros, monitoração de sucesso, possibilidade de utilização em qualquer tabela RMON ou apenas para espelhamento, identificação de frames gigantes, e possibilidade de espelhamento).

## ntop

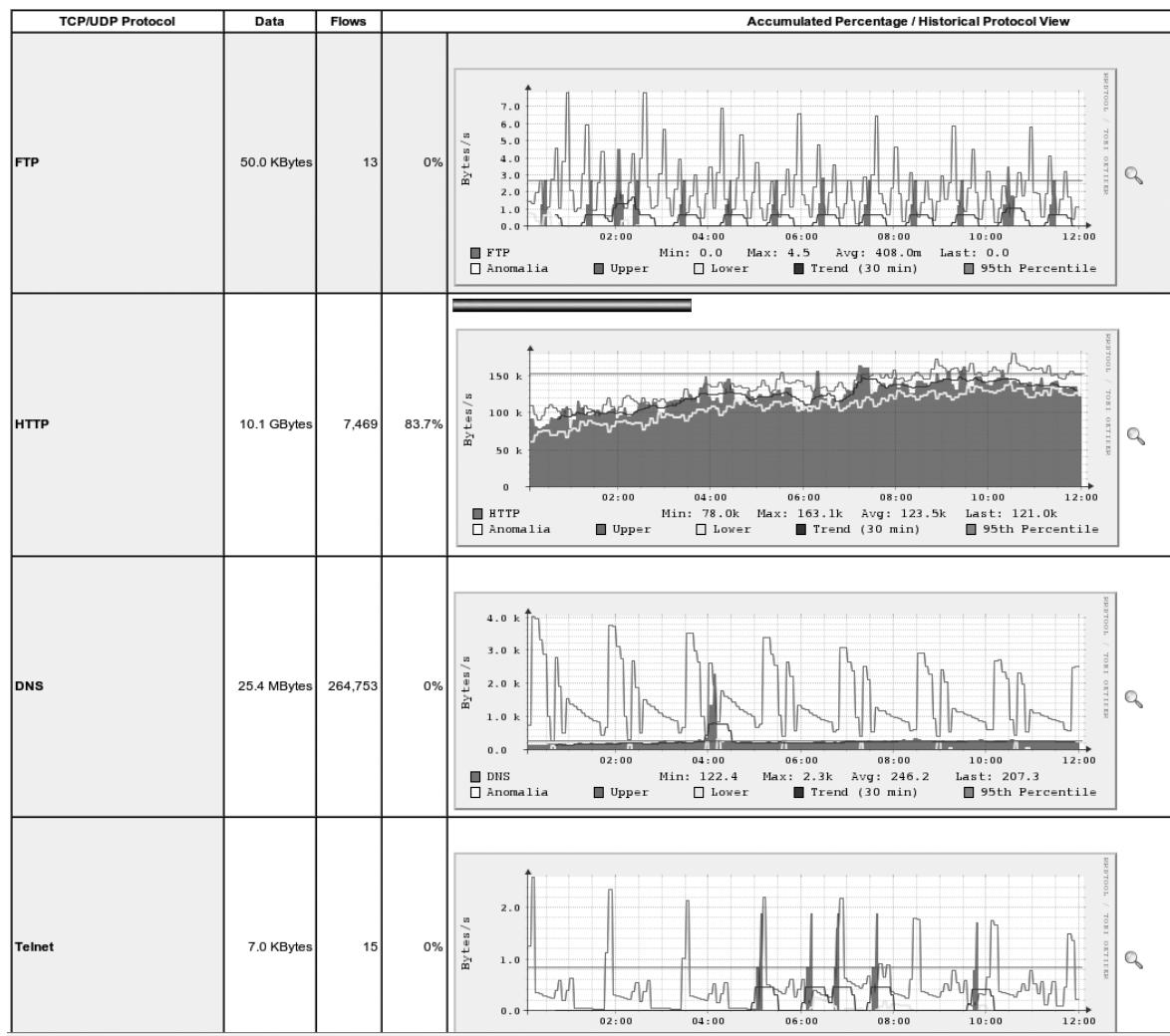


- O que é.
- Características.

O ntop é um monitor de tráfego que mostra a utilização de rede, rodando sobre Unix/Linux e Windows.

Através de um web browser, é possível navegar nas informações de tráfego que ele disponibiliza e ter um snapshot da situação da rede. Sua configuração também é feita via browser. Ele roda normalmente na porta 3000. A figura 4.5 apresenta uma tela do programa ntop.

### Global TCP/UDP Protocol Distribution



Apesar de não suportar o padrão RMON, o ntop se comporta como um monitor remoto com uma interface web que não consome muitos recursos de CPU e memória.

**Figura 4.5**  
Tela do ntop.

Características:

- Ordena a informação de tráfego segundo diversos critérios: protocolo (vários); origem/ destino; matriz de subredes;
- Apresenta estatísticas similares ao RMON;
- Apresenta estatísticas de domínios internet, AS (Autonomous Systems) e VLAN;
- Armazena as informações em disco (formato RRD);
- Identifica passivamente usuários e OS;
- Possui APIs para acesso remoto em Perl/PHP/Python;
- Possui gráficos;
- Possui suporte a HTTPS.



## Gerenciamento de hosts e sistemas

- Hosts representam parcela relevante no tempo de atendimento.
- Precisam ser apropriadamente:
  - configurados (planejamento de capacidade).
  - protegidos (segurança).
  - monitorados (contabilização de uso e gerenciamento proativo).
- Para monitorar a situação de hosts: MIB Host-Resources (RFC 2790).

O termo “host”, no contexto de gerência de redes, implica em um computador qualquer que se comunica com outros computadores interligados via internet, e é diretamente utilizado por usuários humanos. É preciso monitorar as aplicações críticas nos hosts e conter em tempo degradações de serviço isolando aplicações, segmentos de rede que estejam causando problemas de desempenho ou prevenindo uso frívolo da rede.

Gerenciar um host é diferente de dispositivos de rede, porque um host tem mais “inteligência” e tem mais dispositivos a ele conectados. Nesse tipo de equipamento, estão disponíveis mais informações sobre sua atividade, como por exemplo, em seu arquivo de log. Certos produtos (chamados watchdogs) analisam as informações contidas em arquivos de log e geram eventos para plataformas de gerência.

Outra facilidade disponível em hosts é o uso de scripts que automatizam tarefas no próprio sistema. A gerência de sistemas pode ser feita sem usar SNMP, mas existe também a possibilidade de gerenciar hosts usando SNMP mediante o uso da host resource MIB (RFC 2790) para esse propósito.

A MIB II oferece informações básicas que servem para gerenciar hosts. Os grupos TCP e UDP proporcionam informações sobre o volume de dados que são enviados para as aplicações, tal como ilustrado a seguir:

Exemplo de objetos do grupo TCP da MIB II usados para gerenciamento de performance	
Objeto	Informação usada para gerenciamento de performance
tcpActiveOpens	Número de vezes em que o sistema abriu uma conexão.
tcpPassiveOpens	Número de vezes em que o sistema recebeu um pedido de abertura de conexão.
tcpInSegs	Número total de segmentos TCP recebidos.
tcpOutSegs	Número total de segmentos TCP emitidos.
tcpConnTable	Tabela das conexões TCP.

**Tabela 4.1**  
Grupo TCP da MIB II  
para gerenciamento  
de performance.

As informações da tabela tcpConTable também podem ser usadas para gerenciamento de segurança, pois permitem o conhecimento dos sistemas que acessam recursos via TCP tal como referido a seguir:

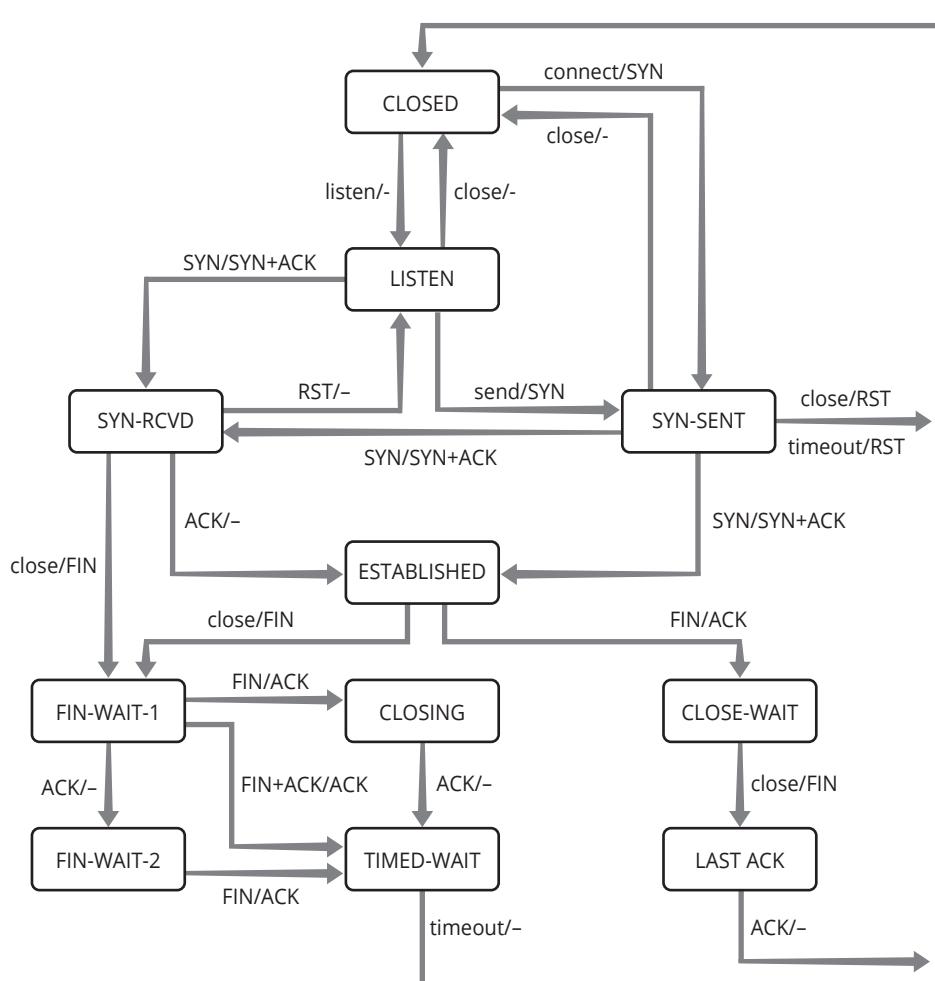
### Exemplo de objetos do grupo TCP da MIB II usados para gerenciamento de segurança

Objeto	Informação usada para gerenciamento de segurança
tConnState	Estado da conexão, que pode ser qualquer um entre os estados assumidos pela conexão TCP: LISTEN, SYN-SENT, SYNRECEIVED, ESTABLISHED, FIN-WAIT-1, FIN-WAIT-2, CLOSE-WAIT, CLOSING, LAST-ACK, TIME-WAIT e CLOSED, ilustrado na figura 4.6.
tConnLocalAddress	Endereço TCP local (valor 0.0.0.0 é usado no caso de conexões que estão no estado de LISTEN, ou seja, aptas a aceitarem conexões de qualquer endereço IP associado).
tConnLocalPort	Porta IP local da conexão.
tConnRemoteAddress	Endereço TCP remoto.
tConnRemotePort	Porta IP remota.

O campo *tcpConLocalPort* determina a porta usada pelas aplicações que estão ativas para receberem conexões no host. Essas aplicações fizeram um “open passivo”, que implica em serem ativadas e a porta a elas associadas passou do modo CLOSED para o modo LISTEN, tal como ilustrado no diagrama de estado do protocolo TCP (figura 4.6). A consulta a esse objeto permite o conhecimento de quais sistemas estão aptos a receber conexões.

O campo *tcpConRemAddress* determina o endereço do sistema remoto que está conectado à entidade. A consulta frequente a esse campo permite o conhecimento de quais sistemas usam os recursos da rede e durante quanto tempo.

**Tabela 4.2**  
Grupo TCP da MIB II  
para gerenciamento  
de segurança.



#### Saiba mais

Muitas aplicações TCP usam portas bem definidas, tais como 25-email, 22-ssh, 20 e 21-ftp, tornando possível determinar quais aplicações estão fazendo ou recebendo conexões TCP.



**Figura 4.6**  
Diagrama de estado  
do protocolo TCP.

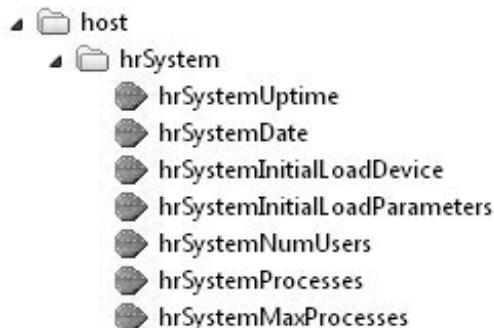
O grupo UDP também oferece um conjunto de objetos relevante para gerenciamento de host, pois diversas aplicações são acessadas mediante o uso desse protocolo, tal como o serviço de nomes (DNS-63) e a própria gerência de rede (SNMP-161).

**Tabela 4.3**  
Grupo UDP  
da MIB II para  
gerenciamento de  
aplicações.

Exemplo de objetos do grupo UDP da MIB II usados para gerenciamento aplicações	
udpInDatagrams	Taxa de datagramas recebidos.
udpOutDatagrams	Taxa de datagramas enviados.
udpNoPorts	Taxa de datagramas que não foram enviados para uma porta válida.
udpInErrors	Taxa de datagramas UDP recebidos com erro.

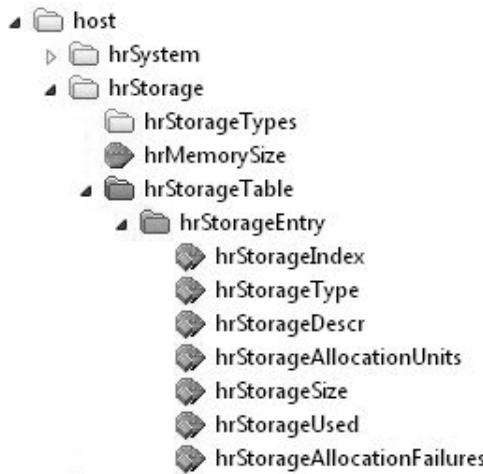
O objeto udpNoPorts informa quando a entidade está recebendo datagramas destinados a uma aplicação que não está ativada. O objeto udpNoPorts informa quando a máquina está recebendo datagramas destinados a uma aplicação que não está ativa. Um volume muito alto de datagramas UDP pode resultar em problemas de performance, e no caso de múltiplas tentativas de acesso a serviços em portas UDP inativas, normalmente indica que o host está sendo objeto de uma varredura de portas (udp port scan) que é usualmente o primeiro passo dos hackers sondando máquinas na tentativa de encontrar aplicativos que tenham vulnerabilidade passíveis de serem exploradas. Em adição às informações disponíveis na MIB ÍCONE INTERNET, foi definida uma Host MIB (RFC 2790 que contém informações relevantes para a gerência de hosts). Essa MIB contém os seguintes grupos de objetos gerenciados:

- **System group:** data do sistema, dispositivo e parâmetros de boot, número de usuários logados, número de processos em execução. Os objetos gerenciados desse grupo são:

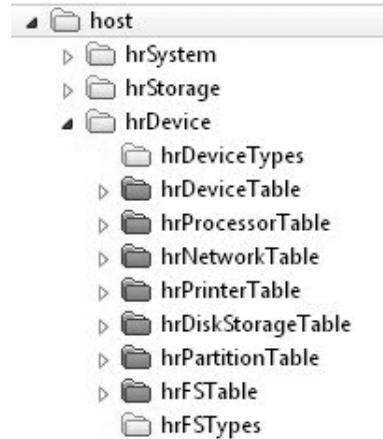


- **Storage group:** quantidade de memória principal, tabela de dispositivos de armazenamento, tipos de dispositivos, descrição, unidades de alocação, tamanho das unidades de alocação usadas, bem como eventuais falhas de alocação. Esses objetos permitem monitorar continuamente os servidores e detectar se algum recurso está sendo consumido até seu limite. A falta de área de disco é um problema que pode impedir o funcionamento de algumas aplicações.



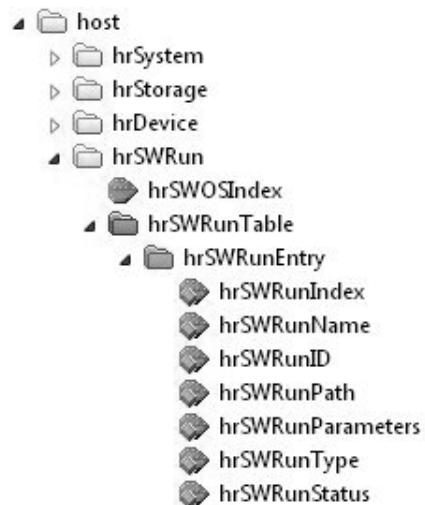


- **Device group:** tabela de dispositivos, tipo, descrição, número de erros, tabela de processadores, utilização, placas de rede, tabela de impressoras, status, estado de erro detectado, tabela de discos e de partições, tabela de sistemas de arquivos, ponto de montagem, permissões de acesso, se é “bootável” ou não, datas de backup parcial e completo;

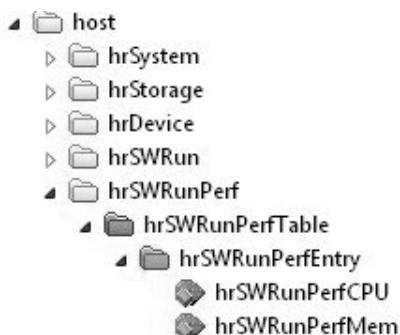


- **Running software group:** descrição do software carregado em memória (física ou virtual) e pronto para rodar; inclui Sistema Operacional, drivers, aplicações, variáveis da tabela, nome do software, path do arquivo executável, parâmetros e tipos de execução, status;

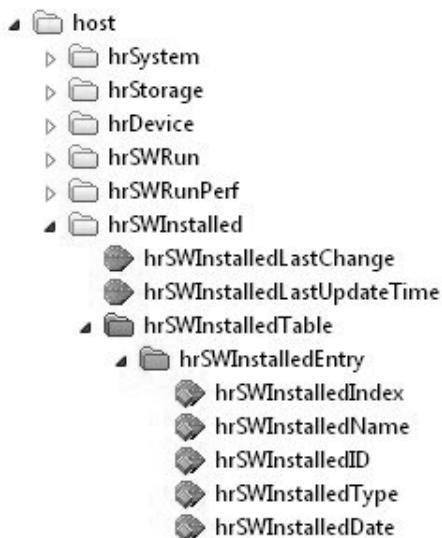




- **Running software performance group:** para cada processo em execução, são apresentadas o consumo de CPU (em centésimos de segundo) e de memória do processo. Esses objetos permitem monitorar remotamente os aplicativos sendo executados na máquina, e assim avaliar se algum deles está consumindo recursos de forma anormal.



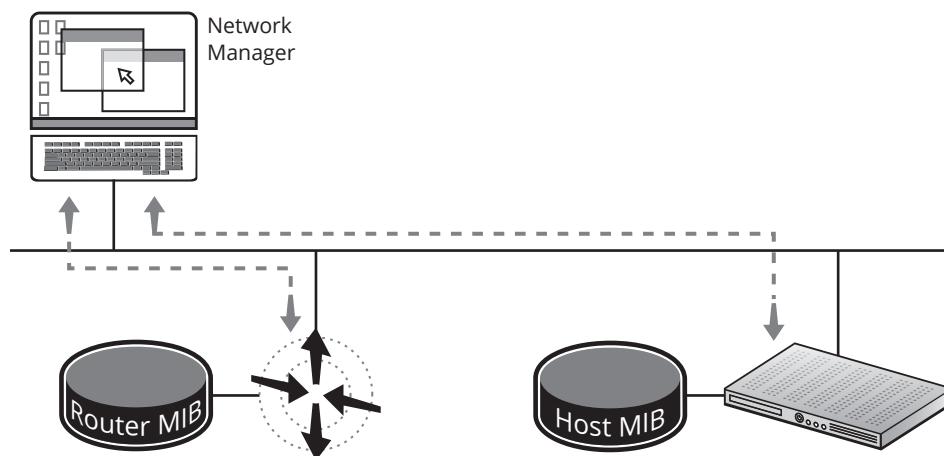
- **Installed software group:** uma tabela tem uma entrada para cada software instalado: nome do software, tipo (operating system, driver, application) e data de instalação. Esse grupo de objetos permite auditar remotamente um hpst e averiguar quais softwares estão instalados naquela máquina.



## Gerenciamento de aplicações

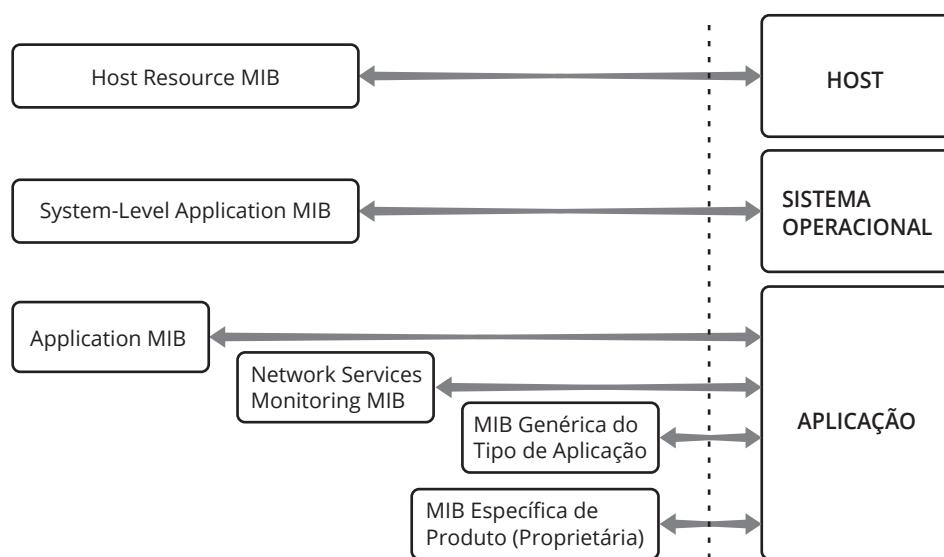
O propósito das tecnologias de informática é a execução de aplicações, que precisam de recursos para funcionar.

É importante poder configurar aplicações, detectar falhas, monitorar o desempenho de aplicações e acompanhar a aplicação ao longo de sua vida, e tudo isso pode ser monitorado mediante o uso da Host MIB. É importante ressaltar que a análise da rede deve ser feita de forma holística, isto é, integrando todos os aspectos, pois eles têm impacto mútuo. Um roteador descartando pacotes pode provocar sobrecarga em um servidor e, reciprocamente, um servidor que esteja recebendo uma carga de tráfego superior à sua capacidade de processamento/armazenamento pode causar congestionamento na rede em função das inúmeras perdas de pacote e consequentes retransmissões que ocorreriam.



**Figura 4.7**  
Usando a Host MIB.

A gerência de aplicações pode ser feita com software especializado, que frequentemente faz parte do próprio software da aplicação (monitores, logs, consoles etc.). Todavia, em função das informações disponibilizadas pelo Host MIB, a monitoração dos recursos do host pode ser feita a partir de qualquer plataforma de gerência de rede, pois as informações necessárias poderão ser buscadas mediante o uso do protocolo SNMP. A integração de diversas estratégias de gerenciamento é desejável para que uma visão completa do cenário seja possível.

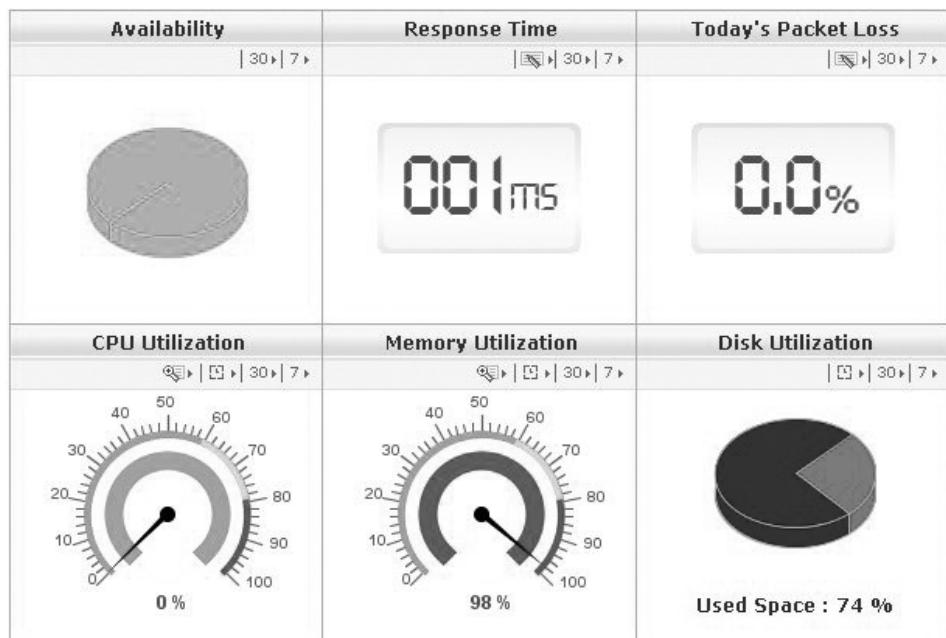


**Figura 4.8**  
Integração de gerenciamento de redes e de sistemas.

Algumas formas de gerenciar a aplicação:

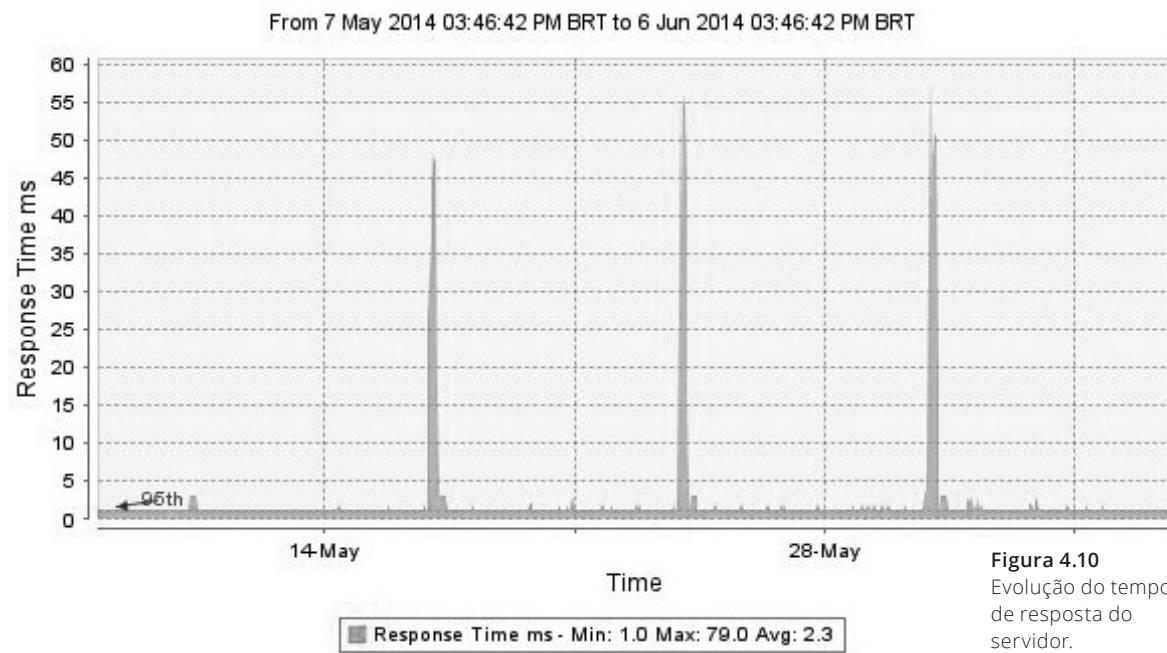
- Sem instrumentação: usando comandos do Sistema Operacional ou utilitários genéricos (por exemplo: netstat, top, lsof etc.), para ver recursos utilizados, status de processos, logs etc.;
- Com instrumentação da aplicação: permite uma gerência mais eficaz, através de ferramentas específicas para a aplicação.
- Há esforços de elaboração de MIBs para possibilitar a utilização de SNMP para monitoramento e controle de aplicações:
  - System-level application MIB (RFC 2287): MIB para gerenciamento de aplicações sem instrumentação;
  - Application Management MIB (RFC 2564): complementa a anterior;
  - Outras MIBs específicas de aplicações típicas (DNS, Mail, diretório, serviço web etc.).

A figura seguinte ilustra parte da tela de uma plataforma de gerência de rede que exibe dados sobre um servidor, obtidos mediante o uso de SNMP, tanto da MIB ÍCONE INTERNET, como da Host MIB. Cada um dos dados exibidos no gráfico da figura 4.9 pode também ser apresentado na forma de um gráfico que exiba a evolução de determinado parâmetro ao longo do tempo. O gráfico da figura 4.10 é um exemplo dessa outra forma de exibição do parâmetro "Response Time" dessa mesma estação ao longo de 30 dias.



**Figura 4.9**  
Exemplo de  
relatório de  
gerência de host.





**Figura 4.10**  
Evolução do tempo de resposta do servidor.

Observando esse gráfico, percebe-se que há picos periódicos em que o tempo de resposta aumenta. Sabendo que backups semanais são realizados como rotina da instalação, deduz-se que tais picos de aumento no tempo de resposta devem coincidir com o período em que está sendo realizado o backup semanal.



# 5

## Aspectos e aplicações de plataformas de gerência

objetivos

Conhecer os modelos de gerência; Aprender sobre Network Management System (NMS); Saber como é a arquitetura de um sistema de gerência; Usar uma análise FCAPS na definição de um NMS; Conhecer as Network Management Systems disponíveis no mercado.

conceitos

Modelos de gerência; Network Management System (NMS); Arquitetura de sistema de gerência; Análise FCAPS na definição de um NMS.

### Modelos de gerência

Quando se planeja a criação de uma infraestrutura de gerência de redes, um bom início é uma análise do modelo de gerência desejado.

- Gerência de Serviços do Negócio (BSM).
- Gerência de Processos do Negócio (BPM).
- Gerência de Serviços de TI (ITSM).
- Gerência de Rede e Sistemas (NMS).



#### Saiba mais

Devido à dependência crescente da infraestrutura de TI por parte das organizações, a escolha da ferramenta de gerência está cada vez mais passando ao largo de uma decisão puramente técnica para se tornar uma decisão estratégica dentro das organizações.

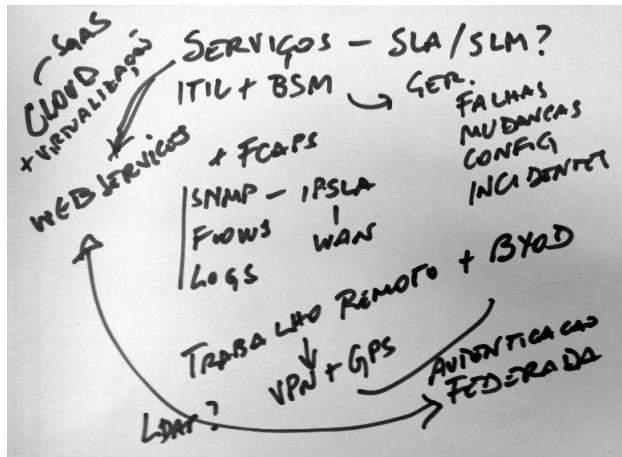
Quando se identifica a necessidade de gerência de uma rede e serviços, normalmente a base da análise está voltada unicamente para a infraestrutura da rede. Entretanto, deve-se inicialmente identificar as reais necessidades de gerência da empresa ou instituição de uma forma mais metódica.

Em um passado não muito distante, a decisão de quais ferramentas usar para ter controle da rede baseava-se em um critério unicamente técnico, já que era somente uma forma de o próprio administrador da rede possuir algum conhecimento geral da infraestrutura para saber qual a causa de uma falha e solucioná-la através de uma intervenção manual (reconfiguração, reinício ou substituição de equipamento, solicitação de reparo por parte das operadoras de telecomunicação etc.).

Decisões como virtualização, contratação de serviços na nuvem, Software as a Service (SaaS), Service Oriented Architecture (SOA), Bring Your Own Device (BYOD), cobertura WiFi, trabalhadores remotos, ampliação do parque de computadores e até mesmo a escolha por um atendimento a clientes e vendas via internet têm se tornado corriqueira em empresas de todo o porte, e parte das definições administrativas relativas ao modelo de negócios da empresa.



Esse portfólio de serviços necessita de um acompanhamento quanto à qualidade do serviço prestado (**SLA**) e, obviamente, uma gerência sobre a sua disponibilidade, desempenho, utilização e a satisfação dos clientes e usuários. Casos como integração da telefonia tradicional com serviços de voz sobre a rede IP (VOIP), serviços de videoconferência e necessidade de controle de uma vasta gama de serviços dependentes da rede justificam investimentos financeiros de vários milhares de reais, geralmente acima da alçada financeira do próprio diretor de TI das instituições. Esse investimento tem como objetivo gerenciar essa complexa infraestrutura de equipamentos e serviços que agora mantém toda a empresa operacional e direcionar novos investimentos e estratégias de negócio.



### SLA

Do inglês Service Level Agreement (Acordo de Nível de Serviço), é um acordo que geralmente é firmado entre um cliente e uma empresa, que descreve o serviço que será oferecido, suas metas de nível de serviço, além dos papéis e responsabilidades das partes envolvidas no acordo.

**Figura 5.1**  
Gerência de Redes  
e Gerência de  
Serviços de TI.

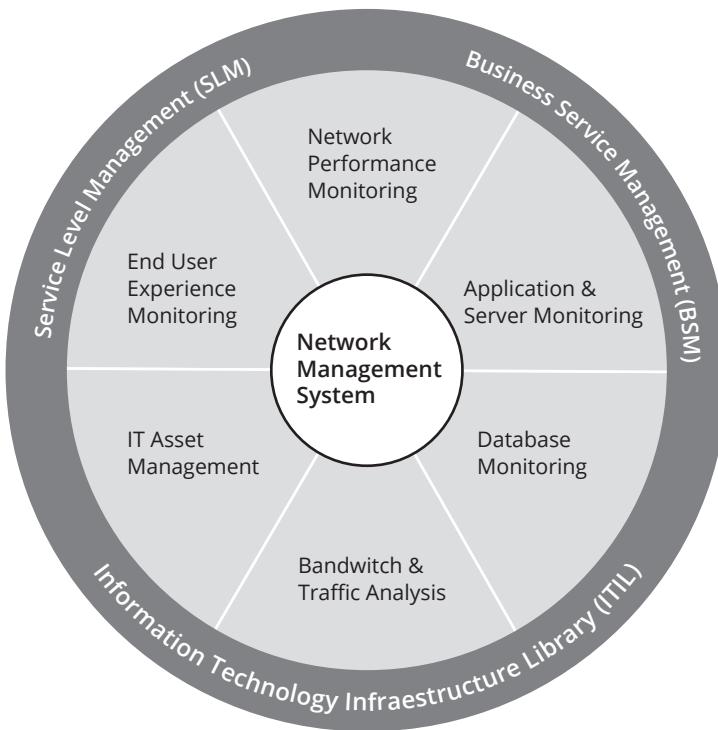
Aquele mundo distante, onde a falha em um equipamento era rapidamente notada pelo administrador da rede, hoje fica oculta pela resiliência planejada para a própria infraestrutura, fibras ópticas em anel, redundâncias de todo o tipo: nobreaks, geradores, servidores,平衡adores de carga, cabeamento, switches, roteadores, circuitos de comunicação e, em alguns casos já nem tão comuns, a redundância de todo o datacenter. Esses são fatos cada vez mais comuns, devido à dependência que as empresas hoje possuem para manter todo o seu negócio. Em um futuro próximo, conceitos como Internet of Things (IoT) serão aplicados, gerenciando a utilização da energia (lâmpadas), sistemas de ar-condicionado do datacenter e até mesmo onde se encontram fisicamente os dispositivos móveis da empresa (Pads, celulares etc.). Tudo isso tende a ser gerenciado utilizando-se uma plataforma única como uma maneira de diminuir a complexidade associada a várias plataformas de gerência independentes.

Nesse contexto, muitas instituições necessitam não somente de uma gerência da infraestrutura básica baseada nos principais elementos da rede, mas sim da possibilidade de crescimento dessa gerência até que ela consiga responder de forma uníssona a todos os fatores de TI que possam influenciar a missão da empresa.

Uma falha não diagnosticada no sistema de ar-condicionado do datacenter pode

- ! causar o desligamento de todo o datacenter. O atraso na substituição das baterias do nobreak pode diminuir a autonomia do datacenter nos casos de falhas de energia.





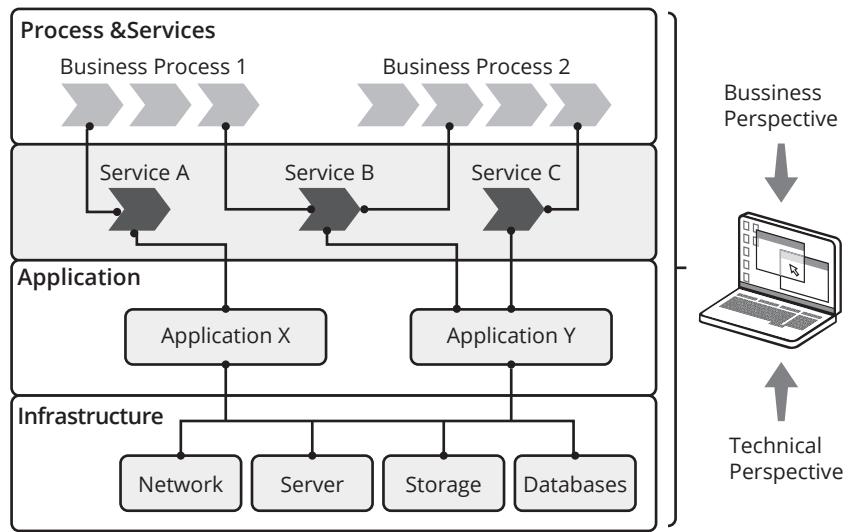
**Figura 5.2**  
BSM, SLM e ITIL  
(fonte: <http://www.manageengine.ca/it360.aspx>).

O primeiro ponto a se estabelecer quando se fala da definição de gerência da rede é justamente delimitar a gerência esperada para a instituição, traçando as linhas básicas e as melhores práticas demandadas para a gerência de serviços do negócio (Business Service Management – BSM) em relação à gerência de TI da empresa (Information Technology Service Management – ITSM), e destas para a gerência da rede propriamente dita (Network Management System – NMS). Uma definição importante aqui é a da integração desejável entre esses elementos.

Um dos modelos frequentemente utilizados nas empresas é o Information Technology Infrastructure Library (ITIL). Nesse modelo, os processos de gestão de TI têm foco no cliente e permitem definir entre outras coisas a relação de dependência de cada um dos elementos de TI da empresa (roteadores, switches, servidores etc.) para cada um dos serviços prestados aos clientes.

Já as ferramentas do BSM são projetadas para auxiliar a organização quanto à visão da empresa a respeito da área de TI, fornecendo recursos a esta, de maneira a melhor manter os serviços prestados pela área à empresa e seus clientes. Em suma, BSM auxilia na gestão de TI através de uma abordagem baseada nos serviços de TI (ITSM). Novos sistemas BSM utilizam uma visão unificada de um datacenter, permitindo que os administradores possam gerenciar desde as aplicações individualmente até os menores eventos da rede de forma unificada (em um mesmo monitor), conseguindo antecipar ou visualizar possíveis problemas antes que os usuários os percebam.





**Figura 5.3**  
Mapeamento de dependência entre processos de negócio e elementos da rede.

Enquanto no modelo de gerência de redes mais tradicional, a proposta é simplesmente aplicar o modelo IETF-SNMP para gerenciar os componentes da rede de uma forma mais técnica, sem se preocupar em mensurar exatamente o peso que cada componente possui dentro de todo o contexto da empresa, a maioria dos setores de TI já possui alguma forma de gerência baseada nos modelos ITIL, BPM ou outros, mas em sua maioria de forma não integrada, isto é, gerências distintas para serviços e elementos distintos da rede. Nesse ambiente, é frequente que as diversas equipes como as de infraestrutura de rede física (cabeamento), infraestrutura L2 (switching), infraestrutura L3 (roteamento), banco de dados, telefonia, serviços em nuvem, atendimento ao usuário etc., tenham um conjunto próprio de ferramentas para responder aos requisitos demandados.

Sendo assim, o primeiro passo dado em direção à definição de uma plataforma de gerência de redes deve ser a análise do que deverá ser gerenciado pela empresa no futuro. Esse passo é especialmente importante por justificar o custo-benefício das soluções mais completas, já que normalmente o esforço de implementação, treinamento e ajustes de uma plataforma de gerência até seu uso confiável situa-se em um tempo medido em anos, incorporando-se nesse tempo várias melhorias.

Na maioria dos casos, a migração de uma plataforma para a outra é quase sempre traumático, significando a perda de todo o conhecimento, tempo e esforço utilizado na plataforma anterior. Geralmente esse tempo é medido por customizações, como a criação de regras de negócio, relações de dependências de serviços e equipamento etc., e impactam diretamente em pontos como confiabilidade, estabilidade, capacitação de equipe e custo de horas técnicas dedicadas à implementação da nova solução. A contabilização desse custo-benefício é uma decisão mais administrativa do que técnica.

Uma vez definidas as linhas básicas pela administração da empresa, uma avaliação mais técnica/gerencial do modelo a ser usado deve ser realizada. Existem diversos modelos de gerência de redes que podem ser implementados. A definição de um modelo é baseada principalmente na necessidade do que se deseja gerenciar, como:

- Uma gerência mais voltada ao negócio e serviços da empresa;
- Gerência visando uma operação específica, como a das empresas de telecomunicações;
- Uma gerência mínima dos elementos de TI da empresa.

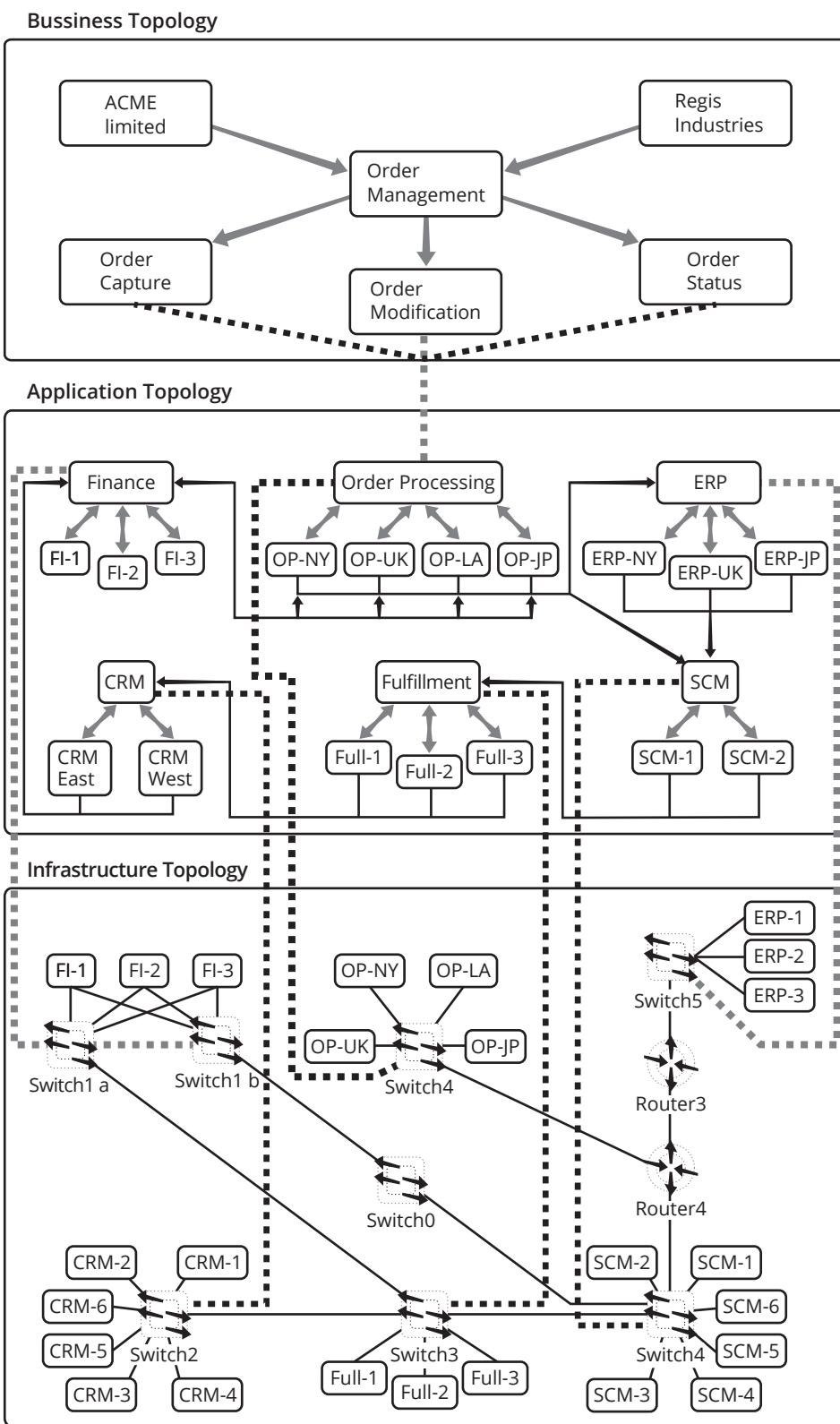
FCAPS	Falhas	Configuração	Contabilização	Desempenho	Segurança
<b>TMN</b>					
Gerência do Negócio	Não	Sim	Sim	Sim	Sim
Gerência de Serviços	Sim	Sim	Sim	Sim	Sim
Gerência de Redes	Sim	Não	Sim	Sim	Sim
Gerência de Elementos	Sim	Não	Sim	Sim	Sim
<b>OAM&amp;P</b>					
Operação	Não	Não	Sim	Sim	Sim
Administração	Não	Sim	Sim	Sim	Sim
Manutenção	Sim	Não	Sim	Sim	Sim
Provisionamento	Sim	Não	Não	Não	Não
<b>TOM /e TOM</b>					
Implantação	Sim	Não	Não	Não	Não
Garantias	Não	Sim	Sim	Sim	Sim
Faturamento	Não	Sim	Sim	Não	Não
Supporte e Facilidades de Operação	Não	Não	Não	Não	Não
IETF SNMP (TCP/IP)	Sim	Sim	Sim	Sim	Sim
ISO CMIP/CMS	Sim	Sim	Sim	Sim	Sim

**Figura 5.4**  
Alguns modelos de gerência.

As gerências voltadas ao negócio envolvem a definição prévia dos processos do negócio, para em um segundo momento definir-se quais as ferramentas serão necessárias para estabelecer os níveis de serviço desejados e o nível de integração do sistema ou sistemas de gerência da rede com outros, de forma semelhante ao que realiza com o sistema de Enterprise Resource Management (ERP) da empresa. Normalmente esse processo decisório leva anos de planejamento, mas se essa é a linha a ser seguida, a escolha de um NMS mais robusto e escalável justifica o investimento.

Existe uma tendência nas empresas de grande porte em uma gerência mais voltada a serviços, enquanto empresas menores tendem a manter um isolamento entre os negócios da empresa e a infraestrutura de TI. Boa parte dessa decisão é associada aos custos de implementação dessa gerência integrada e, claro, outra parte por desconhecimento dessas possibilidades ou pelas empresas simplesmente possuírem uma complexidade dentro dos limites humanos de compreensão dos detalhes e inter-relacionamento de diversos elementos de infraestrutura que compõem determinado serviço.





**Figura 5.5**  
Camadas de gerência.

Existem várias funcionalidades das diversas plataformas de gerência, e a forma mais comum de caracterizar suas funções é utilizando o modelo Fault, Configuration, Accounting, Performance e Security (FCAPS).



## Network Management System (NMS)

- Simplificam e sumarizam as informações de gerência coletadas.
- Integram-se com outros sistemas, criando uma interface única.
- Organizam dados de gerenciamento, apresentando-os de maneira mais adequada (gráficos tabelas e relatórios).
- Plataforma gratuitas versus comerciais.

Em um conceito simples, um sistema de gerência de redes (NMS) é uma combinação de hardware, software e protocolos utilizados para administrar todos os elementos da rede.

Os diferentes elementos a serem gerenciados variam de uma organização para outra e a sua definição é um dos pontos básicos para a escolha de um sistema de gerência com o objetivo de mapear todos os elementos que são importantes para a organização e, preferencialmente, conseguir visualizar seus relacionamentos de forma conjunta.

Entre as centenas de sistemas de gerência existentes, é preciso que o administrador da rede consiga escolher qual, ou quais, melhor de adaptam às suas necessidades.

- As ferramentas escolhidas para o NOC devem ser úteis e utilizáveis;
- Muitas ferramentas são excelentes, mas exigem alto nível de conhecimento para tirar proveito delas;
- Uma boa ferramenta permite tanto customização quanto reconhecimento automatizado da rede;
- Ideia fundamental: quanto menos informação o usuário precisar inserir, melhor.

O ponto fundamental que rege a escolha do NMS a ser implementado é sem dúvida que ele seja adequado às necessidades e expectativas que se tem. Investir em ferramentas caras nem sempre é a melhor solução, já que as inúmeras funcionalidades dispendem um conhecimento não desprezível para o seu uso.

- A interface gráfica deve mostrar claramente as falhas da rede;
- Ter suporte a web e níveis de usuários é importante.

Um dos pontos fundamentais para essa definição é sempre a interface de uso, já que existem sistemas que exigem a instalação de um cliente específico para realizar algumas funções, fazendo distinção entre um acesso web e um acesso pelo cliente instalado (exemplo: edição dos mapas da rede).

Outro ponto relevante é a possibilidade de definir-se perfis e grupos de usuários como administrador e operador, o que evitaria que configurações da gerência de determinados segmentos (exemplo: banco de dados) sejam alterados por outra equipe de trabalho (exemplo: equipe de suporte a redes). Grupos e subgrupos de trabalho são bastante úteis, mas são poucas ferramentas que os implementam de uma forma efetiva e integrada usando plataformas de autenticação, como Ldap.

- O software escolhido é passível de integração?
  - Um bom NMS permite integração com outros padrões da indústria, permitindo interação via mail, sistema de tickes, SMS etc. Quanto maior a possibilidade de integração, melhor.

Na maioria das vezes, não é possível adquirir uma solução de gerência totalmente integrada e completa que abarque todos os elementos e serviços da rede. Nesse ponto, a possibili-

dade de integração com as ferramentas existentes acaba sendo necessária. Um fato comum é uma nova solução de gerência ter de se integrar com um sistema de registro de problemas já existente na empresa. Um caso é a necessidade de abertura de tickets via API ou e-mail para realizar essa integração.

- Existe uma boa interface de relatórios:

- É possível extrair diversos relatório de forma facilitada, customizável e em tempo real.



Em outros pontos a operação da empresa possui portal de clientes e gostaria de disponibilizar os dados da gerência e seus relatórios diretamente para os clientes nesse portal ou emitir-los diretamente e de forma regular aos seus clientes. Algumas ferramentas possuem módulos específicos para esse fim (exemplo: integração com Crystal Reports ou geração de relatórios em PDF ou planilhas de Excel), enquanto em outras os relatórios precisam ser extraídos manualmente.

- Modularidade:

- Possibilidade de agregação em módulo de novas funcionalidades prevendo aumento da gerência;
  - Permite acompanhar futuras demandas da empresa.



A modularidade das ferramentas é um item obrigatório para empresas maiores ou que pretendem adotar um NMS como uma plataforma de gerência institucional. Na prática, a modularidade da ferramenta é o que permitirá a integração de novas funcionalidades na gerência da rede, permitindo a gerência de novos serviços. Alguns módulos são desenvolvidos pelo próprio fabricante e outros por empresas terceiras baseado em APIs que o fabricante do software provê. Alguns exemplos módulos que podem ser integrados a posteriori são:

- Auditoria de sistemas;
- Service Desk e sistema de registro de problemas;
- Gerência de bens e inventário;
- VOIP e Telefonia IP;
- Gerência de Configuração;
- Gerência de Impressão;
- Gerência de Processos e Nível de serviço (SLA).

Cada rede tem suas necessidades, defina o seu ideal de gerência:

- O que você quer gerenciar: processo, serviços etc.;
- Defina as áreas em que você quer agir (FCAPS?);
- Defina se uma gerência integrada é ou não vital.
  - Estabeleça seu orçamento;
  - Defina o tempo para implantação;
  - Planeje o tempo de vida da sua solução;
  - Defina e treine recursos humanos.



É imprescindível definir anteriormente “o que” se deseja gerenciar dentro da instituição, levando em conta que a cada nova gerência existe um custo considerável associado, tanto para compra do produto (se for o caso), quanto para a manutenção da solução e o treinamento da equipe responsável pelo processo.



Entre a extensa lista de possibilidades de ferramentas, a máxima integração deve ser um dos objetivos. Infelizmente, ela acaba sendo cerceada pelo custo das plataformas, fazendo com que a utilização de várias soluções gratuitas seja sempre considerada como algo factível, muitas vezes se desprezando o custo de manutenção de uma solução múltipla em vez de uma solução integrada. Nesse ponto, alguns itens devem ser considerados, como o tempo de vida que você planeja para a solução e o custo de pessoal. Algo que é frequentemente desconsiderado na adoção de plataformas de gerência de redes gratuitas é que um software gratuito possa ser descontinuado ou ser incompatível com a versão anterior, fazendo com que o trabalho de configuração da gerência tenha de ser refeito. Outro ponto a considerar é que a curva de aprendizagem de muitas ferramentas freeware não são desprezíveis.

- Plataformas comerciais:
  - Integradas;
  - Curva de aprendizado relativamente rápida;
  - Intuitivas.
- Plataformas gratuitas:
  - Funcionalidades básicas;
  - Reaprendizado constante;
  - Risco de descontinuidade.

Quando se chega nesse ponto, uma boa saída é novamente a avaliação do modelo FCAPS para definir quais as funcionalidades são necessárias para definir o que precisa ser gerenciado na instituição. A análise inevitavelmente passa pelo fator financeiro, e deve-se ter em mente que muitas vezes existe somente uma substituição entre custo da ferramenta e custo de pessoal para manter ferramentas gratuitas ou de baixo custo.

Os NMSs de primeira linha geralmente possuem uma plataforma totalmente modular, permitindo o crescimento e melhorias no gerenciamento das organizações através da aquisição de novos módulos de gerência. As plataformas comerciais de primeira linha em geral disponibilizam centenas de módulos e plug-ins que aumentam a lista de elementos que podem ser gerenciados, possibilitando o reconhecimento de praticamente todos os softwares e hardwares conhecidos.

Entre a lista dos NMS comerciais mais completos, podemos citar:

- CA Spectrum ([www.ca.com/spectrum](http://www.ca.com/spectrum));
- HP IMC – Intelligent Management Center;
- HP NNM: Network Node Manager ([www.hp.com/go/nnm](http://www.hp.com/go/nnm));
- IBM Tivoli Netcool Network Management ([www.ibm.com/tivoli](http://www.ibm.com/tivoli));
- BMC Suite.

Existem ainda NMSs que não são tão completos no que se refere à integração com diferentes modelos e processos de negócio, mas possuem um crescimento mais limitado e enfoque mais técnico (para administradores de sistemas e gerentes de rede).

Alguns NMS têm custos acessíveis para empresas pequenas e médias, como por exemplo:

- OPManger;
- SNMPC;
- WhatsUP Gold;
- Zabbix;
- Nagios-XI é a plataforma paga do Nagios com várias facilidades de uso.

Em redes menores, existe ainda a possibilidade de utilização de ferramentas comerciais que permitem o uso limitado de funcionalidades, geralmente disponibilizado como uma versão gratuita ou como um opensource ou freeware. Normalmente essas limitações são quanto à quantidade de serviços e funcionalidades disponibilizados nessas plataformas gratuitas, mas em alguns casos podem ser oferecidas o conjunto completo de funcionalidades, porém para um número limitado de dispositivos ou elementos gerenciáveis.

Versões opensource/freeware de ferramentas comerciais:

- HP INM permite uso completo para até 25 nodos;
- OPManger permite uso gratuito para até 10 elementos de rede;
- Zabbix possui uma versão comercial e outra gratuita com restrição;
- Nagios – versão opensource.



Existem ainda NMSs para fins específicos, ou seja, atreladas a algum fabricante de hardware, em geral switches ou roteadores. Essas plataformas normalmente são disponibilizadas junto com a compra de determinados equipamentos, para a gerência específica daqueles equipamentos. Na maioria dos casos, essas ferramentas permitem a gerência limitada para equipamentos de outros fabricantes (exemplo: somente suporte a MIB-II e ICMP), e têm de ser abandonados no momento da troca dos equipamentos (hardware) da rede.

Alguns exemplos de ferramentas de gerência específicas:

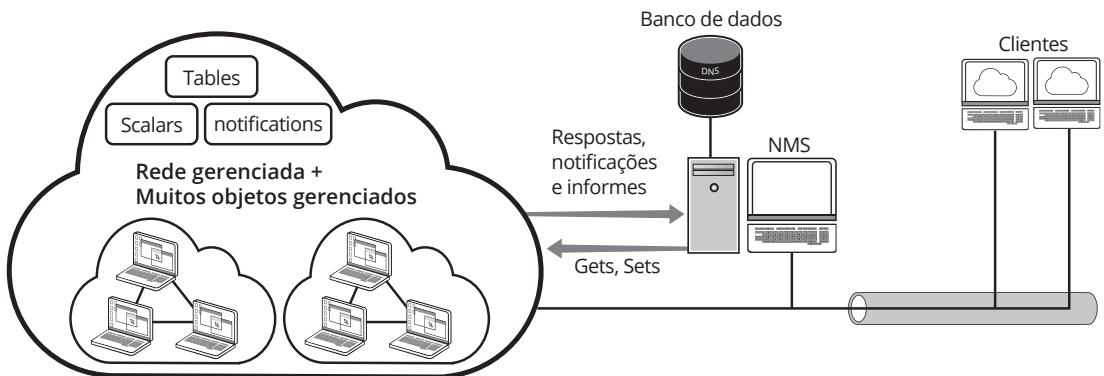
- Ericson IPECS NMS (Network Management Solution);
- Extreme Networks RIDGELINE Network and Service Management;
- HP Procurve Manager;
- Cisco Works LMS.



## Arquitetura de um sistema de gerência

Uma solução de gerência precisa prover uma visão da infraestrutura de forma confiável e escalável. Para que se saiba identificar essas características em muitas plataformas, é necessário entender como essa arquitetura de funcionamento da própria plataforma costuma ser planejada. Em um modelo simplista, o NMS é responsável por receber informações não solicitadas (TRAPS) ou questionar os agentes remotos por alguma informação (GET) e eventualmente solicitar que alterem alguma de suas configurações (SET), porém, como resultado final, o usuário do NMS espera que cada uma das atividades realizadas sejam contabilizadas, tarefas repetitivas sejam automatizadas e resolvidas sem a interferência direta do operador, além, é claro, que seja possível exibir o estado da rede através de um mapa comprehensível e multinível, implementando um processo definido para controle dos problemas e gerando relatórios capazes de prever possíveis problemas baseados no conhecimento armazenado ao longo da operação do sistema, e tudo de forma mais inteligente e automática possível.





**Figura 5.6**  
Solução de gerência SNMP.

Para que toda essa automação seja realidade, a solução torna-se bastante complexa, exigindo que cada um dos processos e conhecimento seja inicialmente ensinado para o NMS e, claro, que esse tenha em sua complexa arquitetura as aplicações que suportam essa inteligência. Nesses pontos é que se encontram os principais diferenciais das arquiteturas e implementações dos diversos NMSs do mercado.

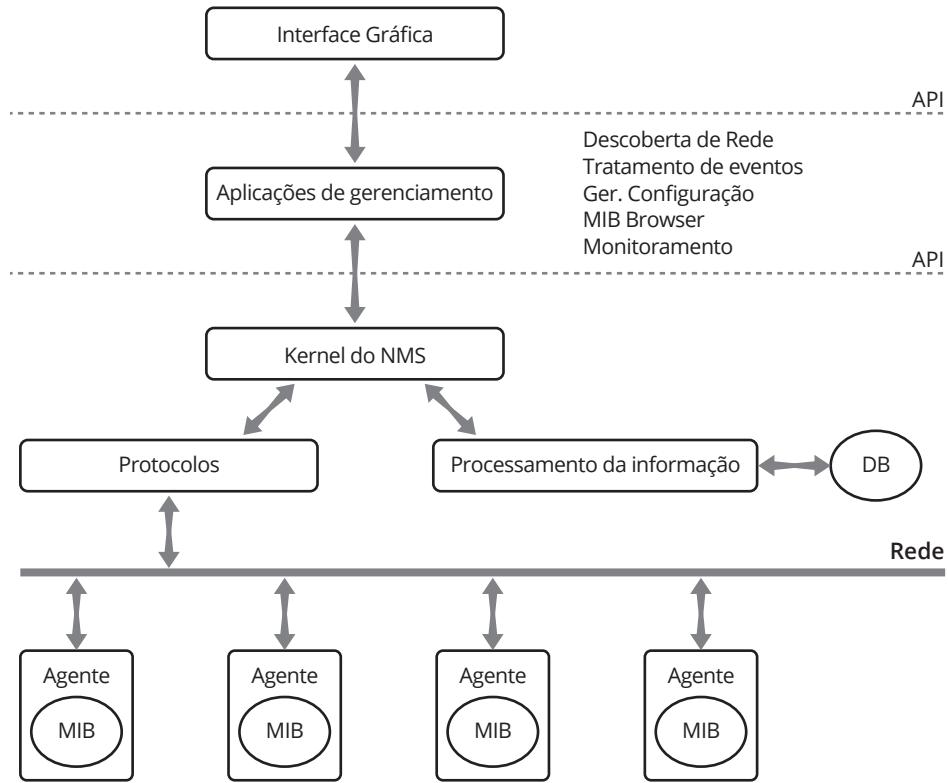
Componentes de um sistema de gerência:

- Core do sistema de gerência;
- Interface com o usuário;
- Banco de dados;
- Agentes a agentes SNMP;
- Geração de relatórios;
- Criação de mapas;
- Correlação de alarmes;
- Descoberta da rede;
- Módulos e Plug-ins.

A arquitetura básica das plataformas de gerência tem no seu ponto mais importante o Kernel do sistema de gerência de redes, responsável por implementar os protocolos de gerência e armazenar todas as informações coletadas em todos os agentes da rede. Essas informações são então armazenadas para uso pelo sistema, geralmente através de uma série de aplicativos que as processam e permitem melhor exibição para o usuário.

Vários sistemas freeware permitem a utilização de frontends distintos, mantendo sempre um mesmo kernel do NMS. Um exemplo bastante comum é o NAGIOS, que possui inclusive uma versão de aplicativos e interfaces pagos, além das inúmeras implementações gratuitas. Essa estrutura em camadas é bastante útil, já que as interfaces com o usuário e aplicativos de gerência costumam ser bastante modificados, para incluir novas plataformas (celulares, por exemplo) e novas funcionalidades, como novos padrões de desenvolvimento web, além de permitirem a implementação de plataformas mais seguras, como as baseadas em clientes proprietários instalados nas máquinas dos próprios administradores da rede.

Outro ponto que possui variações distintas nas NMSs e dependentes do ambiente para o qual o sistema de gerência foi destinado é o banco de dados. Na maioria dos NMSs, ele é suportado pelos BDs mais conhecidos (exemplo: BerkeleyDB, mysql, Oracle, Postgress etc.). Isso é um ponto importante pelas características que cada um desses bancos de dados possui, como controle de concorrência e atomicidade, a tradição da organização em determinado DB e o custo associado a ele.



**Figura 5.7**  
A arquitetura de um NMS e suas APIs.

Muitas plataformas de gerência permitem customização no ambiente do cliente através do fornecimento de várias APIs (Application Program Interface) nas diferentes camadas de software do sistema de gerência. Caso a customização da solução seja um dos objetivos desejáveis para a gerência da rede, como fornecer uma interface de acesso do cliente a determinados gráficos através de um portal, esse pode ser um dos critérios relevantes na definição da plataforma a ser utilizada.

Normalmente as plataformas possuem APIs para customização da interface de acesso ao cliente ou para a criação de scripts que permitam realizar tarefas como a pré-análise e classificação de problemas antes da geração de um determinado ticket.

Um exemplo desse processamento é a definição de incidente que pode ou não ser classificado como um problema, dependendo do horário onde ele ocorre, ou até mesmo a programação de tratamento de um alerta, gerando uma tarefa automática em algum sistema/servidor, como remover determinados arquivos de forma automática no caso de falta de disco em uma partição, evitando que haja queda no serviço. Alguns exemplos de aplicativos desenvolvidos a partir dessas APIs no nível de aplicações do NMS são o próprio MIB-browser, integrado em algumas plataformas de gerência ou outras aplicações, capazes de receber e tratar um TRAP SNMP que avisa da mudança de configuração de um roteador e automaticamente providencia o salvamento desta, registrando quem a realizou e em que horário ela foi realizada. Esse tipo de API é especialmente importante para o caso de o usuário desejar uma gerência mais granular de sua rede, querendo a possibilidade de programar a ação da estação de gerência para o caso de recebimento dos diversos TRAPs enviados pelos equipamentos.

Componentes de um sistema de Gerência:

- Controle de SLA;
- Descoberta automática de topologia;

- Arquitetura de plug-ins;
- Suporte a múltiplos fabricantes de hardware e software;
- Inventário;
- Gráficos com visão da rede em camadas (físico, enlace e rede);
- Controle de acesso em múltiplos níveis;
- Gerência baseada em políticas;
- Amplo suporte a gerência de serviços (ldap, bancos de dados diversos, www, mail etc.).

Outro ponto que diferencia muitas das soluções de gerência de redes são os inúmeros aplicativos existentes, geralmente na forma de módulos ou plug-ins que permitem o suporte a novos hardwares, softwares e MIBs proprietárias, além de agregar outras funcionalidades à plataforma (exemplo: sistema de inventário).

Componentes de um sistema de Gerência:

- Amplo suporte a MIBs do tipo enterprise-específic;
- Possibilidade de realizar a compilador de MIBs e definir novos objetos.

Uma plataforma de gerência comumente possui funções que incluem:

- A descoberta da rede;
- Mapeamento da topologia dos elementos gerenciados;
- Manuseio dos eventos;
- Coleta e graficação de parâmetros de desempenho.

## Obtendo dados da atividade da rede

De uma forma geral, os sistemas de gerência de redes (NMS) trabalham realizando a coleta dos diversos dados de gerência SNMP (traps, gets) e realizando poolings periódicos em outros protocolos ou aplicações (ICMP, http, ssh etc.). Essas informações juntas espelham a atividade da rede. Nesse ponto, algumas ferramentas têm a capacidade de coletar informações com origem em SNMP e ICMP, enquanto em outras é possível obter informações de fluxos e de serviços (ssh, http, https, bancos de dados, syslog etc.). De uma forma geral, é importante analisar quais serviços a plataforma em questão é capaz de tratar, e se para esse tratamento ela os faz através de agentes snmp padrão ou se necessita que seja realizada a instalação de um agente proprietário no equipamento/servidor monitorado.

## Descoberta da rede

- Forma ativa.
- Forma passiva.

A descoberta da rede é um item particular de algumas plataformas e possui características bastante diferenciadas entre as plataformas. Tem como base dois algoritmos fundamentais, um que permite a pesquisa de forma ativa (pesquisando por serviços e hosts a partir de um conjunto de endereços IP), ou simplesmente observando o tráfego de rede e descobrindo cada um dos elementos da rede.

A forma ativa envia a informação na rede, na tentativa de descobrir cada um dos elementos, usando vários protocolos distintos para descobrir os dispositivos e a topologia, como ARP, ICMP, SNMP, HTTP, SSH etc. Esses protocolos são indicados pelo administrador da rede no momento da descoberta.

A forma passiva se utiliza de informações do próprio tráfego coletado pela estação de gerência ou através de uma consulta a um equipamento central, como um switch ou roteador de core da rede. Através das informações contidas nesse equipamento (tabelas de rotas, tabelas ARP, informações de interfaces, agentes RMON etc.), novos equipamentos são descobertos e novamente sondados de forma a montar um mapa de interconexão de todos os equipamentos da rede a partir desse ponto central.

- ▣ **Monitoração passiva de pacotes ARP:** a interface fica em modo promíscuo e escutando todos os pacotes ARP, além de construir uma lista de endereços MAC/IP. Só funciona para subredes conectadas diretamente na estação de gerência ou de sensores espalhados pela rede;
- ▣ **Monitoração ativa de pacotes ARP:** a estação de gerência envia pacotes IP/UDP e monitora as respostas;
- ▣ **Scan sequencial de endereços IP com pacotes ICMP:** a estação de gerência gera vários pacotes ICMP (ping) tentando obter uma resposta de cada um dos IPs, na tentativa de saber se existe ou não algum host naquele IP. Essa estratégia é pouco eficiente se utilizada de forma isolada, dado que muitos Sistemas Operacionais realizam filtragem de ICMP para estações de usuários.

Assim como estas, existem várias outras estratégias de descoberta de rede. Em geral, as melhores ferramentas do mercado possuem um algoritmo misto que utiliza ambas as estratégias para complementar as informações nos seus bancos de dados topológicos.

### Filtrando as atividades da rede baseando-se em limiares

Outro ponto a ser observado é se a plataforma em questão permite ou não a definição de diversos limiares, e quão granular essa definição pode ser. Em algumas plataformas, é possível realizar a definição de determinados templates e associá-los a conjuntos de interfaces. Por exemplo: é possível definir-se que nenhuma interface de longa distância pode ficar mais de 30 minutos com utilização superior a 95%. Em outras plataformas, somente é possível definir que se uma única amostra passe de 95%, o alarme funcione. Plataformas de gerência que permitem a definição baseada em políticas normalmente possuem essa característica.

 A gerência baseada em políticas normalmente é uma aplicação à parte dos NMSs.

Para as ferramentas comerciais, significa um custo adicional.

### Correlação de alarmes

Eventos são “informações relevantes” da atividade da rede, que podem ou não representar uma falha, assim como uma única falha da rede pode desencadear inúmeros eventos. Cabe à plataforma de gerência reconhecer e agrupar esses eventos de forma a definir qual é o problema em questão e, dessa forma, gerar um ou vários alarmes para o operador da rede. Deve portanto haver filtragem de eventos capazes de gerar alarmes:

- ▣ Filtros baseados em limiares determinam um alarme;
- ▣ Filtros que agrupam eventos identificando um único alarme;
- ▣ Filtros que associam criticidade aos eventos;
- ▣ Filtros que configuram uma ação no caso de vários alarmes.

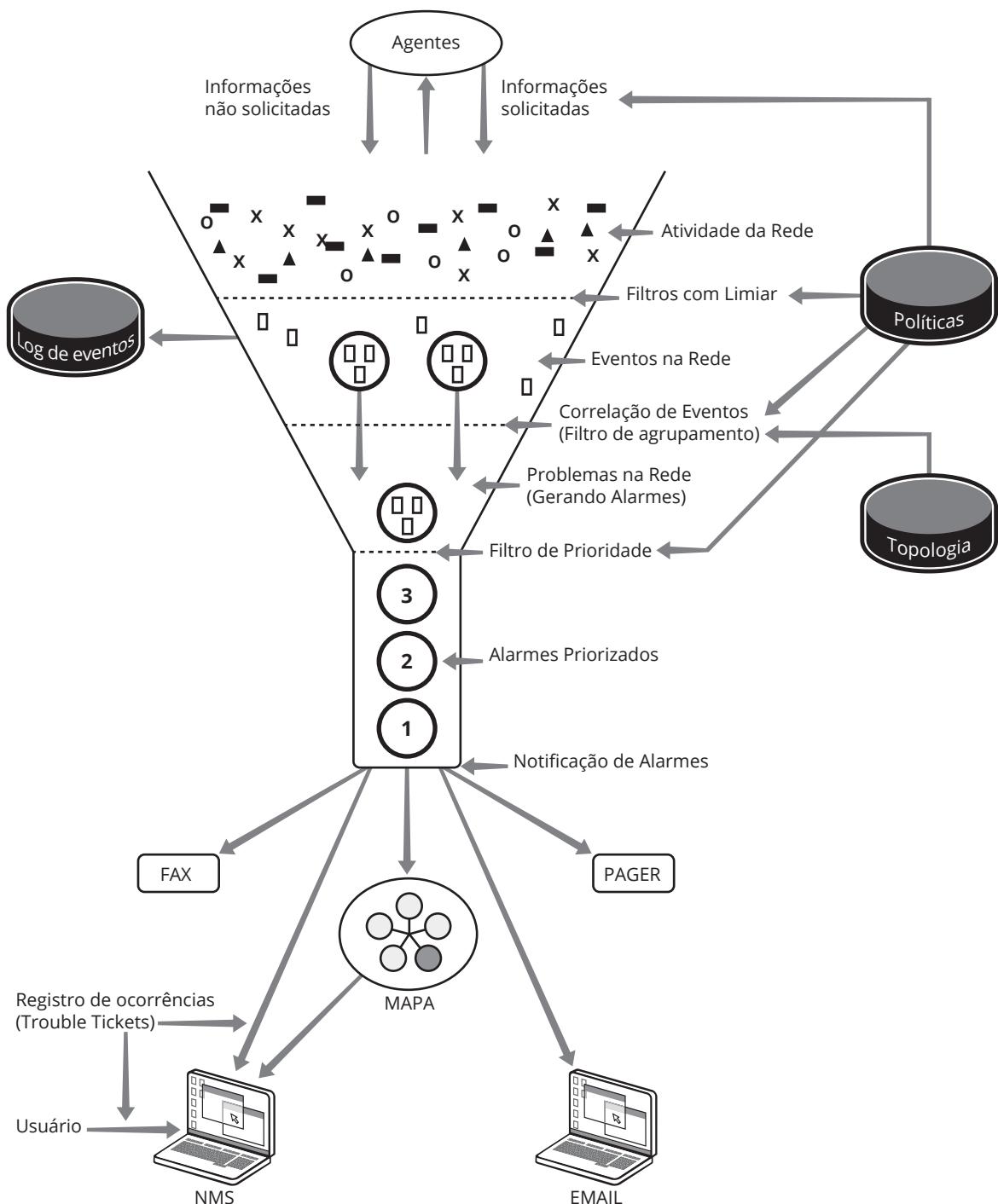
Uma vez que todas as informações das atividades da rede tenham sido devidamente filtradas e agrupadas (exemplo: o caso de um switch que cai e deixa sem acesso vários servidores), os problemas gerados podem acionar um ou vários alarmes, e a capacidade da



plataforma em questão de permitir ou não a correlação desses alarmes ou a definição de uma dependência entre esses alarmes é um item valioso a ser considerado.

A correlação de alarmes, como é chamado, é capaz de identificar a dependência de uma porção da rede de um único equipamento, filtrando vários alarmes e gerando um único problema a ser tratado (exemplo: vários servidores e serviços dependem de um único roteador). As plataformas aqui variam na “inteligência” que o software possui para detectar sozinho essa interdependência baseado na topologia da rede, ou que o administrador minimamente possa definir que essa dependência existe, ou, ainda, em plataformas mais simples os objetos com falhas simplesmente são colorizados na interface e cabe ao administrador entender a sua interdependência.

**Figura 5.8**  
Fluxo de informações dentro de um NMS.



## Priorização de atividades

Uma vez que tenham sido agrupados os vários alarmes, ainda há uma atividade possível na automatização do tratamento dos eventos da rede: a priorização. Essa prioridade normalmente se refere à forma e a quem os eventos serão encaminhados de forma automática.

Nesse item há várias possibilidades, como a simples priorização dos tickets de problema a serem gerados para o service desk, até o controle do tempo de resolução do problema e escalonamento para o gestor da equipe, em caso de demora no restabelecimento do problema. Por exemplo, caso um alarme do site WWW da empresa continue ativo mesmo depois de uma hora fora de funcionamento, o mesmo evento é encaminhado para o gerente do service desk, para que este também fique ciente de que o serviço encontra-se fora do ar.

A priorização das atividades normalmente está atrelada às políticas definidas no módulo de políticas do sistema, e a definição prévia dos escopos para cada uma dessas falhas encontra-se na correlação de alarmes.

## Escalabilidade

- Modelo centralizado.
- Modelo Flat e distribuído.
- Modelo hierárquico.



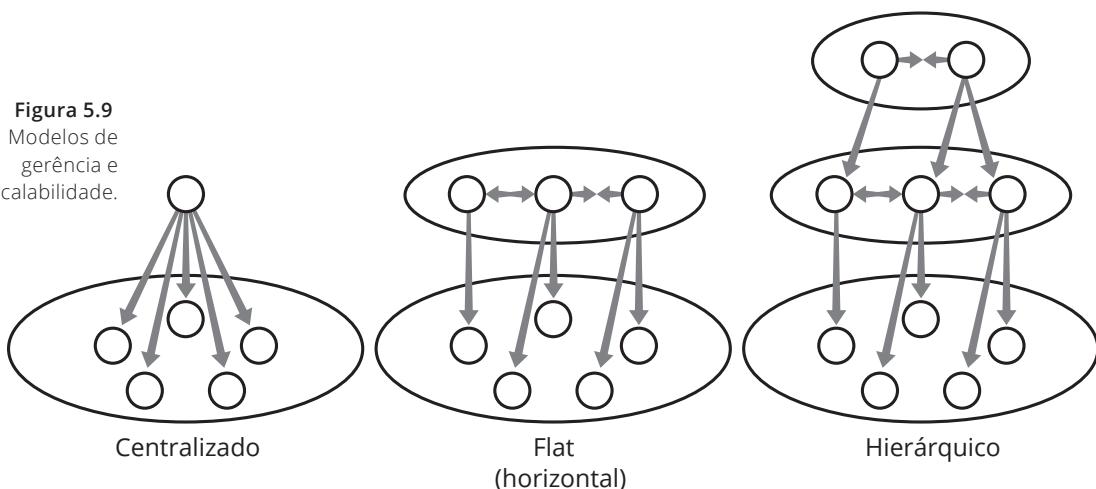
Um dos pontos importantes a observar, principalmente em redes maiores, é a possibilidade que a ferramenta escolhida possua para replicar ou manter de forma distribuída as informações de gerência de redes.

Em grande parte das plataformas, somente é possível gerenciar de forma centralizada. Nesse modelo existe, somente uma estação de gerência que realiza o pooling de todos os equipamentos e diversos objetos em cada um desses dispositivos. O problema aqui reside na capacidade dessa plataforma de realizar todas as medições necessárias na janela de tempo solicitada pelo administrador. Por exemplo, em uma rede 20 mil objetos gerenciáveis e com amostragem de 5 em 5 minutos é necessário que a estação de gerência consiga realizar o pooling em toda a rede a uma taxa aproximada de 70 objetos/segundo, o que pode ser um tempo curto para o caso de atrasos e timeout na resposta de determinados equipamentos.

Se a escalabilidade da rede for algo relevante, é importante analisar quais as outras alternativas existentes de modelos fornecidos pelo desenvolvedor da solução. Em geral, muitas ferramentas têm o conceito de gerente e subgerente, permitindo a criação de uma solução mais robusta com estações de gerência redundantes e com agregadores locais (como diferentes regionais do país ou do continente), ou mesmo através de bancos de dados distribuídos.

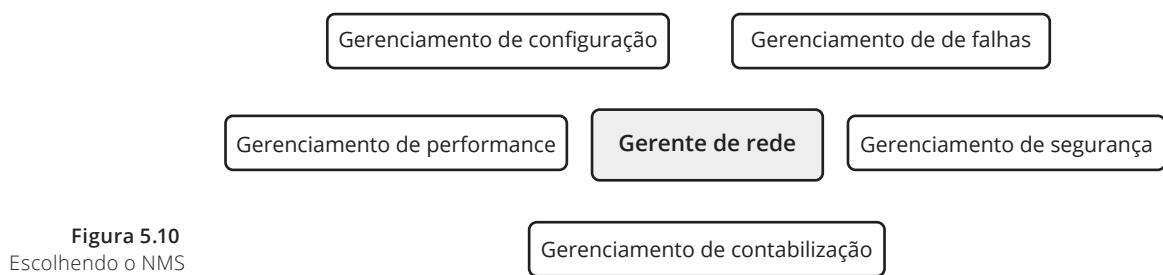


**Figura 5.9**  
Modelos de gerência e escalabilidade.



## Usando uma análise FCAPS na definição de um NMS

Uma abordagem que pode auxiliar na hora de se escolher uma solução de gerência de redes é partir da definição de quais gerências se pretende manter na instituição, e para isso a lembrança das gerências de Falhas, Configuração, Contabilização, Desempenho e Segurança (FCAPS) são um bom ponto de saída.



**Figura 5.10**  
Escolhendo o NMS  
usando FCAPS.

Frequentemente, quando se analisa uma solução de gerência, é difícil realizar uma comparação, e mais difícil ainda tentar se basear em uma análise ou comparativos em sites e revistas. A gerência de redes é sempre um problema de nicho de mercado, e mesmo comparações dentro de mesmos nichos (exemplo: educacional) podem levar à definição de produtos diferentes, por isso é sempre recomendável que a instituição faça ela mesma uma abordagem e uma tabulação dos produtos e de suas necessidades. Uma boa saída para essa avaliação é a definição de quais gerências a ferramenta deve manter, e as funcionalidades esperadas dentro daquela gerência.

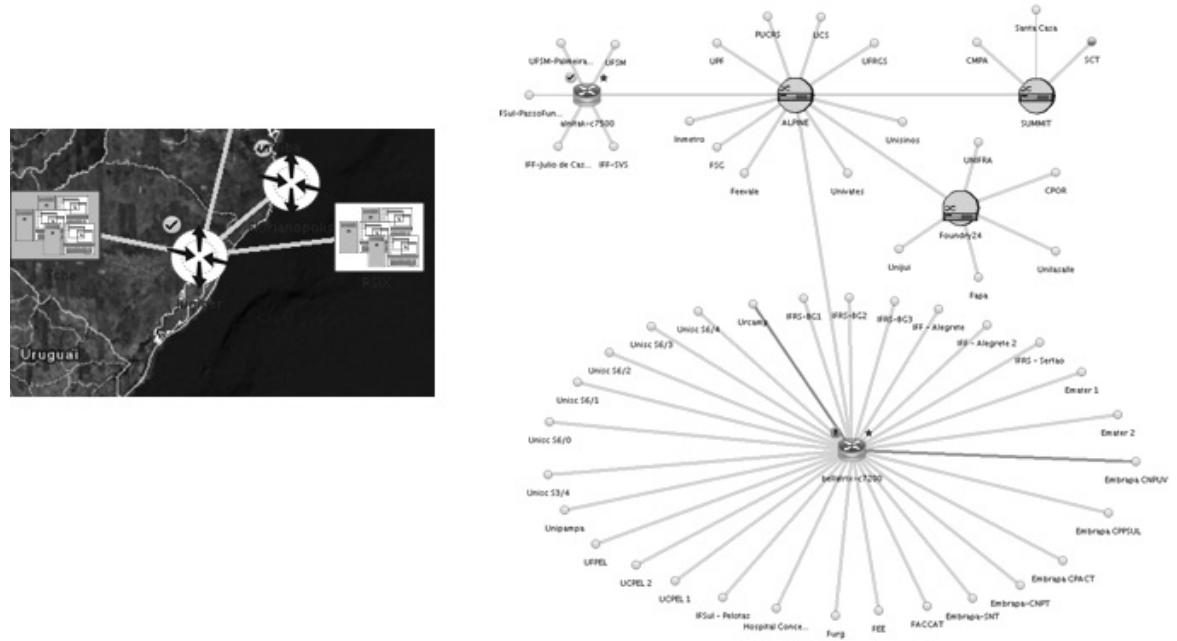
### Gerência de falhas

- Exibir graficamente as falhas da rede.
- Permite visão em tempo real dessas falhas.
- Permite configurar limiares.
- Capacidade para gerenciar exceções filtrando “ruídos” na rede.
- Permite tratar falsos positivos.
- Desejável que a solução alimente diretamente um sistema de tickets com as falhas detectadas.



Um bom NMS possui uma interface clara e multinível, que exibe qual ou quais elementos da rede possuem problemas, permitindo configurar limiares para esses problemas (exemplo: erros e descarte máximo em uma interface de rede).

Algo desejável é a criação de templates, como, por exemplo, circuitos de conexão de clientes MPLS, atrelando a esse conjunto de interfaces determinados limiares de erros e descartes e, é claro, permitindo estabelecer-se exceções que permitam filtrar características já conhecidas da rede, assim como de tratar falsos positivos de falhas.



Outro ponto importante no tratamento de falhas é que cada falha gerada seja capaz de gerar um registro de problemas para o tratamento de incidente, mesmo que a falha tenha sido temporária e tenha se resolvido sem a intervenção do administrador.

Geralmente você vai querer mais do que simplesmente uma mensagem "Device XX is down". Sua solução deve prover informações sobre o que aconteceu antes da falha:

- ▣ Mudanças de configuração;
- ▣ Alterações no inventário de hardware;
- ▣ Mensagens do syslog;
- ▣ Dados sobre performance etc.

#### Gerência de Contabilização

- ▣ Envolve o registro da utilização dos recursos da rede;
- ▣ Tem a função de determinar o baseline da rede.

A gerência de contabilização normalmente é a primeira gerência implementada em qualquer instituição, por conseguir responder perguntas básicas, como: "A rede não está funcionando bem, o que está acontecendo nesse momento que não estava ocorrendo no passado?"

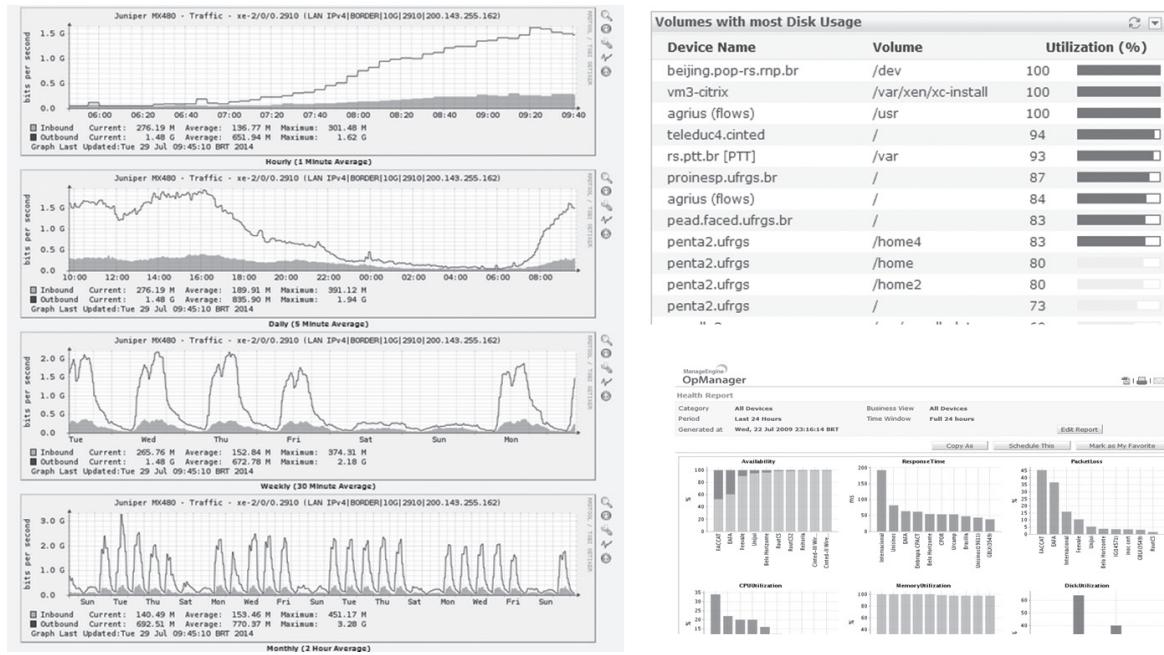
Em geral as ferramentas para a gerência de contabilização são bastante simples de instalar e configurar (cacti, mrtg, etc.) e tem como principal funcionalidade registrar a utilização dos recursos da rede e permitir conhecer o volume demandado em um momento de operação normal da rede (baseline).

**Figura 5.11**  
Analizando uma falha na estação de gerência.



#### Saiba mais

Em geral, as ferramentas para a gerência de contabilização são bastante simples de instalar e configurar (cacti, mrtg etc.) e têm como principal funcionalidade registrar a utilização dos recursos da rede e permitir conhecer o volume demandado em um momento de operação normal da rede (baseline).



**Figura 5.12**

Ferramentas para gerência de contabilização.

Um ponto importante a se observar na gerência de contabilização é a granularidade que se deseja manter dos dados amostrados ao longo do tempo. Em geral, muitas ferramentas não mantêm os dados históricos na forma como foram coletados, mas sim utilizam algoritmos que o “resumem” em estatísticas por horas ou dias, dessa forma armazenando as médias por intervalos de tempo das amostras para facilitar a graficação dos dados (exemplo: mrtg e rrdtool). Caso sua necessidade aponte para manter as séries históricas assim como foram coletadas em cada período, é necessário observar o comportamento da ferramenta em análise e considerar que esses dados consomem recursos não desprezíveis de armazenamento em banco de dados e em recursos computacionais para visualizar esses mesmos dados (CPU e memória da estação de gerência). Praticamente todas as soluções de gerência integradas realizam a gerência de contabilização, cabendo somente ao administrador determinar o período de tempo que a série histórica será mantida (um ano, um mês).

## Gerência de Configuração

Analisando a configuração dos elementos gerenciáveis, você vai desejar informações de inventário para facilitar a abertura de chamados.

- Hardware: tipo de dispositivo, portas, módulos, memória RAM e Flash.
- Dados do bem: part number, número serial.
- Dados do contrato: suporte, SLA, id do circuito etc.

Uma boa ferramenta de gerência de configuração permitirá obter esses dados em tempo real.

Para uma rede sem nenhuma gerência, a gerência de configuração acaba por ser algo para um segundo momento. Entretanto, se o seu levantamento inicial apontou-a como algo necessário para a sua instituição, é importante considerá-la com cuidado na sua definição de ferramenta. Normalmente, ela é uma gerência de implementação particularmente difícil, se considerada a diversidade de equipamentos na instituição. Outro ponto relevante aqui é se ela vai se ater somente aos ativos de rede ou a todos os servidores (hardware e software), backup e configuração dos Sistemas Operacionais e serviços destinados ao usuário final.

Então, aqui é especialmente importante que se faça essa definição.

Caso tenha havido algum problema devido a mudança de configuração você deve facilmente:

- Apontar qual a alteração que causou o problema;
- Saber quem foi o responsável pela mudança;
- Conseguir rapidamente reverter as configurações conforme necessário.

Em alterações na rede, deve ser possível:

- Fazer mudanças de configuração simultaneamente em um grande número de dispositivos;
- As alterações devem realimentar algum tipo de contabilização ou gerenciador de mudanças;
- Deve ser possível manter um histórico dessas mudanças de forma a poder revertê-las em caso de necessidade.



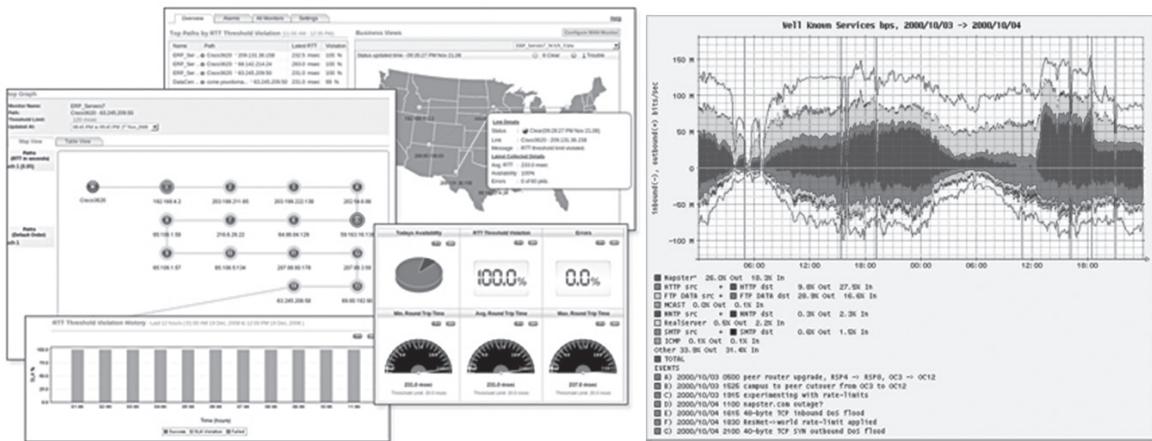
**Figura 5.13**  
Gerência de configurações com o Cisco Config Manager.

## Gerência de Performance

- Capacidade de estabelecer uma baseline com thresholds que notifiquem quando a performance começar a cair.
- Devemos conseguir analisar uma performance vertical abordando o serviço fim-a-fim através de toda a rede:
  - Por exemplo: se existe serviço de VOIP, seu NMS deve conseguir medir a qualidade de voz, jitter, latência etc.
  - Se você usa marcação de QoS, deve ser possível visualizar a quantidade de tráfego que está sendo classificada em cada uma das categorias. Nesse caso, você pode desejar características como Netflow e IP-SLA.



A gerência de desempenho demanda um esforço adicional de configuração por parte do gerente da rede. Embora alguns problemas possam ser detectados automaticamente – como é o caso da gerência de falhas, a gerência de desempenho tem suas próprias nuances. Em geral é necessário que o administrador estabeleça quase que individualmente para elemento, ou classe, determinando quais os limites aceitáveis para o desempenho e em que casos as violações serão consideradas uma falha pelo administrador. Um exemplo típico dessa dificuldade é o uso de uma partição do disco rígido: em um disco de 2GB de dados, 91% de uso é algo preocupante, já em uma partição de 10TB, ter somente 9% de espaço disponível dificilmente geraria um alarme e seria considerado falha. Provavelmente só seria considerado no relatório de crescimento normal dos dados e exigiria atenção no curto ou médio prazo.



**Figura 5.14**  
Gerência de desempenho.

### Network Management Systems disponíveis no mercado

Existe uma gama muito grande de plataformas de gerência de rede no mercado. Como uma avaliação simplificada dessas plataformas comerciais, podemos dividi-las entre as grandes e modulares plataformas, geralmente na faixa de centenas de milhares de dólares, as plataformas médias, também modulares e na faixa de alguns milhares de dólares, e as plataformas menores, geralmente gratuitas, fornecidas por algum fabricante de hardware ou mesmo custando algumas centenas de dólares.

NMSs comerciais.

- Plataformas de alto custo:
  - Bastante completas e modulares;
  - Alta escalabilidade e customização;
  - Custo na ordem de centenas de milhares de dólares;
  - Exemplo: HP-IMC, HP-NMM, Spectrum.
- Plataformas de médio custo:
  - Completas para determinadas tarefas;
  - Alguma escalabilidade;
  - Customização limitada.

- Plataformas gratuitas ou de baixo custo:
  - Funcionalidade limitada a determinados equipamentos ou tipo de gerência;
  - Em geral, somente permitem gerência centralizada (menos de 10 mil objetos);
  - Algumas possuem limitações à diversidade de equipamentos;
  - Em geral, atendem somente a um tipo de gerência específica e necessita ser complementada com outras ferramentas (múltiplas gerências).



## HP IMC – Intelligent Management Center

O HP-IMC tem sua origem nos antigos softwares de gerência especializada para equipamentos da 3Com e HP-Procurve. Em sua evolução natural, restam hoje poucas diferenças em relação a outro software da mesma empresa, o HP-NNM. Normalmente, o HP-IMC tem maior cunho técnico e é voltado para empresas médias e pequenas, enquanto o HP-NNM possui melhor escalonamento para grandes empresas.

Apesar da sua origem, hoje é um produto voltado para redes heterogêneas e possui uma arquitetura voltada a serviços (SOA).

Entre os diferentes tipos de gerência e características dessa ferramenta, podemos citar:

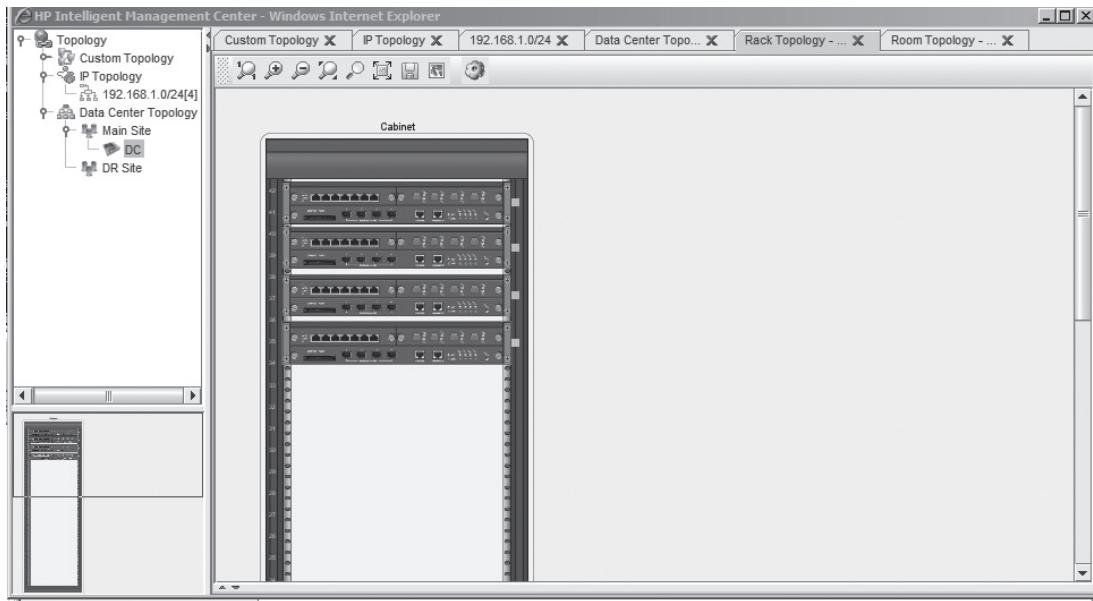
- Suporte a Linux e Windows;
- Interface web para acesso;
- Gerência de falhas e performance com ICMP e SNMP;
- Gerência de configuração (backup, restore e comparação de versões).
- Interface com diferentes CLI de equipamentos para administração de VLANs e ACLs;
- Permite a criação de topologias L2 e L3 entre os diversos elementos da rede;
- Escalabilidade com um modelo hierárquico;
- Captura e análise de flows (netflow);
- Captura e análise de LOGs (syslog);
- Gerência de virtualização (VMWare e Hyper-V);
- Descoberta automatizada da rede;
- Suporte a redes SDN (exemplo: openflow).



Existe uma versão de testes (60 dias), disponível em  
[http://h17007.www1.  
hp.com/us/en/  
networking/products/  
network-management/  
IMC\\_ES\\_Platform/  
\(AVA\).](http://h17007.www1.hp.com/us/en/networking/products/network-management/IMC_ES_Platform_(AVA).)

O software possui várias novidades, além das conhecidas em todas as outras plataformas, como permitir que o datacenter seja desenhado em 3D e se tenha a visão dos equipamentos nos seus racks, integrados à topologia da rede e mostrando quais dos equipamentos possuem algum tipo de alarme.





**Figura 5.15**

Interface  
do HP-IMC  
(fonte: <http://packetpushers.net/review-hp-imc-intelligent-management-center/>).

## HP NNM: Network Node Manager

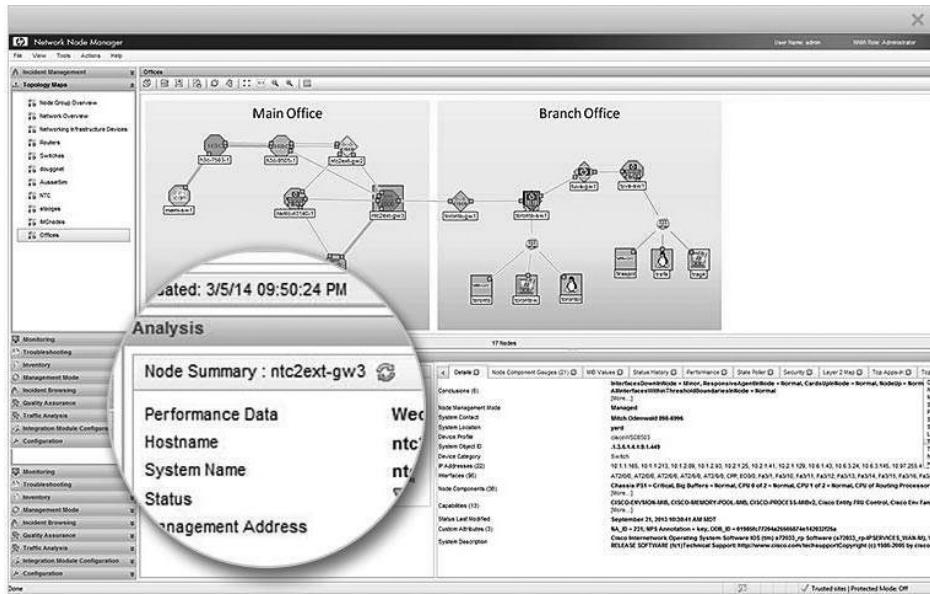
- Sucessor do antigo HP-Openview.
- Software comercial com versão free para até 25 nodos.
- Plataforma com suporte a Business Service Management (BSM).
- Unifica várias gerências.
- Altamente customizável.
- Escalável.
- Plataforma Modular.
- Extensa lista de plug-ins.

O HP-NNM tem a sua origem em um dos primeiros e mais difundidos NMSs, o HP-Openview. Ele teve grande parte do seu código reescrito e hoje em dia possui todas as suas antigas funcionalidades em um sistema baseado em web, podendo ser instalado em sistemas Windows e Linux (Debian e RH). A HP permite o acesso a uma versão freeware do NNM para até 25 dispositivos em <http://www.hp.com/go/nnm>. O HP-NNM é um dos mais completos sistemas de gerência de rede do mercado, é voltado para grandes empresas com um foco maior em Business Service Management (BSM), embora o seu sistema básico seja semelhante a todos os outros sistemas de gerência (ICMP, SNMP, RMON etc.).

Dentre as funcionalidades do software, podemos citar:

- Instalação possível em Linux e Windows, com interface de gerência web;
- Gerência de Falhas com ferramentas automatizadas para a análise (exemplo: roteamento, hardware etc.);
- Descoberta da Rede e mapeamento da topologia camada 2 e camada 3;
- Gerência de performance e geração de relatórios automatizados para SLA e outros;
- Gerência de Configuração e integração com outras ferramentas (CiscoWorks LMS etc.);
- Análise de LOGs e Flows (Netflow);
- Possibilidade de criação de scripts para o tratamento automatizado de alertas;

- Suporte a milhares de diferentes dispositivos;
- Suporte a gerência de protocolos Multicast e IPv6;
- Plataforma modular, com suporte à criação de iSPIs (Plug-in), como para controle de telefonia de vários fabricantes.



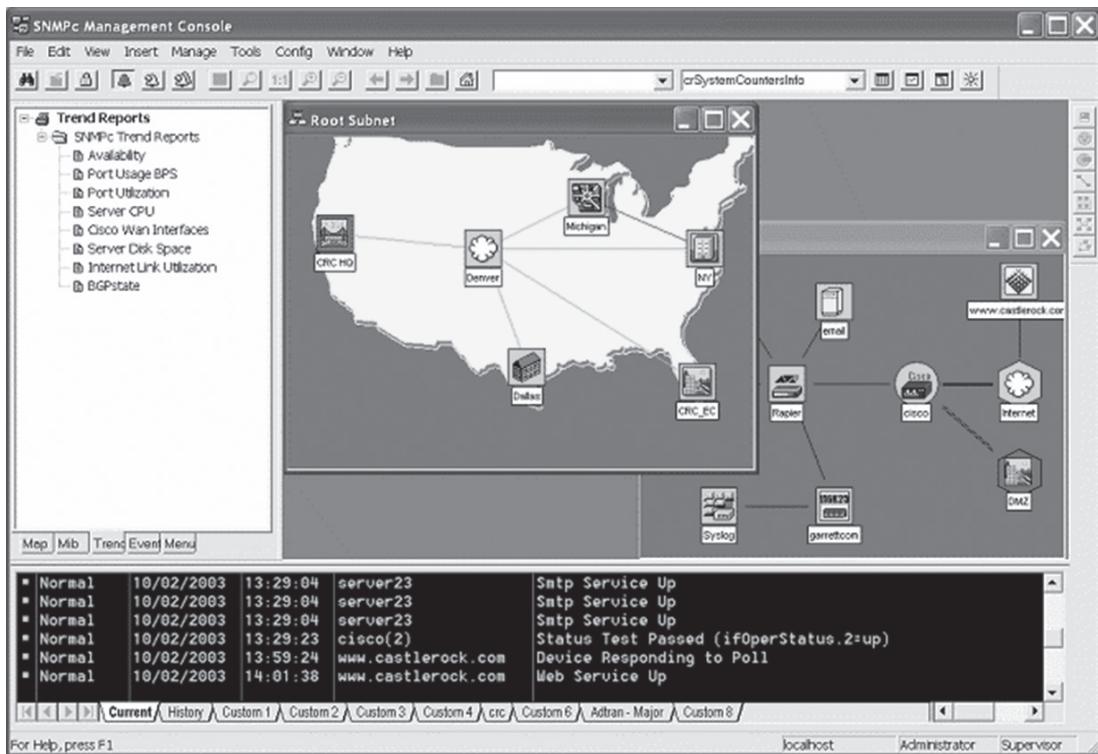
**Figura 5.16**  
Interface do HP-NNM (fonte: <http://www8.hp.com/us/en/software-solutions/network-node-manager-i-network-management-software/>)

## SNMPc da Catlerock

- Possui suporte a SNMPv1, v2 e v3.
- Gerência remota via cliente Windows e Java.
- Visualização da MIB em tempo real.
- Compilador de MIBs.
- Geração automática de relatórios diários, semanais e mensais.
- Interface programável.
- Estrutura distribuída.
- Descoberta de rede automática.

O SNMPc é uma ferramenta de gerência para redes SNMP, que tem como diferencial a ideia de coletar e gerenciar informações em tempo real para hubs, switches, roteadores e servidores, funcionando como um painel de controle da rede para um administrador individual. Ela permite salvar os dados em tempo real, gerando relatórios e alarmes sonoros, e-mails e SMSs em casos de falhas.





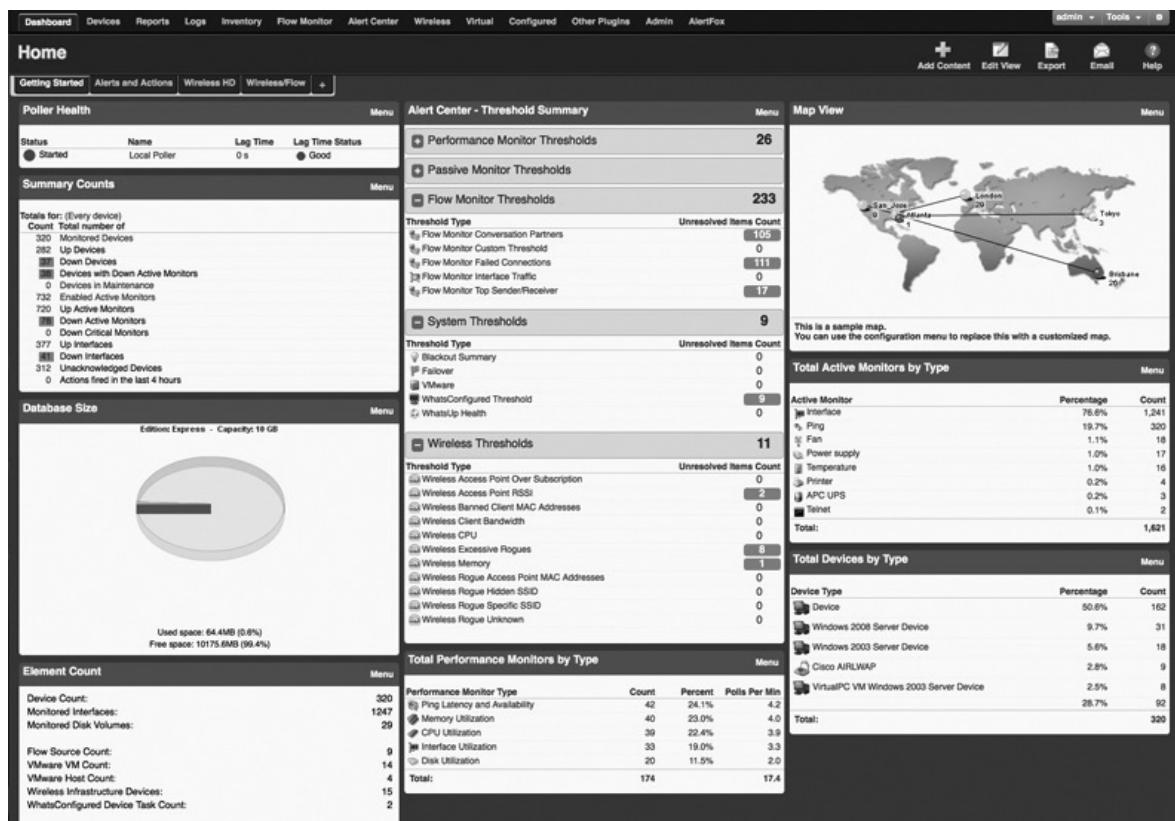
**Figura 5.17**  
Interface do SNMPC

(fonte: [www.castlerock.com](http://www.castlerock.com))

### WhatsUp da IPSwitch

- Descoberta e atualização automática dos mapas da rede.
- Suporte a redes wifi.
- Arquitetura distribuída podendo atender vários datacenters.
- Gerência centralizada.
- Escalabilidade para mais de 20 mil dispositivos.
- Suporte a SNMP e WMI.

Outra ferramenta de custo razoável para gerência de redes pequenas, médias e de datacenters com mais de 15 anos no mercado de gerência. Possui foco na gerência de infraestrutura, como switches, roteadores, servidores e redes wifi.



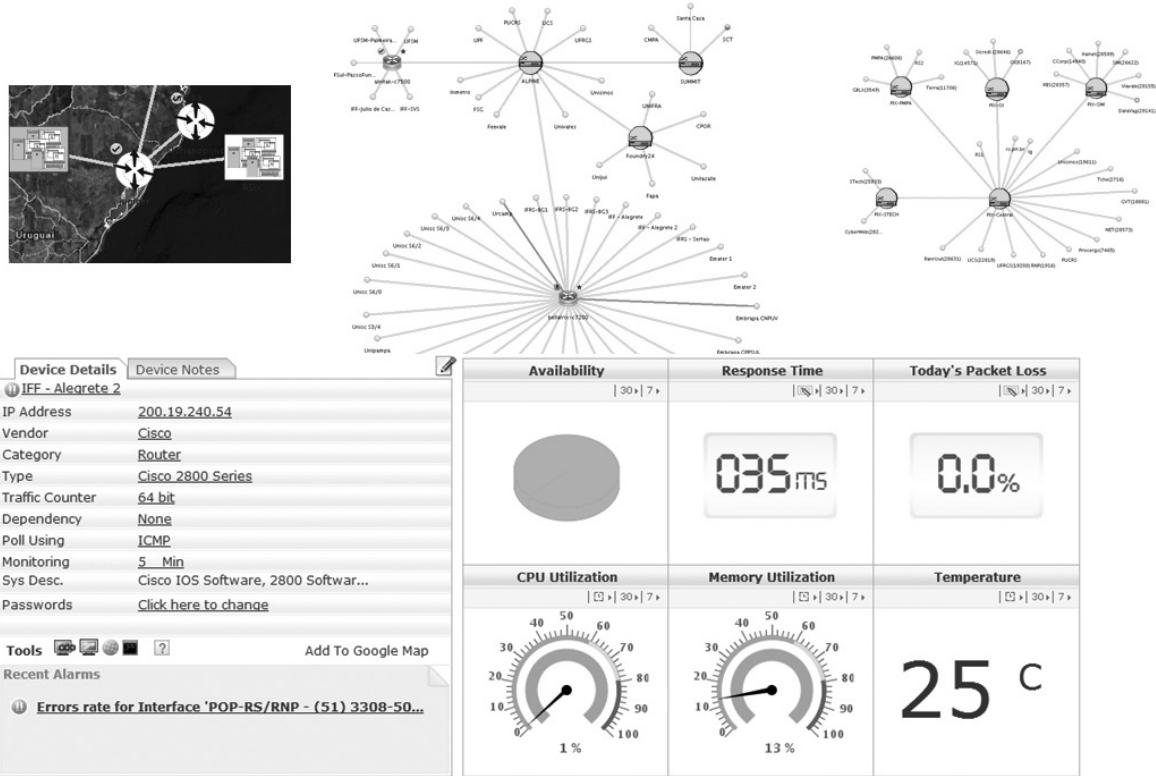
**Figura 5.18**  
WhatsUP Gold  
(fonte: [www.whatsupgold.com](http://www.whatsupgold.com)).

## OPManager, da ManageEngine

- Plataforma modular com suporte a Netflow, IP-SLA etc.
- Descoberta de rede automática.
- Permite a criação de mapas físicos, L2 e L3.
- Monitoramento de serviços.
- Interface totalmente web.
- Suporte a vários plug-ins.
- Suporte a VOIP e Virtualização.
- Geração automática de relatórios.

Essa ferramenta na realidade é parte de um suíte de ferramentas de gerência de TI, servindo como uma alternativa para as principais ferramentas do mercado (HP, IBM e CA). Em uma versão gratuita, permite gerenciar até 10 dispositivos. Na versão básica, a ferramenta é acessível custando a partir de R\$ 10 mil.





**Figura 5.19**

OPManager  
 (fonte: <http://www.manageengine.com/>).





# 6

## Aplicações e plataformas de gerência em software livre

objetivos

Conhecer as soluções em Software Livre para gerência, Descrever o Framework FCAPS. Conhecer as características das ferramentas livres disponíveis.

conceitos

Soluções OSS/FS, Framework FCAPS, Ferramentas livres, Zabbix, Nagios, OpenVAS, OCS Inventory, Spice Works, NTOPIng.

### Soluções OSS/FS

Permite ao usuário executar o programa para qualquer propósito. Entre os principais projetos OSS/FS, destacam-se:

- Linux kernel, Apache, Samba, GNOME, GIMP, MySQL, PostgreSQL, PHP, XFree86, bind, Mozilla e OpenOffice.org.

Na área de gerenciamento rede, existem grandes projetos open source:

- Nagios, OpenNMS, OpenQRM, Zenoss, Netdisco, Ntop, Cacti etc.

Soluções OSS/FS (Open Source Software/Free Software) para gerenciamento substituem soluções monolíticas de fabricantes através de uma abordagem modular. Isso permite que seja montado somente o necessário, de acordo com as especificações, sem necessidade de adquirir sistemas completos e caros.

Soluções OSS/FS são desenvolvidas de forma aberta; por isso, normalmente são aderentes a padrões. Essa característica facilita a interoperabilidade e futuras evoluções. Além disso, o mundo de software aberto garante total transparência no processo de evolução, garantindo que os verdadeiros usuários dos sistemas ditem os rumos do software.



Para saber mais:  
<http://www.gnu.org/philosophy/free-software-for-freedom.html>

Na área de sistema de gerenciamento existe grande potencial de trabalho conjunto das comunidades de software aberto com empresas e grandes clientes interessados em boas alternativas para sistemas de gerenciamento, que pesam muito no orçamento.

A gestão global dos recursos de tecnologia da informação de uma organização é um requisito fundamental. Funcionários e clientes necessitam e contam com os serviços de TI, onde a disponibilidade e o desempenho são obrigatórios, e os problemas devem ser rapidamente identificados e resolvidos, diminuindo, assim, o tempo médio para reparo (MTTR).



## Framework FCAPS

Framework Fault, Configuration, Accounting, Performance e Security (FCAPS).



- ▣ Desenvolvido pela International Organization for Standardization (ISO).
- ▣ Tem como objetivo definir um modelo de gestão de recursos de telecomunicações e de rede.
- ▣ Serve como base para os demais modelos de gestão, como: TMN, OAMP&P, TOM, CMIP/CMIS, ITIL etc.

Na área de sistema de gerenciamento existe grande potencial de trabalho conjunto das comunidades de software aberto, com empresas e grandes clientes interessados em boas alternativas para sistemas de gerenciamento, que pesam muito no orçamento.

A gestão global dos recursos de tecnologia da informação de uma organização é um requisito fundamental. Funcionários e clientes necessitam e contam com os serviços de TI, onde a disponibilidade e o desempenho são obrigatórios, e os problemas devem ser rapidamente identificados e resolvidos, diminuindo, assim, o tempo médio para reparo (MTTR).

O framework Fault, Configuration, Accounting, Performance e Security (FCAPS) foi desenvolvido pela International Organization for Standardization (ISO) e tem como objetivo definir um modelo de gestão de recursos de telecomunicações e de rede. Esse modelo serviu de base para todos os demais modelos de gestão (como exemplo: Telecommunications Management Network – TMN; Operation, Administration, Maintenance and Provisioning – OAM&P; Telecom Operations Map – TOM; Common Management Information Protocol/ Common Management Information Service – CMIP/CMIS; e outros) por definir as áreas funcionais da gerência de redes, que são: a gerência de falhas, a gerência de configuração, a gerência de contabilidade, a gerência de desempenho e a gerência de segurança.

Seguindo o framework FCAPS, abordaremos cada uma das áreas funcionais apresentadas, sugerindo ferramentas que atendam suas necessidades.

## Gerência de Falhas (Fault)



O objetivo do gerenciamento de falhas é reconhecer, isolar, corrigir e registrar as falhas que ocorrem em uma rede. Seus principais componentes são:

- ▣ Detecção de Falhas.
- ▣ Notificação.
- ▣ Tendência.
- ▣ Registro.
- ▣ Resposta.

Uma falha é um evento que tem um significado negativo. O objetivo do gerenciamento de falhas é reconhecer, isolar, corrigir e registrar as falhas que ocorrem em uma rede.

Seus principais componentes são:

- ▣ **Detecção de Falhas:** capacidade de reconhecer um erro quando ele ocorrer;
- ▣ **Notificação:** capacidade de notificar quando a falha ocorre;
- ▣ **Tendência:** permitir a configuração de uma linha de base sobre a operação normal do dispositivo, com o objetivo de detectar desvios ou situações anormais;
- ▣ **Registro:** registrar todas as falhas para contabilidade e auditoria futura;



- ▣ **Resposta:** podendo ser humana ou automatizada.

Alguns exemplos de ferramentas proprietárias são: EMC Smarts, CA Spectrum, HP Network Node Manager i, IBM Tivoli Network Manager, WhatsUpGold etc.

Para um gerenciamento de falhas básico, existem várias ferramentas no mundo livre para a tarefa:

- ▣ **CACTI:** geração de gráficos a partir de dados de rede com RRDTool (<http://www.cacti.net>);
- ▣ **Etherape:** traça gráficos da atividade de rede em formato web (<http://etherape.sourceforge.net/>);
- ▣ **Flow-tools:** ferramentas para tratamento de dados netflow (<http://code.google.com/p/flow-tools/>);
- ▣ **fprobe:** obtém dados do tráfego e exporta-os em formato netflow (Cisco) (<http://sourceforge.net/projects/fprobe/>);
- ▣ **Microsoft Network Monitor:** software de captura de pacotes para análise. Muito parecido com o Wireshark, sua diferença é possibilitar realizar filtros de captura pelo nome do serviço que está gerando tráfego no host local (<http://www.microsoft.com/en-us/download/details.aspx?id=4865>);
- ▣ **monit:** monitoramento em formato web e disparo de ações em função das respostas obtidas (<http://mmonit.com/>);
- ▣ **MRTG:** traça gráficos de informações SNMP obtidas de dispositivos de rede. Possui versões Windows e Linux (<http://oss.oetiker.ch/mrtg/>);
- ▣ **Munin:** estatísticas de tráfego em formato web (<http://munin-monitoring.org/>);
- ▣ **Net-SNMP:** conjunto de aplicações usadas para implementar o protocolo SNMP versões 1, 2c e 3 (<http://www.net-snmp.org/>);
- ▣ **NETXMS:** software para Windows (desktop) e Linux (web) para gerência de dispositivos e alertas, com possibilidade de criação de gráficos (<http://www.netxms.org/>);
- ▣ **ntopng:** sonda (probe) de monitoramento de tráfego que mostra a utilização da rede. Apresenta os dados em formato web (<http://www.ntop.org/products/ntop/>);
- ▣ **RRDTool:** ferramentas para armazenamento de dados e criação de gráficos (<http://oss.oetiker.ch/rrdtool/>);
- ▣ **SNMPPT:** tradutor de traps SNMP escrito em Perl, que utiliza os programas Net-SNMP/UCD-SNMP e snmptrapd (<http://www.snmpppt.org/>);
- ▣ **SpiceWorks:** ferramenta para Windows gratuita para gerenciamento de hosts, que possibilita a inclusão de novos plug-ins gratuitos. (<http://www.spiceworks.com>);
- ▣ **The Dude:** ferramenta para Windows de fácil utilização, para criação de mapas e monitoramento de dispositivos (<http://www.mikrotik.com/thedude>).

## Gerência de configuração (Configuration)

Responsável pelo registro e manutenção da configuração dos dispositivos/serviços de uma rede. Seus principais componentes são:

- ▣ Coletor.
- ▣ Depósito.
- ▣ Trilha.
- ▣ Automatização.

A gerência de configuração é responsável pelo registro e manutenção dos parâmetros de configuração dos dispositivos/serviços de uma rede, tendo como principal objetivo controlar mudanças e configurações, documentando todo o ciclo de vida de um sistema de informação.

Seus principais componentes são:

- ▣ **Coletor:** reúne configurações de todos os ativos monitorados;
- ▣ **Depósito:** local onde são armazenadas as configurações dos dispositivos, para fins de cópia de segurança e recuperação;
- ▣ **Trilha:** monitorar e relatar alterações nas configurações;
- ▣ **Automatização:** capacidade de fazer, automaticamente, alterações de configuração em vários dispositivos, sem interferência de um usuário.

São exemplos de soluções proprietárias: HP Procurve Manager e Cisco Prime. Exemplos de soluções open source:

- ▣ **Configurador Automático e Coletor de Informações Computacionais (CACIC):** ferramenta de inventário desenvolvida pelo governo brasileiro. Suporta geração automática de inventário detalhado de estações Windows e Linux. ([http://www.softwarepublico.gov.br/en/ver-comunidade?community\\_id=3585](http://www.softwarepublico.gov.br/en/ver-comunidade?community_id=3585));
- ▣ **NetCanner:** é uma ferramenta de gerenciamento de configuração que pode ser utilizada para mostrar as configurações dos dispositivos e suas relações com outros dispositivos da rede (<http://bangj.com/>);
- ▣ **OCSInventory:** ferramenta de inventário de hardware e software para estações Windows e Linux (<http://www.ocsinventory-ng.org/>);
- ▣ **OneCMDB:** solução Configuration Management Database (CMDB) compatível com frameworks de tecnologia, como ITIL/COBIT (<http://www.onecmdb.org>);
- ▣ **Puppet:** framework utilizado para gerenciamento de configuração e armazenamento centralizado de dados (<http://www.puppetlabs.com/>);
- ▣ **Rancid:** ferramenta para gerência de configuração de roteadores de diversos fabricantes. Faz uso de um controle de versão (exemplo: CVSweb) para sinalizar e mostrar alterações de configuração (<http://www.shrubbery.net/rancid/>);
- ▣ **Webmin:** é uma interface web de administração de sistemas Unix/Linux. É permitida a configuração de inúmeros itens dos sistemas (contas de usuários, configuração de serviços como Apache, DNS, compartilhamento de arquivos etc. – <http://www.webmin.com/>).



Uma lista de exemplos de ferramentas open source para gerenciamento de configuração está disponível para consulta em: [http://en.wikipedia.org/wiki/Comparison\\_of\\_open-source\\_configuration\\_management\\_software](http://en.wikipedia.org/wiki/Comparison_of_open-source_configuration_management_software).

## Gerência de contabilidade (Accounting)

Responsável por coletar estatísticas e usá-las para definir ou auditar cotas de utilização, medir ou regular o uso de um serviço ou dispositivo. Seus principais componentes são:

- ▣ Gerenciamento de contas, senhas e permissões.
- ▣ Sistema de logs de auditoria.
- ▣ Execução de cópias de segurança de dados críticos.



Gerência responsável por coletar estatísticas de uso dos usuários e usar estas estatísticas para definir/auditar cotas de utilização, medir ou regular o uso de um serviço/dispositivo.



Seus principais componentes são:

- Gerenciamento de contas, senhas e permissões.
  - Serviços de diretórios: Microsoft Active Directory, OpenLDAP etc.;
  - Autenticadores: TACACS e FreeRADIUS etc.;
  - Sistemas de armazenamento de senhas: KeePass (<http://keepass.info/>), Password Manager Pro ([www.manageengine.com/products/passwordmanagerpro/](http://www.manageengine.com/products/passwordmanagerpro/)).
- Sistema de logs de auditoria:
  - Sistema de auditoria básica disponível no Windows (Event Viewer) e Linux (syslog);
  - Syslog-ng: Log: consolidação, análise e envio através de rede de logs (<http://www.balabit.com/network-security/syslog-ng>);
- Execução de cópias de segurança de dados:
  - Microsoft System Center Data Protection Manager e Symantec;
  - Backup Exec and Veritas Netbackup;
  - Windows backup and restore;
  - Amanda (<http://www.zmanda.com/>);
  - Bacula (<http://www.bacula.org/en/>).

## Gerência de desempenho (Performance)

Responsável pela medição e disponibilização das informações sobre aspectos de desempenho dos serviços de rede. Seus principais componentes são:

- Coleta de Dados.
- Visualização.
- Análise de Tendências.
- Sumarização.

A gerência de desempenho é responsável pela medição e disponibilização das informações sobre aspectos de desempenho dos serviços de rede. Esses dados são usados para garantir que a rede opere em conformidade com a qualidade de serviço acordado com seus usuários. Também são usados para análise de tendência.

Seus principais componentes são:

- **Coleta de Dados:** armazena informações de desempenho para recuperação;
- **Visualização:** transforma dados coletados em imagens para uma rápida compreensão;
- **Análise de Tendências:** monitora dados com o objetivo de reconhecer padrões de comportamento;
- **Sumarização:** consolida os dados e informações sobre dispositivos ou serviços.

Tipos de ferramentas open source para gerência de desempenho:

- **Medição de Desempenho:** Cacti, MRTG e SmokePing;
- **Análise Forense:** Wireshark, NTOP, Netstumbler e TCPDump;
- **Gerador de Carga:** Iperf, D ITG e Internet 2 NDT.



## Gerência de segurança (Security)

Responsável por restringir o acesso à rede e impedir o uso incorreto por parte dos usuários. Seus principais componentes são:

- Políticas e procedimentos.
- Sistemas de varredura de vulnerabilidades.
- Segurança física.
- Firewalls.
- Antivírus, Trojan e proteção contra códigos do tipo Malware.

Gerência responsável por restringir o acesso à rede e impedir o uso incorreto por parte de seus usuários, de forma intencional ou não.

Seus principais componentes são:

- Políticas e procedimentos;
- Sistemas de varredura de vulnerabilidades;
- Segurança física;
- Firewalls;
- Antivírus, Trojan e proteção contra códigos do tipo Malware.

A área de segurança é vasta e possui muita complexidade. Realizar gerenciamento de segurança é tarefa para especialistas.

São citadas a seguir algumas ferramentas populares no mundo livre:

- **BackTrack**: distribuição Linux utilizada para testes de penetração e perícia forense (<http://www.backtrack-linux.org/>);
- **Nmap**: scanner de serviços em rede. Pode varrer serviços de várias formas e até encontrar detalhes dos servidores que os implementam (<http://nmap.org/>);
- **The Open Source Security Infrastructure Management System (OpenSIMS)**: conjunto de várias ferramentas de segurança agregadas em um único pacote (<http://opensims.sourceforge.net/>);
- **OpenVas**: scanner de vulnerabilidades em serviços derivado do Nessus (<http://www.openvas.org/>);
- **Osiris**: Monitor de integridade de sistemas. Detecta mudanças ocorridas em arquivos do Sistema Operacional (<http://osiris.shmoo.com/>);
- **Ossec**: Monitor de integridade de sistemas. Detecta mudanças ocorridas em arquivos de instalação e análise de logs (<http://www.ossec.net/>);
- **Snort**: IDS e sniffer (<http://www.snort.org/>);
- Mais ferramentas para segurança: <http://sectools.org/>



## Ferramentas livres

Até o momento, foram relacionadas ferramentas disponíveis para cada uma das áreas do FCAPS, com o intuito de contribuir com maior proximidade entre essas soluções e as necessidades existentes. Serão abordadas nesse tópico a instalação e a utilização básica das ferramentas que abordam mais de uma área do FCAPS ou que possuem grande quantidade de usuários em suas comunidades. São elas:

- ▣ Zabbix.
- ▣ Nagios.
- ▣ OpenVas.
- ▣ SpiceWorks.
- ▣ Ntop.
- ▣ OCSInventory.

Alguns dos sistemas livres para gerenciamento já possuem certa sofisticação e funcionalidades avançadas, que os tornam indicados para vários cenários de gerenciamento. Qualquer lista atual conteria algumas das opções:

- ▣ **Zabbix**: software que permite a monitoração de aplicações e serviços, podendo ser usado para registrar, monitorar, planejar capacidade, disponibilidade e performance dos ativos de uma rede (<http://www.zabbix.com/>);
- ▣ **Nagios**: popular aplicação de monitoração de rede que permite monitorar tanto hosts quanto serviços, alertando-o quando ocorrerem problemas e também quando os problemas forem resolvidos (<http://www.nagios.org/>);
- ▣ **OpenVas**: software que possui um repositório on-line atualizado de vulnerabilidades, as quais podem ser verificadas nos hosts escolhidos pelo usuário. Ao final do procedimento, a ferramenta informa qual a severidade das vulnerabilidades encontradas por host;
- ▣ **SpiceWorks**: ferramenta web para Windows gratuita, permite o Discovery da rede, realizando inferências por ICMP, WMI e SNMP. Possui comunidade ativa, que desenvolve plugins, possibilitando, por exemplo, a geração de gráficos ou o envio de SMS;
- ▣ **Ntop**: ferramenta que captura e analisa os fluxos da rede, gerando estatísticas gráficas dos maiores fluxos capturados;
- ▣ **OCSInventory**: solução de gerenciamento de configuração que centraliza em uma única ferramenta todas as informações de servidores e estações da rede. É necessário instalar um cliente nos hosts a serem inventariados, podendo ser Windows, Linux, Android, Windows Mobile e MacOS.

### Zabbix

O Zabbix é uma solução livre de NMS, sob licença GPL, com monitoramento de dispositivos SNMP e outros, geração de alarmes e alertas, gráficos de desempenho, armazenamento em base MySQL ou Postgres.

O servidor deve ser uma máquina Linux que monitora sistemas clientes em diferentes plataformas e elementos de rede. Um recurso útil do Zabbix é a criação de templates de máquinas. Usando esses templates, é possível herdar características comuns e personalizar dispositivos.

Seu funcionamento é atestado em várias distribuições Linux, sendo que em muitas delas já há pacotes de instalação prontos.



Seu site oficial (com Wiki, documentação e fórum com postagens):  
<http://www.zabbix.org/>



Características do Zabbix:

- Escalável (testado com 5 mil dispositivos, com vários tipos de verificação por segundo);
- Monitoramento em tempo real (desempenho, disponibilidade e integridade);
- Flexíveis de alerta (e-mail, SMS, avisos sonoros e comandos remotos);
- Geração de relatórios e estatísticas;
- Fácil integração com outros módulos de terceiros;
- Service Level Management (SLM), serviços de TI hierárquicos e relatórios;
- Uso de agentes extensíveis para várias plataformas (Unix/Linux, Windows e MacOS);
- Execução automática de comandos remotos;
- Monitoramento com e sem agentes (IPMI, SNMP e traps);
- Segurança (permissão de usuários e autenticação por IP);
- Fácil administração: configuração centralizada e documentação on-line.

#### IPMI

Intelligent Platform Management Interface é um frontend WEB para gerenciamento remoto. Exemplos são: IMM, da IBM; iLO, da HP; DRAC, da Dell.

## Instalação do Zabbix

- Acesse o Terminal;
- Execute o comando a seguir para acessar o terminal como usuário root, informando a sua senha;

```
sudo su
```

- Atualize os pacotes do seu Ubuntu:

```
apt-get update
```

- Instale os pacotes necessários para o Zabbix (será instalado o Apache, o MySQL e o PHP). Informe a palavra “root” como senha do mysql.

```
apt-get install apache2 php5 libapache2-mod-php5 mysql-server mysql-client php-pear libgd-tools libipc-sharedcache-perl lm-sensors php5-mysql
```

- Baixe e instale o Zabbix-server:

```
wget http://repo.zabbix.com/zabbix/2.2/ubuntu/pool/main/z/zabbix-release/zabbix-release_2.2-1+trusty_all.deb
```

```
dpkg -i zabbix-release_2.2-1+trusty_all.deb
```

```
apt-get update
```

- Instale e configure a conexão do Zabbix com o mysql. Nesse procedimento, é pedido o usuário/senha do Mysql (root/root) e a senha para o front-end (zabbix):

```
apt-get install zabbix-server-mysql zabbix-frontend-php
```

- Edite o arquivo de configuração do PHP e coloque as informações a seguir:

```
gedit /etc/php5/apache2/php.ini
```

```
max_execution_time = 300  
memory_limit = 128M  
post_max_size = 16M  
upload_max_filesize = 2M
```



```

max_input_time = 300

date.timezone = America/Sao_Paulo

■ Instale o Zabbix Agent, processo responsável pela coleta das informações:
apt-get install zabbix-agent

■ Habilite o Zabbix Server no Apache:
ln -s /etc/zabbix/apache.conf /etc/apache2/conf-enabled/zabbix.conf

■ Reinicie o serviço do Zabbix e do Apache:
service apache2 restart

service zabbix-server restart

■ Abra um Browser em sua máquina virtual e acesse o site a. O usuário padrão é "admin" e a senha "zabbix":

```

## Utilizando o Zabbix

O Zabbix está estruturado por:

- **Hosts:** são os objetos a serem gerenciados;
- **Itens:** são os itens de monitoração do Hosts;
- **Templates:** possui as configurações de que itens serão gerenciados pelo host;
- **Trigger:** são regras associadas aos itens, as quais são criadas dentro de hosts ou templates para gerar alertas;
- **Discovery rules:** são regras para detectar determinados itens em um host;
- **Actions:** são ações atribuídas para triggers que possam executar comandos, como execução de script, envio de SMS, e-mail etc.

Em nosso exemplo, utilizaremos a monitoria por SNMP. Para realizar isso, primeiramente precisamos configurar um template com a nossa comunidade SNMP, que será atribuído a um Host.

- Acesse “Configuration” > “Templates” em sua interface do Zabbix e clique sobre “Template SNMP Device”, conforme a figura 6.1.

<input type="checkbox"/> <a href="#">Template OS Mac OS X</a>	<a href="#">Applications</a> (10)	<a href="#">Items</a> (19)	<a href="#">Triggers</a> (11)	<a href="#">Graphs</a> (3)	<a href="#">Screens</a> (1)	<a href="#">Discovery</a> (1)	<a href="#">Web</a> (0)	<a href="#">Template App Zabbix Agent</a>
<input type="checkbox"/> <a href="#">Template OS OpenBSD</a>	<a href="#">Applications</a> (10)	<a href="#">Items</a> (29)	<a href="#">Triggers</a> (14)	<a href="#">Graphs</a> (5)	<a href="#">Screens</a> (1)	<a href="#">Discovery</a> (2)	<a href="#">Web</a> (0)	<a href="#">Template App Zabbix Agent</a>
<input type="checkbox"/> <a href="#">Template OS Solaris</a>	<a href="#">Applications</a> (10)	<a href="#">Items</a> (27)	<a href="#">Triggers</a> (14)	<a href="#">Graphs</a> (5)	<a href="#">Screens</a> (1)	<a href="#">Discovery</a> (2)	<a href="#">Web</a> (0)	<a href="#">Template App Zabbix Agent</a>
<input type="checkbox"/> <a href="#">Template OS Windows</a>	<a href="#">Applications</a> (9)	<a href="#">Items</a> (18)	<a href="#">Triggers</a> (9)	<a href="#">Graphs</a> (2)	<a href="#">Screens</a> (1)	<a href="#">Discovery</a> (2)	<a href="#">Web</a> (0)	<a href="#">Template App Zabbix Agent</a>
<input checked="" type="checkbox"/> <a href="#">Template SNMP Device</a>	<a href="#">Applications</a> (2)	<a href="#">Items</a> (6)	<a href="#">Triggers</a> (0)	<a href="#">Graphs</a> (0)	<a href="#">Screens</a> (0)	<a href="#">Discovery</a> (1)	<a href="#">Web</a> (0)	<a href="#">Template SNMP Generic Template SNMP Interfaces</a>

**Figura 6.1**  
Visualização e configuração de templates.

Na próxima tela, clique em “Full clone” para gerar uma cópia desse template, conforme a figura 6.2.



- Altere o nome do template a clique em "Save". Em nosso exemplo, escolhemos o nome "Template\_SNMP\_Device\_RNP\_ESR".

**Figura 6.2**  
Geração de cópia de template.

- Você será encaminhado para a tela anterior, mas agora aparecerá seu novo template. Sobre esse template, clique em "Items", conforme a figura 6.4.

**Figura 6.3**  
Cópia de template.

<input type="checkbox"/>	Template OS Windows	Applications (9)	Items (18)	Triggers (9)	Graphs (2)	Screens (1)	Discovery (2)	Web (0)	Template App Zabbix Agent
<input type="checkbox"/>	Template SNMP Device	Applications (2)	Items (6)	Triggers (0)	Graphs (0)	Screens (0)	Discovery (1)	Web (0)	Template SNMP Generic, Template SNMP Interfaces
<input type="checkbox"/>	Template SNMP Device_RNP_ESR	Applications (2)	Items (6)	Triggers (0)	Graphs (0)	Screens (0)	Discovery (1)	Web (0)	Template SNMP Generic, Template SNMP Interfaces
<input type="checkbox"/>	Template SNMP Disks	Applications (1)	Items (0)	Triggers (0)	Graphs (0)	Screens (0)	Discovery (1)	Web (0)	-

- Na tela dos "items", selecione todos os itens clicando no canto superior esquerdo e escolha a opção "Mass update", como mostrado na figura 6.5.

**Figura 6.4**  
Listagem de templates.



<a href="#">« Template list</a> <b>Template:</b> <a href="#">Template SNMP Device_RNP_ESR</a> <a href="#">Applications (2)</a> <a href="#">Items (6)</a> <a href="#">Triggers (0)</a> <a href="#">Graphs (0)</a> <a href="#">Screens (0)</a>										
<a href="#">Discovery rules (1)</a> <a href="#">Web scenarios (0)</a>										
<input checked="" type="checkbox"/>	Wizard	Name	↑	Triggers	Key	Interval	History	Trends	Type	Application
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Template SNMP Generic: <a href="#">Device contact details</a>			sysContact	3600	7		SNMPv2 agent	General
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Template SNMP Generic: <a href="#">Device description</a>			sysDescr	3600	7		SNMPv2 agent	General
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Template SNMP Generic: <a href="#">Device location</a>			sysLocation	3600	7		SNMPv2 agent	General
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Template SNMP Generic: <a href="#">Device name</a>			sysName	3600	7		SNMPv2 agent	General
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Template SNMP Generic: <a href="#">Device uptime</a>			sysUpTime	60	7	365	SNMPv2 agent	General
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Template SNMP Interfaces: <a href="#">Number of network interfaces</a>			ifNumber	3600	7	365	SNMPv2 agent	Interfaces
<a href="#">Mass update</a>			▼	<a href="#">Go (6)</a>						

Zabbix 2.2.4 Copyright 2001-2014 by Zabbix SIA | Connected as 'Admin'

**Figura 6.5** Execução de update nos itens de um template.

- Escolha a opção “SNMP community” e altere a comunidade para “esr-rnp”, como mostrado na figura 6.6.

**ZABBIX** Help | Get support | Print | Profile | Logout 127.0.0.1

Monitoring Inventory Reports Configuration Administration

Host groups | Templates | Hosts | Maintenance | Actions | Screens | [Slide shows](#) | [Maps](#) | [Discovery](#) | [IT services](#) Search

Configuration of templates » Dashboard » Configuration of host groups » Configuration of History: templates » Configuration of items

**CONFIGURATION OF ITEMS**

« Template list   **Template:** [Template SNMP Device\\_RNP\\_ESR](#)   [Applications \(2\)](#)   [Items \(6\)](#)   [Triggers \(0\)](#)   [Graphs \(0\)](#)   [Screens \(0\)](#)   [Discovery rules \(1\)](#)   [Web scenarios \(0\)](#)

**Mass update**

Type  Original  
 SNMP community    
 Context name  Original

**Figura 6.6** Alteração dos itens de um template.

- Terminada a etapa de configuração do template, podemos adicionar nossos hosts utilizando a comunidade configurada. Para isso, acesse o menu “Configuration” > “Hosts” e clique em “Create Hosts”, na parte superior direita.
- Nesta interface, informe o nome do dispositivo, os grupos os quais esse dispositivo faz parte, seu FDQN, IP e principalmente a porta de gerência, que em nosso exemplo é a porta do SNMP (161).



**CONFIGURATION OF HOSTS**

Host	Templates	IPMI	Macros	Host inventory																									
Host name: servidor1	Visible name:																												
Groups		In groups: Linux servers	Other groups: Discovered hosts, Hypervisors, Templates, Virtual machines, Zabbix servers																										
		<input type="button" value="&lt;&lt;"/> <input type="button" value="&gt;&gt;"/>																											
<b>New group:</b> Servidores <table border="1"> <thead> <tr> <th>IP address</th> <th>DNS name</th> <th>Connect to</th> <th>Port</th> <th>Default</th> </tr> </thead> <tbody> <tr> <td><a href="#">Add</a></td> <td>200.132.0.188</td> <td>servidor.dominio.br</td> <td><input type="radio"/> IP <input type="radio"/> DNS</td> <td>161</td> </tr> <tr> <td><a href="#">Add</a></td> <td></td> <td></td> <td></td> <td><input checked="" type="radio"/> Remove</td> </tr> <tr> <td><a href="#">Add</a></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td><a href="#">Add</a></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>					IP address	DNS name	Connect to	Port	Default	<a href="#">Add</a>	200.132.0.188	servidor.dominio.br	<input type="radio"/> IP <input type="radio"/> DNS	161	<a href="#">Add</a>				<input checked="" type="radio"/> Remove	<a href="#">Add</a>					<a href="#">Add</a>				
IP address	DNS name	Connect to	Port	Default																									
<a href="#">Add</a>	200.132.0.188	servidor.dominio.br	<input type="radio"/> IP <input type="radio"/> DNS	161																									
<a href="#">Add</a>				<input checked="" type="radio"/> Remove																									
<a href="#">Add</a>																													
<a href="#">Add</a>																													
Monitored by proxy: (no proxy) Status: Monitored																													
<input type="button" value="Save"/> <input type="button" value="Cancel"/>																													

- Na segunda aba, inclua o template cadastrado anteriormente e salve seu host.

**Figura 6.7**  
Inclusão de host no Zabbix.

**CONFIGURATION OF HOSTS**

Host	Templates	IPMI	Macros	Host inventory									
<table border="1"> <thead> <tr> <th>Linked templates</th> <th>Name</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td colspan="3">No templates linked.</td> </tr> <tr> <td colspan="3">           Link new templates: <input type="text" value="Template SNMP Device_RNP_ESR"/> <input type="button" value="Select"/>  <input type="button" value="Add"/> </td> </tr> </tbody> </table>					Linked templates	Name	Action	No templates linked.			Link new templates: <input type="text" value="Template SNMP Device_RNP_ESR"/> <input type="button" value="Select"/> <input type="button" value="Add"/>		
Linked templates	Name	Action											
No templates linked.													
Link new templates: <input type="text" value="Template SNMP Device_RNP_ESR"/> <input type="button" value="Select"/> <input type="button" value="Add"/>													
<input type="button" value="Save"/> <input type="button" value="Cancel"/>													
<b>Zabbix 2.2.4 Copyright 2001-2014 by Zabbix SIA</b>													

- Agora é possível verificar o status de nossos servidores. Observe que a coluna "Availability" (figura 6.9) mostra como a coleta das informações de nossos hosts é realizada. No exemplo, o servidor1 está monitorado por SNMP, enquanto o Zabbix server está sendo monitorado pelo "Zabbix Agentd".

**Figura 6.8**  
Inclusão de host no Zabbix.



**Figura 6.9**

Hosts monitorados pelo Zabbix.

Ferramenta que atende a Fault e Performance no modelo FCAPS. Entre suas principais características, podemos destacar:

- Detecção rápida de falha de infraestrutura;
- Mais de 10 anos de desenvolvimento ativo;
- Escalas para monitorar milhares de nós;
- Monitora serviços de rede (SSH, SMTP, POP3, HTTP, NNTP, ICMP e SNMP) e possibilita a integração de plugins para monitoria de outros serviços;
- Monitoração remota suportada através de túneis criptografados SSH ou SSL;
- Checagem dos serviços paralelizados;
- Possibilidade de criação de mapas;
- Rotação automática de log;
- Suporte para implementação de monitoração redundante;
- Software Open Source;
- Lançado sob a licença GPL;
- Alertas por SMS;
- Interação com Banco de Dados;
- Possibilita a utilização de outros frontends, incluindo soluções para mobile.

Existem outras ferramentas que integram novas interfaces e funcionalidades utilizando o nagios, como por exemplo o Centreon.

### Instalação do Nagios

- Para instalá-lo, siga os procedimentos a seguir:
 

```
apt-get install nagios3
```
- Durante a instalação, será solicitado a senha de administração do Nagios. Digite "admin";
- Para acessar o nagios, digite no seu browser web o endereço: <http://localhost/nagios3> com usuário "nagiosadmin" e senha "admin";
- Se o serviço não responder, verifique se o apache está em execução;



- Para ativar o Nagios, digite:

```
a2enconf nagios3  
etc/init.d/apache2 start
```

## Customizações do software

- Nesta etapa, vamos realizar algumas configurações básicas do software Nagios. Para isso, siga atentamente os passos a seguir.
  - 1. Definir um dispositivo a ser monitorado: (Máquina Virtual do Professor);
  - 2. Definir os serviços que serão monitorados;
  - 3. Inserir o dispositivo em um grupo de monitoração.

### 1. Definindo um dispositivo e seus serviços:

Edite o arquivo a seguir, inserindo as linhas informadas, alterando de acordo com sua estrutura de rede:

```
gedit /etc/nagios3/conf.d/servidor1.cfg  
;----- inicio  
;  
; Define uma estação a ser monitorada  
  
define host{  
    use generic-host ; Nome do template ( template definido no arquivo  
    "generic-host_nagios3.cfg"  
    host_name <SERVIDOR1.DOMINIO> - Nome do host a ser monitorado  
    alias <SERVIDOR1> - Apelido do host a ser monitorado  
    address <IPSERVIDOR> - Endereço IP do host  
    parents localhost - Hierarquia utilizado para traçar o gráfico veja  
    a aba "Status Map" no menu do Nagios  
    hostgroups esr, all - Grupos que a estação faz parte  
    notification_interval 120 - Tempo entre notificações  
    notification_period 24x7 - Período que as notificações serão  
    enviadas  
    notification_options d,u,r - Eventos que geram notificações "d"own,  
    "u"nreachable, and "r"ecovered  
    contact_groups admins - Para onde as notificações serão enviadas e  
    como (mail, pager) veja arquivo "contact_groups"  
}
```



2. Definindo um serviço a ser monitorado. Incluindo as linhas a seguir no mesmo arquivo:

```
define service {  
    service_description SNMP  
    use generic-service ;  
    host_name SERVIDOR1.DOMINIO ; host a ser monitorado  
    service_description snmp  
    check_command check_snmp! -C esr-rnp -o sysUpTime.0 ;  
        comunidade SNMP do host  
}
```

Não se esqueça de salvar as alterações.

3. Edite o arquivo que contém a definição dos grupos de hosts que serão monitorados:

```
gedit /etc/nagios3/conf.d/hostgroups_nagios2.cfg #  
Edite o arquivo adicionando as linhas a seguir no final do arquivo:  
;---- inicio  
define hostgroup {  
    hostgroup_name esr  
    alias ESR  
    members <SERVIDOR1.DOMINIO> - Altere, informando o nome do host  
    definido anteriormente  
}  
;----fim
```

Não esqueça de salvar as alterações.

- Adicione as seguintes linhas no arquivo */etc/nagios3/commands.cfg*:

```
gedit /etc/nagios3/commands.cfg  
; ---- inicio  
# 'check_snmp' command definition  
define command{  
    command_name check_snmp  
    command_line $USER1$/check_snmp -H $HOSTADDRESS$ $ARG1$  
}  
;----fim
```

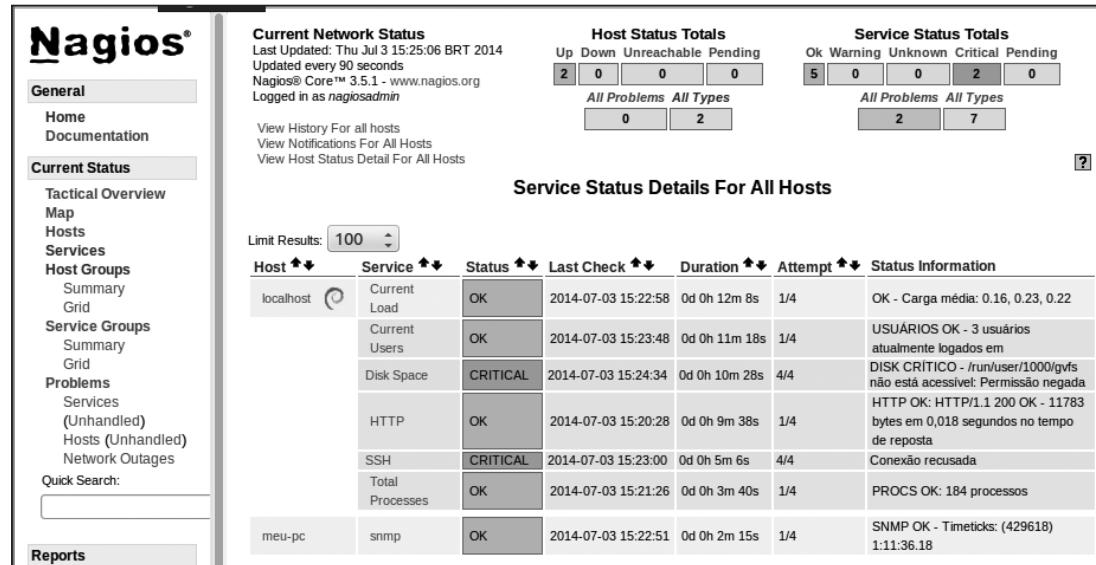
Não esqueça de salvar as alterações.

- Para que as configurações entrem em vigor, reinicie o serviço da seguinte maneira:

```
/etc/init.d/nagios3 restart
```



- Visualize no browser e note as alterações. Verifique o menu “Services”, conforme a figura 6.10.



- O intervalo de verificação do sistema é de 90 segundos, configurável no arquivo: `/etc/nagios3/cgi.cfg` na diretiva `refresh_rate=90`
- Para observar quais serviços podem ser monitorados, acesse a página de plugins do nagios: `nagios_plugins`;
- É possível utilizar o Nagios com o banco de dados Mysql, no entanto, essa não é a configuração padrão. O módulo `perfparse` facilita a utilização do banco de dados Mysql pelo nagios.
- Demais informações:
- O Nagios possui uma estrutura de diretórios para a definição de suas diretivas de configuração, representada a seguir:

```

Plugins: /usr/lib/nagios/plugins/
Arquivos de Configuração: /etc/nagios3
Documentação: /usr/share/nagios3/htdocs
Plugins: /etc/nagios-plugins/config/
CGI: /usr/lib/cgi-bin/nagios3
Binarios: /usr/sbin
Script de inicialização: /etc/init.d/nagios3
  
```

**Figura 6.10**  
Verificação de serviços monitorados por host.

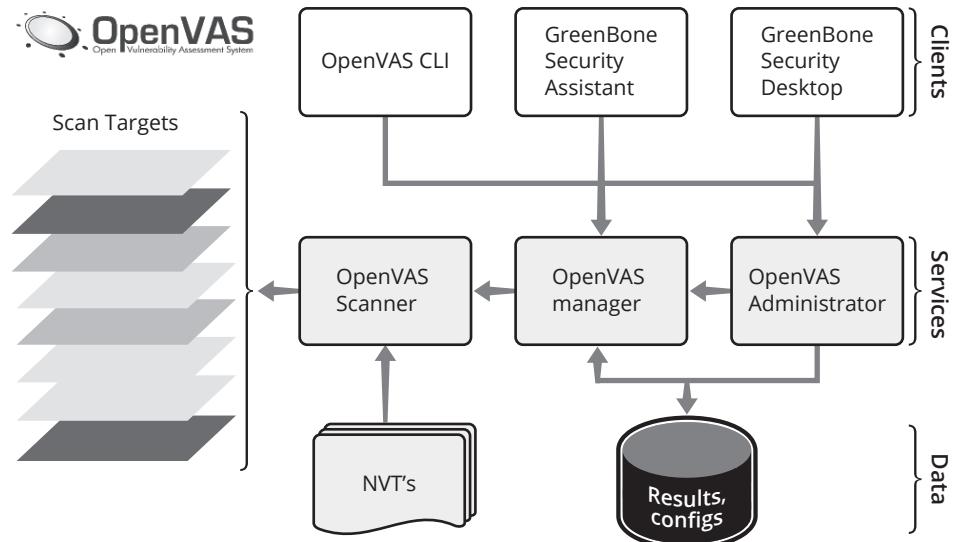
## OpenVAS

O Open Vulnerability Assessment System (OpenVAS) é uma estrutura de vários serviços e ferramentas que oferecem uma solução abrangente e poderosa para varredura, identificação e gerenciamento de vulnerabilidades nos ativos conectados a uma rede. O projeto OpenVas nasceu como um subproduto do Nessus, isso porque os plugins do Nessus deixaram de ser fornecidos sobre licença GPL.

O sistema de varredura de segurança é alimentado diariamente através de uma rede de desenvolvedores de plugins (NVTs), tendo mais de 30 mil plugins (abril de 2013).



Todos os produtos do OpenVAS são software livre, e a maioria dos componentes está licenciado sob a GNU General Public License (GNU GPL).



**Figura 6.11**  
Estrutura do  
OpenVAS.

Para utilizar o openvas, basta cadastrar qual a rede que será analisada. Para isso, acesse “Configuration” > “Targets” e informe um nome para sua rede e quais os hosts que serão analisados (figura 6.12). Observe que o host ou rede informada aqui necessariamente necessita ser a qual você analisará.

**Figura 6.12**  
Criação de Targets.

- Como próxima etapa, acesse “Scan Management” > “New Tasks” e crie uma tarefa informando o nome e o Scan Target criado anteriormente.



**New Task** ?

Name	Análise rede Local
Comment (optional)	
Scan Config	Full and fast
Scan Targets	rede-local
Escalator (optional)	-
Schedule (optional)	-
Slave (optional)	-
Observers (optional)	

**Scan Intensity**

Maximum concurrently executed NVTs per host	4
Maximum concurrently scanned hosts	20

**Create Task**

**Figura 6.13**  
Criação de Task.

- Após criada a tarefa, selecione o botão de play:

- Após o término do scan, no menu “Asset Management” aparecerá a relação de hosts que possuem vulnerabilidades que merecem a atenção do administrador do sistema, conforme a figura 6.15.

**Figura 6.14**  
Execução de tarefa no openvas.

IP	High	Medium	Low	Last Report	OS	Ports	Apps	Distance	Prognosis	Reports	Actions
10.10.1.2	0	3	2	Oct 13 2013	Ubuntu	4	1	1	Medium	1	[Search] [Details]
10.10.1.5	0	1	1	Oct 13 2013	Windows 7	1	0	1	Medium	1	[Search] [Details]
10.10.1.100	0	1	3	Oct 13 2013	Ubuntu	4	0	1	Medium	1	[Search] [Details]
127.0.0.1 (localhost)	1	5	5	Oct 13 2013	Ubuntu	5	5	0	High	1	[Search] [Details]

Total: 4

**Figura 6.15**  
Resultado da verificação de vulnerabilidades.



## OCS Inventory

OCS Inventory é um software livre utilizado para inventariar o hardware e software de estações de trabalho e servidores conectados a uma rede através de um agente. Para tal, ele usa um agente, que deve ser instalado em cada estação de trabalho, responsável por realizar o inventário em computadores cliente e enviar essas informações para um servidor de gerenciamento central que consolida os resultados do inventário permitindo a geração de relatórios.

A comunicação entre agentes e servidor de gerenciamento é feita usando os protocolos HTTP/HTTPS. Todos os dados são formatados em XML e comprimidos utilizando a biblioteca Zlib para reduzir a média de tráfego de rede.

Ele permite identificar, apenas, softwares instalados via yum ou apt (no linux) ou registrados pelo Windows Installer (no Windows).

Como já informado, os agentes devem ser instalados nos computadores-clientes, podendo ser implantados através de login scripts ou Active Directory GPO no Windows. No Linux, o agente deve ser instalado manualmente.

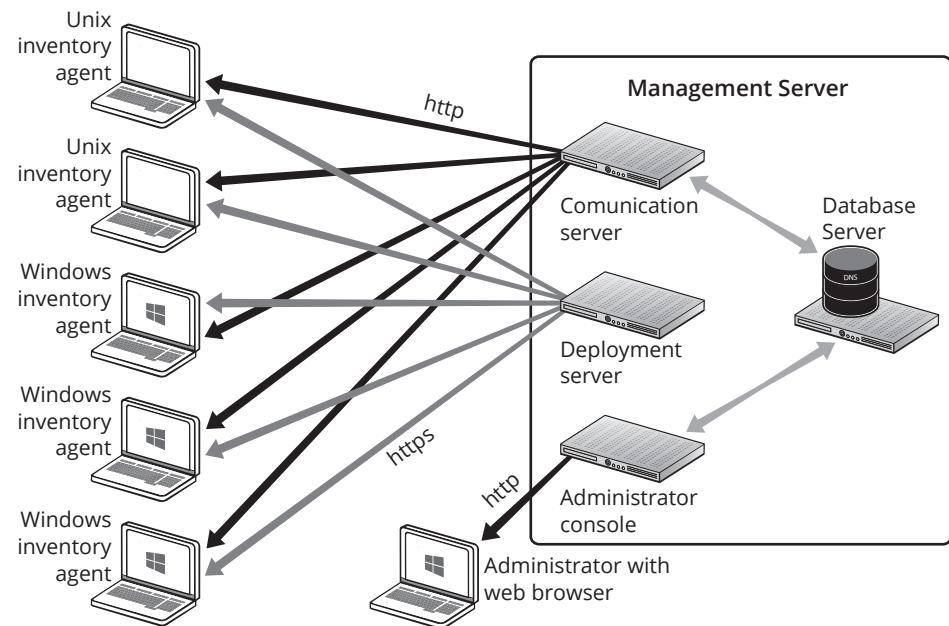


Figura 6.16  
Estrutura do OCS Inventory.

O servidor de gerenciamento contém quatro componentes principais:

- **Serviço de banco de dados:** para armazenamento das informações enviadas pelos agentes;
- **Serviço de Comunicação:** que vai tratar das requisições HTTP entre o servidor de banco de dados e agentes;
- **Serviço de Implantação:** que armazena todas as configurações de implantação de pacote (exige HTTPS);
- **Console de administração:** que permitirá aos administradores elaborar relatórios a partir das informações armazenadas no servidor.

Esses quatro componentes podem ser hospedados em um único computador ou em computadores diferentes, para permitir balanceamento de carga. Acima de 10 mil computadores inventariados, é interessante usar pelo menos dois servidores diferentes, um para o serviço de banco de dados e comunicação e outro para réplica do banco de dados, administração e deployment.



## Instalação do OCS Inventory

- Para instalar o OCS inventory, execute:

```
apt-get install ocsinventory-reports ocsinventory-server
```

- Após a instalação, acesse <http://localhost/ocsreports/> para continuar a instalação.  
Informe o usuário e a senha root/root.

localhost/ocsreports/

DB configuration not completed. Automatic install launched

WARNING: You will not be able to build any deployment package with size greater than 8MB  
You must raise both post\_max\_size and upload\_max\_filesize in your php.ini to increase this limit.

WARNING: If you change default database name (ocsweb), don't forget to update your ocs engine files

MySQL login:

MySQL password:

Name of Database:

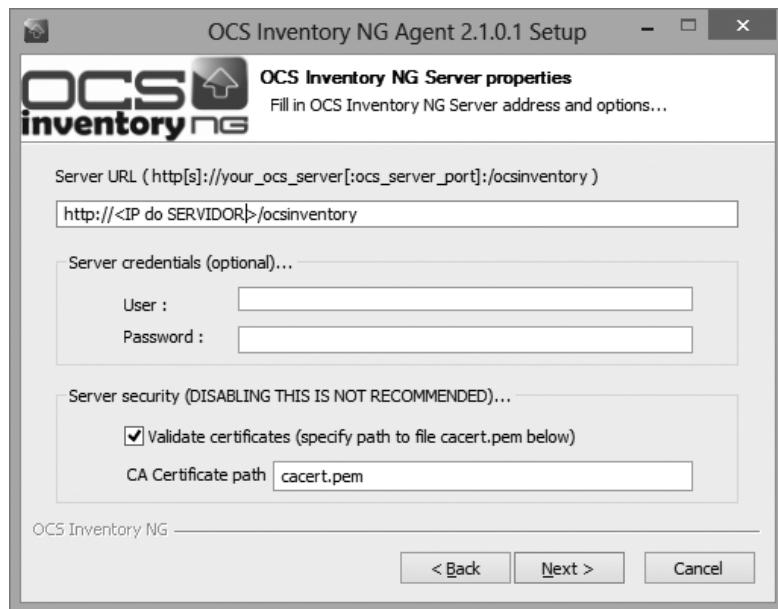
MySQL HostName:

Send

**Figura 6.17**  
Continuação da instalação do OCS Inventory.

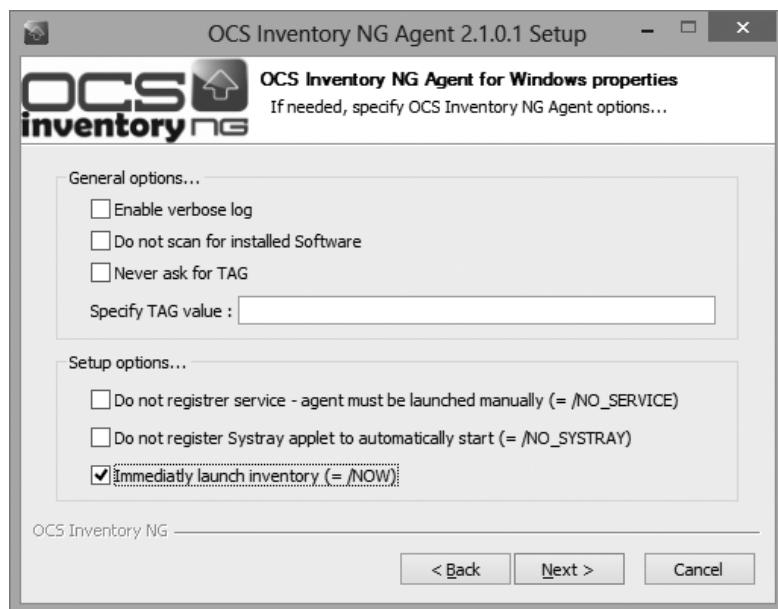
- Terminada a instalação, acesse a interface com o usuário admin e senha admin;
- Instale o cliente o ocs-inventory em seu Windows para enviar o inventário para o servidor.  
Baixe o agente no site: <https://launchpad.net/ocsinventory-windows-agent/2.x/2.0.5/+download/OCSNG-Windows-Agent-2.0.5.zip>
- Durante a instalação do agente, informe o IP do servidor.





**Figura 6.18**  
Instalação do agente do OCS Inventory.

- Selecione a última opção para realizar um inventário logo após acabar a instalação.



**Figura 6.19**  
Instalação do agente do OCS Inventory.

- Agora, acesse a interface de gerência do OCSInventory para visualizar o inventário realizado. (<http://localhost/oscreports>).



The screenshot shows a software interface for managing computer inventories. At the top, there's a toolbar with various icons. Below it is a search/filter bar with fields for 'Show' (set to 20), 'Restrict view', 'Filter', and 'Add column'. A message box indicates '1 Result(s) (Download)'. The main area displays a table with one row of data:

Account info: TAG	Last inventory	Computer	User	Operating system	RAM (MB)	CPU (MHz)	Select	Delete
NA	2012-10-12 19:45:57	LOUREIRO ULTRA	CesarAugusto	Microsoft Windows 8 Pro	4008	1601	<input type="checkbox"/>	

**Figura 6.20**  
Visualização do inventário do OCSinventory.

## Spice Works

Spiceworks é um MNS que, além de gerenciar falhas e performance, possibilita o gerenciamento de configuração do modelo FCAPS. Sua distribuição é gratuita, contudo, possui propagandas. Entre suas principais características, podemos destacar:

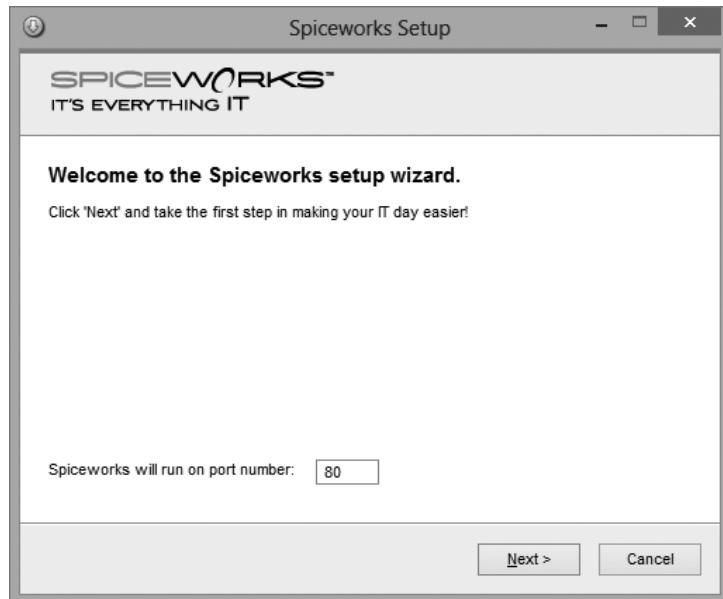
- Comparação de configurações (hardware/software) de duas ou mais estações de trabalho;
- Serviço de descoberta automática de sistemas e dispositivos;
- Possibilidade de anexar documentação extra, notas e anotações customizadas em qualquer dispositivo;
- Carrega todos os dados do Active Directory, facilitando o gerenciamento.
- Monitora Microsoft Exchange;
- Alertas pré-programados;
- Controle de licenças de software;
- Controle de chamados, integrando usuário, chamado e equipamentos, possibilitando criação de chamado por e-mail.
- Possibilita a integração de plugins de forma gratuita.

### Instalação do SpiceWorks

Realize o download do SpiceWorks no endereço: <http://download.spiceworks.com/Spiceworks.exe>. (existe uma cópia na pasta do curso em seu desktop).

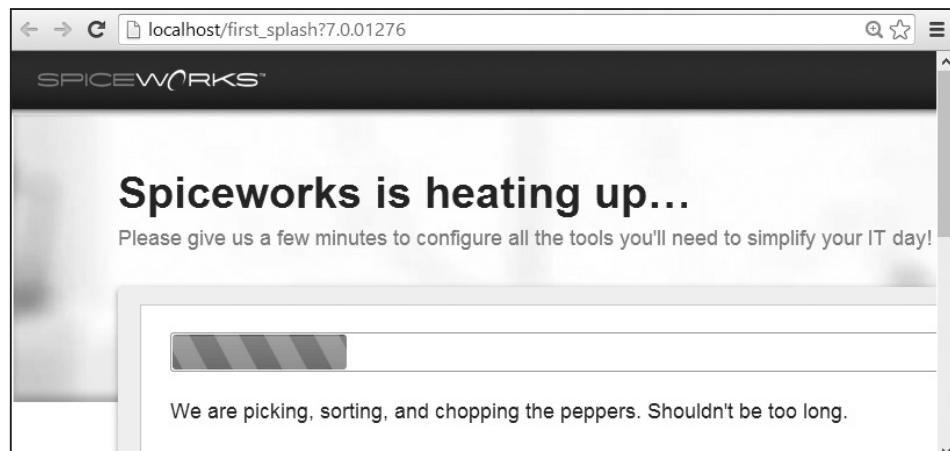
- Execute o arquivo e responda afirmativamente para as opções solicitadas, clicando em "Next", conforme a figura 6.21.





**Figura 6.21**  
Instalação do  
Spiceworks.

- Após a instalação dos arquivos, será criado um serviço web em seu computador e será acessado automaticamente pelo instalador, para terminar o processo de instalação, conforme a figura 6.22.



**Figura 6.22**  
Segunda etapa  
da instalação do  
Spiceworks.

- Ao final desse procedimento é necessário criar uma conta. Retire a seleção das duas perguntas, conforme a figura 6.23.



SPICEWORKS

First Name:

Last Name:

Username (Email): user@company.com

Password: 5 character minimum

Confirm Password:

Company name:

Industry: Select one

Receive special offers on IT products and services.

Participate in periodic surveys about IT topics.

Already have an account?

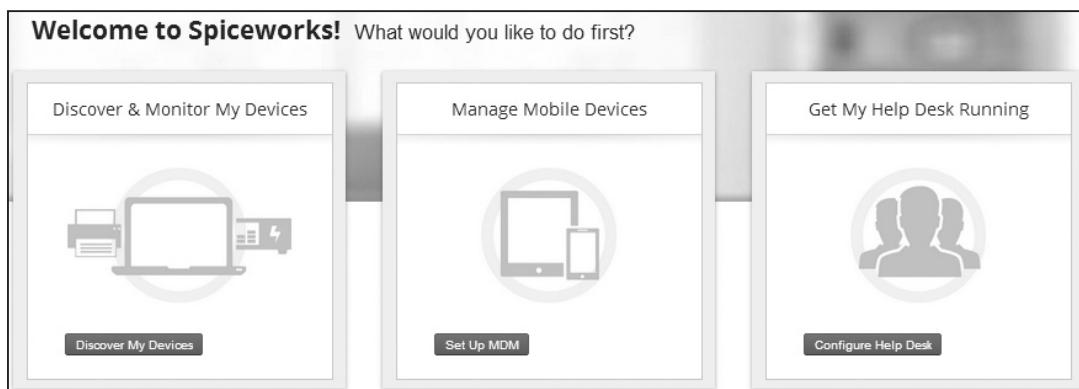
**Figura 6.23**  
Criação de conta no SpiceWorks.

**Saiba mais**

Para gerenciar as estações Windows, o SpiceWorks utiliza o Windows Management Instrumentation (WMI). Para habilitá-lo, é necessário configurar o acesso, que não será abordado neste material. Para mais informações, acesse a comunidade do SpiceWorks: <http://community.spiceworks.com>

- Selecione Discovery My devices para descobrir os dispositivos da rede local, como mostrado na figura 6.24.

**Figura 6.24**  
Opções iniciais do SpiceWorks.



- Aparecerão todos os dispositivos encontrados da rede. Clique em "Enter Credentials", informe a comunidade SNMP "rnp-esr" e clique em "Start Scan", conforme a figura:

**Figura 6.25**  
Configuração das credenciais do discovery.

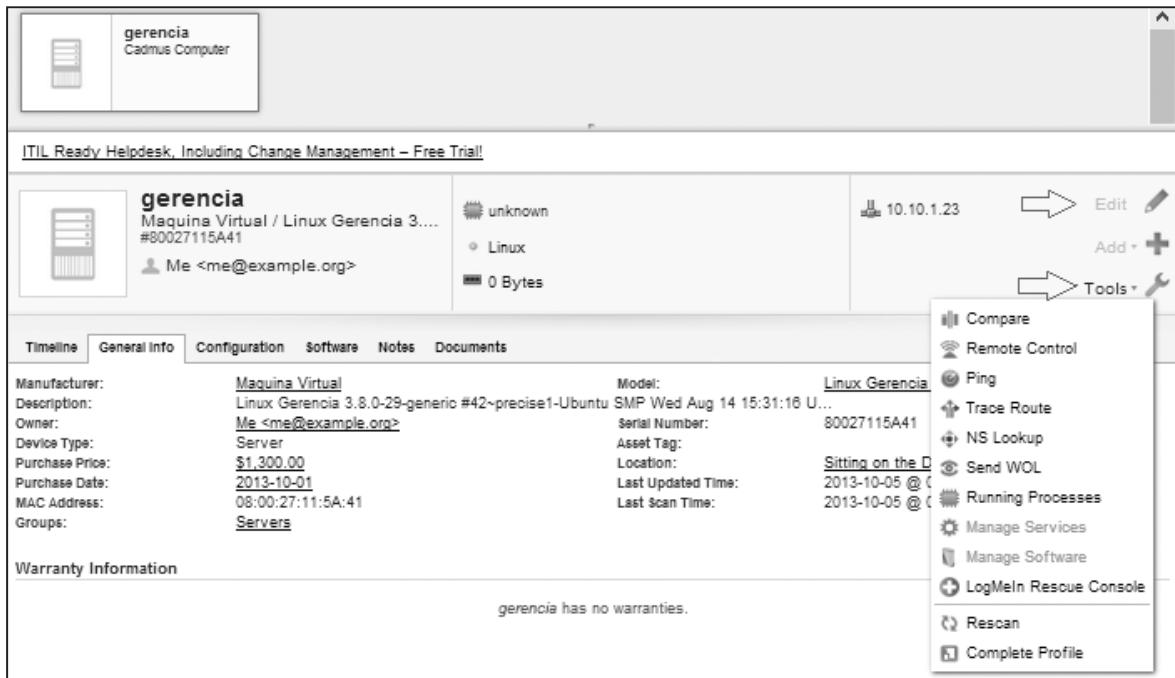
Discovery      Needs Credentials      Inventoried      Profit!

Windows WMI Credentials  
Using a domain admin account will give you the best results.  
Username:   
Password:

Unix and Mac (2) SSH Credentials  
 I Don't Have Any  
 Specify Credentials  
Username:   
Password:

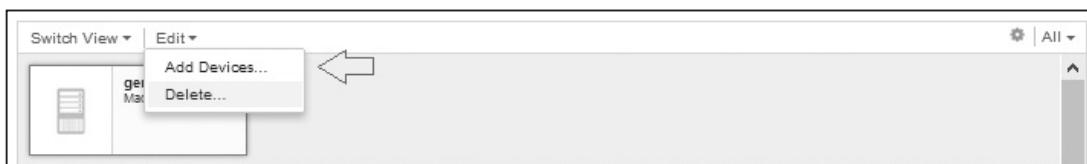
Printers and Switches SNMP Credentials  
 Use "public"  
 Specify Community String  
SNMP Community String:    
Description:

- Terminado o processo, será apresentado os dispositivos reconhecidos. Você pode repetir o processo acessando o menu “Inventory” > “Scan”;
- Para visualizar os dispositivos reconhecidos, acesse “Inventory” > “Devices”. Sobre o dispositivo escolhido, você pode editar alguma configuração/informação ou realizar procedimentos de gerenciamento através da opção “Tools”, conforme as setas existentes na figura 6.26.



**Figura 6.26**  
Visualização, edição e gerenciamento de dispositivos.

- Na parte superior da própria interface de “Inventory” > “Devices”, você pode acrescentar um novo dispositivo, acessando “Edit” > “Add Devices”, conforme a figura 6.27.

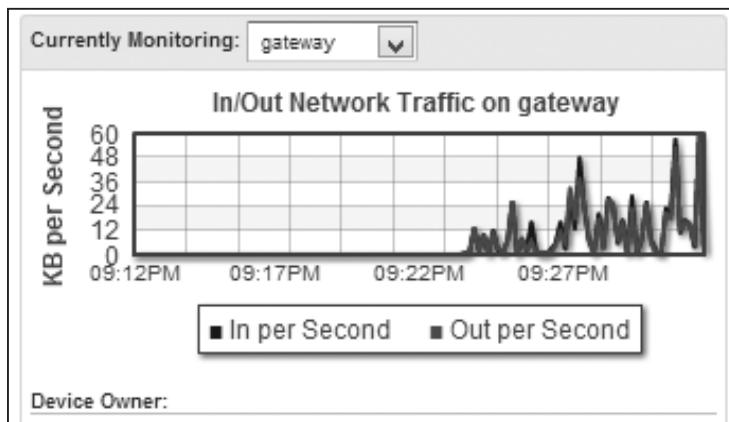


**Figura 6.27**  
Adição de Dispositivos.

- É possível, através de plugins, alterar a linguagem do software, acessando “Inventory” > “Setting”.
- Nesta interface, acesse “Advanced & International Options” para alterar a linguagem, e configurações regionais.

**Figura 6.28**  
Configurações Regionais.

- Você pode adicionar plugins através de “Inventory” > “Tools”, como por exemplo o “Bandwidth Monitor”, que inclui do DashBoard informações de tráfego gerado por dispositivos, conforme a figura 6.29.



**Figura 6.29**  
Plugin Bandwidth Monitor.

## NTOPng

O Network Traffic Probe: New Generation (NTOPng) atua como um analisador de rede, gerando estatísticas em tempo real do tráfego recebido pela interface de rede monitorada.

Possibilita geolocalização dos hosts.

- Mostrar distribuição de tráfego IP entre os vários protocolos;
- Identifica os fluxos de rede;
- Pode funcionar como um analisador de fluxos exportados por roteadores, como Cisco e Juniper;
- Produz estatísticas do tráfego da rede através da interface gráfica utilizando Ajax/HTML5;
- Armazena as estatísticas de tráfego de rede em formato Round Robin Database (RRD).

### Instalação do NTOPng

- Como pré-requisito para a instalação do NTOPng, execute os comandos a seguir:

```
apt-get install subversion rrdtool libxml2-dev libglib2.0-dev
libpcap-dev libgeoip* redis-server autoconf libsqlite3-dev libtool
```

- Execute a instalação do NTOPng:

```
svn co https://svn.ntop.org/svn/ntop/trunk/ntopng/
cd ntopng
./autogen.sh
./configure
make geoip
make
make install
```

Para executar o ntop, passe como parâmetro a interface de rede que deverá se monitorada.

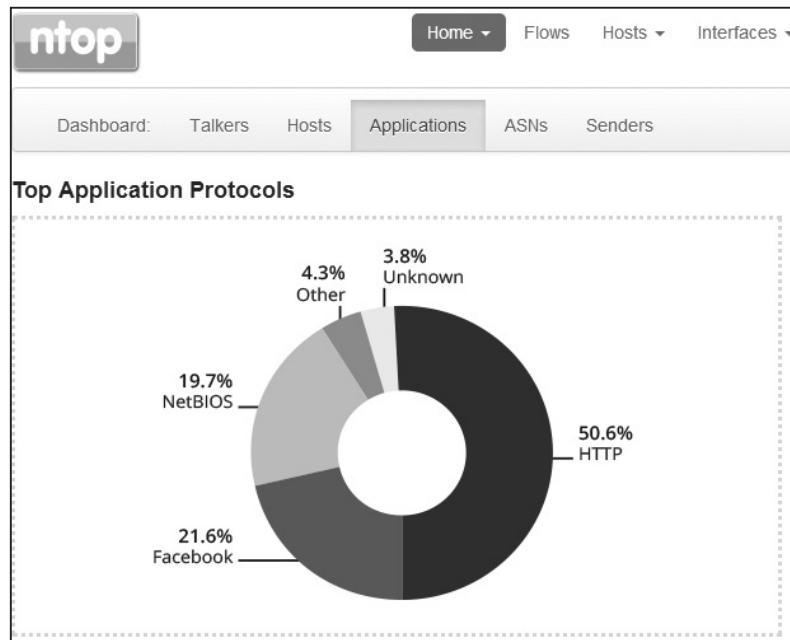
```
Exemplo: ntopng -i eth0
```

Com esse comando, será habilitada a captura dos fluxos na interface de rede e disponibilizado o acesso através da porta 3000, que poderá ser acessada pelo browser, como demonstrado na figura 6.30.

Exemplo: `http://127.0.0.1:3000`

Usuário: admin

Senha: admin



**Figura 6.30**  
Dashboard  
do Ntop.

- Através da interface do Ntop, podemos ter acesso de maneira simples às informações de fluxos, como aplicações e hosts como maior tráfego ou verificar os fluxos ativos, conforme mostrado na figura:

Active Flows								
Info	Application	L4 Proto	Client	Server	Duration	Breakdown	Throughput	Total Bytes
Info	NetBIOS	UDP	10.10.1.17:137	10.10.1.255:137	12 min, 21 sec	Client	0 bps	60.55 KB
Info	FaceBook	TCP	10.10.1.21:50315	edge-star-shv-03-gru...:443	10 min, 22 sec	Client Server	0 bps	53.5 KB
Info	HTTP	TCP	10.10.1.21:51453	economia.terra.com.br:80	1 sec	Client		39.16 KB
Info	HTTP	TCP	10.10.1.21:49946	idrops.terra.com.br:80	11 sec	Client Server	0 bps	7.18 KB
Info	DropBox	UDP	10.10.1.17:17500	10.10.1.255:17500	12 min, 4 sec	Client	0 bps	4.54 KB
Info	HTTP	TCP	10.10.1.21:37283	comments.wsv.terra.c...:80	1 sec	Client Server	0 bps	2.12 KB
Info	HTTP	TCP	10.10.1.21:52359	tr2.terra.com:80	32 sec	Client Server	0 bps	1.63 KB
Info	Unknown	UDP	10.10.1.1:520	10.10.1.255:520	11 min, 59 sec	Client	0 bps	1.61 KB
Info	MDNS	UDP	fe80::a00:27ff:feae::5353	ff02::fb:5353	1 min, 20 sec	Client	0 bps	1.54 KB
Info	HTTP	TCP	10.10.1.17:9057	10.10.1.21:3000	1 sec	Client Server	0 bps	1.35 KB

Showing 1 to 10 of 127 rows

← First | Prev | 1 | 2 | 3 | 4 | 5 | Next | Last →

**Figura 6.31**

Fluxos ativos  
do Ntop.





# 7

## Tratamento de registros de ocorrências (logs) e fluxos de dados

objetivos

Conhecer o tratamento de Logs; Aprender sobre monitoramento de fluxos.

conceitos

Tratamento de Logs; Monitoramento de fluxos.

### Importância do registro de ocorrências

Tudo deve ser registrado.

- Para comprovar uma falha no sistema.
- Debug e Troubleshooting.
- Como evidência de um crime.

Todos os equipamentos conectados à rede possuem funcionalidades que permitem registrar problemas e informações relevantes sobre a operação do dia a dia. Essas informações são imprescindíveis para detectar mau funcionamento ou esgotamento de recursos do equipamento na execução das suas atividades. Falhas de equipamentos, erros de configuração ou até mesmo atividades ilícitas possuem indícios que têm suas atividades registradas pelos equipamentos nos mais diversos níveis (debug, erros, alertas etc.) na forma de trilhas para a auditoria (LOGs).

Os LOGs gerados pelos diversos equipamentos possuem vida útil bastante variada, que pode ser desde alguns minutos – para comprovar um erro de configuração durante algum teste, ou de vários anos – para o caso de algum processo judicial em andamento. Nesse último caso, vários cuidados precisam ser tomados visando a sua preservação ao longo dos anos.

- O registro de LOGs deve ser constante;
- A inspeção dos registros deve ser rotineira.

Um ponto importante é que tanto a geração quanto a inspeção desses LOGs sejam realizadas sistematicamente visando manter registros sobre o mau funcionamento ou mal uso dos equipamentos e sistemas. O registro de atividades de Log é considerado uma forma complementar de gerência da rede, e cobre os casos onde não existe a previsão ou a configuração de Traps SNMP ou outros objetos gerenciados pelo fabricante para aquela falha específica. Esse é um caso comum em falhas no software básico dos equipamentos.



Um desses casos pode ser visualizado a seguir, onde é registrado nos logs de um roteador a falta de memória para que esse realize as suas funções.

```
2013-04-11T14:58:31-03:00 c12000 94211: C12000 RP/0/4/CPU0:Apr 11 14:58:29.886 :  
wdsysmon[429]: %HA-HA_WD-6-IN_DEPENDENCY_LIST : Process devb-eide-prp2 pid 40991 is  
present in Dependency List and will not be killed.  
  
2013-04-11T14:58:31-03:00 c12000 94212: C12000 RP/0/4/CPU0:Apr 11 14:58:29.886  
: wdsysmon[429]: %HA-HA_WD-4-MEMORY_STATE_CHANGE : New memory state:  
Severe  
  
2013-04-11T14:58:42-03:00 c12000 94230: C12000 RP/0/4/CPU0:Apr 11 14:58:42.722  
: l2fib[291]: %OS-SHWIN-2-ERROR_ENCOUNTERED : SHWIN: Error encountered:  
System memory state is severe, please check the availability of the system memory  
  
2013-04-11T14:58:58-03:00 c12000 94231: C12000 RP/0/4/CPU0:Apr 11 14:58:58.171 :  
fib_mgr[215]: %OS-SHWIN-2-ERROR_ENCOUNTERED : SHWIN: Error encountered:  
System memory state is severe, please check the availability of the system memory
```

**Figura 7.1**  
Exemplo de LOGs.

Complementarmente, quando se trata de um crime em investigação pelo sistema judiciário, é importante que tudo o que for registrado venha sendo realizado de forma periódica e constante, eliminando qualquer dúvida sobre a possibilidade de perseguição contra o indivíduo em questão, algo que pode ser questionado no caso de provas pontualmente geradas, iniciadas próximo ao momento do delito.

## Como registrar os LOGs

- LOGs na máquina local não são confiáveis.
- Em uma invasão.
  - Podem ser removidos ou alterados.
  - O sistema de LOGs pode ser desabilitado.
- O registro remoto é necessário.
  - O conceito de LOGHOST.



Muitos equipamentos, como roteadores e switches, em razão do custo da sua memória de armazenamento, não mantêm seus registros de auditoria por muito tempo. Em geral, estes utilizam uma pequena fração de sua memória volátil (RAM) para armazenamento, utilizada como um buffer circular. Nesse caso, as mensagens mais antigas acabam sendo sobrepostas pelas mais atuais. É normal que esses equipamentos mantenham somente os registros de atividades da ordem de dias ou semanas, o que torna-se necessário que esses LOGs sejam enviados para um equipamento remoto com maior capacidade de armazenamento, em geral um servidor.

## O servidor de LOGs (LOGHOST)

- É um "Bastion Host".
- Firewall customizada.
- Kernel mínimo e customizado.
- Os poderes do root são limitados (exemplo: securelevel do FreeBSD/OpenBSD).
  - Alterações de configuração exigem reboot.
  - Sistema de arquivos (partições append-only)



O equipamento que é utilizado para armazenar os logs enviados por outros equipamentos é normalmente conhecido pelo nome de LOGHOST. Nesse modo de operação, é normal que os logs sejam gerados em duplidade: uma cópia no servidor local e outra no Loghost. Essa abordagem é especialmente importante em casos de invasões de hackers em servidores, onde normalmente a primeira atividade do invasor é remover e adulterar os registros de auditoria da máquina invadida. O uso de um servidor de LOGs externo permite que os registros de auditoria sejam mantidos a salvo no servidor remoto. Esse tipo de atividade é fortemente recomendado e é reconhecida como parte necessária à reconstrução da linha do tempo dos eventos ocorridos no caso de uma auditoria na rede ou sistemas.

Pelo tipo e importância dos dados mantidos pelo servidor de Logs, o servidor que será escondido como Loghost deve ser preparado para suportar sozinho um ataque, independente de outras linhas de defesa que possam existir na rede (exemplo: firewall, IPS etc.). Ele deve ser construído obedecendo às mesmas diretivas da construção de um “bastion host”, ou seja, deve ser dedicado para a atividade de Logs, suportando somente aplicativos diretamente relacionados a essas atividades e estar preparado para ataques da mesma forma que uma máquina totalmente exposta à internet. Deve possuir seu próprio firewall (exemplo: iptables, ipfilter etc.) e não rodar nenhum serviço ou protocolo desnecessário. Também é aconselhado que se faça uso de outros dispositivos de segurança, como um kernel customizado e proteção adicional ao sistema de arquivos, onde estarão os arquivos de LOGs. Outro ponto importante é a restrição no acesso remoto a esse servidor e as pessoas que eventualmente têm um login para acesso.



Para saber mais sobre o SELinux: <http://selinuxproject.org>

Controles adicionais de segurança, conhecidos como MLS (Multi-Level Security), são implementados em diversos Sistemas Operacionais, como módulos do próprio Sistema Operacional, ou através da customização direta do kernel. No Linux, uma boa opção é o uso do SELinux, enquanto nos kernels BSD-Like uma boa opção é a ativação do mecanismo “securelevel” no momento do boot do sistema, associado com o uso de flags atribuídos aos arquivos e partição de logs (comando chflags do BSD). Entre esses flags, os mais úteis para a criação de um servidor de LOGs são:

- ▣ **sappnd, sappend:** somente é permitido o append nos arquivos – não é possível fazer edição;
- ▣ **uunlink, unlink:** não é possível deletar arquivos;
- ▣ **uchg, uchange, immutable:** não é possível reverter os flags atribuídos aos arquivos ou filesystems.

Esses mecanismos de segurança permitem controle mais granular sobre o acesso de escrita, leitura e acesso ao sistema de arquivos, além de diminuir os poderes do superusuário do sistema (root). Isso impede o próprio root de alterar regras de firewall ou remover e editar os arquivos de Logs. Mais informações podem ser obtidas em <http://www.manpages.info/freebsd/securelevel.8.html>.

Outro ponto relevante é que o servidor de LOGs faça uso de um GPS ou que minimamente sincronize seu horário via o protocolo Network Time Protocol (NTP), e que ele mesmo insira seu registro de tempo para cada uma das mensagens recebidas de outros servidores, mesmo que esse timestamp seja colocado em duplicidade nos Logs.

Em empresas maiores, o Network Operation Center (NOC) divide as tarefas de segurança com o Security Operations Center (SOC), e normalmente a administração e uso das informações do servidor de LOGs são compartilhados entre essas equipes, dada a natureza das informações armazenadas e a necessidade de seu uso no dia a dia. Nesse ponto, a administração do LOGHOST fica a cargo do SOC, sendo o NOC um usuário dessas informações.



Dessa forma, cria-se uma estratégia de vigilância cruzada, onde o reinício do servidor de LOGs (indício de uma possível violação) ficará registrado na estação de gerência (NMS) administrada pelo NOC, e as alterações nos equipamentos administrados pelo NOC e equipes de sistemas ficarão registrados no servidor de Logs (LOGHOST) administrado pelo SOC. Complementarmente a essas implementações de segurança, é relevante que o acesso físico ao servidor de LOGs seja controlado. Em ambientes mais críticos, o servidor de Logs é tratado de forma semelhante a uma unidade certificadora (acesso físico restrito e sem login remoto).

## LOGHOST

- É exclusivo para o serviço de LOGS.
  - Não roda outros serviços.
  - O acesso remoto é restrito ou inexistente.
- Geralmente é administrado pelo SOC.
  - Diferentes equipes dificultam ocultação de vestígios.
  - O NOC é usuário desse serviço.
- O horário de Timezone corretos é vital (NTP).
- Centraliza logs de switches, roteadores, servidores e aplicações.



O LOGHOST tem papel vital na operação da rede, mantendo uma cópia confiável e centralizada do registro de todas as operações realizadas nos diversos níveis, desde o Sistema Operacional da máquina até a camada de aplicação e controles de acesso e autenticação de usuários:

- Sistema Operacional (Windows, unix-like);
- Switches e roteadores (login e log de comandos);
- Controle de acesso à rede e servidores (Radius, Diameter, Ldap);
- Acesso a serviços (Xen, Vmware, dns, ftp, http etc.);
- Logs dos diversos firewalls.

Essas informações, com horário preciso e centralizado em um mesmo local, são vitais para uma investigação ou análise forense, principalmente se obtidas de um local confiável, no caso de uma violação da rede e sistemas.

## O volume de informações

- Depende do nível de segurança da instituição:
  - Militar.
  - Educacional.
  - Comercial.
- Volume de informações gerado é considerável.
- A análise manual é inviável.
- Uma empresa média gera em torno de 5GB/dia de logs (~100 milhões de eventos distintos).



A quantidade de logs gerada por uma instituição depende muito de quais atividades serão elegíveis para armazenamento e por quanto tempo se pretende guardá-las. O Marco Civil da Internet (Lei nº 12.965, de 23 de abril de 2014) exige que provedores de acesso mantenham logs de acesso de seus clientes por 12 meses. Outras recomendações, como a da Anatel (Resolução nº 614, de 28 de maio de 2013), indica que as empresas de Serviço de Comunicação Multimídia (SCM) devam manter informações dos registros de conexão e dados dos



usuários pelo prazo mínimo de um ano (Artigo 53). Já o CGI-BR recomenda que as informações de acesso à rede sejam mantidas pelo prazo mínimo de três anos (<http://www.cgi.br/publicacoes/documentacao/desenvolvimento.htm>).

O Marco Civil (Lei nº 12.965, de 23 de abril de 2014) especifica:

- “Art. 13. Na provisão de conexão à internet, cabe ao administrador de sistema autônomo respectivo o dever de manter os registros de conexão, sob sigilo, em ambiente controlado e de segurança, pelo prazo de 1 (um) ano, nos termos do regulamento.”
- “Art. 15. O provedor de aplicações de internet constituído na forma de pessoa jurídica e que exerce essa atividade de forma organizada, profissionalmente e com fins econômicos deverá manter os respectivos registros de acesso a aplicações de internet, sob sigilo, em ambiente controlado e de segurança, pelo prazo de 6 (seis) meses, nos termos do regulamento.”

Considera-se que esses logs podem posteriormente ficar armazenados de forma off-line, ou seja, gravados em uma mídia externa (CD-ROM, DVD-ROM) e guardados em local seguro.

Para ter-se ideia do volume das informações geradas, uma empresa média gera em torno de 5GB/dia de logs (~100 milhões de eventos distintos), considerando-se logs de firewall, e-mail, acesso a sites etc. Entretanto, essas informações são altamente compactáveis, já que se tratam de informações textuais. Em ambientes com requisitos maiores de segurança, esse volume pode ser multiplicado várias vezes, podendo ser registrados inclusive as informações de conexões e acessos realizados por cada um dos elementos da rede (exemplo: log de flows, proxies e NAT).

## A difícil tarefa de interpretar as mensagens de LOGs

Ter e manter um servidor de Logs centralizado não é uma tarefa fácil, e como qualquer serviço exige cuidado constante para verificar se todos os novos servidores e equipamentos de rede estão sendo corretamente configurados para fazê-lo. Entretanto, existe um agravante: realizar pesquisas nesses logs não é uma tarefa simples, dado que os logs gerados pelos diversos equipamentos e serviços não seguem nenhum padrão. Em geral, as mensagens são definidas pelos próprios programadores segundo algumas regras básicas da própria empresa em que trabalham. Alguns exemplos de logs de equipamentos:

### Exemplo de um roteador Cisco

```
Sep 18 03:28:55.101 UTC: %LINK-3-UPDOWN: Interface FastEthernet1/0/0,  
changed state to down  
  
Sep 18 03:29:40.123 UTC: %SEC-6-IPACCESSLOGP: list 112 denied tcp  
192.168.0.2(6012) -> 192.168.0.4(23), 1 packet  
  
Sep 18 03:42:50.212 UTC: %FAN-3-FAN_FAILED: Fans had a rotation  
error reported
```

### Exemplo de um roteador Juniper

```
Sep 18 03:36:16 mib2d[152]: SNMP_TRAP_LINK_DOWN: ifIndex 79,  
ifAdminStatus down(2),ifOperStatus down(2), ifName ge-3/0/0  
  
Sep 18 03:40:23 mgd[4334]: UI_COMMIT: User ‘bertholdo’ requested  
‘commit’ operation (comment: none)  
  
Sep 18 03:55:37 Modifying fan speed as high temp is now 53 C
```

### **Exemplo de um switch extreme**

```
Alpine SYST: Error: Disk full or allocation exceeded  
Alpine SYST: Port 1:2 link down  
Alpine USER: admin logged in through telnet (192.168.0.225)  
Alpine TRXDIAG: Current temperature reading [9] is 37C.
```

### **Exemplo de um servidor Linux**

```
Sep 18 16:22:54 intranet sshd[21519]: Accepted password for berthold  
from 192.168.1.228 port 52064 ssh2  
  
Sep 18 16:22:54 intranet sshd[21519]: pam_unix(sshd:session):  
session opened for user berthold by (uid=0)  
  
Sep 18 16:23:27 intranet su[21817]: Successful su for root by  
berthold  
  
Sep 18 16:23:27 intranet su[21817]: + /dev/pts/0 berthold:root  
  
Sep 18 16:23:27 intranet su[21817]: pam_unix(su:session): session  
opened for user root by berthold(uid=10002)  
  
Sep 18 16:25:07 intranet kernel: [195362.552961] IN=eth2 OUT= MA  
C=ff:ff:ff:ff:ff:2c:27:d7:94:55:b5:08:00 SRC=192.168.0.152  
DST=192.168.0.191 LEN=150 TOS=0x00 PREC=0x00 TTL=128 ID=27708  
PROTO=UDP SPT=17500 DPT=17500 LEN=130
```

Essa impossibilidade de padronizar os logs de cada um dos sistemas ou equipamentos, aliado à grande quantidade de informações produzidas, torna inviável a análise manual dos LOGs, salvo em alguma auditoria especializada e procurando por atividades em uma janela de tempo específica – caso comum somente no campo da análise forense. Na atividade corriqueira do NOC/SOC, esses logs serão analisados por ferramentas específicas para análise, como o AWSTAT, LOGWATCH, SARG, OSSEC e Splunk.

## **Definindo uma estratégia de LOGs**

- O syslog server coleta todos os LOGs (514/UDP, 514/TCP).
- Roteadores, switches e servidores em geral implementam o protocolo syslog.
  - RFC 3164 (The BSD syslog Protocol).
  - RFC5424 (The Syslog Protocol), que suprecede a RFC3164.
  - RFC 5426 (Transmission of Syslog Messages over UDP).
  - RFC 6587 (Transmission of Syslog Messages over TCP).
  - RFCs 5425 (Transport Layer Security (TLS) Transport Mapping for Syslog).

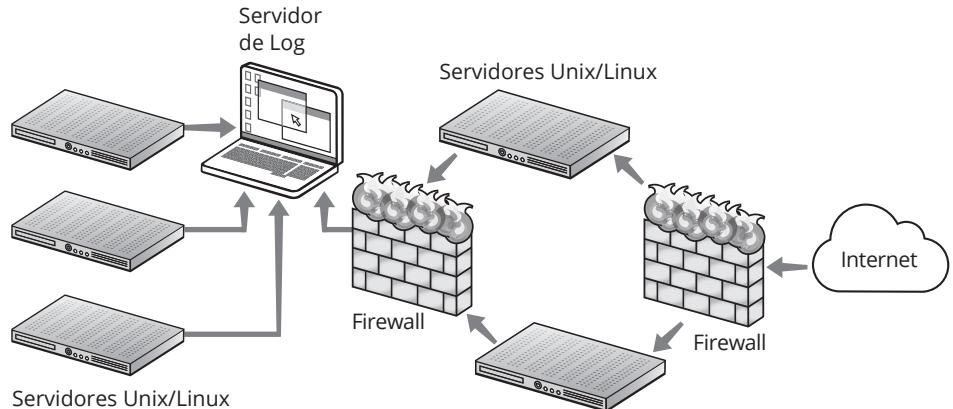


Um dos pontos importantes quando se define a estratégia de logs a ser utilizada na instituição é a verificação da possibilidade de manter toda a informação em um único servidor de forma centralizada. Obviamente, essa estratégia depende de uma boa conexão de dados entre os dispositivos e o servidor de LOGs.

Deve-se observar que o servidor de logs deve minimamente estar situado após a primeira barreira de segurança da instituição, ou seja, o seu firewall.

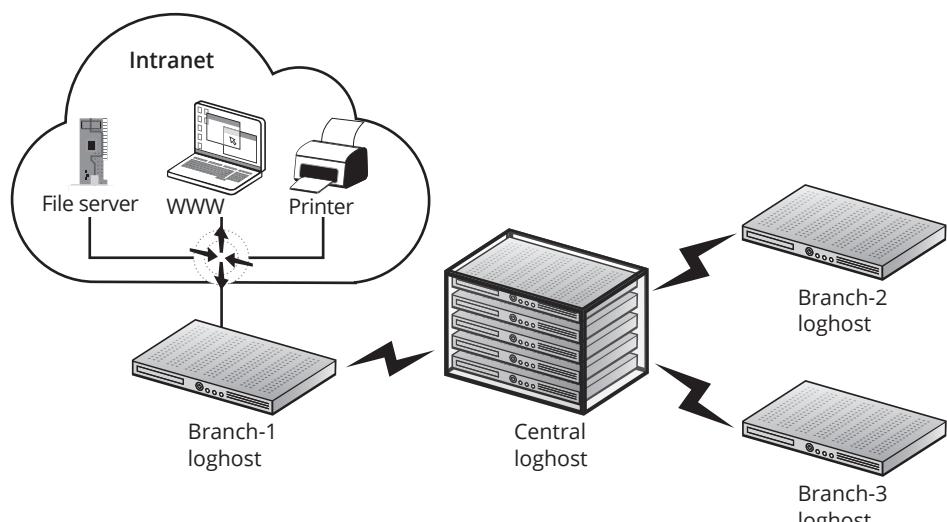


Seguindo a ideia de um conjunto de logs centralizados, assim como o próprio firewall, os servidores externos e o próprio roteador da instituição também devem enviar seus logs para o LOGHOST.



**Figura 7.2**  
Servidor de Logs  
e sua posição  
na rede.

Para os casos onde a rede da instituição é distribuída por uma conexão de baixo desempenho, é possível optar por servidores de logs distribuídos, criando a possibilidade de logs locais para os casos onde o site esteja off-line. Uma boa estratégia nesses casos é providenciar a transferência dos logs rotacionados para o servidor principal em determinados períodos de tempo. Essa estratégia aumenta a complexidade da solução e somente deve ser utilizada em casos especiais.



**Figura 7.3**  
Servidores de logs  
distribuídos.

Normalmente, a solução de LOGs é realizada usando como LOGHOST um equipamento Unix, embora algumas soluções comerciais permitam a uso de servidores Windows. A maioria dos fabricantes desenvolvem seus clientes minimamente obedecendo a RFC 3164 (The BSD syslog Protocol), já existem definições para a RFC5424 (The Syslog Protocol) que suprecedem a RFC3164 e a RFC 5426 (Transmission of Syslog Messages over UDP). Outras melhorias previstas já foram definidas pelas RFC 6587 (Transmission of Syslog Messages over TCP) e RFC 5425 (Transport Layer Security (TLS) Transport Mapping for Syslog).

- Gerência In-band x Gerência Out-Of-Band.
- Logs preferencialmente utilizam gerência ou-of-band.

Como dito, o mais comumente usado é mesmo o transporte utilizando a porta 514/UDP, e nesse caso existe sempre a possibilidade de ataques de negação de serviço e ou de falsificação de mensagens para o servidor de LOGs. Outro problema com essa abordagem é a perda eventual de mensagens, já que o protocolo UDP não possui garantia de entrega.

Um dos contornos possíveis para esses problemas é o uso da rede de gerência out-of-band (OOB) para transporte também dos logs de equipamentos e servidores, enquanto as novas soluções ainda não são difundidas. A gerência OOB é caracterizada por cada equipamento possuir uma interface física distinta para a gerência da rede – nessa interface não há tráfego de produção, somente tráfego de controle (gerência SNMP, Backup, acesso SSH para configuração etc.). Outra característica que existe na maioria das implementações de gerência OOB é que não existe roteamento, todo o acesso é realizado em camada 2.

Servidores Unix-like utilizam:

- Rsyslog, syslogd e syslog-ng

Servidores Windows utilizam o Event Log:

- Necessitam de software adicional
- Winsyslog, syslog-win32, ...



Nos servidores Unix, existe uma diversidade de pacotes capazes de manipular as mensagens do syslog, sendo os mais comuns o syslogd, rsyslogd e o syslog-ng. Este último ainda possui uma versão open source (OSE) e uma segunda versão comercial chamada de premium (PE). A versão syslog-ng é preferida por já implementar suporte TCP (RFC 5424) e outras facilidades de filtragem e suporte a bancos de dados.

Os servidores Windows por sua vez utilizam por default o Windows Event Log Service e, no caso de uma rede mista, não existe uma solução nativa da Microsoft para usar o protocolo syslog. Nesse caso podem ser utilizados aplicativos freeware, como o Winsyslog (<http://www.winsyslog.com>) e o syslog-win32 (<http://sourceforge.net/projects/syslog-win32/>).

## O protocolo Syslog



- Tipos de eventos (facility).
- A severidade dos eventos.

Apesar das inúmeras variações das RFCs mas atuais, a base do protocolo syslog ainda é formada por um código básico que duas informações básicas: o tipo de evento – conhecido como “facility” e a severidade do evento que está sendo informado (fonte RFC5424):

Código	Facilidade	Descrição
0	Kern	Mensagens do Kernel
1	User	Mensagens de nível do usuário
2	Mail	Sistema de e-mail
3	Daemon	Serviços do Sistema Operacional (daemons)
4	Auth	Mensagens de segurança/autorização/autenticação
5	Syslog	Mensagens geradas pelo próprio syslogd
6	Lpr	Subsistema de impressão do Sistema Operacional



Código	Facilidade	Descrição
7	News	Subsistema de News
8	Uucp	Subsistema UUCP
9	Cron	Mensagens do serviço de clock/cron do sistema
10	Authpriv	Mensagens de segurança/autorização/autenticação
11	ftp	Serviço de transferência de arquivo
12	ntp	Subsistema NTP
13	audit	Logs do sistema de auditoria
14	console	Mensagens da console do sistema
15	Cron2	Mensagens do serviço de clock/cron do sistema
16	Local0	Uso local pelos aplicativos/uso livre
17	Local1	Uso local pelos aplicativos/uso livre
18	Local2	Uso local pelos aplicativos/uso livre
19	Local3	Uso local pelos aplicativos/uso livre
20	Local4	Uso local pelos aplicativos/uso livre
21	Local5	Uso local pelos aplicativos/uso livre
22	Local6	Uso local pelos aplicativos/uso livre
23	Local7	Uso local pelos aplicativos/uso livre

**Tabela 7.1**  
Tabela de  
Facilidades dos  
protocolos syslog.

Quanto ao nível de prioridade, o protocolo define um nível de severidade do erro para cada um dos subsistemas, segundo o impacto e a necessidade de intervenção do administrador no equipamento ou subsistema em questão. São eles:

Código	Prioridade	Descrição
0	Emerg	O sistema está inutilizado. Usualmente conhecido como “panic condition”. Necessita de Intervenção imediata do administrador.
1	Alert	O sistema ainda opera, mas necessita de intervenção imediata.
2	Crit	Condição crítica do sistema, geralmente falha em um sistema secundário que, se não tratado, pode afetar o serviço.
3	Error	Condição de erro do subsistema em questão.
4	Warning	Mensagem de exceção que indica que o sistema precisa de acompanhamento
5	Notice	Mensagem considerada importante sobre a operação do subsistema.
6	Informational	Mensagem de informação sobre o processamento do subsistema. Não requer ação do administrador.
7	Debug	Mensagens de debug do sistema.

**Tabela 7.2**  
Tabela de  
Severidade dos  
protocolos syslog.



Segue o exemplo de um tcpdump de uma mensagem do sistema syslog, formada pelo nome do programa que envia a requisição (proftpd), a facilidade utilizada (daemon) e a prioridade ou severidade da mensagem (6=informativa).

```
192.168.1.80.514 > 192.168.1.79.514: SYSLOG, length: 125
    Facility daemon (3), Severity info (6)
    Msg: Sep 23 12:20:59 proftpd[46143]: 192.168.1.80 (192.168.75.186
[192.168.75.186]): Preparing to chroot to directory '/mirror'
```

## Configurando diversos clientes syslog

- Configurando um roteador cisco.
- Configurando um roteador juniper.
- Configurando um cliente Linux (rsyslog).
- Configurando um cliente Windows (winsyslog).



As configurações dos diversos clientes de syslog seguem a mesma linha básica, sendo necessário configurar os parâmetros utilizados pelo protocolo:

- O endereço IP/IPv6 do servidor de logs remoto;
- A "facility" a ser utilizada para identificar aquele equipamento;
- A prioridade ou criticidade mínima a ser enviada para o servidor.

No caso específico da prioridade, todas as mensagens de mais alta prioridade são sempre enviadas, ou seja, se o administrador definiu que devam ser enviadas mensagens de prioridade equivalente a 3(error), serão enviadas para o servidor as mensagens 0(emerg), 1(crit), 2(alert) e 3(error).

Em equipamentos Cisco (IOS), a configuração mínima é somente o nome ou endereço IP/IPv6 do servidor de logs. Mas normalmente é recomendada alguma configuração adicional relativa ao servidor de tempo, o formato das mensagens a ser enviado, o endereço da interface que identificará o equipamento, o código do serviço e a prioridade a ser enviada para o loghost. Um exemplo dessa configuração é o que segue, considere que o endereço 192.168.1.79 é o IP do Logserver.

```
ntp server 192.168.1.1
service timestamps log datetime localtime
logging 192.168.1.79
logging facility local2
logging source-interface Loopback0
```

Em equipamentos Juniper, a configuração estabelece parâmetros similares:

```
server 192.168.1.1 version 3 prefer;
set system syslog host 192.168.1.79 authorization any
set system syslog host 192.168.1.79 daemon any
set system syslog host 192.168.1.79 security any
set system syslog host 192.168.1.79 change-log any
```

```

set system syslog host 192.168.1.79 facility-override local2
set system syslog host 192.168.1.79 source-address 192.168.1.2

```

Em sistemas unix-like, basta acrescentar a linha a seguir no final do arquivo de configuração do syslogd. Geralmente o arquivo `/etc/syslogd.conf`(\*BSD) ou `/etc/rsyslog.conf`(Linux).

*.*	@192.168.1.79
-----	---------------

Essa configuração enviará uma cópia de todos os logs gerados pelo servidor para o loghost definido para a rede (@loghost).

Em máquinas Windows, como dito anteriormente, é necessário realizar a configuração de um software adicional. A tela a seguir refere-se à configuração do servidor de logs utilizando o software freeware Winsyslog (<http://www.winsyslog.com>). Outra alternativa para os servidores Windows é o eventlog-to-syslog (<http://code.google.com/p/eventlog-to-syslog/>).

Para instalá-lo, execute como administrador do sistema os seguintes comandos:

```

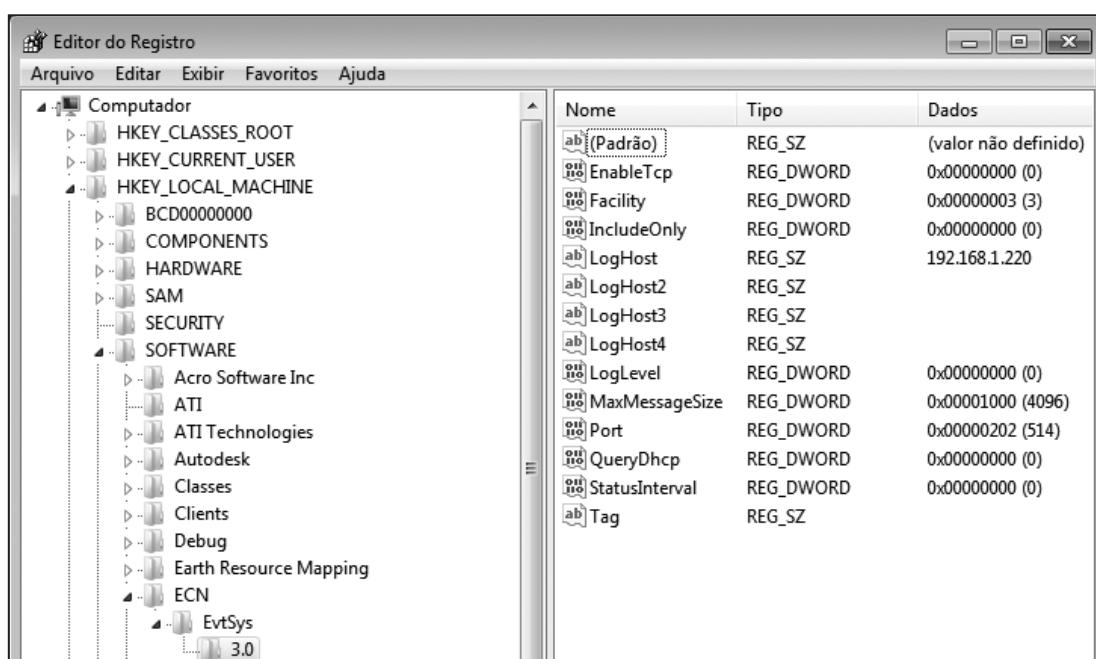
c:\> evtsys.exe -i -l 0 -t WINDOWS -h endereço.ip.do.logserver
c:\> copy evtsys.exe c:\Windows\System32

```

A instalação do software fará a configuração de alguns registros do sistema. Esses registros são instalados em `HKLM\SOFTWARE\ECN\EvtSys\3.0`:

- Facility (DWORD)Default: 3
- IncludeOnly (DWORD)Default: 0
- LogHost (String) Default: N/A
- LogHost2 (String) Default: <empty>
- LogLevel (DWORD)Default: 0
- Port (DWORD)Default: 514
- QueryDhcp (DWORD)Default: 0
- StatusInterval (DWORD)Default: 0

**Figura 7.4**  
Registro EvtSys  
no Windows.



Uma vez resolvidos os problemas relativos à segurança da máquina servidora de LOGs e configurado individualmente cada um dos clientes (roteadores, switches e servidores), o passo seguinte é a instalação e configuração do servidor de LOGs (loghost). O processo de instalação é relativamente simples, embora seja necessário que se faça alguma adequação aos requisitos locais no que trata da forma como serão agrupadas as informações da instituição – geralmente atrelada a permissões para cada uma das equipes de trabalho (exemplo: NOC, Sistemas e Segurança).

## Analisando os LOGs

Análise Manual:

- Busca por causas de problemas detectados (gerência reativa).
- Busca por problemas em andamento ainda não detectados.
- Busca por indícios de novos problemas (gerência proativa).
- Correlacionamentos.



A ideia básica de se manter um LOG completo e centralizado é fundamentalmente a de prover uma fonte rica e confiável que possibilite um olhar no passado com o objetivo de compreender o que exatamente ocorreu em um determinado evento ou incidente.

Uma atividade bastante comum na área de gerência de rede é a busca por mais informações sobre um determinado problema ou identificação de um erro de configuração ou mal funcionamento dos equipamentos. Como visto anteriormente, nesta sessão, é bastante usual que equipamentos com algum tipo de problema deixem algum vestígio registrado. Isso é comum em problemas como falta de memória, problemas de sobreaquecimento, falta de espaço em disco e muitos outros específicos de cada equipamento.

Quando se trata de uma identificação de problema em um equipamento específico (gerência reativa), o sistema de LOGs é uma ferramenta indispensável. Em casos mais extremos, é bastante comum habilitar algum tipo de DEBUG naquele equipamento ou subsistema e realizar uma leitura mais detalhada do problema.

Os diversos fabricantes costumeiramente proveem esse tipo de funcionalidade e solicitam que o cliente realize algum procedimento de DEBUG como primeiro passo da solução do problema, como um debug no processo OSPF de um cisco (debug ospf 1 hello) ou inspecionando o fluxo de informações que passa através do firewall de um roteador Juniper (debug flow basic). Essa mesma análise se aplica a outros processos e serviços, um bastante comum é o serviço de e-mails, onde por exemplo se deseja identificar os problemas no envio/recepção de e-mail para um determinado servidor:

```
# grep -w joe.ufrrgs.br mail.log
2013-09-03T10:13:08-03:00 beta postfix/smtpd[22983]: connect from
joe.ufrrgs.br[143.54.1.200]
2013-09-03T10:13:08-03:00 beta postfix/smtpd[22983]: 25F0718D4A1:
client=joe.ufrrgs.br[143.54.1.200]
2013-09-03T10:13:08-03:00 beta postfix/smtpd[22983]: disconnect from
joe.ufrrgs.br[143.54.1.200]
```



Entretanto, para termos as informações que desejamos, é necessária uma inspeção periódica ou eventualmente pontual nesse banco de informações. Uma forma rápida de realizarmos essa busca é a procura utilizando comando do Unix, tais como *Find*, *grep*, *egrep*, *cut*, *paste*, *sed* e uma boa dose de conhecimento de expressões regulares. Ou então utilizando alguma linguagem com boa capacidade de processamento de textos, como PERL ou PYTHON.

Algumas vezes, essas buscas podem exigir um conhecimento razoável. Por exemplo, observe o comando utilizado para obter todos os IPs que tentaram acessar via ssh um determinado servidor e erraram a senha ou então tentaram logar com um nome de usuário inválido:

```
# cat auth.log | grep ssh | egrep "invalid user|Failed password" |  
grep -o '[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}' | sort -u
```

## Automatizando a análise de LOGs

Análise Automatizada:

- Awstat
- Sarg
- Splunk
- Elsa



Embora a análise manual dos logs seja algo relativamente corriqueiro nas atividades de operação de redes e servidores, e com valia para gerência reativa, o grande volume de dados oculta muitas informações importantes, dificultando a análise proativa e correlação das informações de diversos serviços ou servidores, tornando-se imprescindível um ferramental que realize uma análise automatizada desses dados com o objetivo de identificar principalmente problemas menores que tendem a se manifestar com o passar do tempo.

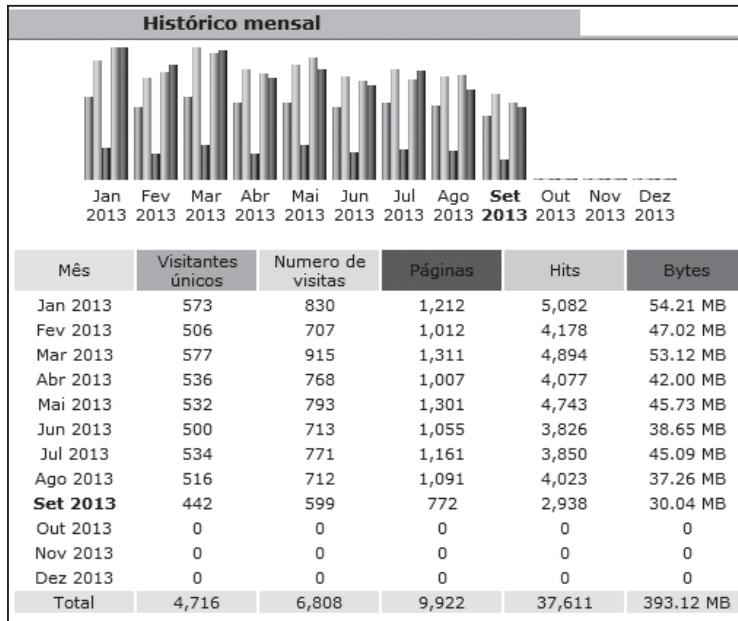
Várias ferramentas existem nesse sentido, sendo algumas delas bastante conhecidas. Infelizmente não existe nesse segmento uma ferramenta totalmente gratuita e completa. O grande problema é a falta de padronização de partes simples das mensagens de log, como por exemplo o próprio identificador de tempo das mensagens, que varia entre inúmeros servidores conhecidos (codificação ISO, expresso em inteiro, ponto flutuante, ...).

Entre o ferramental conhecido existem alguns que podem ser referenciados como opções para os diferentes problemas de análise de logs. Basicamente são softwares com algum tipo de interface web e que necessitam de acesso aos arquivos de log do sistema:

### AWSTAT

Awstats (<http://awstats.sourceforge.net/>) é uma excelente ferramenta para análise de websites, permitindo gerar relatórios baseados em históricos de acesso aos sites. Informações como a origem dos acessos, páginas mais acessadas e navegadores mais utilizados para acesso ao site são visualizadas em forma gráfica e transparente pela ferramenta.

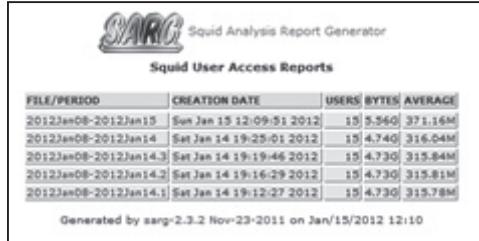




**Figura 7.5**  
Exemplo de tela do AWStats.

## SARG

Squid Analysis Report Generator (SARG), <http://awstats.sourceforge.net/>, é uma ferramenta para análise de logs do Squid (proxy web), permitindo gerar relatórios de uso individual para usuários, além de várias outras informações, como uso de banda e sites mais acessados. Uma opção possível é utilizá-lo para monitorar acessos de usuários em um firewall pfSense.

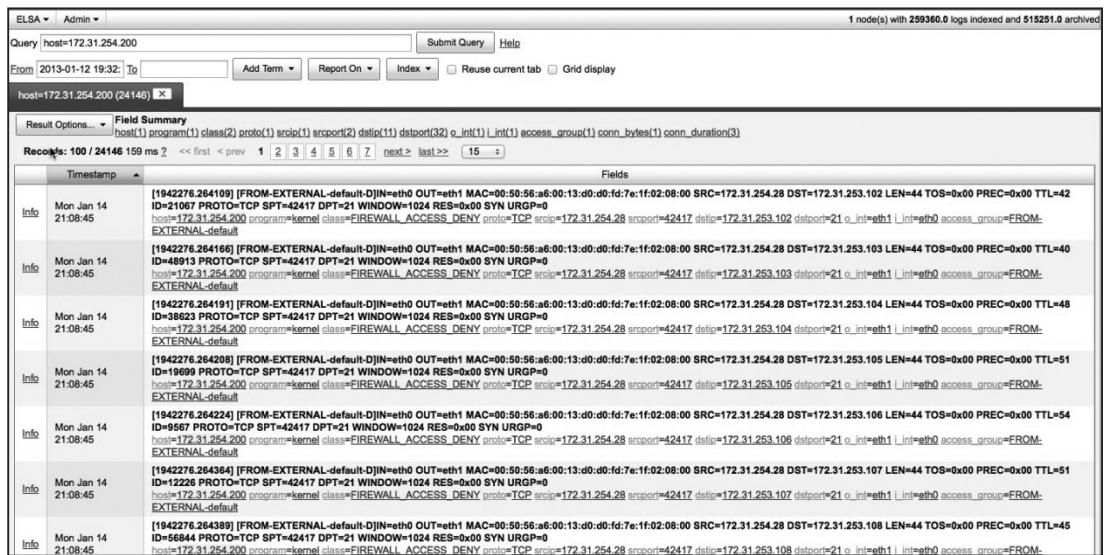


**Figura 7.6**  
Exemplo de tela do SARG.

## ELSA

Enterprise Log and Search Archive (ELSA), <https://code.google.com/p/enterprise-log-search-and-archive/>, é um projeto relativamente novo e tem por meta processar os logs centralizados com o syslog-NG em um LOGHOST. A ideia básica do software é facilitar a vida do administrador, que necessita gerar relatórios e buscar informações nos arquivos de logs sem necessitar de programação script (shell, perl) ou ter de dispensar um tempo considerável pensando em expressões regulares complexas para buscar o que procura. A interface é relativamente simples. Esse software é uma alternativa ao SPLUNK.





**Figura 7.7**

Exemplo de tela  
do ELSA.

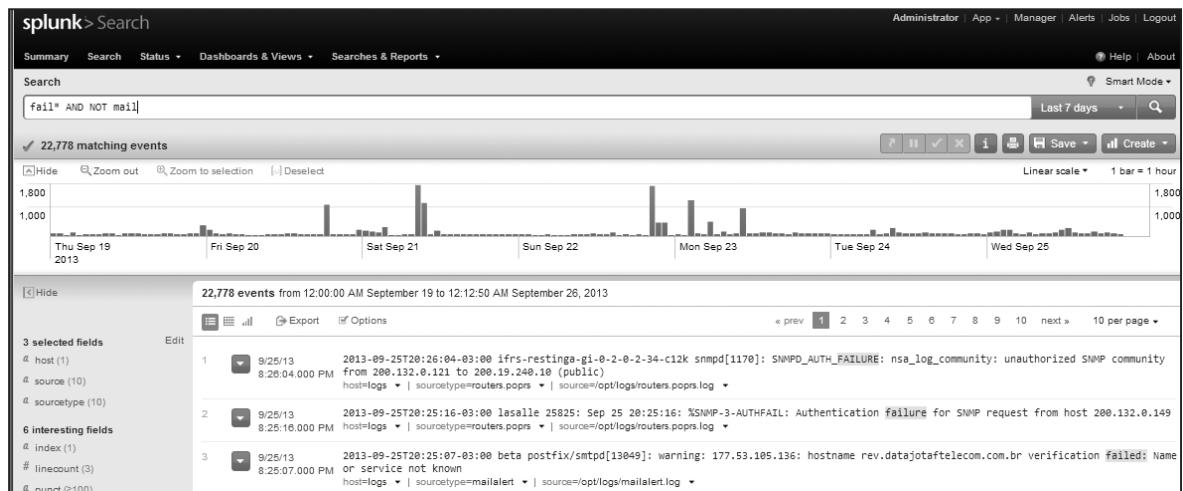
## SPLUNK

SPLUNK, <http://www.splunk.com>, é uma ferramenta comercial que tem por objetivo indexar em um banco de dados todos os arquivos de LOGs de um servidor (exemplo: LOGHOST) e permitir buscas rápidas nessa base de dados, além de permitir gerar alarmes de maneira semelhante a um Log-based Intrusion Detection System (LIDS). Ele possui uma interface relativamente simples, que permite a geração de relatórios e buscas rápidas nos LOGs baseado em operações lógicas simples para o usuário.

Uma funcionalidade interessante do software é a existência de um “Dashboard” que o usuário pode configurar previamente com as consultas e gráficos que julgue mais relevantes, como um gráfico instantânea de todas as mensagens de erro ou warnings que estão sendo locados por todos os servidores naquele momento. O software tem uma integração muito boa da parte gráfica e da pesquisa nos logs, tendo inclusive vários add-ons para interpretação de logs de Cisco, Juniper, Linux, Mysql, Xen, VMWare, NAGIOS e muitos outros. Porém, um grande ponto negativo é que a licença de uso gratuita somente permite a análise diária de até 500 MB.

**Figura 7.8**

Exemplo de tela  
do Splunk.



# Monitoramento de fluxos

## Introdução

Monitoramento de fluxos:

- Traça perfis de tráfego com base em fluxos.
- Coleta em pontos-chave da rede.

Fluxo de rede entre nós finais:

- Sequência unidirecional de pacotes.

Implementações:

- Conjunto de informações e protocolo.

Monitoramento de fluxos: é uma abordagem de monitoramento de rede que traça perfis de tráfego baseado em estatísticas de fluxos coletados de pontos-chave da rede.

Um fluxo é uma sequência unidirecional de pacotes entre dois nós finais. Como exemplo, se há uma comunicação entre os nós A e B, há um fluxo de A para B e outro fluxo de B para A.

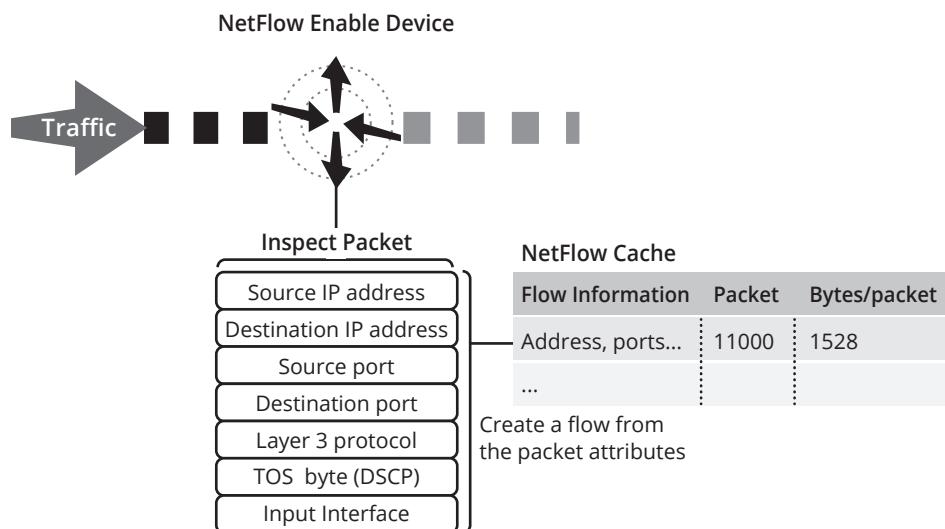


Figura 7.9  
Um fluxo de dados é sempre unidirecional.

As implementações definem um conjunto de informações a serem derivadas dos fluxos e um protocolo que permite a exportação dessas informações para um equipamento de coleta e análise.

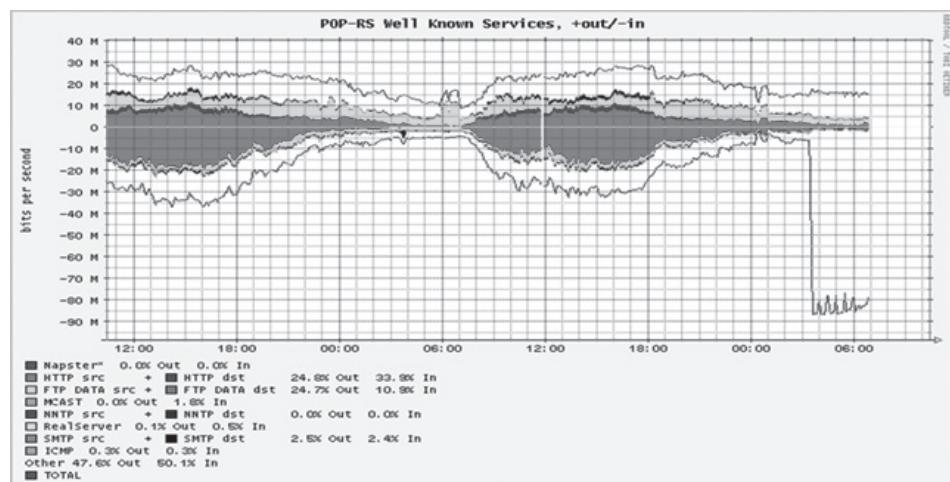
Para que serve?

- Informar o que está acontecendo na rede:
  - Contabilizar o uso da rede.
  - Detectar anomalias e atividade maliciosa.
  - Monitorar as atividades da rede.
  - Facilitar o planejamento da rede.
- Alertar os administradores.
  - Envio de e-mail.



O armazenamento e análise dos fluxos traz inúmeros benefícios, entre eles a identificação facilitada de problemas na rede ocasionados por mau uso (exemplo: DoS ou DDoS).

Com as informações dos cabeçalhos IP/UDP/TCP coletadas a partir dos fluxos, torna-se possível realizar pesquisas sobre esses dados a respeito de um determinado IP, conexão ou volume de tráfego entre determinadas redes. Essa facilidade é de uso corriqueiro na engenharia de tráfego: uma consulta aos fluxos informa quais os sistemas autônomos (ASNs) são as maiores origens ou destinos do tráfego da instituição, ou até mesmo quais provedores de acesso têm a maioria dos clientes que acessam o website da empresa. Dessa forma, pode-se predizer qual é a melhor operadora a ser contratada e qual a dimensão aproximada do circuito de dados que deve ser contratado. Além disso, os flows possibilitam que sejam mostrados em gráficos os seus dados utilizando algumas ferramentas complementares, como NFSen e FlowScan, permitindo facilmente identificar variações na baseline do tráfego.



**Figura 7.10**  
Detectando um  
DoS através  
da análise gráfica  
dos fluxos.

Alguns outros exemplos de uso dos flows podem ser considerados em análises como:

- Identificar o tráfego demasiado alto entre nós não relacionados;
- Planejar e avaliar a adição de tráfego;
- Interfaces de rede sobrecarregadas;
- Tráfego de software malicioso – worm;
- Ataques de negação de serviço de rede;
- Enumeração de máquinas e serviços disponíveis – port scan;
- Envio de e-mail alertando os administradores de problemas.

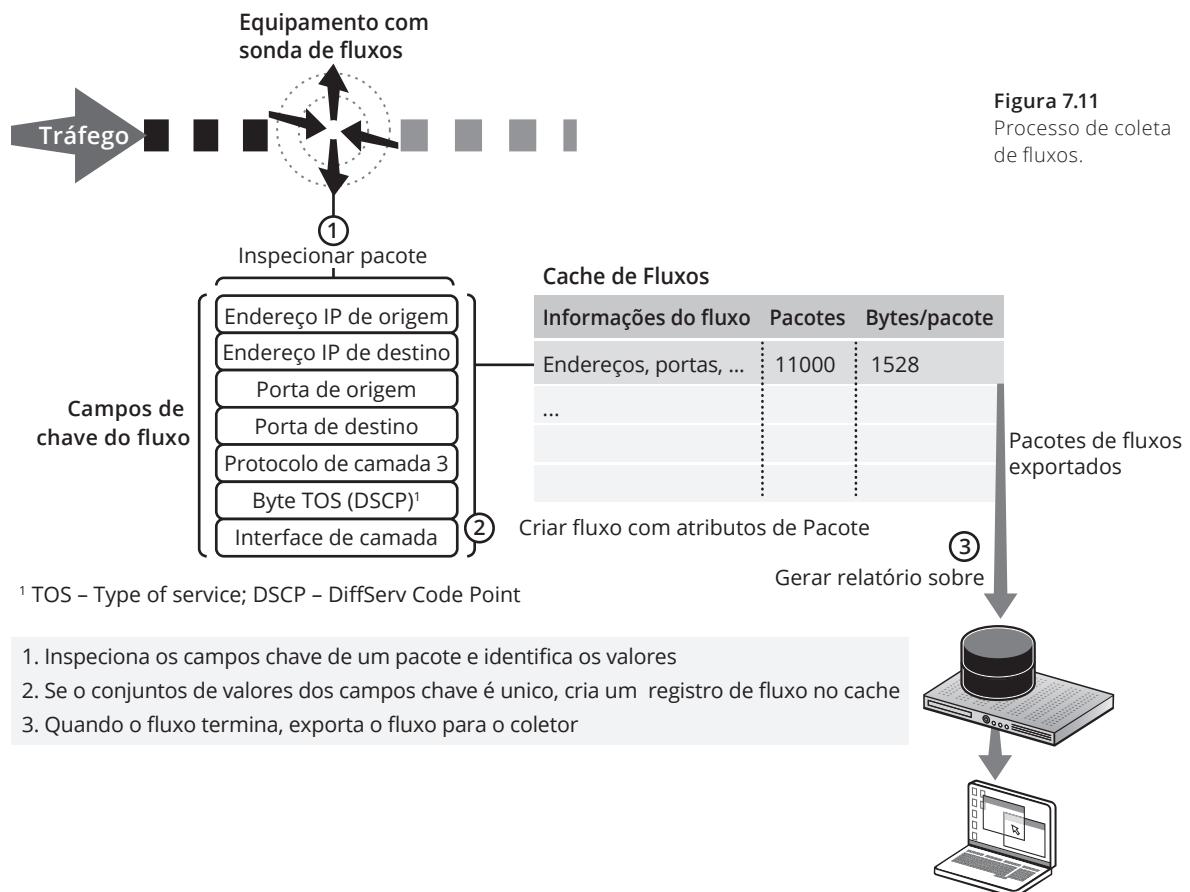
## Como é realizada a coleta dos fluxos

- Sondas são espalhados na rede e identificam os fluxos de dados.
- Os fluxos são:
  - Capturados.
  - Contabilizados (pacotes, bytes).
  - Exportados para um servidor de fluxos.

Para obter as informações sobre os fluxos, é necessário identificá-los univocamente e registrá-los. Essa tarefa fica a cargo de sondas ativadas em equipamentos de rede, tais como: roteadores, switches, gateways etc. As sondas são projetadas para manter os registros em um cache pequeno e volátil. A leitura dos registros do cache permite ter um retrato imediato do que está passando pelo equipamento. No entanto, normalmente as sondas estão instaladas em máquinas que não conseguem armazenar muitos registros, por isso, de tempos em tempos, esses fluxos são exportados para uma máquina coletora, deixando no cache somente o registro dos fluxos mais recentes.

As informações de fluxo são coletadas e guardadas em uma máquina com maior capacidade de armazenamento, permitindo manter um histórico dos fluxos da rede.

- Sonda ativada em equipamento de rede:
  - Identificação de fluxos;
  - Registro de poucos fluxos;
  - Cache volátil para consultas imediatas;
  - Exportação de tempos em tempos.
- Coletor fica em servidor de rede:
  - Captura fluxos exportados;
  - Armazena em mídia não-volátil;
  - O limite é o disco da máquina;
  - Mantém histórico de fluxos coletados.



Cada pacote que passa pela sonda incrementa os contadores dos fluxos existentes ou, no caso uma nova identificação única de fluxo, ele é eleito para criar uma nova entrada na tabela de fluxos. Essa identificação é feita através da inspeção de campos-chaves contidos em um pacote recebido.

Nas primeiras versões das implementações, os campos-chave eram fixos:

- Endereço IP de origem;
- Endereço IP de destino;
- Porta de origem da camada de transporte;
- Porta de destino da camada de transporte;
- Protocolo da camada de rede;
- Byte TOS/DSCP (Type of Service/DiffServ Code Point);
- Interface de entrada (Identificador da interface – mesmo obtido pelo SNMP ifIndex).

Nas implementações mais recentes, é possível definir quais campos serão considerados chave para a criação de um novo registro no cache de fluxos. As informações contidas no registro também podem variar de acordo com a implementação, mas no mínimo contêm os seguintes registros originais das primeiras versões:

- Data e hora de início e fim;
- Endereços da origem e do destino;
- Portas de origem e destino;
- Protocolo (TCP, UDP, ICMP etc.);
- Interface de entrada e de saída (valor referente a OID ifIndex da MIB-II do SNMP);
- Contadores de bytes transferidos e pacotes usados;

Além dessas informações, também podem aparecer:

- Código ICMP;
- Tipo de serviço (Type of Service – TOS);
- Endereço do próximo salto (gateway);
- Máscara de rede da origem e do destino;
- Número do Sistema Autônomo (ASN) de origem e de destino;
- Rótulos Multiprotocol Label Switching (MPLS);
- Endereço físico (MAC) dos quadros de origem e de destino;
- VLAN IDs dos quadros de origem e de destino;
- Identificação (login) do usuário;
- Em fluxos TCP ainda podem aparecer os flags TCP.

Fluxo só é exportado quando termina.

- No caso do TCP:
  - A flag FIN ou RST marca o término do fluxo.

Demais protocolos:

- Por inatividade na comunicação.
- Usando contador de tempo de espera configurável.
- Na casa das dezenas de segundos.
- Contador reiniciado para cada pacote do fluxo.



Assim que um pacote chega à sonda, ele é inspecionado para determinar se os valores dos campos-chave coincidem com o fluxo existente. Se coincidir, o pacote é contabilizado nesse fluxo. Caso contrário, é criada uma nova entrada no cache. Quando o fluxo termina, ele é exportado. O processo de término é o mesmo dos protocolos UDP e TCP, timeout ou encerramento de conexão, existindo ainda a possibilidade de um fluxo mais antigo ser eleito para envio ao coletor nos casos de falta de memória em cache para armazenar os novos fluxos que estão chegando. Quando enviados para a máquina coletora, os registros são armazenados em disco para posterior processamento e análise.

A exportação do fluxo só pode acontecer quando o fluxo termina. No caso do TCP, o término é marcado por um pacote com a flag FIN ou RST. No entanto, outros protocolos não possuem um marcador de término natural, nesse caso, supõe-se que o fluxo terminou por inatividade na comunicação. O tempo de espera para supor esse término é configurável e fica normalmente na casa das dezenas de segundos. Para todo pacote visto, é iniciado um contador de inatividade, se aparecer um novo pacote antes do contador atingir a quantidade de tempo configurada, o contador é reiniciado. Caso contrário, o fluxo é exportado.

A principal finalidade do uso de fluxos é determinar em tempo hábil a situação da rede, mas fluxos muitos longos só aparecem no coletor após seu término, mascarando a situação real da rede até que eles terminem. Por essa razão, a sonda tem uma pré-configuração que força a finalização do fluxo em um prazo máximo. Isso causa o registro de dois ou mais fluxos pertencentes a um único fluxo. Apesar de ser uma informação incorreta, os processadores conseguem facilmente associar dois fluxos consecutivos ao fluxo correto. Esse tempo de vida máximo comumente varia entre 1 a 30 minutos, mas podem ser configurados tempos maiores.

A finalização forçada também é usada para diminuir quantidade de recursos utilizados pela sonda no equipamento.

Registro por amostragem:

- ▣ Memória da sonda é pequena;
- ▣ O processador tem outras tarefas a realizar;
- ▣ Amostra de tráfego visto.
  - ▣ Exemplo: 1 a cada 1000 pacotes;
  - ▣ Tendências do tráfego são mantidas.



Os equipamentos de interligação (roteadores) não foram feitos especificamente para registrar fluxos, e isso pode eventualmente exaurir seus recursos durante esse registro. Para evitar isso, é possível configurar o equipamento para registrar os fluxos por amostragem do tráfego, ou seja, o equipamento registra informações apenas para uma fração dos pacotes vistos, por exemplo, 1 a cada 1000 pacotes.

Como o coletor terá apenas uma amostra do tráfego, as informações derivadas serão imprecisas, mas indicarão as tendências do tráfego.

Registro de fluxos agregados:

- ▣ Sumário de fluxos.
  - ▣ Critério de agregação.
  - ▣ Feito pela sonda;
  - ▣ Usa cache principal e caches de agregação;
  - ▣ Parâmetros configuráveis na sonda.





- Exportação.
  - Gera número menor de pacotes;
  - Usa menos banda;
  - Diminui o trabalho da máquina coletora.

Algumas vezes, apenas um sumário do tráfego é suficiente para entender o que está acontecendo na rede, por exemplo, agregação dos fluxos por sistema autônomo de origem ou destino. O registro do fluxo agregado deve conter a somatória das informações contidas nos fluxos que atendem ao critério de agregação. A agregação é feita pela sonda, através do uso de múltiplos caches, um cache principal e caches auxiliares de agregação. Os registros terminados ou expirados no cache principal são acumulados nos cache de agregação.

O critério de agregação e expiração de fluxos dos caches de agregação são configuráveis na sonda. Devido à agregação, o número de pacotes exportados é menor, assim, há uma diminuição do uso da banda e dos recursos usados pela máquina coletora para processar e armazenar os fluxos.

As sondas são normalmente implementadas por equipamentos de interligação, tais como: roteadores e switches, mas também podem ser implementadas por software em Sistemas Operacionais de uso geral, como o Linux, por exemplo.

Tanto as sondas, quanto as coletoras e analisadoras, podem ser implementados por equipamento especificamente projetado para essas funções, com ganhos de desempenho e facilidade de implantação.



Registros personalizados:

- Formato definido por modelo (template);
  - Campos-chave.
  - Número de fluxos aumenta proporcionalmente.
- Campos para registro;
- Informado de tempos em tempos;
  - Registros de fluxos.
  - Contém informações mínimas sobre o modelo (template).

As versões mais recentes das implementações são muito flexíveis. Elas permitem uso de registros personalizados, onde é possível definir quais campos são chave para criação de novas entradas no cache e quais campos devem ser registrados nos fluxos. O número de fluxos criados é proporcional ao número de campos-chave do registro.

A personalização é feita através de modelos (templates) que são enviados da sonda para o coletor de tempos em tempos. Dessa forma, o coletor passa a ter os subsídios para interpretar os dados contidos nos fluxos recebidos.

Os registros de fluxos propriamente ditos contêm informações mínimas sobre o modelo para que seja possível relacionar o conjunto de dados do fluxo ao modelo (template) recebido previamente.

## Saiba mais



No caso das máquinas coletoras, é comum usar um Sistema Operacional de uso geral, onde uma ferramenta coleta e armazena os fluxos e outras ferramentas processam, analisam e geram relatórios e gráficos sobre os fluxos armazenados.

## Métodos de exportação

A exportação dos fluxos pode ocorrer de várias formas. A transmissão pode ser em unicast (um destinatário – um coletor) e em multicast (grupo de destinatários – grupo de coletores). Já em relação ao protocolo, apesar da predominância do uso do User Datagram Protocol (UDP), também é possível usar o Transmission Control Protocol (TCP) e o Stream Control Transmission Protocol (SCTP).

Depois que os fluxos estão armazenados, então é possível processá-los para derivar informações mais sintéticas sobre a massa de dados coletados, através da aplicação de filtros, agregação de fluxos e correlacionamento de fluxos. Com isso, geram-se relatórios textuais e gráficos com base nos fluxos coletados, tais como protocolos mais usados, banda utilizada por máquina etc. Algumas dessas ferramentas são de tal forma inteligentes que são capazes de determinar se há algum tipo de tráfego abusivo com base em tráfegos considerados normais previamente coletados ou em limiares predefinidos pelo administrador. As ferramentas NFSen e Plixer Scrutinizer são exemplos de ferramentas com tal inteligência.

## Onde ativar as sondas

O melhor lugar é ativá-las são em pontos centralizadores do tráfego da rede, tais como roteadores e switches, mas nada impede que elas sejam ativadas em servidores e estações de trabalho rodando softwares como o ntop-ng.

Muitos switches possuem o recurso de espelhamento de porta, que permite a cópia do tráfego visto em uma ou mais portas para uma outra porta do equipamento. Esse recurso também é encontrado sob os nomes Switched Port Analyzer (SPAN), na Cisco, e Roving Analysis Port (RAP), na 3Com. Com isso, é possível colocar uma sonda (exemplo: um servidor rodando ntop-ng) nessa porta espelhada.

Espelhamento de porta:

- Porta de espelhamento, SPAN ou RAP;
- 1:1 – 1 porta espelho e 1 porta de produção;
- N:1 – 1 porta espelho e N portas de produção.



O espelhamento pode ser do tipo 1:1, ou seja, 1 porta de espelhamento copia o tráfego de uma porta de produção; ou pode ser N:1, onde 1 porta de espelhamento copia o tráfego de N portas de produção do switch.

Existem também equipamento de fim específico para a coleta de fluxos de forma transparente e com capacidade de exportar os registros de fluxos para um coletor.

## Implementações

Tecnologias concorrentes:

- NetFlow (Network Flow), da Cisco Systems.
  - Deu início a tudo e hoje em dia é muito flexível.
  - Versão 5 é a mais usada, mas existe a versão 9.
- sFlow (Sampling Flow), da InMon Corporation.
  - Introduziu o conceito de amostragem de tráfego.
  - Implementada por CHIP dedicado adicionado ao hardware.
  - Tipos de mensagens diferentes. Está na versão 5.



- IPFIX (IP Flow Information Export) da IETF:
  - Baseada na versão 9 do NetFlow.
  - Introduziu o uso de templates (modelos).
  - Pretende ser o padrão no monitoramento de fluxos.

Entre as tecnologias para monitoramento de fluxo, existem três que se destacam:

- **NetFlow (Network Flow)**: desenvolvida pela Cisco Systems. Outros fabricantes dão nomes diferentes para a mesma tecnologia. Exemplo: Juniper Networks: jFlow, 3Com – NetStream etc. Deu início a tudo, sendo a plataforma mais usada atualmente. Evoluiu bastante com o tempo e aplica todas as facilidades de monitoramento de fluxo. A versão 5 é a mais usada, atualmente está na versão 9, também conhecida pelo nome Flexible Netflow. É muito flexível, pois permite a definição dos campos-chave e dos dados a serem registrados;
- **sFlow (Sampling Flow)**: desenvolvida pela InMon Corporation. Introduziu o conceito de amostragem de tráfego. É implementada por CHIP dedicado adicionado ao hardware, liberando o processador principal. Podem ser usados tipos de mensagens diferentes, flexibilizando a definição de quais dados serão registrados. Está na versão 5;
- **IPFIX (IP Flow Information Export)**: desenvolvida pela Internet Engineering Task Force (IETF), baseada na versão 9 do NetFlow. Foi desenvolvida com base no Netflow versão 9. Introduziu o conceito de templates (modelos) para definir os dados a serem enviados para o coletor. Pretende ser o padrão no monitoramento de fluxos.

Sondas por Hardware:

- Sondas embutidas nos roteadores e switches;
  - Netflow: Cisco, Juniper Networks etc.
  - sFlow: Foundry, Extreme Networks etc.
- Stand-alone Appliances.
  - Nmon nBox.
  - Invea Tech FlowMon Probe.

As sondas implementadas por hardware comumente vêm instaladas em equipamentos de interligação de rede, tais como roteadores e switches. Os equipamentos da Cisco Systems e Juniper Networks são exemplos que suportam Netflow. Já os equipamentos da Foundry e Extreme Networks são exemplos que suportam sFlow.

Além dessas, existem também máquinas autosuficientes de fim especial (Stand-alone Appliances) para realizar essa tarefa. Esses equipamentos são preparados para suportar uma grande quantidade de tráfego, e assim evitar a perda de fluxos da rede.

A seguir, dois exemplos desses equipamentos:

- Nmon nBox;
- Invea Tech FlowMon Probe.





**Figura 7.12**  
Hardwares especializados em coleta de fluxos na rede.

Sondas por software:

- Rodam em S.O. de uso geral.
  - Linux, FreeBSD, Solaris etc.
- Usam a libpcap.
- Os formatos de fluxo suportados variam.

As sondas implementadas por software podem ser instaladas e configuradas em Sistemas Operacionais de uso geral, tais como Linux e FreeBSD. A maioria delas usa a biblioteca libpcap para fazer a captura dos pacotes da rede para depois processá-los e finalmente exportar o fluxo.

Alguns exemplos de sondas, formatos suportados e respectivos endereços na internet:

	Netflow v5	Netflow v9	sFlow v5	IPFIX
fprobe	X			
softflowd	X	X		
pmacct	X	X	X	
nProbe	X	X	X	X

**Tabela 7.3**  
Formatos suportados por cada solução.

Coletores:

- Appliances;
  - Interface de administração e operação remota.
  - Com código proprietário ou aberto.
- Proprietários;
  - Plixer International Scrutinizer
  - ManageEngine Netflow Analyzer
  - Lancope StealthWatch
  - Arbor Networks Peakflow
  - Código aberto
    - Flow-tools
    - NFDump/NFSen
    - Flowscan

O coletor é normalmente implementado em software. Existem dezenas de coletores disponíveis do mercado, tanto proprietários quanto de código aberto.

Alguns integradores embutem software em equipamentos de uso geral, adicionam interfaces gráficas para administração e operação para comercializá-los como appliances de análise de fluxos. O software embutido nesses appliances pode ser proprietário ou aberto.

Entre os softwares de código proprietário, destacam-se os seguintes produtos:

- Scrutinizer FlowAnalyzer
- ManageEngine Netflow Analyzer
- Lancope StealthWatch
- Arbor Networks Peakflow

Entre os softwares de código aberto, os seguintes estão entre os mais utilizados

- Flow-tools
- NFDump/NFSen
- Flowscan
- SiLK

## Exemplos de Configurações

- Sonda: Cisco Systems IOS (configuração)

```
interface FastEthernet0/0
description Access to backbone
ip address 192.168.0.1 255.255.255.0
ip route-cache flow ! IOS ver < 12.4
ip flow [ingress|egress] ! IOS ver >= 12.4
duplex auto
speed auto
!
ip flow-export version 5
ip flow-export destination 192.168.0.2 2055
ip flow-cache timeout active 5
```

- Sonda: Cisco Systems IOS (comandos)

- show ip flow export: mostra situação, configuração e estatísticas sobre o exportador
- sh ip cache flow: mostra estatísticas e valores dos registros armazenados no cache
- sh ip flow top-talkers: mostra os fluxos por ordem decrescente de quantidade de bytes transferidos.

Os equipamentos da Cisco Systems rodam os Sistemas Operacionais IOS (roteadores) e CatOS (switches), eles implementam o Flexible Netflow e podem exportar os formatos mais comuns de pacotes Netflow.

O exemplo ao lado se aplica a um roteador, a configuração habilita o Netflow na interface FastEthernet0/0, configura o exportador para mandar os pacotes Netflow v5 para o coletor hospedado no endereço 192.168.0.2 e estabelece que o tempo de vida máximo do fluxo é 5 minutos.

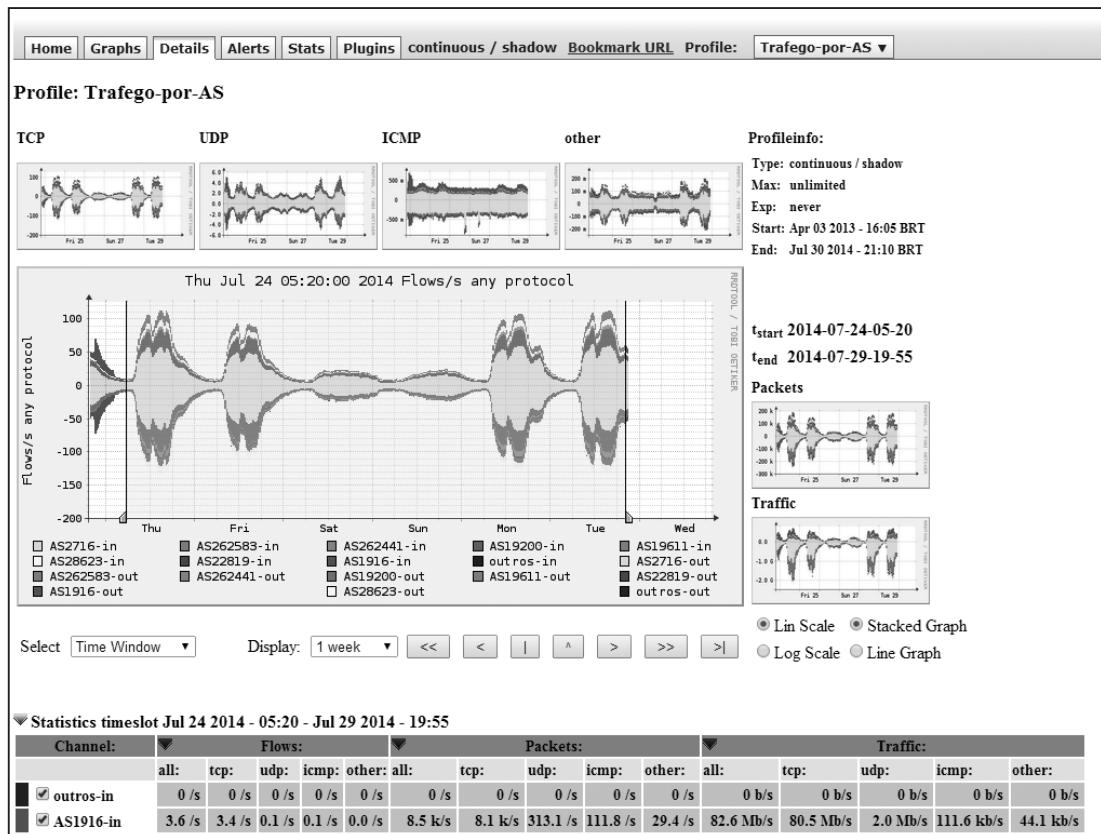


- As sondas fprobe e softflowd rodam em sistemas Linux e FreeBSD e fazem uso da biblioteca libpcap para interceptar o tráfego. A seguir seguem os endereços da página web destas ferramentas:
  - fprobe: <http://fprobe.sourceforge.net/>
  - softflowd: <http://www.mindrot.org/projects/softflowd/>
- O Flow-tools é uma biblioteca e um conjunto de ferramentas de linha de comando usados para coletar e processar fluxos NetFlow e para gerar relatórios sobre esses fluxos.
  - flow-capture – coleta, comprime, armazena e mantém o espaço utilizado no disco.
  - flow-cat – concatena arquivos que contém os fluxos coletados.
  - flow-dscan – ferramenta simples para detectar alguns tipos de varreduras de rede e ataques de negação de serviço.
  - flow-expire – expira fluxos usando as mesmas políticas disponíveis no flow-capture.
  - flow-export: exporta fluxos para o formato ASCII ou cflowd.
  - flow-fanout – replica os pacotes Netflow para destinos unicast e multicast, permitindo o uso de múltiplos coletores simultâneos.
  - flow-filter – filtra fluxos com base em qualquer dos campos contidos no registro do fluxo.
  - flow-gen – gera dados de teste.
  - flow-header – mostra meta informações contidas no arquivo de fluxos.
  - flow-import – importa fluxos no formato ASCII ou cflowd.
  - flow-log2rrd: processa as linhas STAT (estatísticas) do log provenientes do flow-capture e flow-fanout e converte para RRD
  - flow-mask: aplica rótulos a arquivos de fluxos
  - flow-merge – junta arquivos de fluxos em ordem cronológica.
  - flow-nfilter: filtra fluxos com base em qualquer dos campos contidos no registro do fluxo. Filtros definidos em arquivo.
  - flow-print: mostra os fluxos em ASCII usando um formato predefinido. O formato é selecionável entre vários.
  - flow-receive: recebe fluxos exportados no formato Netflow sem armazenar no disco
  - flow-report – gera relatórios sobre o conjunto de dados coletados.
  - flow-rpt2rrd: converte a saída do flow-report separado por vírgula para o formato RRD
  - flow-rptfmt: converte a saída do flow-report separado por vírgula para ASCII ou HTML formatado
  - flow-send – envia dados pela rede usado o protocolo NetFlow.
  - flow-split: parte arquivos de fluxos em arquivos menores com base em tamanho, data e hora ou rótulo.
  - flow-stat: gera relatórios com os dados dos fluxos capturados.
  - flow-tag: rotula fluxos com base no endereço IP ou número do AS, facilitando o agrupamento desses fluxos.
  - flow-xlate: executa a tradução de alguns campos de registro de fluxos.
- O NFDump é um conjunto de ferramentas de linha de comando para coleta e processamento de fluxos de dados.
  - nfcapd – capturador de fluxos no formato NetFlow



- sfcapd – capturador de fluxos no formato sFlow
- nfdump – leitor de fluxos capturados
- nfprofile – processador de filtros para criação e armazenamento de perfis de fluxos.
- nfreplay – replicador de fluxos
- nfclean.pl – coletor de lixo; removedor de fluxos antigos
- **NFSEN** é uma interface web para apresentação dos dados coletados pelo NFDump, que permite o manuseio dos fluxos de uma forma mais simplificada, permitindo entre outras coisas:
  - Visualização dos fluxos em relatórios e gráficos
  - Navegação pelos fluxos
  - Sumário de estatísticas
  - Aplicação simplificada de filtros
  - Configuração de perfis de tráfego
  - Permite o uso de plugins
  - Permite a configuração de alertas
  - Re-configuração fácil com o comando ‘nfsen’

**Figura 7.13**  
Interface para análise de fluxos do NFSEN.



## Considerações finais

- Problemas
  - Os fluxos não são classificados pelo conteúdo
  - Fluxos perdidos tornam a informação imprecisa
  - O espaço em disco consumido no coletor é grande
  - Os relatórios e gráficos não substituem o humano na análise



- Qual o valor agregado com o uso de fluxos?
  - Granularidade nas informações
  - Inspeção pouco intrusiva
  - Melhor conhecimento da rede
- Visão ampla, pontual e histórica
- Apoio na análise de problemas de rede

O uso de fluxos traz muitas vantagens para os administradores de rede, mas a tecnologia não é perfeita.

As informações derivadas dos fluxos são baseadas puramente em cabeçalhos, ou seja, o conteúdo não é inspecionado. Essa característica é interessante, pois há pouca intrusão no tráfego passante, porém isso também evita a verificação profunda do tráfego, tornando a classificação imperfeita, feita com base apenas nos protocolos e portas utilizadas.

Os fluxos perdidos causam imprecisões nos relatórios. As implementações atuais possuem métodos para descobrir se houve perda de fluxo e complementar a informação mostrada para o administrador. A perda de fluxos pode ser minimizada com um bom acoplamento do desempenho da sonda e do coletor em relação à quantidade de fluxos a ser tratada.

Um fator impeditivo no uso de monitoramento de fluxos é a quantidade de espaço em disco utilizada pelo coletor para armazenar o histórico de fluxos. Redes de alta capacidade geram uma quantidade de fluxos absurdamente alta, muitas vezes, levando ao uso de amostras de tráfego e consequentemente imprecisão nos relatórios.

Ainda que os fluxos revelem informações valiosas, eles de nada servem sem um humano com a capacidade analisá-los corretamente.

A justificativa para se usar fluxos é que eles permitem uma granularidade melhor que o uso de medidores de tráfego de interface, por exemplo. Enquanto uma medição por SNMP fornece apenas a informação da uso de uma interface, o monitoramento de fluxo diz quais máquinas clientes usaram e a quantidade de banda utilizada por cada uma.

Como o monitoramento de fluxos não inspeciona o conteúdo, ele é menos intrusivo que outras tecnologias e menos sujeito a reclamações de invasão de privacidade por parte dos usuários.

O melhor conhecimento da rede calca-se no fato ser possível verificar facilmente picos de uso, tendências, desempenho de aplicações, atividade maliciosa etc.

Com a tecnologia de fluxos é possível ter uma visão ampla da situação, mas também é possível verificar problemas pontuais e até mesmo ter uma visão do passado para se ter um referencial de comparação.

Com tudo isto, o uso de fluxos é mais uma facilidade para resolver problemas de rede, as informações deles derivadas podem indicar imediatamente um problema específico ou podem ser usadas em conjunto com outras ferramentas para se chegar a uma solução.



## Referências

- SINGER, A.; BIRD, T. Building a Logging Infrastructure. The USENIX Association. 2004. ISBN: 978-1-9319-7125-6
- KENT, K.; SOUPPAYA, M. Guide to Computer Security Log Management: Recommendations of the National Institute of Standards and Technology. NIST Special Publication 800-92. 2006. <http://csrc.nist.gov/publications/nistpubs/800-92/SP800-92.pdf>
- REID, Gavin. Cisco CSIRT on Advanced Persistent Threat. 2011. Cisco Blogs. <http://blogs.cisco.com/security/cisco-csirt-on-advanced-persistent-threat/>. Disponível em 23/09/2013.
- Cisco IOS Netflow: <http://www.cisco.com/web/go/netflow>
- InMon sFlow: <http://www.sflow.org/>
- IPFIX: <http://www.ietf.org/html.charters/ipfix-charter.html>
- Port Mirroring: [http://en.wikipedia.org/wiki/Port\\_mirroring](http://en.wikipedia.org/wiki/Port_mirroring)
- Libpcap: <http://www.tcpdump.org/>
- nBox: <http://www.nmon.net/nBox.html>
- FlowMon Probe: <http://www.invea-tech.com/products/flowmon-probes>
- fprobe: <http://fprobe.sourceforge.net/>
- Softflowd: <http://www.mindrot.org/projects/softflowd/>
- Pmacct: <http://www.pmacct.net/>
- nProbe: <http://www.ntop.org/nProbe.html>
- Plixer Scrutinizer: <http://www.plixer.com/products/scrutinizer.php>
- ManageEngine Netflow: <http://www.manageengine.com/products/netflow/>
- Lancope StealthWatch: <http://www.lancope.com/products/>
- Arbor Networks Peakflow: <http://www.arbornetworks.com/>
- Flow-tools: <http://www.splintered.net/sw/flow-tools/>
- NFDump: <http://nfdump.sourceforge.net/>
- NFSen: <http://nfsen.sourceforge.net/>
- SiLK: <http://tools.netsa.cert.org/silk/>
- FlowScan: <http://net.doit.wisc.edu/~plonka/FlowScan/>





# 8

## Gerenciamento de performance e qualidade de serviço

objetivos

Entender o gerenciamento de qualidade de serviços (QoS). Aprender os fundamentos de QoS. Entender o QoS na Internet. Conhecer as características do provisionamento de tráfego. Entender os Tipos de Serviços. Conhecer o mecanismo de Service Level Agreements (SLA). Conhecer os mecanismos de implementação de QoS. Entender as métricas e os padrões para o gerenciamento de rede. Conhecer as ferramentas de diagnóstico/monitoração da rede.

conceitos

Visão Geral Sobre QoS, Qos na Internet, Serviços Diferenciados, Condições de provisionamento de tráfego, Tipos de Serviços, Provisionamento e configuração, Service Level Agreements, Mecanismos para implementar QoS, Gerenciamento de Congestionamento, Mecanismos de policiamento e conformação, Métricas para o gerenciamento de rede, Padrões para Monitoração da Rede, Fluxos, Disponibilidade, Ferramentas de diagnóstico e monitoração da rede

### Introdução

Embora a capacidade de banda atualmente disponível seja bem superior ao que havia há cinco anos, por exemplo, em função dos backbones em fibra ótica, ainda assim é insuficiente, pois o tráfego continuamente cresce até o limite da capacidade instalada. É preciso gerenciar continuamente a performance da rede e desencadear medidas paliativas quando necessário.

Expandir a capacidade nem sempre é a solução possível, especialmente por razões de custo, mas que também pode ser inibida por limites da tecnologia disponível. Por isso, é necessário avaliar se o uso da rede é apropriado, se não está ocorrendo uso supérfluo, com aplicações não prioritárias ou que não apoiam a missão da instituição, gerando tráfego, em detrimento de aplicações relevantes e necessárias. Isso leva à necessidade de definição e implantação de alguma forma de provimento de serviços com níveis de qualidade diferenciada. A qualidade de serviço pode ser gerenciada na internet com o apoio de soluções que permitam atender diferentes parcelas do tráfego com estratégias de priorização diferenciadas.

O Gerenciamento da Qualidade de Serviço, que costuma ser designado como QoS (Quality of Service), precisa ser acompanhado de gerenciamento de banda. Os mecanismos de Qualidade de Serviço a serem utilizados são derivados do que é combinado no contrato de prestação de serviço, e esse acordo é usualmente referido como SLA (Service Level Agreement).



Necessidade de redes com o conceito de qualidade de serviços (QoS):

- Novas aplicações:
  - Videoconferência e telemedicina;
  - educação a distância.
- Não funcionam adequadamente em redes baseadas em melhor esforço, como a internet.

A questão é: qual é o melhor mecanismo para implementar QoS? Na década de 1990, foram propostos modelos para internet, como o Integrated Services e o Differentiated Service (DiffServ), os quais abordam vários tipos de serviços, incluindo serviços best-effort e real-time, além de permitir reserva de banda.

O IntServ ou Serviços Integrados é um modelo de implementação do QoS desenvolvido para garantir a qualidade do serviço para fluxos individuais de tráfego, usando para tanto a sinalização fim-a-fim e a reserva de recursos por toda a rede, dos roteadores intermediários até o roteador de destino. O modelo de serviços integrados é caracterizado pela reserva de recursos. Antes de iniciar uma comunicação, o emissor solicita ao receptor a alocação de recursos necessárias para definir-se uma boa qualidade na transmissão dos dados. O protocolo Resource Reservation Protocol (RSVP) é utilizado, nesse modelo, para troca de mensagens de controle de alocação dos recursos. A alocação de recursos diz respeito à largura de banda e ao tempo em que será mantida a conexão. Nesse período de tempo, o emissor daquele serviço tem uma faixa da largura de banda disponível para transmitir seus dados (Santos, 1999).

Modelo atual na internet:

- Serviço do tipo “best-effort”, onde todo o tráfego é tratado da mesma forma e o melhor esforço é empregado para entregá-lo.

Modelo esperado pela comunidade internet:

- Aplicações sensíveis ao tempo, como voz e vídeo.
- Tratamento preferencial para alguns tipos de tráfego de aplicações.

## Demonstração

O modelo de serviços diferenciados (DiffServ) implementa QoS com base na definição de tipos de serviços. No cabeçalho de um pacote IP, existe um campo chamado *TOS* (*Type of Service*), que pode representar o tipo do serviço. Esse campo inclui poucos bits, conhecidos como IP Precedence, que são usados para priorizar tráfego através de enfileiramento diferenciado dentro de roteadores. Por exemplo, tráfego de alta prioridade, indicado pelo valor mais alto do campo *IP Precedence*, deve ser colocado na fila de alta prioridade no roteador e repassado antes das filas de baixa prioridade.

O DiffServ é uma arquitetura que foi inicialmente implantada na internet<sup>2</sup>, mas que se popularizou e atualmente é oferecida na maioria dos roteadores. A gerência da Qualidade de Serviço da rede (Service Level Management ou SLM) demanda duas estratégias, uma para implantar e configurar os equipamentos que vão proporcionar o tratamento diferenciado ao tráfego e outra para monitorar a performance da rede com ou sem priorização. No primeiro caso, é preciso conhecer os mecanismos usados na solução DiffServ e, no segundo caso, utilizar métricas e ferramentas para a medição da performance. Empregar QoS na rede implica em (Cisco 1999):

- **Controle sobre os recursos:** controlar quais são os recursos que estão sendo usados (bandwidth, equipamentos, facilidades de wide-area etc.). Por exemplo, limitar a banda consumida sobre um backbone por uma transferência de File Transfer Protocol (FTP) ou dar prioridade a uma base de dados importante;



- ▣ **Serviços particulares:** no caso de um ISP, o controle e a visibilidade providos pelo QoS habilita o oferecimento de serviços diferenciados a seus clientes;
- ▣ **Coexistência de aplicações de missão crítica:** o QoS garante que a rede será utilizada eficientemente por aplicações de missão crítica; que o bandwidth e atrasos requisitados por aplicações sensíveis ao tempo estarão disponíveis; e que outras aplicações, utilizando o link, não afetarão o tráfego de missão crítica.

## Visão Geral Sobre QoS

Quality of Service ou Qualidade de Serviço: a qualidade necessária para satisfazer o usuário de uma dada aplicação.

Aplicações necessitam de QoS diferentes:

- ▣ Telefonia;
- ▣ Videoconferência;
- ▣ Download de arquivos;
- ▣ TV.

A necessidade de QoS na internet é um fato. Até hoje, a internet tem oferecido apenas serviço do tipo “best-effort”. Outros modelos para internet:

- ▣ Integrated Services;
- ▣ Differentiated Service (DiffServ).

Qualidade de Serviço refere-se à habilidade da rede em prover melhores serviços a um tráfego de rede selecionado, sobre vários tipos de tecnologias. Em particular, características de QoS proveem melhores e mais serviços de redes, uma vez que:

- ▣ Suportam bandwidth dedicado;
- ▣ Possuem melhorias em relação à perda;
- ▣ Mecanismos para evitar e gerenciar congestionamento de rede;
- ▣ Mecanismos para conformação do tráfego da rede;
- ▣ Mecanismos para configuração de priorização de tráfego através da rede.

Dentro da arquitetura de QoS, os seguintes componentes são necessários para viabilizar o atendimento do tráfego com qualidade diferenciada em redes:

- ▣ Mecanismos de QoS dentro de um único elemento de rede, o qual inclui funcionalidades de enfileiramento, tratamento diferenciado de filas e conformação de tráfego;
- ▣ Técnicas de sinalização QoS fim a fim entre os elementos de rede;
- ▣ Funcionalidades de policiamento e gerenciamento de QoS para controlar e administrar tráfego fim a fim através da rede.

Nem todas as técnicas são apropriadas a todos os roteadores da rede, porque roteadores de borda e roteadores de backbone necessariamente não realizam as mesmas operações; assim, as tarefas de qualidade de serviços podem ser diferentes (CISCO 1999).

Em geral, roteadores de borda realizam as seguintes funções de QoS:

- ▣ Classificação de pacotes;
- ▣ Controle de admissão;
- ▣ Gerenciamento de configuração.

Em geral, roteadores de backbone realizam as seguintes funções de QoS:

- Gerenciamento de congestionamento;
- Função de evita congestionamento.

## O modelos de serviços QoS fim a fim

- IntServ 1994 – Integrated Services.
- Protocolo RSVP: Resource Reservation Protocol.
- Fluxo em tempo-real e fluxo best effort.
- Gerenciamento de QoS no nível de micro fluxos.



Cada aplicação que requeira algum tipo de garantia precisa fazer um pedido de reserva, e os roteadores ao longo da rota podem enviar respostas concordando com o pedido de reserva.

Um modelo de serviço, também chamado nível de serviço, descreve um conjunto de características QoS fim a fim. O QoS fim a fim é a habilidade da rede em entregar requisitos de serviço para um tráfego de rede específico, de um fim da rede a outro. Encontramos três tipos de modelos de serviços QoS: best-effort, integrated e differentiated services.

- **Best-Effort Service:** o Best-effort é um modelo de serviço único no qual uma aplicação envia dados quando desejar, em qualquer quantidade, e sem requisitar permissão ou informar primeiro a rede. Para serviços best-effort, a rede entrega os dados se ela puder, sem qualquer tipo de segurança de entrega, atraso associado ou throughput;
- **Serviços Integrados:** é um modelo de serviço múltiplo que acomoda múltiplos requisitos de QoS. Nesse modelo, a aplicação requisita um específico tipo de serviço da rede antes de enviar os dados. A requisição é realizada através de sinalização; a aplicação informa a rede do seu perfil de tráfego e requisita um tipo particular de serviço. A aplicação envia dados apenas depois que ela recebe a confirmação da rede. Envia dados de acordo com as regras descritas no perfil de tráfego. A rede realiza controle de admissão, baseada na informação da aplicação e recursos de rede disponíveis. Ela realiza a manutenção por estado do fluxo e então realiza classificação de pacotes, policiamento e enfileiramento inteligente baseado nesse estado. Esse modelo utiliza o protocolo Resource Reservation Protocol (RSVP) para sinalizar seus pedidos de QoS para o roteador;
- **Serviços Diferenciados:** é um modelo de serviços múltiplos que pode satisfazer diferentes tipos de requisitos de QoS. Entretanto, diferentemente do modelo de Serviços Integrados, uma aplicação usando Serviços Diferenciados explicitamente não sinaliza o roteador antes de enviar o dado. Para os serviços diferenciados, a rede tenta entregar um tipo particular de serviço, baseado na específica QoS de cada pacote. Essa especificação pode ocorrer de diferentes maneiras. Por exemplo: usando o bit IP Precedence setado em pacotes IP ou endereços de fonte e destino. A rede usa a especificação QoS para classificar, conformar e policiar tráfego, e para realizar enfileiramento inteligente.

O modelo de serviços diferenciados é usado por inúmeras aplicações de missão crítica e para prover QoS fim a fim. Tipicamente, esse modelo de serviço é apropriado para fluxos agregados porque ele realiza classificação de tráfego. Características do modelo de Serviços Diferenciados incluem:

- O Committed Access Rate (CAR) realiza classificação de pacotes através do IP Precedence e conjuntos de regras de QoS. O CAR realiza medições e policiamento de tráfego, proporcionando gerenciamento de banda;



### **Weighted random early detection (WRED)**

Random Early Detection (RED), também conhecido como descarte preliminar aleatório, é uma disciplina de enfileiramento fila para um programador de descartes na rede que visa evitar congestionamentos. Em Weighted RED você pode ter diferentes probabilidades para diferentes prioridades (precedência IP, DSCP) e/ou filas.

### **Weighted fair queuing (WFQ)**

Weighted Fair Queueing (WFQ) é uma técnica de programação de envio (Scheduling) de pacotes de dados que permite diferentes prioridades de agendamento para os fluxos de dados multiplexados estatisticamente.

- Esquemas inteligentes de enfileiramento, tal como **Weighted random early detection (WRED)** e **Weighted fair queuing (WFQ)**, podem ser utilizados com CAR para entregar serviços diferenciados.

O objetivo principal da nova internet é suportar o avanço das novas aplicações de rede. Ao contrário do que deveria ser, muitas dessas aplicações não são visíveis na internet atual, pelo fato de o modelo presente de entrega “best-effort” não prover a mínima solicitação de performance fim a fim assegurada. Para habilitar essas aplicações, a nova internet deve prover qualidade de serviço (QoS) funcional que permita a estas reservarem recursos de redes-chave sem causar impacto no tráfego best-effort.

Com o passar dos últimos anos, as aplicações internet e a comunidade de engenheiros têm identificado um conjunto de requisitos para QoS na internet, baseado em necessidades de aplicações e de engenheiros. Na verdade, o internet2 QoS Working Group tem estudado esses requisitos e recomendado a adoção de Differentiated Services (DiffServ) para QoS.

O framework DiffServ tem emergido nos últimos anos como uma forma simples e escalar de QoS que provê serviços significativamente fim a fim através de múltiplas nuvens de rede administrativamente separadas, sem necessidade de complexidade. Objetivos que se igualam ao da internet2 QoS – ênfase em simplicidade, escalabilidade, interoperabilidade e administrabilidade.

## **QoS na internet**

Na internet atual, cada elemento ao longo do caminho do pacote IP não faz nada mais que o melhor esforço para entregar o pacote a seu destino. Se a fila do roteador é sobre carregada, pacotes são descartados com pouca ou nenhuma distinção entre tráfego de baixa prioridade e tráfego urgente. Isso é conhecido como serviço best-effort.

Para funcionar corretamente, muitas aplicações avançadas necessitam o máximo de banda garantida e o mínimo de atraso do pacote (latência), os quais o meri best-effort não pode disponibilizar. Por exemplo, ferramentas remotas de colaboração, videoconferência ou tele-medicina geralmente têm requisitos de qualidade mais exigentes que podem ser descritos em termos de demanda de banda e latência mínimos. Se tais requisitos não forem atendidos, a aplicação pode ser inviabilizada.

Com vistas a poder oferecer QoS na internet, o internet2 QoS Working Group identificou alguns requisitos:

- Habilitar aplicações avançadas;
- Escalabilidade;
- Administração;
- Medição;
- Admitir múltiplas e interoperáveis implementações de pedaços individuais de equipamentos e nuvens;
- Suporte de Sistemas Operacionais e middleware.

Quando alguém pergunta qual é a qualidade de serviço que uma aplicação necessita da rede, desenvolvedores de aplicações costumam dizer coisas como: “Eu necessito de toda a banda que puder ser dada com pouca latência, pouco jitter e pouca perda.” Essa resposta na verdade é uma utopia. A realidade das redes faz com que os responsáveis pelo desenvolvimento de aplicações procurem tornar o software capaz de ajustar-se a uma grande variedade de throughputs. Mas tudo tem limite. Se a rede não conseguir proporcionar um conjunto de requisitos de qualidade mínimo, mesmo uma aplicação flexível e tolerante não poderá funcionar.

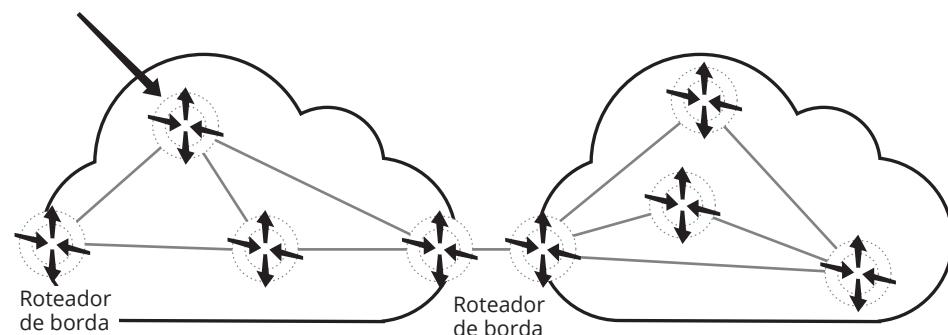


Para dar suporte ao desenvolvimento de aplicações ditas avançadas, o protocolo de transporte internet, TCP, tem sido aperfeiçoado nos últimos anos com vistas a adaptar-se e tentar compensar os efeitos de congestionamento na rede. Mas nesse contexto não há limite para os pedidos de conexão que vão sendo ativados. O que ocorre quando esse número aumenta é que a falta de recursos para atendimento de toda a demanda provoca uma degradação gradual na performance das conexões. Por outro lado, em uma rede com QoS habilitada, o usuário perceberá um modelo de serviço parecido com o sistema de telefonia. Ao tentar estabelecer uma conexão, precisa iniciar a conexão e reservar os recursos necessários. Se isso for possível, o usuário obtém um canal livre para transmitir, com os requisitos solicitados. Alternativamente, no momento de call setup, o usuário talvez receba um sinal ocupado e lhe seja negado o privilégio de conectar em um nível de QoS desejado.

Uma dimensão fundamental de qualquer requisito de QoS de aplicação é o conjunto de parâmetros de transmissão. Os parâmetros de transmissão mencionados como requisitos básicos são banda passante e latência.

Existe um aspecto importante a considerar na implementação de um serviço com QoS garantida relativo ao tempo de duração do serviço. Para gerenciar a ativação e o ajuste dos mecanismos de garantia de QoS, é necessário utilizar funcionalidades de gerenciamento de configuração para propagar na rede os parâmetros relativos aos mecanismos de tratamento diferenciado que são inerentes a um sistema de oferta de serviços de rede com QoS. Adicionalmente, existem questões a considerar quando o serviço cresce e precisa ser oferecido ao longo de uma trajetória que envolva mais de um domínio administrativo.

A figura seguinte ilustra os elementos integrantes do sistema de oferta de serviços de rede com QoS. O tráfego ingressa na rede através de algum dos roteadores de borda, proveniente de algum equipamento final de usuário ou de alguma outra rede interconectada. Nos roteadores de borda, os pacotes são inspecionados e eventualmente marcados para que os roteadores de trânsito da rede tenham informações que apoiem as decisões de tratamento diferenciado ou não para os diversos fluxos de pacotes.



### Saiba mais

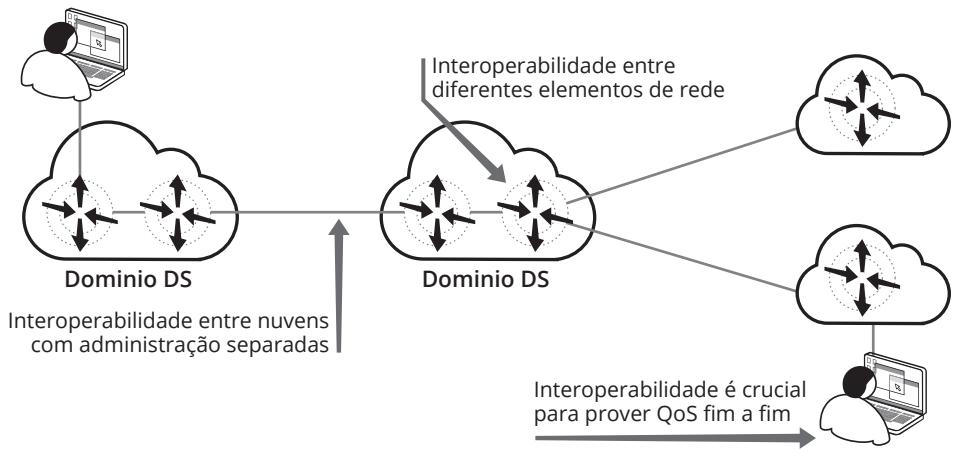
Algumas aplicações requerem também jitter limitado (variação no atraso dos pacotes), embora a maioria mascara o jitter através de buffers.

**Figura 8.1**  
Elementos integrantes de uma rede com QoS.

Como qualquer outro recurso, existe a necessidade de mecanismos para alocação e contabilização para QoS. Esses mecanismos devem operar eficientemente, provendo aos usuários o acesso rápido às características de QoS que a rede é capaz de prover.

Considerando a possibilidade de que instituições e eventualmente usuários venham a pagar por serviços de QoS, deve haver maneiras para que o usuário possa medir e auditar a performance da rede. Os requisitos de medição implicam não apenas a necessidade de ferramentas de medição, mas também uma necessidade por métricas de performance de rede. Provedores de rede podem necessitar de ferramentas de medição adicionais para assistir o provisionamento e depuração de serviços de QoS, e possivelmente suportar mecanismos de controle de admissão baseados em medidas automatizadas.

Qualquer tipo de QoS escolhido para ser implementado deve ser suportado por um ou por muitos fornecedores de equipamentos. Em redes heterogêneas grandes, como a internet, a interoperabilidade entre equipamentos de diferentes fabricantes é absolutamente essencial. Adicionalmente, a habilitação QoS em fluxos e sinalização de call setup deve ser tratada de maneira padronizada em ligações entre redes sob administrações diferentes, mesmo que cada uma das redes possa implementar QoS internamente de diferentes maneiras. Implementações internas de QoS podem variar, dependendo das tecnologias empregadas na rede, do policiamento interno e das decisões de provisionamento.



Essa é uma situação típica de ambiente na nova internet. Como as interconexões são separadas por diferentes controles administrativos, existe a necessidade de padronizar a noção de QoS através rede interoperantes.

## Serviços Diferenciados

A arquitetura de serviços diferenciados (referida como Diffserv) procura prover um espectro de serviços na internet sem ter de manter estados de fluxos para cada roteador. Isso ocorre através da união de fluxos dentro de um pequeno número de agregados, aos quais é oferecido um tratamento diferenciado pela rede. O Diffserv elimina a necessidade de reconhecimento e armazenamento de informações sobre cada fluxo individual no roteador do core. Cada fluxo é policiado e marcado no primeiro roteador por onde for encaminhado e que integre o serviço Diffserv. Isso acontece de acordo com o perfil de serviço contratado. Na visão do administrador de rede, o primeiro roteador é um roteador-de borda na periferia da rede. Esse roteador é responsável por realizar o policiamento e marcação dos pacotes recebidos dos hosts usuários da rede. Quando uma decisão de controle de admissão local for feita pela rede, o roteador é configurado com o perfil de contrato do fluxo do serviço. No trajeto a partir desse roteador, todo o tráfego integrante daquele perfil é tratado como agregado.

É definida através de um modelo simples no qual tráfego que entra na rede é:

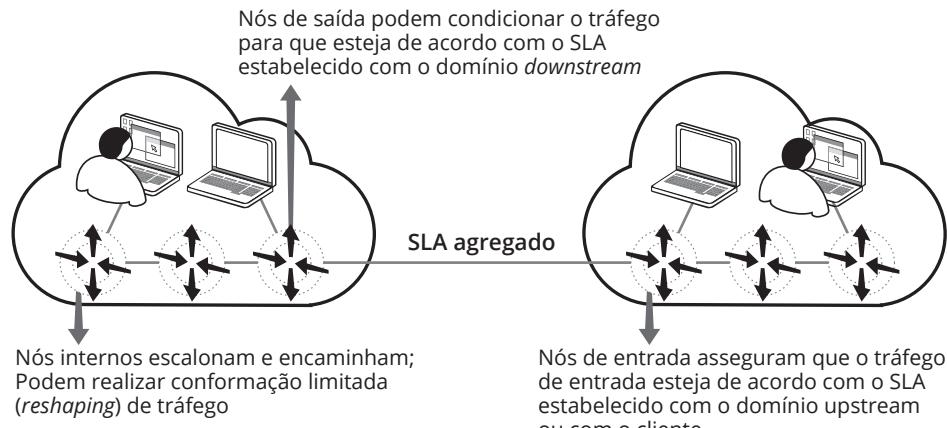
- Classificado;
- Possivelmente condicionado na borda da rede;
- Atribuído a diferentes agregações de comportamento.

Cada agregação de comportamento é definida por um único DS field.

Dentro do core da rede, os pacotes são encaminhados de acordo com o per-hop behavior associado com o DS field.

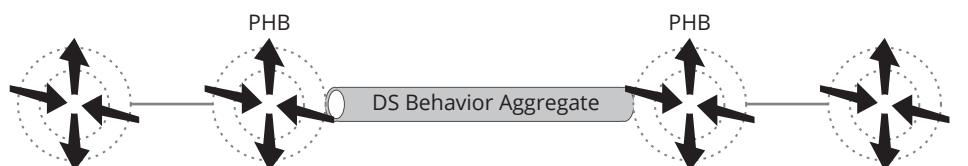


A arquitetura de serviços diferenciados é baseada em um modelo simples no qual o tráfego que entra na rede é classificado, possivelmente condicionado na borda da rede e atribuído a diferentes agregações de comportamento. Cada agregação de comportamento é definida por um único DS codepoint. Dentro do core da rede, os pacotes são encaminhados de acordo com o per-hop behavior associado com o DS codepoint.



**Figura 8.3**  
Arquitetura física de Serviços Diferenciados.

Essa arquitetura é composta por vários elementos funcionais implementados em nodos de rede, incluindo um pequeno conjunto de per-hop behaviors, funções de classificação de pacotes e funções de condicionamento de tráfego incluindo: medição, marcação, conformação e policiamento. Essa arquitetura alcança escalabilidade implementando classificação complexa e funções de condicionamento apenas nos nodos de borda de rede, e implementando per-hop behaviors para agregações de tráfego as quais são apropriadamente marcadas usando o campo DS. O per-hop behaviors (PHB) é definido para permitir uma média granular razoável de alocação de buffer e recursos de largura de banda em cada nodo.



**Figura 8.4**  
Comportamento de encaminhamento.

Tratamento de encaminhamento que os pacotes recebem nos roteadores:

#### **PHB EF (Expedited Forwarding)**

Encaminhamento expresso (acelerado).

Baixa perda, retardo e variação do retardo (jitter).

Preferência total de encaminhamento.

DSCP = 101110

#### **PHB AF (Assured Forwarding)**

Grupo de PHBs de encaminhamento assegurado.

Quatro classes de serviços com três níveis de descarte.

Define tratamentos diferenciados aos pacotes, do tipo “melhor que o melhor esforço”.

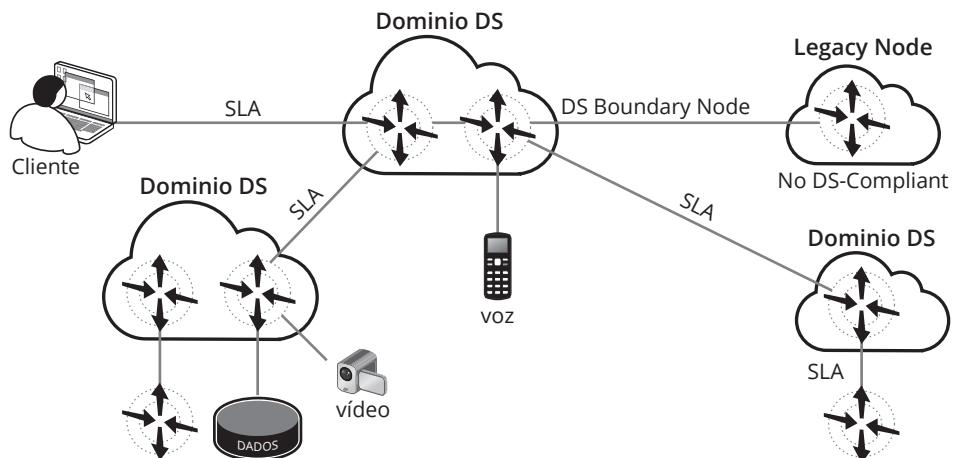


Assim, o PHB é responsável pelo tratamento de encaminhamento que os pacotes recebem nos roteadores. Há dois tipos básicos de PHBs (Nichols 98):

- PHB EF (expedited Forwarding):
  - Encaminhamento expresso (acelerado);
  - Pouca perda, retardo e variação de retardo (jitter);
  - Preferência total de encaminhamento.
- PHB AF (Assured Forwarding):
  - Grupo de PHBs de encaminhamento assegurado;
  - Quatro classes de serviços com três níveis de descarte;
  - Define tratamento diferenciado aos pacotes, do tipo “melhor que o melhor esforço: BBE”.

Esses serviços são entregues após uma negociação de contrato entre o provedor e o cliente, respeitando os serviços a serem providos. Esse contrato é conhecido como Service Level Agreements (SLAs).

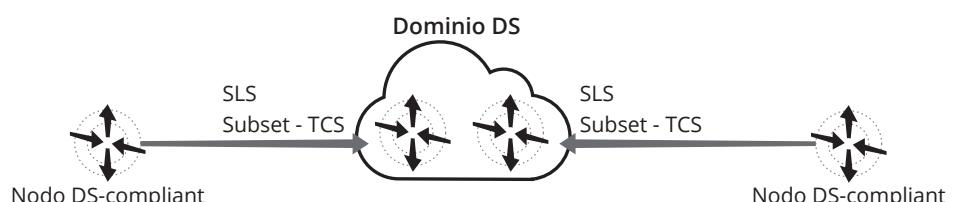
Um domínio DS é um conjunto contíguo de nodos DS os quais operam com um serviço de provisionamento de policiamento comum e um conjunto de grupos de PHB implementados em cada nodo. Um domínio DS normalmente consiste de uma ou mais redes dentro de uma mesma administração. A administração de um domínio tem a responsabilidade de assegurar que recursos adequados sejam provisionados e/ou reservados para suportar o SLA oferecido pelo domínio.



**Figura 8.5**  
Arquitetura  
lógica de Serviços  
Diferenciados.

### SLSs e TCSs

Para cada serviço, diferentes aspectos técnicos do serviço a ser provido são definidos em forma de um Service Level Specification (SLS), que especifica todas as características e a performance esperadas pelo cliente. Devido ao fato de os serviços DS serem unidirecionais, as duas direções de fluxo devem ser consideradas separadamente. Um subset importante do SLS é o “Traffic Conditioning Specification – ou TCS.



**Figura 8.6**  
TCS Traffic  
Conditioning  
Specification.



O TCS especifica perfis de tráfego e ações para pacotes dentro do perfil (in-profile) e fora do perfil (out-of-profile). Perfis de tráfego são responsáveis por especificar regras para classificar e medir um fluxo, identificar quais são elegíveis e definir regras para determinar se um pacote está dentro ou fora do perfil. Um pacote dentro do perfil pode ser adicionado a uma agregação de comportamento diretamente, enquanto um pacote fora do perfil pode ser conformado antes da entrega da seguinte forma:

- Pode ser atrasado até que esteja dentro do perfil;
- Pode ser descartado.

Entre os parâmetros de serviço para cada nível de serviço que o TCS especifica, temos:

- Parâmetros de performance, tais como: throughput, probabilidade de descarte e latência;
- Indicação do escopo de cada serviço nos pontos de ingresso e saída;
- Perfis de tráfego;
- Disposição do tráfego submetido em excesso ao perfil especificado;
- Marcação do serviço proporcionado;
- Conformação do serviço proporcionado.

### Serviços quantitativos e qualitativos

A arquitetura de Serviços Diferenciados pode suportar uma grande variedade de diferentes tipos de serviço. Classificar esses serviços significa associar um SLS a um serviço respectivo. Alguns serviços podem ser claramente classificados como qualitativos ou quantitativos, dependendo do tipo de parâmetros de performance oferecidos. Serviços qualitativos são aqueles que oferecem garantias relativas que somente podem ser avaliadas por comparação. Como exemplo de serviços qualitativos, temos:

- O tráfego oferecido no nível de serviço A será entregue com baixa latência;
- O tráfego oferecido no nível de serviço B será entregue com baixa perda.

Serviços quantitativos são aqueles que oferecem garantias concretas que podem ser avaliadas por medições convenientes, independentes de outros serviços. Como exemplos de serviços quantitativos, temos:

- 90% do tráfego entregue dentro do perfil no nível de serviço C não terá mais do que 50ms de latência;
- 95% do tráfego entregue dentro do perfil no nível D será efetivamente entregue.

Como serviços que possuem quantificação relativa, temos:

- O tráfego oferecido no nível de serviço E terá duas vezes mais banda do que o nível F;
- O tráfego com drop precedence AF12 tem uma prioridade de entrega maior de que o tráfego com drop precedence AF13.

De uma forma geral, quando um provedor oferece um serviço quantitativo, será necessário especificar perfis de policiamento quantitativo.

### SLS dinâmico vs. estático

Os SLSs podem ser estáticos ou dinâmicos. Os SLSs estáticos são a norma atualmente, e são um resultado da negociação entre o provedor e o cliente. Um SLS estático é definido por um acordo de data de início e pode ser periodicamente renegociado (em ordem de dias, semanas ou meses). Todavia, o SLS pode especificar que o nível de serviço mude em certas horas ao dia ou certos dias na semana, mas o contrato permanece estático.



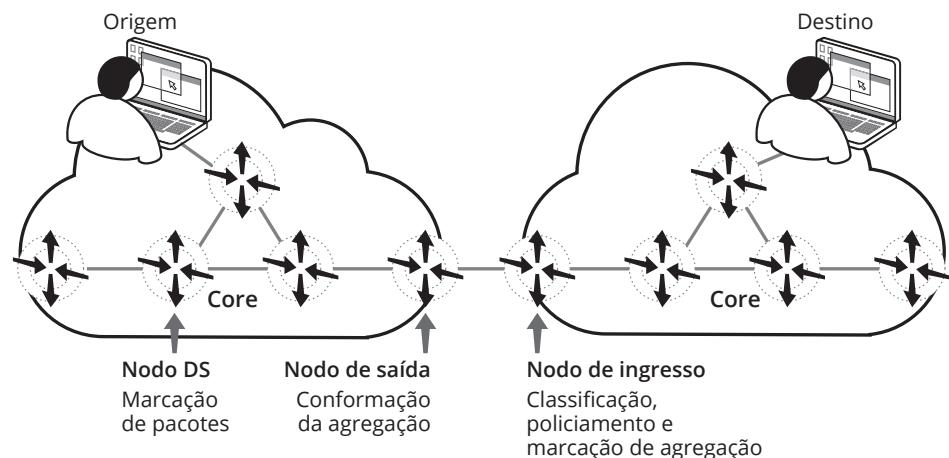
O SLS dinâmico, ao contrário, pode mudar frequentemente. Tais mudanças podem resultar, por exemplo, de variações na carga de tráfego oferecida, relativa a thresholds ou de mudanças no preço oferecida pelo provedor. Os SLSs dinâmicos mudam sem intervenção humana e requerem protocolos automatizados.

### Condições de provisionamento de tráfego em dispositivos de borda para provedores de serviços

Uma vez que um SLS tenha sido negociado, o provedor de serviço (e opcionalmente o cliente) vai configurar componentes de condicionamento de tráfego no limite das duas redes. Assim, o provedor de serviço tem como obrigação garantir os recursos ao cliente, porém sem que os recursos excedam os termos do TCS. E o cliente, ao contrário, tem como objetivo fazer o melhor uso do serviço adquirido do provedor. O tráfego do cliente talvez seja autenticado por conexões físicas através das quais ele chega ou por sofisticada criptografia.

Os quatro componentes de condicionamento de tráfego são:

- Medidor;
- Marcador;
- Conformador;
- Descartador.



**Figura 8.7**  
Componentes do condicionamento.

A combinação e interação de componentes de condicionamento de tráfego são selecionadas em bases de pacotes por pacotes pelo DS codepoint. Os parâmetros de configuração para componentes em cada codepoint são determinados por policiamento e perfis aplicados; dessa forma, o condicionador policia o tráfego baseado no Behavior Aggregate (BA) especificado pelo codepoint. Os medidores medem o tráfego submetido de acordo com o perfil de tráfego contratado (TCS), provendo controle de entrada para os outros componentes os quais implementam o policiamento:

- Os conformadores policiam atrasando alguns ou todos os pacotes de uma sequência de tráfego, de modo a levar o fluxo a tornar-se complacente com o perfil de tráfego. Um conformador tem geralmente um tamanho de buffer finito e pacotes podem ser descartados se não houver espaço de buffer para assegurar o atraso dos pacotes;
- Os descartadores policiam, descartando alguns ou todos os pacotes de uma sequência de tráfego, de modo a levar o fluxo a tornar-se complacente com o perfil de tráfego. Esse processo é conhecido como policiamento de fluxo. Note que o descartador pode ser implementado como um caso especial de um conformador, setando o tamanho do buffer para zero (ou quase);



- Os marcadores policiam o tráfego remarcando o tráfego com um codepoint particular, somando o pacote a um comportamento DS particular. Isso ocorre:
  - Mapeando codepoint/PHB específico do domínio;
  - Rebaixando o fluxo fora do perfil de tráfego.

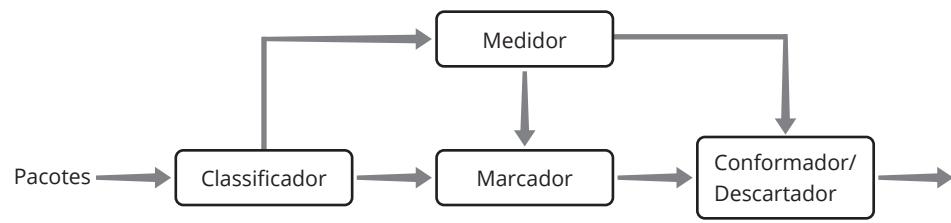
Em soma a esses quatro componentes, classificadores de tráfego são requisitados para separar o tráfego submetido dentro de diferentes classes. Os classificadores podem separar o tráfego baseado no campo DS ou podem fazê-lo baseado em múltiplos campos dentro do cabeçalho do pacote e até a partir do payload do pacote. Na maioria da vezes, o tráfego vai chegar ao limite de um DS-domain pré-marcado e pré-modelado. Mesmo que o tráfego do cliente venha pré-marcado e pré-modelado, o provedor de serviço poderá realizar policiamento de tráfego no limite do ponto de ingresso, de modo a atender os próprios interesses do domínio. Isso pode resultar em que o tráfego seja remarcado ou descartado.

Um condicionador de tráfego inclui:

- Classificador;
- Medidor;
- Marcador;
- Suavizador;



Condicionadores de tráfego podem ser encontrados dentro de um domínio DS, na borda de um domínio. Nem todos os quatro elementos do condicionamento precisam estar presentes em todos os nós de borda. Uma visão lógica de condicionamento de tráfego é mostrada na figura a seguir:



**Figura 8.8**  
Visão lógica de um classificador e condicionador de tráfego.

Note que o condicionamento de tráfego pode não necessariamente conter todos os quatro elementos. Por exemplo, em casos onde não há um profile de tráfego, pacotes podem apenas passar por um classificador e um marcador.

- Medidor (meter):** mede o fluxo para verificar se está de acordo com o perfil de tráfego contratado (TCS).



O provedor deve provisionar nodos internos na sua rede, de modo a atender as garantias oferecidas pelos SLSS negociados no limite da rede. Para fazer isso, o provedor pode usar mecanismos de condicionamento de tráfego similares aos usados no limite da rede. O provedor pode policiar periodicamente dentro da rede, por remodelagem, remarcção ou descarte de tráfego.

A arquitetura de serviços diferenciados propõe que um serviço fim a fim pode ser construído pela concatenação de serviços de domínios e SLAs associados ao cliente-provedor para cada um dos domínios onde o tráfego venha a passar.

## Tipos de serviços

Todo tipo de transmissão pode ser tratada como dados. Uma vez que um sinal analógico é convertido para um sinal digital, ele pode ser tratado como se fosse um pedaço de dados. Entretanto, diferentes tipos de transmissão podem possuir diferentes tipos de requisitos.

Tanto voz como transmissões de vídeo de baixa qualidade apresentam alta tolerância a erros. Se um pacote ocasionalmente é descartado, a fidelidade na reprodução de voz e vídeo não será severamente afetada. Em contraste, pacotes de dados têm baixa tolerância a erros. Um bit errado pode mudar o significado dos dados.

Transmissão de voz, vídeo e dados também têm diferentes requisitos em relação a atrasos. Para que uma voz, que foi encapsulada em um pacote, possa ser traduzida para um sinal analógico, o atraso de rede para esses pacotes devem ser constantes e baixos. No caso de pacotes de dados, o atraso de rede pode variar consideravelmente. Pacotes de dados podem ser transmitidos de forma assíncrona através da rede, sem se importar com o tempo entre o emissor e o receptor. Em contraste, a transmissão de vídeo deve possuir uma relação de tempo entre o emissor e o receptor.

Pacotes de vídeo e voz, ocasionalmente, podem ser perdidos ou descartados. Em casos de eventos de excessivo atraso na rede, os pacotes podem ser descartados porque já não possuem utilidade. Essa perda não afeta severamente a fidelidade da voz, se a perda de pacotes for menor que 1% do total de pacotes transmitidos.

A transmissão de voz e vídeo também requer um tamanho de fila pequeno nos nodos da rede, de modo a reduzir o atraso e torná-lo previsível. Um tamanho de fila de pacotes de voz pequeno pode prevenir um overflow ocasional, o qual poderia resultar na perda de pacotes. Entretanto, pacotes de dados requerem uma fila de tamanho grande, de modo a prevenir que pacotes possam ser perdidos em condições de overflow.

A seguir serão descritos exemplos de serviços e como eles podem ser suportados por específicos PHBs. Lembremos que tais exemplos têm caráter tão-somente ilustrativo, em se considerando a grande quantidade de serviços que podem ser empregados usando o modelo de serviços diferenciados.

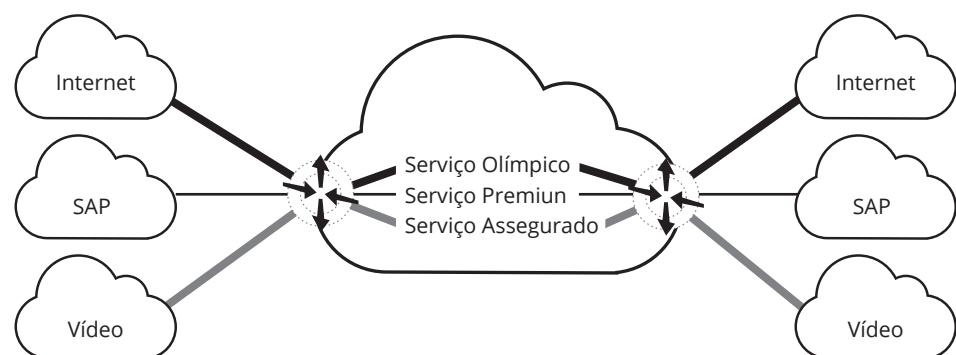


Figura 8.9  
Tipos de serviços diferenciados.

### Serviço melhor que best-effort

Esse é um serviço quantitativo que promete transportar tráfego de webservers em uma prioridade maior que a usada no método best-effort. Esse tipo de serviço oferece perda de performance (não quantificada) relativa de um dado ponto de ingresso a qualquer outro ponto de saída. Também é conhecido como serviço do tipo Olímpico, no qual o contrato refere-se ao serviço “melhor”, relativo a quem paga menos. Tem as seguintes características:



- ▣ O PHB nesse caso descarta as classes inferiores primeiro (AF);
- ▣ Tem como regra de policiamento descartar ou remarcar pacotes fora de perfil;
- ▣ Também é chamado de Classe of Service;
- ▣ Tem como classes “olímpicas” de serviço de melhor esforço:
  - ▣ Ouro;
  - ▣ Prata;
  - ▣ Bronze.

Os pacotes submetidos para o serviço BBE devem ser marcados com o codepoint do campo DS correspondente a AF11 PHB. O provedor tem a promessa de transportar o tráfego de 1 Mbps do ponto de ingresso para qualquer ponto de saída a uma prioridade maior que a do best-effort. Uma classe de serviço menor, correspondente a AF13 PHB, será aplicada ao tráfego submetido pela AF11 PHB, que exceder 1 Mbps.

O provedor tem de prover um policiamento no ponto de ingresso. O tráfego submetido até o limite de 1 Mbps será direcionado ao AF11 PHB. O tráfego submetido em excesso ao 1 Mbps será remarcado para o AF13 PHB. Note que o esquema será preservado ordenando os pacotes desde que a AF11 e a AF13 usem uma única fila.

De modo a prover esse serviço, o provedor terá de implementar a AF11 PHB e a AF13 PHB no equipamento do core da rede, que também deverá ser devidamente provisionado para recebê-las.

A AF11 PHB e a AF13 PHB podem ser implementadas, por exemplo, usando uma fila do tipo RIO (Red with In or Out). Provisionando parâmetros do tipo RED, por exemplo, o provedor está apto a controlar a prioridade do tráfego AF11 relativo ao tráfego AF13 em cada nodo da rede.

### Serviço de emulação de linhas privadas

Esse é um serviço quantitativo que emula o serviço de linhas privadas tradicionais. Ele promete entregar o tráfego do cliente com pouca latência e pouquíssima probabilidade de descarte, até a taxa negociada. Acima dessa taxa, o tráfego é descartado. Esse tipo de serviço é encontrado tipicamente entre dois pontos específicos. Ele se encaixa para muitas aplicações de clientes. Entretanto, devido à alta garantia de qualidade, ele acaba tendo um preço mais alto que serviços alternativos. Assim, ele acaba sendo utilizado apenas por aplicações que necessitam realmente desse tipo de serviço. Um exemplo de aplicação é a telefonia IP.

Esse serviço também é conhecido como serviço do tipo Premium, no qual o contrato se refere à emulação de linha dedicada a uma taxa de pico específica, e tem as seguintes características:

- ▣ O PHB nesse caso tem como regra encaminhar o pacote primeiro (EF);
- ▣ Tem como regra de policiamento o descarte de pacotes fora de perfil;
- ▣ Na saída, os domínios devem conformar agregações EF para mascarar rajadas.

Como um exemplo, considere-se um cliente com três redes geograficamente dispersas, interconectadas via um único provedor de rede. Os pontos de conexão do cliente serão identificados como A, B e C. Em cada ponto conectado, um SLS descreve o serviço de linha privada a ser provido aos outros pontos. A informação a seguir representa a informação requisitada no TCS da conexão do ponto A:

- ▣ **EF-Mark: 100 Kbps: ponto de saída B:** descarte de tráfego não conforme;
- ▣ **EF-Mark: 50 Kbps: ponto de saída C:** descarte de tráfego não conforme.



Os pacotes submetidos pelo serviço de linha privada devem ser marcados com o codepoint no campo DS correspondente a EF PHB [EF]. Do ponto de ingresso A para o ponto de saída B, o provedor promete transportar até 100 kbps de tráfego. O tráfego excedente será descartado. Do ponto de ingresso A, para o ponto de saída C, o provedor promete transportar 50 Kbps de tráfego. É claro, existem algumas tolerâncias requisitadas em policiamento de tráfego, como jitter e tamanho de rajada. Entretanto, para serviços de linha privada, o primeiro parâmetro de perfil de tráfego pode ser o sustained traffic rate.

O provedor provisionará policiamento no ponto de ingresso A para limitar o tráfego destinado ao ponto de saída B a 100 Kbps. Similarmente, um policiamento será configurado, de modo a limitar o tráfego destinado ao ponto de saída C a 50 Kbps. Esses policiamentos requerem classificação baseada no DS-mark e o endereço de destino em cada pacote.

A fim de prover esse serviço, o provedor terá de implementar a EF PHB no equipamento do core da rede. A EF PHB pode ser implementada usando “strict priority queuing” ou, alternativamente, aplicando pacotes marcados com EF no esquema WFQ (heavily weighted queue). O provedor terá de provisionar equipamentos no core da sua rede. Por exemplo, roteadores transportando tráfego entre o ponto A e ponto B e/ou C terão que ser provisionados considerando-se os recursos comprometidos pelo TCS no ponto A. Isso significa que um roteador o qual essa no caminho de A e B e de A e C, terá de ser considerado como tendo comprometido 150 Kbps de sua largura de banda como resultado do TCS colocado em A. Um roteador, apenas no caminho entre A e B, terá de ser considerado como tendo um comprometimento de 100 Kbps como resultado do TCS. É claro, o roteamento está sujeito a mudar, falhas nos caminhos podem também ser provisionadas. Para aumentar a segurança oferecida pelos serviços EF, os provedores podem empregar mecanismos de roteamento, como: route pinning mechanisms ou QoS routing mechanisms.

### Serviço quantitativo assegurado: Media Playback

Esse serviço oferece menor garantia que o serviço de linha privada recém-descrito, mas ele ainda é considerado um serviço quantitativo. Em particular, ele promete entrega de tráfego com alto grau de confiança e com latência variável, porém limitada, até a taxa negociada. Acima dessa taxa, o tráfego é sujeito a um atraso ou descarte significativo. Esse tipo de serviço é tipicamente oferecido entre um conjunto específico de pontos e é empregado em muitas aplicações de clientes.

Devido à sua variação de latência, ele acaba saindo mais em conta que o serviço de linha privada. Entretanto, devido ao seu limite de latência e alto grau de entrega, ele acaba tendo preço maior que outros serviços alternativos. Tal serviço é destinado particularmente a playback de áudio ou vídeo, no qual uma largura de banda considerável é necessária em bases contínuas, mas a natureza não interativa do tráfego torna-o um pouco tolerante a atrasos.

Também é conhecido como serviço do tipo Assegurado, no qual o contrato afirma que a rede parece estar “levemente carregada” para tráfego em perfil especificado (taxa e rajada). Tem as seguintes características:

- ▣ O PHB nesse caso descarta por último (AF);
- ▣ Tem como regra de policiamento remarcar pacotes fora do perfil para que tenham uma probabilidade de descarte maior;
- ▣ O tráfego em uma classe compartilha fila única;
- ▣ Na saída, os domínios podem também visualizar agregações AF.



Os pacotes submetidos ao serviço de playback confiável devem ser marcados com o code-point do campo DS correspondendo a AF11 PHB. Do ponto de ingresso A para o ponto de saída B, o provedor promete um transporte até 100 Kbps do tráfego tolerado (sustained traffic) com rajadas (burst) de 100 Kbps de tamanho e taxa de pico de 200 Kbps. Rajadas de tráfego excedentes serão marcadas com o codepoint AF12 e o tráfego fora do perfil será transportado com o codepoint AF13. Tão logo essas condições sejam encontradas, a latência será limitada a um segundo. Note-se que para esse serviço o perfil de tráfego é descrito usando um conjunto completo de parâmetros de token bucket. Uma vez que o limite de latência para tal serviço é menos rigoroso que no serviço de linhas privadas, um certo grau de traffic burstiness pode ser tolerado.

O provedor deve suportar as AF11, AF12 e AF13 PHBs nos roteadores do core da rede. Essas PHBs podem ser providas, por exemplo, direcionando o tráfego marcado com AF11, AF12, AF13 para uma única fila RIO com alto limite de descarte. Os policiais na borda limitarão a competição de tráfego na linha com o TCS, de modo a assegurar que a latência possa ser encontrada. O provedor de serviço terá de provisionar dispositivos no core da rede.

O provisionamento discutido em linhas privadas pode ser aplicado aqui, entretanto, em geral, o provedor de serviço tem a liberdade de ser menos conservativo no provisionamento e realizar melhores ganhos estatísticos.

## Provisionamento e configuração

O provisionamento de serviços diferenciados requer provisionamento e configuração cuidadosa, e refere-se à determinação e alocação de recursos necessários em vários pontos na rede. O provisionamento pode:

- Ditar a soma ou a remoção de recursos físicos em vários pontos (provisionamento físico);
- Definir a modificação de parâmetros operacionais dentro de equipamentos existentes na rede, de modo a alterar relativos compartilhamentos de recursos de rede os quais são alocados a uma ou outra classe de serviço (provisionamento lógico).

A configuração refere-se à distribuição de parâmetros operacionais apropriados para equipamentos de rede, de modo a alcançar objetivos de provisionamento. A configuração pode ser feita utilizando protocolos como SNMP, CLI, RSVP, COPS e LDAP.

### Provisionamento e configuração: Borda vs. Interior

- Medidor, suavizador e descartador.
- Classificador e marcador.
- Mecanismos que permitem ler e escrever conteúdo de campos.



De modo a ser breve, consideremos o termo “provisionamento” como referência a provisionamento e configuração. O importante notar aqui é que provisionamento na borda da rede deve ser tratado separadamente de provisionamento no interior da rede. Desde que o provedor de serviços diferenciados vende um contrato (SLA) na borda da rede, podemos considerar o provisionamento de borda, o qual suporta SLSs, como sendo o responsável em determinar o provisionamento do interior do provedor. Por exemplo, um operador de rede não pode oferecer um SLS o qual não pode localizar recursos disponíveis no interior da rede. De uma forma geral, o processo geral de provisionamento interage entre bordas e interior. De agora em diante, referenciaremos o provisionamento em respeito a TCS em vez de SLS, já que o TCS é um componente do SLS que define detalhes de parâmetros de manipulação de tráfego.



## Provisionamento de borda

No mínimo, o provedor deve assegurar que recursos físicos suficientes estejam provisionados na borda de modo a poder encontrar os requisitos do TCS. Por exemplo, se a soma dos perfis suportados em um ponto de ingresso permitir 10 Mbps de tráfego, é inaceitável provisionar o link com um acesso T1. Um T3, entretanto, seria suficiente. Uma vez que o provisionamento físico é implementado, é necessário aplicar o provisionamento lógico apropriado. Isso é alcançado via configuração de policiamento que limita a quantidade de tráfego aceito pelo link T3, em cada nível de acesso e para duplos TCSs finais, para o ponto de saída apropriado.

Também pode ser necessário configurar uma quantidade de buffer para as filas usadas para o serviço. O provisionamento similar é também apropriado em cada ponto de saída, se o agregado do perfil provisionado para a saída exceder a capacidade de saída do link.

### Distribuindo informações de configuração

- Classificador.
- Controle de congestionamento.
- Mecanismos de escalonamento de pacotes (disciplinas de serviço).
- Ou seja, enfileiramento.
- Prevenção de congestionamento.
- Técnicas para evitar transbordamento das filas.
- Influência do controle de congestionamento do protocolo TCP (fonte cooperante).



O processo de provisionamento físico é, por necessidade, relativamente estático e não pode ser automatizado, desde que requeira instalação de equipamentos físicos. Entretanto, o provisionamento lógico e configurações podem e devem ser automatizados. Nesta sessão, abordaremos técnicas de distribuição de informações de configuração.

No caso mais simples, os TCSs são estáticos e as bordas e o interior da rede são provisionados estaticamente através do processo de envio da informação de configuração para o nodo de rede apropriado. A configuração dos nodos de borda requer primeiro o envio da informação de policiamento. Nesse momento, os nodos são configurados pelo provedor. O administrador de rede pode usar um dos vários protocolos para fazer isto, incluindo SNMP ou CLI.

De modo a acomodar o tráfego submetido pelo provisionamento de um novo TCS, é necessário provisionar o interior da rede. Nesse caso de configuração top down, as informações de configuração de interior são também enviadas via protocolo de configuração, tal como SMNP ou CLI.

Servidores de policiamento podem ser usados para extrair informações de base de dados e para convertê-las em informações de configuração, as quais são enviadas para nodos individuais.

Nesse cenário, os servidores de policiamento poderiam utilizar protocolos do tipo directory access protocol, tal como LDAP, para buscar informações do diretório e usar um protocolo de configuração como SMNP ou CLI para push down a informação de configuração para o nodo da rede.

### Modificações de base de informações de configuração em medições de tempo real

Um terceiro mecanismo para a configuração de nodos interiores poderia ser baseado em medidas da carga do tráfego corrente nos nodos chaves da rede. A configuração baseada em medidas é mesmo necessária para provisionamento quantitativo, desde que os padrões



de tráfego quantitativo sejam relativamente previstos. Entretanto, ele pode aumentar significativamente a eficiência com a qual o provisionamento qualitativo pode ser alcançado.

O objetivo final em relação a QoS é prover a usuários e aplicações alta qualidade na entrega de serviços de dados. No ponto de vista do roteador, o suporte à qualidade de serviço é dividida em três partes: definição de classes de tratamento de pacotes, especificação da quantidade de recursos para cada classe e classificação de todos os pacotes de entrada da rede dentro de suas classes correspondentes. O modelo DiffServ especifica a primeira e a terceira parte: ele especifica classes de tráfego, bem como provê um mecanismo simples de classificação de pacotes. Já o modelo Bandwidth Broker (BB) especifica a segunda parte, mantendo a informação de alocação atual do tráfego marcado e interpretando novas requisições.

O BB tem responsabilidades internas e externas referentes a gerenciamento de recursos e controle de tráfego. Internamente, um BB pode manter informações de requisições de QoS de usuários individuais e aplicações, e alocar recursos internos de acordo com regras de policiamento usadas para recursos específicos dentro do domínio. Externamente, o BB tem responsabilidades de configurar e manter acordos bilaterais de serviços com os BBs de domínios vizinhos de modo a assegurar a manipulação de QoS do tráfego de dados entre as bordas.

O Bandwidth Broker (BB) é um agente responsável pela alocação de serviços preferenciais para usuários no momento da requisição, e por configurar os roteadores da rede com o comportamento de entrega correto para o serviço definido. Um BB está associado a uma região de confiança particular, um por domínio; tem uma base de dados para policiamento que mantém as informações de quem pode fazer o quê, quando e um método de utilizar a base de dados para autenticação de requisições. Apenas o BB pode configurar o roteador folha para entregar um serviço particular para um fluxo, crucial para o desenvolvimento de um sistema seguro.

Quando uma alocação é desejada para um fluxo particular, uma requisição é enviada para o BB. A requisição inclui o tipo de serviço, a taxa-destino, a rajada máxima e o período de tempo que o serviço será utilizado. A requisição pode ser realizada por um usuário ou ela pode vir de outras regiões de BB. Um BB autentica em primeiro as credenciais do requisição, então, verifica se existe largura de banda disponível suficiente para a requisição. Se a requisição passa por esse teste, a largura de banda disponível é reduzida pela quantidade requisitada e a especificação do fluxo é registrada.

O BB configura o roteador-folha com informações sobre o fluxo de pacotes a ser dado ao serviço no momento que este se inicia.

A ideia do BB foi introduzida como parte da arquitetura de Serviços Diferenciados. O BB está diretamente envolvido com a administração do gerenciamento de recursos de serviços diferenciados. Dois aspectos importantes são:

- Gerenciamento de recursos entre domínios;
- Gerenciamento de recursos dentro do domínio.

## Service Level Agreements

O Service Level Agreements (SLA) provê um mecanismo simples para alocação de blocos de um serviço em particular para um cliente específico. Essa facilidade reserva banda para usuários ou organizações, mas não refere-se à alocação para fluxos específicos. Um SLA é válido para fluxo dentro de uma única região. Uma vez que o SLA tenha sido estabelecido,

porções desse serviço podem ser associadas a fluxos específicos. O SLA inclui as seguintes informações:

- Identificação do cliente;
- Tipo de serviço;
- Parâmetros de tipo de serviço;
- Restrições do serviço.

O Bandwidth Broker assegura que todas as obrigações para qualquer serviço não excede a quantidade desse serviço disponível na região de confiança.

### Requisição de alocação de banda

As requisições de alocação de banda são usadas por clientes de modo a requisitar porções de um serviço alocado por um SLA, para fluxos individuais.

Esses fluxos individuais são descritos por informações de origem/destino/protocolo, além da informação da taxa. O pedido de requisição de banda contém:

- Identificador de usuário;
- O ID do SLA para negociar o SLA;
- Parâmetros de nível de serviço (taxa, rajada máxima etc.);
- Identificador fonte (número da porta, endereço IP e protocolo);
- Destino (número da porta, endereço IP e protocolo);
- Duração da requisição.

O Bandwidth Broker, antes de permitir a requisição, garante que a alocação não vai violar os limites do SLA e também não excederá a quantidade do agregado do serviço na região de confiança.

### Configuração do roteador

O Bandwidth Broker tem de configurar um grupo de roteadores com capacidades de Diff-Serv, de modo a prover o nível desejado de serviço dentro da região.

O BB configura os roteadores-folhas e de saída no seu domínio, de acordo com o Service Level Agreements e a requisição de alocação de banda de entrada para os clientes. Depois da verificação e validação de um SLA, o roteador de saída apropriado é conectado e parâmetros requeridos para a configuração do serviço particular são enviados. Similarmente, na validação de um BAR, o roteador-folha mencionado no BAR é conectado e os parâmetros necessários para marcação e policiamento do fluxo são passados a ele.

### Mecanismos para implementar QoS

Esta sessão tem como objetivo apresentar os mecanismos já desenvolvidos pelo fornecedor Cisco Systems, no que condiz à QoS. São eles:

- Classificação;
- Gerenciamento de congestionamento;
- Mecanismos para evitar congestionamento;
- Policiamento e conformação;
- Sinalização;
- Mecanismos de eficiência de link.



## Classificação

A classificação utiliza um descritor de tráfego para categorizar um pacote dentro de um grupo específico, e para definir o pacote e marcá-lo como acessível para manipulação de QoS na rede. Usando classificação de pacotes, podemos partitionar o tráfego da rede dentro de múltiplos níveis de prioridade ou classes de serviços. Quando os descritores de tráfego são usados para classificar tráfego, a origem concorda seguir os termos do contrato e a rede promete qualidade de serviço. O policiamento de tráfego, tal como a característica de limite de taxa do Committed Access Rate (CAR) e conformação de tráfego, além de Frame Relay Traffic Shaping (FRTS) e Generic Traffic Shaping (GTS), usam um descritor de tráfego de pacotes – classificação – para assegurar o contrato (Cisco 99b).

A classificação de pacotes é primordial para técnicas de policiamento que selecionam pacotes que cruzam elementos de rede ou uma interface particular para diferentes tipos de serviços QoS. Os métodos antigos de classificação eram limitados ao conteúdo do cabeçalho do pacote. Os métodos atuais de marcação de pacote para classificação permitem configurar informações em cabeçalhos de nível 2, 3 ou 4, ou até configurar informações dentro do payload do pacote.

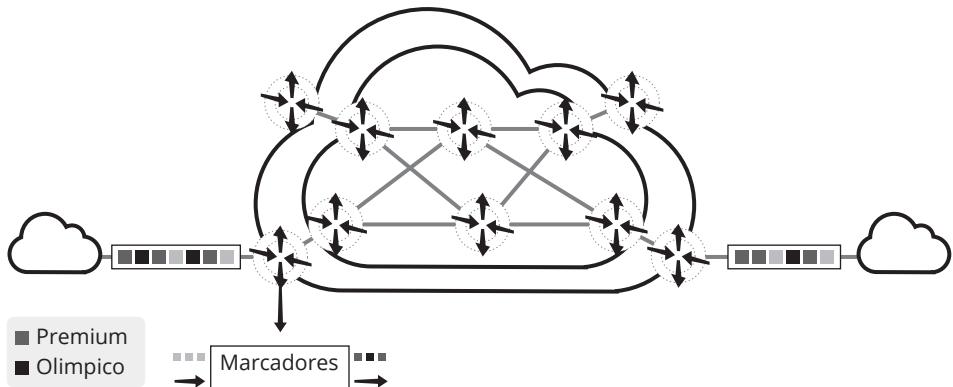
O uso do IP Precedence permite especificar a classe de serviço (CoS) para um pacote. São usados os três bits precedentes do campo ToS no cabeçalho IPv4 para esse propósito.



Figura 8.10  
Histórico do campo IP Precedence

Usando os bits ToS, podemos definir até seis classes de serviços. Outras características configuradas através da rede, podem então usar esses bits para determinar como tratar o pacote, em vez de considerar o tipo de serviço para garantí-lo. Essas outras características de QoS podem fornecer políticas apropriadas de manipulação de tráfego, incluindo estratégias de gerência de congestionamento e alocação de banda. Por exemplo, embora o IP Precedence não seja um método de enfileiramento, métodos de enfileiramento tais como Weighted Fair Queueing (WFQ) e Weighted Random Early Detection (WRED) podem usar o IP Precedence para configurar os pacotes para tráfego priorizado.





**Figura 8.11**  
IP Precedence.

Configurando níveis de precedência em tráfego entrante e usando-os em combinação com características de enfileiramento, podemos criar serviços diferenciados. Igualmente pode-se utilizar características tal como policy-based-routing (PBR) e CAR para configurar a precedência, baseada em classificação por lista de acesso.

Assim, cada elemento da rede pode prover serviços baseados em determinado policiamento - o IP Precedence é geralmente implementado o mais perto da borda da rede. Podemos pensar em IP Precedence com uma funcionalidade da borda que permite o core ou backbone – características de QoS, tal como WRED-, reenviar tráfego baseado em CoS. O IP Precedence pode também ser configurado no host ou na rede do cliente, mas essa configuração pode ser reescrita por policiamento dentro da rede.

As seguintes técnicas podem utilizar o IP Precedence para determinar como o tráfego deve ser tratado:

- Distributed Weighted Random Early Detection (Distributed-WRED);
- Weighted Fair Queueing (WFQ);
- Committed Access Rate (CAR).

Como os bits do IP Precedence são utilizados para classificar pacotes  
DS-Field.

- Pacotes são marcados para receber serviços diferenciados nos Domínios DS.
- Campo TOS do IPv4 ou Traffic Class do IPv6.
- codepoint (DSCP).
  - identifica o PHB (Per-Hop Behavior).

Podemos utilizar os três bits do campo ToS no cabeçalho IP para especificar o CoS dado para cada pacote, partitionar o tráfego em até seis classes – os dois restantes são reservados para uso interno – e então usar mapas de policiamento e lista de acesso para definir políticas de rede, em termos de manipulação de congestionamento e alocação de bandwidth para cada classe. Por razões históricas, cada precedente corresponde a um nome, os quais são definidos no RFC 791. A seguinte tabela lista os números e seus correspondentes nomes.



Número	Nome
0	Rotineira
1	Prioritária
2	Imediata
3	Flash
4	Flash override
5	Crítica
6	Internet
7	Network

**Tabela 8.1**  
Valores do IP Precedence.

Os bits 6 e 7 de IP Precedence são reservados para informações de controle, tal como atualizações de roteamento. Embora características de IP Precedence permitam flexibilidade considerável para dar precedência, é possível definir um mecanismo de classificação específico. Por exemplo, pode ser desejável atribuir precedência baseada em aplicações.

Por default, o valor do IP Precedence não é alterado, preservando-se o valor configurado no cabeçalho. Permite-se assim que os dispositivos da rede possam prover serviços baseados no valor configurado.

Essa política de policiamento segue o padrão, no qual o tráfego de rede deve ser agrupado dentro de vários tipos de serviços no perímetro da rede, e esses serviços devem ser implementados no core da rede. Os roteadores no core da rede podem utilizar os bits do IP Precedence, por exemplo, para determinar a ordem de transmissão, o seu descarte, e assim por diante.

Podemos utilizar um dos seguintes mecanismos para configurar IP Precedence nos pacotes:

- Policy-Based Routing;
- QoS Policy Propagation via Border Gateway Protocol (PB-BGP);
- Committed Access Rate (CAR).

Depois que um pacote tenha sido classificado, pode-se usar outro mecanismo, tal como CAR e WRED, para especificar e forçar policiamento.

## Policy-Based Routing

O PBR permite configurar IP Precedence, para especificar o caminho correto do tráfego, ou o caminho baseado em configurações de policiamento. O PBR permite:

- Classificar o tráfego baseado em critérios de lista de acesso, assim estabelecendo critérios de associação;
- Configurar IP Precedence, dando à rede a habilidade de habilitar diferentes classes de serviços;
- Rotear pacotes para especificar caminhos; rotear para permitir serviços QoS através da rede.

O policiamento pode ser baseado em endereço IP, número da porta, protocolo ou tamanho do pacote. Para um simples policiamento, podemos usar um desses descritores; para um policiamento complexo, podemos usar todos eles.





Por exemplo, a classificação de tráfego por PBR permite identificar tráfego por diferentes tipos de serviço na borda da rede e então implementar QoS definido por cada CoS no core da rede, usando técnicas de prioridade ou weighted fair queueing. Esse processo obviamente necessita classificação de tráfego, explicitamente em cada interface no core da rede.

Todos os pacotes recebidos, com PBR habilitado na interface, são passados através de filtros conhecidos como mapas de rotas. O mapa de rotas usado pelo PBR dita o policiamento, determinando para onde os pacotes devem ser encaminhados. Algumas aplicações ou tráfegos podem ser beneficiados por roteamento; por exemplo, poderíamos transferir registros de estoque para um escritório com alta largura de banda, enquanto transmitirmos aplicações rotineiras tal como e-mail, através de links de baixa velocidade.

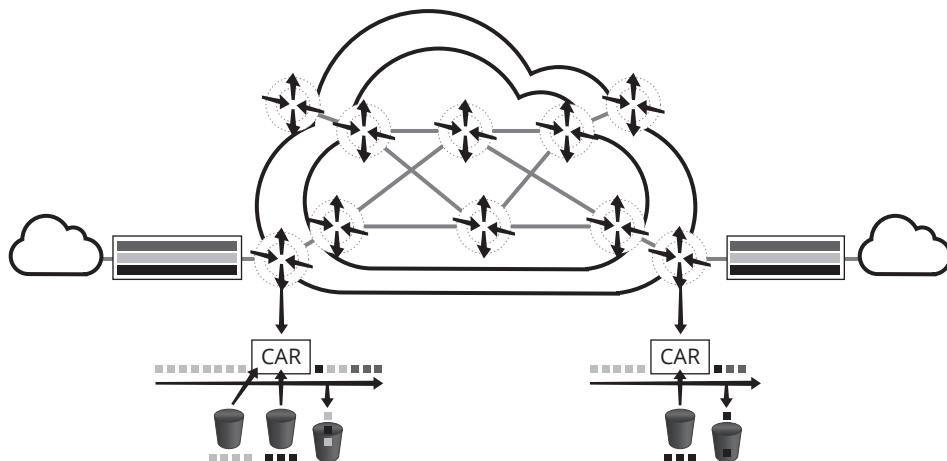
O Border Gateway Protocol (BGP) é um protocolo de roteamento entre domínios, que troca informações de roteamento com outros sistemas BGP (definido no RFC 1163). Políticas de propagação via BGP permitem classificar pacotes baseados em:

- Lista de acesso;
- Lista de community BGP;
- Caminhos de sistemas autônomos BGP;
- IP Precedence;
- Endereços de origem e destino.

Depois de o pacote ter sido classificado usando BGP, pode-se usar características de QoS, tal como CAR e WRED, para especificar policiamento compatíveis com o modelo do negócio da organização. O CAR é um mecanismo que implementa classificação de serviços e policiamento através de limites de taxa. Podemos utilizar serviços de classificação por CAR para configurar IP Precedence em pacotes que entram na rede. Essa característica do CAR permite particionar a rede em múltiplos níveis de prioridade ou classes de serviço. Dispositivos de rede dentro da rede podem utilizar o IP Precedence para determinar como o tráfego deve ser tratado. Depois do pacote ser classificado, a rede pode aceitar ou reescrever e reclassificar o pacote, de acordo com um policiamento específico.

O Committed Access Rate (CAR) tem como objetivo prover ao operador da rede a característica de gerenciamento da alocação da largura de banda, a qual foi determinada na criação da conexão. Entre as ações tomadas, pode ocorrer a mudança de classe (IP precedence) ou o descarte de pacotes (RED-like). Como exemplo de política aplicáveis pode-se citar:

- **Firm CAR:** pacotes que excedem a banda alocada são descartados;
- **CAR + Premium:** pacotes que excedem a banda alocada são “recoloridos” com alta ou baixa preferência;
- **CAR + Best Effort:** pacotes que excedem a banda alocada são “recoloridos” até o estouro do threshold, depois são descartados;
- **Per Application CAR:** diferentes CARs são especificados para diferentes aplicações.



**Figura 8.12**  
Committed  
Access Rate.

## Gerenciamento de congestionamento

As características de gerenciamento de congestionamento permitem o controle de congestionamento, pela determinação da ordem com que os pacotes são transmitidos para fora da interface, baseada em prioridades para esses pacotes. O gerenciamento de congestionamento está diretamente ligado à criação de filas; o direcionamento dos pacotes para essas filas baseia-se na classificação dos pacotes e na programação dos pacotes em uma fila para transmissão. A característica de gerenciamento de congestionamento oferece quatro tipos de protocolos, cada um dos quais permite que se especifique a criação de um número diferente de fila, de acordo com o tipo de tráfego e a ordem na qual os pacotes são transmitidos.

Durante períodos onde não existe congestionamento, os pacotes são transmitidos para fora da interface assim que chegam. Durante períodos de congestionamento, os pacotes chegam mais rápido que a interface de saída pode suportar. Se for utilizado o gerenciamento de congestionamento, os pacotes acumulados na interface são enfileirados até que a interface esteja livre novamente; o envio dos pacotes é então programado para transmissão, de acordo com sua prioridade, e o mecanismo de enfileiramento configurado para a interface. O roteador determina a ordem com que os pacotes são transmitidos, controlando quais pacotes são colocados nas filas e como filas são servidas com respeito as outras.

Existem esses quatro tipos básicos de enfileiramento, os quais constituem os mecanismos de gerenciamento de congestionamento para QoS:

- **First-In, First-Out Queueing (FIFO):** o método FIFO não utiliza o conceito de priorização ou classes de tráfego. Com FIFO, a transmissão de pacotes para fora da interface ocorre de acordo com a chegada destes;
- **Weighted Fair Queueing (WFQ):** o WFQ divide o bandwidth através de filas de tráfego baseado em pesos. O WFQ assegura que todo o tráfego é tratado de acordo com as regras, dado seu peso. Para ajudar a entender como o WFQ trabalha, considere uma fila para pacotes FTP como uma fila coletiva, e uma fila para tráfego de pacotes interativos como uma fila individual. Dado o peso das filas, o WFQ assegura que para todos os pacotes da fila coletiva transmitidos, um número igual de pacotes da fila individual é transmitido. Assim, o WFQ assegura, satisfatoriamente, o tempo de resposta a aplicações críticas, tal como as interativas, aplicações baseadas em transações que são intolerantes a degradações de performance;



- ▣ **Custom Queueing (CQ):** com CQ, o bandwidth é alocado proporcionalmente para cada classe de tráfego diferente. O CQ permite especificar o número de bytes ou pacotes para a fila. Nesse caso, o CQ é usado geralmente para interface de baixa velocidade;
- ▣ **Priority Queueing (PQ):** com PQ, pacotes com prioridade maior são enviados antes que todos os pacotes com prioridade menor, de modo a assegurar o tempo de entrega desses pacotes.

Atualmente, existe uma necessidade real de que o tráfego seja compartilhado entre aplicações, de modo que não venha a afetar a performance. Nesse caso, deve-se considerar cada vez mais o uso de técnicas de gerenciamento de congestionamento para assegurar o tratamento através dos vários tipos de tráfego. Isso é derivado de necessidades tais como:

- ▣ A priorização de tráfego é especialmente importante para aplicações sensíveis ao atraso e transações interativas – por exemplo, vídeo-conferência – que necessitam prioridade maior que aplicações de transferência de arquivos;
- ▣ A priorização é mais efetiva em links WAN nos quais a combinação entre tráfego em rajadas e taxas menores de dados podem causar congestionamentos temporários;
- ▣ Dependendo do tamanho médio dos pacotes, a priorização é mais efetiva quando aplicada a links T1/E1 ou menores;
- ▣ Se os usuários de aplicações que rodam através da rede identificam uma resposta pobre em relação ao tempo, deve ser considerada o uso de características de gerenciamento de congestionamento. Características de gerenciamento de congestionamento são dinâmicas, podendo se ajustar sozinha às condições existentes na rede. Entretanto, considerando que se um link WAN está constantemente congestionado, a priorização de tráfego pode não resolver o problema. A melhor solução seria aumentar o tamanho do link;
- ▣ Se não existe congestionamento no link WAN, não há razão para implementar priorização de tráfego.

## Mecanismos para evitar congestionamento

As técnicas para evitar congestionamento monitoram a carga do tráfego de rede, de modo a antecipar e evitar o congestionamento em épocas de gargalos de rede. Evitar congestionamento tem como base o descarte de pacotes. Entre as mais variadas técnicas de evitar congestionamento usadas, encontramos a Random Early Detection (RED), a qual se destina a redes de transmissão de alta velocidade.

Esta sessão disponibiliza uma descrição dos tipos de características para evitar congestionamento, tais como:

- ▣ **Tail Drop:** essa técnica é a padrão para evitar comportamentos de congestionamento;
- ▣ **Weighted Random Early Detection (WRED):** combina as características do algoritmo RED com IP Precedence.

### Tail Drop

O mecanismo de tail drop trata todo tráfego da mesma maneira e não faz diferenciação entre as classes de serviços. As filas são preenchidas em períodos de congestionamento. Quando a fila de saída é completada e o mecanismo de tail drop está em vigor, os pacotes são descartados até que o congestionamento seja eliminado e a fila não esteja mais cheia.



## Weighted Random Early Detection

- Policiamento de tráfego, de modo a maximizar o throughput em condições de congestionamento.
- O RED trabalha em conjunto com protocolos como TCP, de modo a evitar congestionamento da rede.



Esta sessão de aprendizagem oferece uma introdução breve dos conceitos de RED e endereça o WRED, uma implementação de RED. O mecanismo RED foi proposto por Sally Floyd e Van Jacobson em 1990, para endereçar congestionamento de rede em resposta à maneira tradicional. O mecanismo RED está baseado na premissa de que a maioria do tráfego roda em implementações de transporte de dados, as quais são sensíveis à perda, e em determinados períodos sofre um atraso devido ao descarte do seu tráfego. O TCP, que responde apropriadamente ao descarte do tráfego através de técnicas de atraso no envio deste, permite o uso do RED com um mecanismo de sinalização para evitar congestionamento. É importante considerar que o uso do RED deve ser empregado em transportes de rede tal como TCP, onde o protocolo é robusto, em resposta à perda de pacotes. No caso do protocolo Novell Netware e AppleTalk, nenhum deles é robusto em resposta à perda de pacotes, assim não devemos utilizar RED nesses casos.

O objetivo do RED é controlar o tamanho médio da fila indicando aos hosts quando eles devem transmitir seus pacotes mais lentamente. O RED leva vantagem ao utilizar-se do mecanismo de controle de congestionamento do TCP. Através do descarte randômico de pacotes em períodos de grande congestionamento, o RED conta a origem dos pacotes nos quais deve ocorrer uma diminuição na sua taxa de transmissão. Assumindo que o pacote de origem está utilizando TCP, a fonte vai diminuir sua taxa de transmissão até que todos os pacotes possam alcançar o seu destino, indicando que o congestionamento não ocorre mais. Na verdade, o TCP não para totalmente, ele reinicia rapidamente e adapta-se à taxa de transmissão que a rede pode suportar.

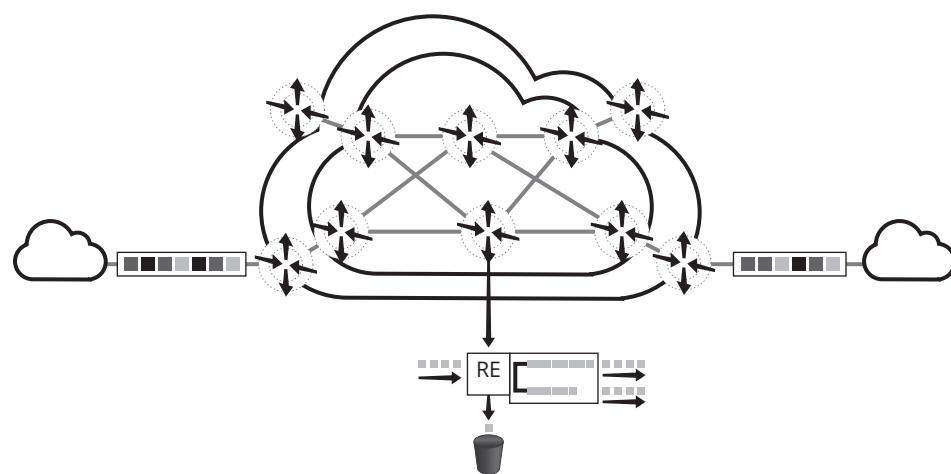


Figura 8.13  
Random Early Detection.

## Como o TCP manipula a perda de tráfego

Quando o recipiente do tráfego TCP – chamado de receptor – recebe o segmento de dados, ele verifica se os quatro octetos, os quais identificam o número de sequência, estão de acordo com o esperado, indicando, assim, que o segmento de dados foi recebido em ordem. Se o número bate, o receptor envia todos os dados para a aplicação destino, para então atualizar o número de sequência de modo a refletir o próximo número, e finalmente transmitir

um reconhecimento (ACK) para o emissor ou programa – um ACK – para ser transmitido após um pequeno período de tempo. O ACK notifica o emissor que o receptor recebeu todos os segmentos.

Os receptores geralmente tentam enviar um ACK, em resposta a alternativos segmentos de dados que recebem; esse envio se dá porque, para muitas aplicações, se o receptor espera mais que o atraso esperado, ele pode enviar um ack de resposta como uma resposta normal para o emissor. Entretanto, quando o receptor recebe um segmento de dados fora de ordem, ele responde imediatamente com um ACK para retransmitir o segmento de dados perdido.

Quando um emissor recebe um ACK, ele realiza as seguintes determinações: pode determinar se o dado foi entregue ou não; pode determinar que o ACK é um Keepalive, utilizado para manter a linha ativa. No caso da não recepção do dado, o ACK determina que o receptor recebeu algum ou nenhum dado. No caso da recepção de algum dado, o ACK determina se novos créditos para envio dos dados serão permitidos. Quando um reconhecimento de ACK é recebido, sendo que não houve dados enviados e não há mais dados a serem enviados, o emissor interpreta o ACK como um ACK repetido. Essa condição indica que alguns dados foram recebidos fora de ordem, forçando com que o receptor envie o primeiro ACK, e que o segundo segmento de dados foi recebido fora de ordem, forçando assim com que o receptor envie o segundo ACK. Na maioria dos casos, o receptor receberá dois segmentos fora de ordem porque um dos segmentos foi descartado.

Quando um emissor de TCP detecta um segmento de dados descartado, ele retransmite o segmento. Então ele ajusta a taxa de transmissão, que é a metade da existente antes do descarte detectado. Esse é o comportamento conhecido como back-off ou slow down. Embora esse comportamento seja apropriado para tratar congestionamento, problemas podem ocorrer quando múltiplas sessões concorrentes TCP encontram-se no mesmo roteador e todos os emissores TCP atrasam a transmissão dos pacotes ao mesmo tempo.

Roteadores podem manipular múltiplas sessões TCP concorrentes. Como os fluxos de rede são adicionados aos poucos, existe uma probabilidade de que o tráfego exceda o Transmit Queue Limit (TQL). Entretanto, existe uma grande probabilidade de que o tráfego excessivo seja temporário e que o tráfego não fique excessivo, exceto nos pontos nos quais ocorre o encontro entre tráfegos ou em roteadores das bordas.

Se um roteador descarta todo o tráfego que excede o TQL, como ocorre quando é usado o mecanismo tail drop, muitas sessões TCP irão simultaneamente se iniciar. Consequentemente, todo o tráfego vai ser afetado e todos os fluxos precisarão ir sendo gradualmente iniciados; essa atividade cria uma condição de sincronismo global.

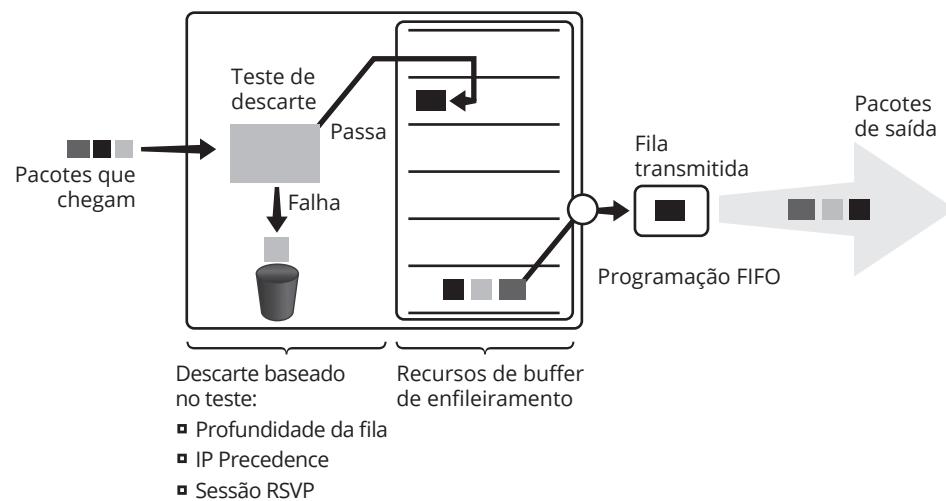
Entretanto, se o roteador não descarta tráfego, como é o caso de mecanismos de filas como fair queueing (FQ) ou custom queueing (CQ), então o dado opera como se fosse armazenado na memória principal, degradando dramaticamente a performance do roteador.

No caso do RED, ele resolve o problema recém-descrito levando uma sessão TCP ao retardado por vez, permitindo o uso completo do bandwidth. Já no caso do mecanismo WRED, ele combina as características do algoritmo RED com o IP Precedence, de modo a prover uma manipulação de tráfego preferencial para pacotes com maior prioridade. O WRED pode seletivamente descartar tráfego de baixa prioridade quando a interface inicia o congestionamento e provê diferentes características de performance para diferentes classes de serviços. Para interfaces configuradas para utilizar Resource Reservation Protocol (RSVP), o WRED escolhe pacotes de outros fluxos a serem descartados em vez de fluxos RSVP.



Também o IP Precedence governa quais pacotes são descartados – o tráfego de baixa prioridade tem uma taxa de descarte maior que a dos de alta prioridade.

O WRED difere de outras técnicas para evitar congestionamento tal como estratégias de filas, porque uma vez que ocorra congestionamento, no lugar de controlá-lo, ele procura se antecipar e evitá-lo.



### Saiba mais

O Weighted RED trabalha com múltiplos thresholds, um para cada classe de serviço. Serviços de baixa prioridade são descartados antes dos serviços de alta prioridade.

**Figura 8.14**  
Weighted Random Early Detection.

O WRED faz a antecipação de detecção de congestionamento possível e provê para múltiplas classes de tráfego. Ele também protege contra a sincronização global. Por essa razão, o WRED é utilizado em qualquer interface de saída na qual é esperada uma ocorrência de congestionamento. Entretanto, o WRED é geralmente usado em roteadores do core da rede, e não nos roteadores da borda. Roteadores de borda marcam precedência IP para pacotes quando eles entram na rede. O WRED usa essa precedência para determinar como tratar diferentes tipos de tráfego. O WRED provê thresholds e pesos separados para diferentes IP Precedence, permitindo prover diferentes qualidades de serviços com descarte de pacotes para diferentes tipos de tráfego. Nesse caso, o tráfego tradicional pode ser descartado mais frequentemente que o tráfego do tipo premium durante períodos de congestionamento.

Em períodos de congestionamento, quando os pacotes começam a ser descartados, o WRED avisa a fonte dos pacotes para decrescer sua taxa de transmissão. Se a fonte dos pacotes está utilizando TCP, ela decrescerá a taxa de transmissão de pacotes até que todos os pacotes alcancem o seu destino, que indica que o congestionamento não existe mais.

O WRED geralmente descarta pacotes seletivamente baseados no IP Precedence. Os pacotes com o maior IP Precedence são descartados em menor proporção do que os de menor precedência. Assim, quanto mais alta a prioridade dos pacotes, mais alta a probabilidade de que os pacotes sejam entregues. O WRED reduz as chances do tail drop através da seleção do descarte dos pacotes quando a interface de saída começa a mostrar sinais de congestionamento. Dropando alguns pacotes antecipadamente, em vez de esperar que a fila complete, o WRED evita um descarte numeroso de pacotes de uma só vez e minimiza as chances de sincronismo global. Assim, o WRED permite que a linha seja usada completamente. O WRED é útil quando o tráfego é do tipo TCP/IP. Com TCP, pacotes descartados indicam congestionamento, então a fonte dos pacotes reduz a taxa de transmissão.



## Mecanismos de policiamento e conformação

Nesta sessão, descreveremos dois mecanismos para regular tráfego: o rate-limite do committed access rate (CAR) para policiamento de tráfego e o Generic Traffic Shaping (GTS) e Frame Relay Traffic Shaping (FRTS) para conformar o tráfego. Implementar esses mecanismos através da rede significa assegurar que um pacote, ou dado origem, siga o contrato estipulado. Tanto o policiamento como o mecanismo de conformação usam o descritor de tráfego para um pacote – indicado pela classificação do pacote – para assegurar o caminho e o serviço.

Policiais e conformadores geralmente identificam as violações no descritor de tráfego de maneiras idênticas. Eles geralmente diferem, entretanto, no modo como respondem à violação. Por exemplo:

- Um policial tipicamente descarta o tráfego. (Por exemplo, o policial da taxa limite no CAR ou descarta o pacote ou reescreve seu IP Precedence, reconfigurando os bits do tipo de serviço do cabeçalho do pacote.);
- Um conformador tipicamente atrasa o tráfego excessivo, usando um buffer ou mecanismos de enfileiramento, para segurar pacotes e conformar o fluxo quando a taxa de dados da origem é mais alta do que o esperado. (Por exemplo, o GTS usa o Weighted fair queue para atrasar os pacotes para conformar o fluxo, e o FRTS usa o Priority Queue (PQ), Custom Queue (CQ) ou first-in, first-out (FIFO) para a mesma situação, dependendo de como é configurado.)

A conformação de tráfego e o policiamento podem trabalhar em conjunto. Por exemplo, um bom esquema de conformação de tráfego poderia ser usado para detectar fluxos com problemas de comportamento. Essa atividade é muitas vezes chamada de “policiamento de fluxo de tráfego”.

O CAR incorpora uma característica de taxa-limite para policiamento de tráfego, somado à sua característica de classificação. Essa característica gerencia o policiamento do acesso ao bandwidth da rede, assegurando que o tráfego, de acordo com os parâmetros de taxa especificados seja transmitido, enquanto descarta pacotes que excedam a quantidade de tráfego ou transmitindo-os com uma prioridade diferente.

As funções do limitador de taxa são:

- Permite controlar a taxa máxima de tráfego transmitido ou recebido em uma interface;
- Dá a habilidade de definir agregados de nível 3 ou taxa-limite de bandwidth de ingresso ou saída e para especificar políticas de manipulação de tráfego quando o tráfego está conforme ou excede taxas- limites especificadas.

O CAR examina o tráfego recebido na interface ou um subset desse tráfego selecionado por critérios da lista de acesso. Ele então compara a taxa de tráfego a um token bucket configurado, e age de acordo com os resultados. Por exemplo, o CAR descartará o pacote ou reescreverá o IP Precedence, reconfigurando os bits type-of-service (ToS).

A primeira razão para se utilizar conformação de tráfego é o controle de acesso para com o bandwidth disponível, assegurando que o tráfego seja conforme ao policiamento estabelecido para ele, e para regular o fluxo do tráfego, de modo a evitar congestionamentos que possam ocorrer quando o tráfego excede a velocidade da interface remota. Por exemplo:



- Controle de acesso ao bandwidth quando o policiamento dita que a taxa de uma dada interface não deveria, na média, exceder uma certa taxa, embora a taxa de acesso exceda a velocidade;
- Configurar conformação de tráfego na interface tendo uma rede com diferentes taxas de acesso. Suponha que os links Frame Relay tenha 128 kbps e 256 kbps. Enviar pacotes a 256 kbps pode causar falha na aplicação.

O conformador de tráfego previne perda de pacotes. O seu uso é especialmente importante em redes Frame relay porque o switch não pode determinar quais pacotes têm precedência, além de quais pacotes devem ser descartados quando o congestionamento ocorre. É de importância crítica para o tráfego do tipo real-time, tal como Voz sobre Frame relay, cuja latência é limitada, limitando assim a quantidade de tráfego e perda de tráfego em links de redes de dados em qualquer tempo, mantendo o dado em roteadores que fazem a garantia. A retenção do dado em roteadores permite que o roteador priorize o tráfego de acordo com a garantia. (A perda de pacotes pode ser resultante da consequência de aplicações real-time e interativas.)

O conformador de tráfego limita a taxa de transmissão de dados através de:

- Uma taxa específica configurada;
- Uma taxa derivada, baseada no nível de congestionamento.

Como mencionado, a taxa de transferência depende de três componentes que constituem o token bucket: tamanho da rajada, taxa resultante e intervalo de tempo. A taxa resultante é igual ao tamanho da rajada dividido pelo intervalo.

Quando o conformador de tráfego é habilitado, a taxa de bits da interface não vai exceder a taxa resultante sobre qualquer intervalo múltiplo do integral. Em outras palavras, durante todo o intervalo, um tamanho máximo de rajada pode ser transmitido. Dentro do intervalo, entretanto, a taxa de bit pode ser mais rápida que a taxa resultante em qualquer tempo dado.

Pode-se especificar qual pacote Frame Relay tem baixa prioridade ou baixa sensibilidade ao tempo e será o primeiro a ser descartado quando um Frame Relay estiver congestionado.

O mecanismo que permite um Frame Relay identificar tal pacote é o bit DE.

Pode-se definir listas de DE que identificam características de pacotes a serem descartados. Uma lista DE pode ser baseada em protocolos ou interfaces e em características, tal como fragmentação de pacotes, um TCP específico ou porta User Datagram Protocol (UDP), um número de lista de acesso ou tamanho de pacote.

## Qualidade de serviço na prática

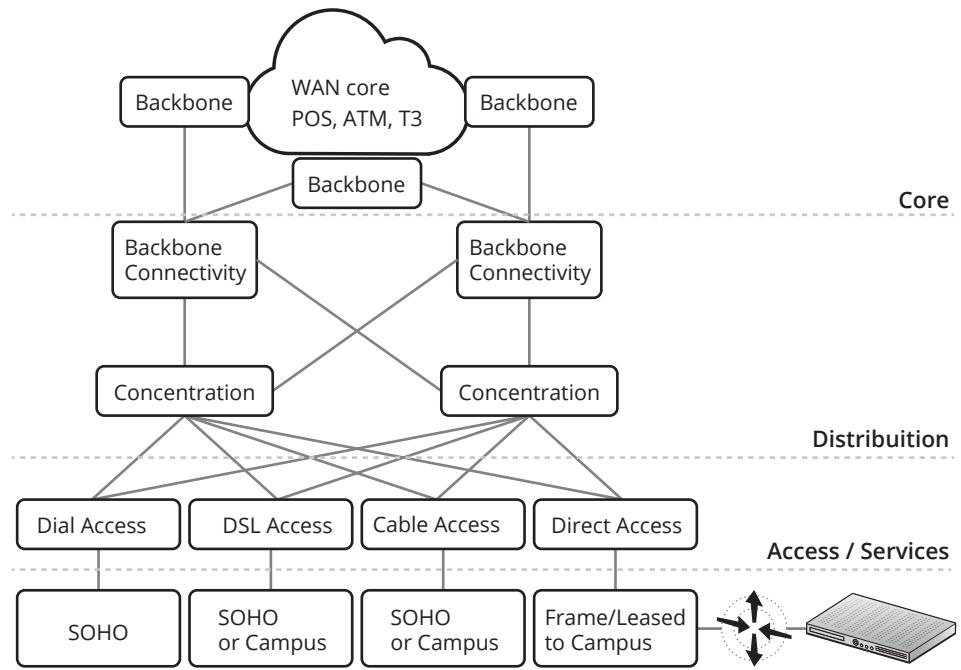
Uma vez que já foram definidas as técnicas para empregar QoS, como também a definição da arquitetura DiffServ, será apresentada a seguir uma solução de Qualidade de Serviço.

Nela o ambiente em questão requer diferenciação de tráfego, devido ao fato da necessidade de cobrança diferenciada deste, bem como garantia do Service Level Agreement.

A topologia de rede da solução é dividida em três níveis.

- Acesso;
- Distribuição;
- Core.





**Figura 8.15**  
Níveis de rede.

### Acesso

É no nível de acesso que o policiamento deve ser empregado. Isso ocorre devido ao fato da necessidade de limitar a taxa do tráfego de cada cliente requerido pelo SLA. Ou seja, cada cliente dentro da solução é cobrado pela banda que utiliza. Assim, para cada cliente o policiamento deve ser empregado.

Nesse nível também é classificado o tráfego do cliente de modo a se enquadrar em um dos seguintes tipos (classificação por DiffServ): Ouro (Gold traffic), Prata (Silver traffic) e Bronze (Bronze traffic). Esses tipos são relacionados usualmente a aplicações específicas:

- **Ouro:** voz e outras aplicações em tempo real;
- **Prata:** comércio eletrônico;
- **Bronze:** e-mail e web.

As técnicas para garantir o SLA do cliente podem envolver o uso de CAR ou WFQ.

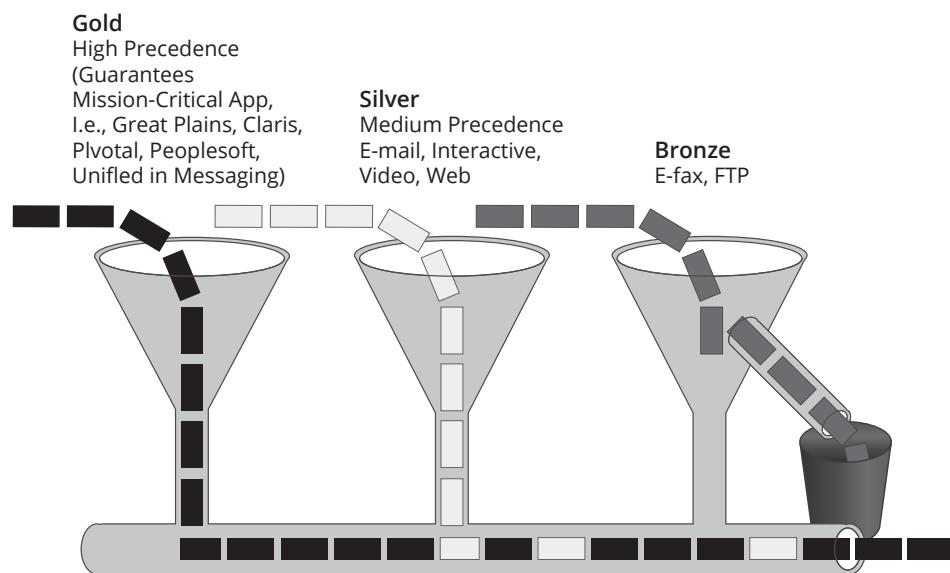
- Committed Access Rate (CAR):
  - **Marcação/Classificação de pacotes:** IP Precedence e QoS group setting;
  - **Gerência de acesso a banda:** limitação de banda (policiamento).
- Class-Based WFQ;
  - Configuração de limite mínimo de bandwidth;
  - Serviço de enfileiramento para controlar a latência.

### Distribuição

O nível de distribuição tem como função definir quais serão as políticas de descarte de rede quando a rede estiver congestionada. Ou seja, a distribuição tem como função evitar e gerenciar o congestionamento da rede. Assim, quando ocorre um congestionamento, o tráfego Silver ou Bronze é descartado, de modo a não afetar o tráfego Gold.



O que vale ressaltar é que o tráfego que será descartado em caso de congestionamento será o tráfego de um agregado de menor priorização, e não aleatório.



**Figura 8.16**  
Técnica de descarte.

## Core

Como o nível de distribuição, o Core também tem a função de prevenir o congestionamento da rede. Esse nível tem também a responsabilidade de interagir com o tráfego de ingresso e de saída. Ou seja, é nesse nível que os acordos bilaterais para fazer valer o QoS entre domínios devem ocorrer.

## Conclusão

A internet global está mudando tudo e todos. O mundo está convergindo para o protocolo da internet e para suas necessidades de rede. Porém, nesse processo, o projeto original do IP tornou-se muito débil. Como resultado, a internet necessita mudar para acomodar a demanda de novas aplicações. A largura de banda é uma solução, mas não é suficiente. De um modo geral, a internet necessita gerenciamento dessa largura de banda, ela necessita de “inteligência”.

Até agora, a internet tem provido apenas o serviço “best-effort”, no qual os recursos são compartilhados igualmente. Somar QoS significa somar “inteligência” à internet, o que é uma necessidade imediata, uma vez que habilita a possibilidade de diferenciação dos serviços.

Nos provedores, essa necessidade torna-se uma prioridade, uma vez que precisamos disponibilizar mais formas de acomodar o mundo da internet em um ambiente rico em serviços. Pois, com esse tipo de arquitetura, os ISPs poderão segmentar seu mercado, entregar mais valor ao cliente, alcançar negócios mais lucrativos e esquentar o crescimento mundial na internet.

## Métricas para o gerenciamento de rede

Está se tornando uma prática comum para os provedores instrumentarem suas redes com ambientes de gerência com suporte a medições ativas e passivas:

- Provedores de serviço:
  - Querem identificar “gargalos” de performance e tendências na rede (disponibilidade, taxa de perdas, utilização de banda, atraso) para fins de planejamento de capacidade.



- Pesquisadores:

- Querem estudar as características das redes que podem ser adotados em modelos de simulação para o desenvolvimento de novos protocolos de rede para aplicações avançadas.

- Usuário final:

- Está interessado em conhecer a performance de rede que está sendo oferecida em seu computador.
  - Ainda não costuma fazer uso das medições ativas, porém ferramentas estão surgindo para facilitar o uso pelo usuário final (exemplo: NDT).

- Aplicações úteis a ele:

- “Por que a qualidade da vídeo-conferência está ruim?”
    - Qual a largura de banda ele consegue utilizar até um provedor?
    - E o suporte a multicast, conectividade com outras redes?

Aplicações avançadas baseadas em rede tais como visualização remota, ferramentas de colaboração e compartilhamento, e escalamento de serviços computacionais em cluster podem ser mais eficientes se tiverem dados de “previsão” de desempenho da rede. Ou seja, a necessidade de realizar medições nas redes é atender as demandas dos provedores na questão de planejamento de capacidade e responder as necessidades dos usuários e dos pesquisadores.

## O que é desempenho de rede

Algo que é intrinsecamente ligado a desempenho de rede é a definição de “lentidão da rede”.

O que é exatamente uma rede lenta, e como se pode definir?

Como determinar quando a rede está lenta, e como se pode fazer isso? Existem usualmente mais questões que respostas quando se está lidando com o desempenho de rede em ambientes de produção.

A maioria dos elementos envolvidos no desempenho da rede pode ser resumida a poucos princípios simples que podem ser medidos, monitorados e controlados pelo administrador de rede.

A maioria das ferramentas de desempenho da rede usa uma combinação de cinco elementos distintos para medir o desempenho:

- Disponibilidade;
- Tempo de resposta;
- Utilização da rede;
- Vazão da rede;
- Largura de banda da rede.

A monitoração da rede através de medições envolve 3 conceitos fundamentais:

- Métrica.
- Metodologia de Medição.
- Medição da rede propriamente dita.



## Métrica, Metodologia de medição e Medição

### Métrica

Uma métrica do contexto da monitoração de redes é definida como:

- Uma propriedade de um componente da rede, definido cuidadosamente e quantificada usando unidades padrão;
- É uma entidade que permite descrever o desempenho, a confiabilidade e o estado operacional de uma rede ou seus elementos;
- É uma descrição formal dos serviços ou das condições operacionais na rede.
- Metodologia de medição.
  - A metodologia de medição é uma forma sistematizada de estimar a métrica;
  - Pode existir mais que uma metodologia para uma mesma métrica.
- Medição:
  - A medição é o resultado da aplicação de uma metodologia.
  - Em geral, uma medição tem incertezas ou erros.
  - O valor de uma métrica é calculado a partir de um ou mais resultados.

### Medição Ativa e Medição Passiva

#### Medição Ativa

- Na medição ativa, um tráfego artificial é trocado entre os nós de monitoramento/medição, com o objetivo de conhecer o desempenho da rede;
- A técnica de medição ativa requer a injeção de pacotes de teste na rede para determinar a performance fim-a-fim de um caminho de rede;
- Pacotes (sondas) são enviados entre um ponto emissor e um ponto receptor na rede e a partir de seu comportamento na rede se determina o desempenho;
- Pontos positivos:
  - Caracteriza melhor a percepção de qualidade da aplicação pelo usuário, pois emula o comportamento de tráfego da aplicação usando pacotes de teste.
- Pontos negativos:
  - Consome largura de banda utilizada pelo tráfego da aplicação;
  - Compete com as aplicações pelo uso de recursos da rede.

#### Medição Passiva

- Na medição passiva, o tráfego da rede ou uma amostra desse é avaliado pelos nós de monitoramento;
- Nesta técnica não existe a necessidade da injeção de pacotes de teste na rede;
- Requer a captura de pacotes e seus correspondentes timestamps por aplicações rodando em dispositivos conectados na rede em diversos pontos;
- Pontos positivos:
  - Não injeta tráfego de teste e os dados são obtidos de dispositivos que estão envolvidos no funcionamento da rede.
- Pontos negativos:
  - Impõe certa sobrecarga nos dispositivos de rede para disponibilizar as informações em adição a sua função de encaminhar pacotes.



## Padrões para monitoração da rede



### O Padrão IPPM

- ▣ IPPM – IP Performance Metrics work group;
- ▣ Desenvolveu um conjunto de métricas padrão que podem ser aplicadas para qualidade, desempenho e confiabilidade dos serviços de entrega de dados da internet;
- ▣ Os padrões do IPPM são úteis para uniformização das medições, que podem ser realizadas na internet ou em redes IP privadas.

Por exemplo, as métricas de atraso de pacotes em um sentido (one-way packet delay) e a perda de pacotes em um sentido (one-way packet loss) já foram utilizadas em plataformas de medições, como Surveyor.

Esses são exemplos de unificação de procedimentos. As medições de um único sentido (one-way) são antagônicas em relação às medições de ida-e-volta (round-trip), pois existem muitas instâncias nas quais os caminhos de ida e volta são diferentes entre si (em termos de rota). Com isto, produz-se valores para métricas de atraso mínimo, variação de atraso ou perda de pacotes distintas.



### O projeto perfSONAR

- ▣ O projeto perfSONAR adota a abordagem de medição ativa e medição passiva, além dos procedimentos e métricas definidas pelo IPPM para obtenção de dados de desempenho unidirecionais em redes IP;
- ▣ A infraestrutura desenvolvida para realizar as medições é composta de dispositivos de medição, servidores de análise e bases de dados;
- ▣ Os dispositivos de medição são equipamentos dedicados, com relógios GPS e Sistema Operacional modificado para melhorar a precisão no relógio do sistema.

## Fatores que influenciam o desempenho de rede



### Bandwidth Delay Product (BDP):

- ▣ O Bandwidth Delay Product (BDP) de um caminho fim-a-fim é o produto da largura de banda de contenção/gargalo e o atraso do caminho.
- ▣ Pode-se pensar BDP como “uma capacidade memória” de um caminho fim-a-fim, isto é, a quantidade de dados que cabem inteiramente no caminho entre os dois sistemas finais.
- ▣ O BDP ajuda a estimar o tamanho ótimo da janela do TCP.

### Unidade Máxima de Transmissão (MTU):

- ▣ A MTU descreve o tamanho máximo de um pacote IP que pode ser transferido sobre um canal de comunicação sem fragmentação
- ▣ Tamanhos comuns de MTU são:
  - ▣ 1500 bytes (Ethernet, 802.11 WLAN).
  - ▣ 4470 bytes (FDDI, padrão para POS e links seriais).
  - ▣ 9000 bytes (Convenção da internet2 e GEANT, limite de alguns adaptadores Gigabit Ethernet, Jumbo Frame).
  - ▣ 9180 bytes (ATM, SMDS).



Exemplo: resultados da medição da vazão máxima obtida entre dois servidores HPML385.

Na tabela 1, a vazão foi medida com Jumbo frame não habilitado, enquanto na tabela 2 o MTU foi definido em 9000 Bytes (Jumbo frame habilitado).

Exemplo: Canal de 2Mbps e atraso RTT = 300ms

$2000000 \times 0,30 = 60.000 / 8 = 75.000 \text{ bytes} = \text{Valor da janela TCP, ou } 2/8 = 0,25 * 0,3 = 0,75\text{M} = 75\text{Kbytes.}$

Canal de 10Mbps, atraso RTT = 100ms

$10000000 \times 0,10 = 1.000.000 / 8 = 125000\text{Bytes} = 125\text{Kbytes} = \text{Valor da janela TCP.}$

Caminhos de redes com BDP alto são chamados de Long Fat Networks ou LFN.

**Tabela 8.2**  
Servidor HP ML385  
sem Jumbo frame.

Exemplo: Exemplo: Redes de satélite tem alta banda e alto atraso.

O BDP é um parâmetro importante para o desempenho de protocolos baseados em janela (window-based protocols). Exemplo: Exemplo: TCP.

#### Vazão e Perdas – Camada 3

[ 3]	569 MBytes	955 Mbits/sec	0.034 ms	1024/406888 (0.25%)
[ 3]	569 MBytes	955 Mbits/sec	0.003 ms	1056/406915 (0.26%)
[ 3]	569 MBytes	955 Mbits/sec	0.012 ms	1088/406946 (0.27%)
[ 3]	569 MBytes	955 Mbits/sec	0.002 ms	1056/406915 (0,26%)

#### Vazão e Perdas – Camada 2

[ 3]	569 MBytes	955 Mbits/sec	0.003ms	1024/406889 (0.25%)
[ 3]	569 MBytes	955 Mbits/sec	0.002ms	1056/406915 (0.26%)
[ 3]	569 MBytes	955 Mbits/sec	0.002ms	1088/406947 (0.27%)
[ 3]	569 MBytes	955 Mbits/sec	0.024ms	1056/406916 (0,26%)

#### Vazão e Perdas – Camada 3

[ 3]	0.0 - 5.0 sec	591 Mbytes	992 Mbits/sec 0.002 ms	25/69704 (0036%)
[ 3]	5.0 - 10.0 sec	591 Mbytes	992 Mbits/sec 0.120 ms	35/69715 (005%)
[ 3]	10.0 - 15.0 sec	591 Mbytes	992 Mbits/sec 0.115 ms	30/69709 (0043%)
[ 3]	15.0 - 20.0 sec	591 Mbytes	992 Mbits/sec 0.022 ms	35/69714 (005%)

#### Vazão e Perdas – Camada 2

[ 3]	0.0 - 5.0 sec	591 Mbytes	992 Mbits/sec 0.002 ms	25/69704 (0036%)
[ 3]	5.0 - 10.0 sec	591 Mbytes	992 Mbits/sec 0.001 ms	35/69714 (005%)
[ 3]	10.0 - 15.0 sec	591 Mbytes	992 Mbits/sec 0.007 ms	30/69709 (0043%)
[ 3]	15.0 - 20.0 sec	591 Mbytes	992 Mbits/sec 0.001 ms	35/69715 (005%)

**Tabela 8.3**  
Servidor HP ML385  
com Jumbo frame.



## Atraso fim-a-fim em um sentido

- O atraso fim-a-fim em um sentido (One-Way Delay – OWD) é o tempo que um pacote leva para atingir seu destino de fim-a-fim.
- É o tempo entre a ocorrência do primeiro bit de um pacote no primeiro ponto de observação (interface de monitoramento transmissora) e a ocorrência do último bit do pacote no segundo ponto de observação.
- Padrão que define a métrica “Atraso fim-a-fim em um sentido”.
- RFC 2679: A One-way Delay Metric for IPPM.
- Formas de medição.
- RFC 3763: One-way Active Measurement Protocol (OWAMP).

O atraso fim-a-fim em um sentido (OWD) pode ser decomposto em atraso por hop, ou seja, o atraso observando entre um salto e outro na rede. O atraso por hop ainda pode ser decomposto em atraso por canal de comunicação (link) e por nó de rede (roteador).

O atraso em um sentido é uma métrica de desempenho definida pelo IPPM segundo o RFC 2679: A One-way Delay Metric for IPPM.

A medição do atraso em um sentido fim-a-fim exige a sincronização precisa dos relógios de ambos os sistemas finais.

O RFC 3763: One-way Active Measurement Protocol (OWAMP) Requirements define como essa métrica pode ser medida.

- Erros e Incertezas na medição;
- Sincronização entre pontos de observação;
- Perda de pacotes;
- Fragmentação;
- Imprecisão dos carimbos de tempo (timestamps);
- Unidade de Medição: ms (milissegundos);
- Aplicabilidade:
  - Detectar sintomas de congestionamento na rede e determinar exatamente em qual sentido da comunicação o congestionamento está ocorrendo;
  - Realizar a medição com alta precisão;
  - Validar certas aplicações na rede, tais como **VoIP** e vídeo.

### VoIP

Voz sobre IP. Os seguintes fatores a seguir contribuem para a existência de erros e incertezas na medição do atraso em um sentido:

- Sincronização entre pontos de observação;
- Perda de pacotes;
- Fragmentação;
- Imprecisão dos carimbos de tempo (timestamps).

A sincronização dos relógios dos pontos de medição em fontes de sincronização de alta precisão do tipo GPS faz com que se obtenha precisão da ordem de microsssegundos.

A medição do atraso em um sentido permite validar certas aplicações na rede, tais como VoIP e Vídeo. Essas aplicações possuem restrições quanto ao atraso máximo em um sentido.

- Composição do OWD:
  - Atraso por canal de comunicação (link):
    - Atraso de serialização;
    - Atraso de propagação.
  - Atraso por nó de rede (roteador)
    - Atraso de enfileiramento;
    - Atraso de encaminhamento (forwarding).
  - Atraso de Serialização:
    - O atraso de serialização é o tempo necessário para separar um pacote em unidades de transmissão sequenciais no canal (link).
  - Atraso de propagação:
    - O atraso de propagação é o tempo de duração para mover os sinais (bits) do transmissor para o receptor de um canal de comunicação.
- Fonte: <http://kb.pert.geant.net/PERTKB/SerializationDelay>

Fibre length	One-way delay	Round-trip time
1m	5 ns	10 ns
1km	5 µs	10 µs
10km	50 µs	100 µs
100km	500 µs	1 ms
1000km	5 ms	10 ms
10000km	50 ms	100 ms

Atraso de propagação na fibra óptica em diferentes distâncias

Fonte: <http://kb.pert.geant2.net/PERTKB/PropagationDelay>

- Atraso por nó de rede;
- Atraso de Enfileiramento:
  - Tempo que um pacote permanece dentro de um nó tal (como um roteador) enquanto espera pela disponibilidade do canal de saída.
- Atraso de encaminhamento:
  - Refere-se ao processamento feito no nó.



#### Saiba mais

O traceroute incrementa o TTL em uma unidade para cada conjunto de pacotes que ele envia, conseguindo assim realizar o rastreamento dos gateways/roteadores pelo caminho.



Link Rate	64 kb/s	1 Mb/s	10 Mb/s	100 Mb/s	1Gb/s
<b>Packet Size</b>					
64 bytes	8 ms	0,512 ms	5.12 µs	5.12 µs	0.512 µs
512 bytes	64 ms	4.096 ms	409,6 µs	40.96 µs	4.096 µs
1500 bytes	185,5 ms	12 ms	1.2 ms	120 µs	12 µs
9000 bytes	1125 ms	72 ms	7.2 ms	720 µs	72 µs



Na tabela são apresentados os tempos de atraso de serialização para diferentes tamanhos de pacotes em canais de comunicação com diferentes capacidades de transmissão ou diferentes taxas de bits por segundo.

O atraso de serialização é o tempo necessário para separar um pacote em unidades de transmissão sequenciais no canal (link).

A unidade sequencial de transmissão em geral é o bit. O atraso de serialização é obtido pela divisão do tamanho do pacote (em bits) pela capacidade do canal em bits por segundo.

Atualmente, com a existência de links com alta taxa de bits, o atraso de serialização é pouco relevante.

## Atraso fim-a-fim bidirecional (ida e volta)

- O atraso bidirecional (ida e volta) ou (Round Trip Time – RTT) é a soma do atraso em um sentido da fonte para o destino, do destino para a fonte e mais o tempo que o destino leva para formular a resposta.
- Padrão que define a métrica:
  - IPPM RFC 2681: A Round-trip Delay Metric for IPPM
- Formas de medição:
  - Injeção de pacotes na rede, podendo usar os protocolos ICMP, UDP ou TCP.
- Unidade de medição:
  - Tipicamente em milissegundos.
- Aplicabilidade:
  - Detectar sintomas de congestionamento na rede.
  - Identificar baixo desempenho em TCP, em canais de alta capacidade.
  - Medir a disponibilidade de dispositivos de rede.
  - Estimar o atraso sem a necessidade de sincronização de relógios.

O RTT pode ser definido também como o período entre o instante de tempo que um pacote de requisição é enviado pelo nó fonte e o instante de tempo que ele recebe o pacote de resposta correspondente. O pacote de resposta deve ser enviado ao nó fonte assim que o pacote de requisição for recebido.

Valores altos de RTT podem causar problemas para o TCP e outros protocolos de transporte baseados em janela.

Esse comportamento do TCP pode ser exemplificado no cálculo da vazão TCP teórica entre o PoP-SC -> internet2.edu.

Ou seja, o atraso entre esses dois sites na internet é da ordem de 200ms – a janela TCP padrão é de 64Kbps. Pela fórmula do BDP, visto anteriormente, conclui-se que a vazão máxima será da ordem de 2,6Mbps.

Exemplo do cálculo do BDP com janela de 64K e atraso de 200ms:



Janela(Bytes) = banda(Bytes) X Delay(seg)

$$\text{Banda} = 65.536 / 0.2$$

$$\text{Banda} = 327.680 / 0.2$$

**Banda = 2.621.440 bits/s ou 2.62 Mbps**

## Variação do atraso – Jitter

- O atraso unidirecional (OWD – One Way Delay) não é constante em uma rede real.
- Define-se variação de atraso, comumente jitter, como sendo a diferença entre o OWD do pacote atual e a média do OWD.
- Segundo o IETF, dados um conjunto de pelo menos dois pacotes entre A e B, a variação do atraso é a diferença do atraso em um sentido (OWD) de um par selecionado de pacotes no conjunto.
- Erros e incertezas na medição:
  - Sincronização entre pontos de observação.
  - Perda de pacotes; Fragmentação.
  - Imprecisão dos carimbos de tempo (timestamps).
- Padrão que define a métrica:
  - RFC 3393: IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)
- Unidade de medição:
  - Tipicamente em milissegundos
- Aplicabilidade:
  - Previsão de desempenho de aplicações sensíveis à variação do atraso (VoIP).
  - Dimensionamento do de-jitter buffer.

O ITU-T define a variação do atraso com a diferença entre o percentil 99,9 do OWD e um atraso referência (aconselhado atraso mínimo do caminho).

## Perda de pacotes em um sentido

- A métrica perda de pacotes é definida como a probabilidade de um pacote ser perdido no trânsito de uma origem A para um destino B.
- A taxa de perda indica o percentual de pacotes perdidos quando observado em um ponto de recepção em relação ao número de pacotes transmitidos em um ponto de transmissão em um intervalo de tempo.
- Padrão que define a métrica:
  - RFC2680: A One-way Packet Loss Metric for IPPM
- Formas de medição:
  - A perda de pacotes pode ser medida de forma ativa através do envio de pacotes de uma fonte para um destino.
  - Esse valor é mensurado através da razão entre o número de pacotes recebidos sobre o número total de pacotes enviados.



- Erros e incertezas na medição:
  - Sincronização entre pontos de observação.
  - Perda de pacotes.
  - Fragmentação.
- Unidade típica de medição:
  - % (percentual) = total de pacotes recebidos/total de pacotes transmitidos.
- Aplicabilidade:
  - Predição de desempenho de aplicações sensíveis a perdas de pacotes.
  - Identificar problemas de comunicação na camada física.
  - As suas principais razões para se ter perdas de pacotes na rede é o congestionamento e os erros.
- Perdas por congestionamento:
  - Quando a carga excede a capacidade de uma parte da rede, pacotes são enfileirados (bufferizados).
  - O congestionamento pode estourar as filas e se conduzir a perdas de pacotes.
- Perdas por erro:
  - Uma razão para a perda de pacotes é a "degeneração" dos dados, onde partes do pacote são modificadas em trânsito.
  - Ruídos das linhas normalmente são detectados pelo checksum (CRC) na camada de enlace na recepção, quando então o pacote é descartado.

Aplicações que requerem transmissão confiável (exemplo: Transferência de arquivos) usam retransmissão, técnica que pode reduzir o desempenho.

Protocolos sensíveis a congestionamento, tal como o TCP padrão, assumem que a perda de pacotes ocorre devido a congestionamento e respondem reduzindo sua taxa de transmissão proporcionalmente.

O congestionamento pode ser causado por condições de carga moderada mantidas por longos períodos de tempo ou pela chegada repentina de uma quantidade muito grande de tráfego (tráfego em rajada).

## Vazão (largura de banda alcançável)



- É a taxa máxima (camada IP) que um fluxo pode alcançar em um caminho na presença de tráfego (com carga).
- A vazão é o montante de tráfego de dados movidos de um nó da rede para outro em um determinado período de tempo.
- Erros e incertezas na medição:
  - Equipamentos para testes disponíveis e adequados.
  - Protocolo de transporte utilizado (TCP/UDP).
- Formas de medição:
  - Realização de testes específicos usando equipamentos adequados nas extremidades do caminho desejado.
  - Unidade de medição.
  - Bits por segundo (bps) ou múltiplos (Mbps, Gbps).



- Aplicabilidade:
  - Certificação do enlace contratado.
  - Dimensionamento de aplicações.
  - Simulação e aplicação rodando na rede.



A largura de banda alcançável ou vazão é a métrica definida como a quantidade máxima de dados (bits de pacotes IP) por unidade de tempo que pode ser transmitida através de um caminho consistindo de múltiplos canais, cada um deles exibindo um nível específico de utilização, entre dois nós finais da rede.

Os seguintes fatores a seguir contribuem para a existência de erros e incertezas na medição do atraso em um sentido:

- **Equipamentos para testes disponíveis e adequados:** para medir a largura de banda alcançável, devemos possuir equipamentos capazes de gerar tráfego na capacidade nominal máxima do canal de comunicação sendo medido. Por exemplo, para a medição do canal/caminho de comunicação de 1Gbps, devemos ter à disposição equipamentos com capacidade para gerar tráfego (TCP ou UDP) nessa capacidade;
- **Protocolo de transporte utilizado (TCP/UDP):** em redes IP geralmente se faz a medição utilizando o protocolo TCP ou UDP. O TCP possui capacidade de recuperação de erros e controle de fluxo, e é útil para estimar a vazão máxima do canal/caminho quando utilizando o protocolo utilizado pela maioria das aplicações. O UDP, protocolo de controle de fluxo, pode ser usado quando se deseja estressar a capacidade máxima do canal/caminho ou mesmo verificar se este é capaz de suportar uma determinada vazão sem apresentar perdas.

## Largura de banda de contenção

- É a métrica que define a capacidade do enlace de menor capacidade em um caminho.
- O que implica ser a taxa máxima (camada IP) que um fluxo pode alcançar em um caminho quando não existe tráfego (sem carga).
- Erros e incertezas na medição:
  - São os mesmos que implicam nos erros e incertezas das métricas de vazão.
- Formas de medição:
  - Realização de testes específicos usando equipamentos adequados nas extremidades do caminho desejado.
- Unidade de medição:
  - Bits por segundo (bps) ou múltiplos (Mbps, Gbps).
- Aplicabilidade:
  - Certificação do enlace contratado.
  - Dimensionamento de aplicações.
  - Simulação e aplicação rodando na rede.



## Métricas de medições passivas

Métricas que podem ser obtidas através de medições passivas:

- Descartes.
- Erros.
- Atraso em um sentido.
- Largura de banda utilizada.
- Fluxos.
- Disponibilidade.

### Descartes

- É um contador representando o número de pacotes descartados para uma determinada interface de rede.
- Coletado via SNMP ou CLI do equipamento.
- Usado para detecção de links congestionados ou mal provisionados (descartes na saída da interface).

É uma métrica normalmente obtida através do SNMP ou CLI que indica o número de pacotes descartados para uma determinada interface de rede. Um descarte pode ser ocasionado em condições normais devido ao congestionamento de um enlace. Na maioria dos equipamentos de rede, é possível configurar as políticas de enfileiramento e descartes, como FIFO, RED, WRED etc.

Erros e incertezas na medição:

- Equipamentos de rede indevidamente configurados;
- Problemas de buffer overflow dos contadores;
- Processador lento.

Formas de medição:

- Através da coleta de dados via SNMP;
- Através da visualização dos contadores via CLI.

Unidade de medição:

- Contador.

Aplicabilidade:

- Detecção de links congestionados ou mal provisionados (descartes na saída da interface);
- Descartes na entrada da interface podem indicar erro de configuração, como, por exemplo, uma vlan incorretamente permitida em uma ponta do link e não existente na outra.

### Erros

- É um contador que representa o número de erros em uma determinada interface (CRC e FCS).
- Coletado via SNMP ou comandos remotos nos ativos de rede.
- Dessa métrica é derivada a taxa de erros, que é a razão entre o número de pacotes que não foram entregues ao destino final e o número total de pacotes, em um determinado período de tempo (unidade em percentual).

- É usado para detecção de links intermitentes ou problemas de hardware.
- Ocorrem devido a:
  - Erros nos pacotes.
  - Erros de transmissão.
  - Insuficiência da taxa de transmissão.



É uma métrica normalmente obtida através de SNMP, que indica o número de erros em uma determinada interface.

- Frame Check Sequence (FCS);
- Cyclic Redundancy Check (CRC).

Valores elevados de erros de FCS/CRC são indicativos de problemas no meio de transmissão, como por exemplo um link WAN fornecido pela operadora ou uma fibra óptica mal conectada em uma rede de Campus.

#### **Taxa de erros:**

É a razão entre o número de pacotes que não foram entregues ao destino final e o número total de pacotes, em um determinado período de tempo.

As perdas de pacotes normalmente ocorrem devido a:

- Erros nos pacotes:
  - Cabeçalhos: descarte em roteadores;
  - Dados (payload): descarte pelo Sistema Operacional;
  - Erros de transmissão em geral, inerentes aos meios de transmissão, falhas em equipamentos ou ocorridos no nível de enlace;
  - Insuficiência da taxa de transmissão (banda de rede) em relação aos requisitos da taxa de transferência: estouros de buffer e descartes de pacotes.

#### **Erros e incertezas na medição:**

- Equipamentos de rede indevidamente configurados;
- Problemas de buffer overflow dos contadores.

#### **Formas de medição:**

- Através da coleta de dados via SNMP;
- Através de comandos remotos nos ativos de rede.

#### **Unidade de medição:**

- A unidade de medição de erros é o contador de erros. Já a taxa de erros é o percentual de pacotes com erros, ou seja, total de pacotes com erros sobre o total de pacotes.

#### **Aplicabilidade:**

- Detecção de links intermitentes;
- Detecção de problemas de hardware.



## Atraso em um sentido (One-Way Delay – OWD)

- É a coleta de cabeçalho de um tráfego real, passando em nós observadores, onde é agregado o timestamp e correlacionados os dados para medir o OWD, resultando em uma métrica em milissegundos.
- Podem ocorrer erros de sincronismos, que podem deturpar os resultados.
- Aplica-se para medir o tempo de propagação de um pacote entre dois pontos na rede.

Em redes de alta velocidade possui limitações de implementação e escalabilidade.

Erros e incertezas na medição:

- Erros de sincronismo de relógio.

### Formas de medição:

- Coleta dos timestamps nos pontos de observação. A ferramenta Coralreef, desenvolvida no contexto do projeto Cooperative Association for internet Data Analysis (CAIDA), pode ser utilizada para realizar esse tipo de medição: <http://www.caida.org>;
- Unidade de medição: Milissegundos;
- Aplicabilidade: medir o tempo de propagação de um pacote entre dois pontos na rede.

## Largura de banda usada

- Quantidade de tráfego (calculada em bits por segundo ou múltiplos) em um enlace em um determinado momento.
- É limitada pela capacidade dos contadores de medição.
- Usada para planejamento de capacidade, contabilização e perfil de comportamento do tráfego e detecção de anomalias.
- Pode ser obtida através de contadores de tráfego dos roteadores, SNMP e CLI.

Quantidade de tráfego em um enlace em um determinado momento. A quantia da capacidade usada por pacotes IP (ambos cabeçalho e dados) sobre uma janela de tempo específica (período) (normalmente 5-15 minutos).

### Erros e incertezas na medição

Capacidades dos contadores. O SNMPv1 possui limitação de capacidade dos contadores. Em redes com muito tráfego, esses podem estourar e reiniciar para zero entre uma coleta e outra.

### Formas de medição

- Contadores de tráfego dos roteadores;
- SNMP e CLI.

Com SNMP, a variável ifInOctets contém o número total de octetos recebidos pela interface e a variável ifOutOctets contém o total de octetos transmitidos pela interface.

### Unidade de medição

- Bits por segundo (bps) ou múltiplos (Mbps e Gbps).



## **Aplicabilidade**

- Planejamento de capacidade;
- Contabilização;
- Perfil de comportamento do tráfego e detecção de anomalias.

## **FLUXOS**

- É utilizada para avaliar o tráfego sendo transmitido na rede (amostra ou total), podendo prover informação de utilização.
- Fluxo: IP fonte, IP destino, protocolo sobre o IP, porta fonte e porta destino.
- É usualmente obtida nos roteadores.
- Pode ser utilizada para determinar, por tipo de fluxo, a duração e a quantidade do tráfego de uma aplicação que está cruzando a rede.
- É medida através de protocolos presentes nos roteadores (Netflow, J-FLow, Sflow) e/ou espelhamento de interfaces.
- Aplica-se principalmente para segurança, contabilização, planejamento de rede e engenharia de tráfego.

Métricas relacionadas a fluxos de dados usualmente obtidas dos roteadores.

### **Erros e incertezas na medição**

- Medição por amostragem, podendo ser em diversas escalas, como 1 para 1, 1 para 200, 1 para 1000, 1 para N;
- Depende do processamento dos roteadores, se estiverem congestionados pode haver perda na medição;
- Pode existir perda na hora de exportar os fluxos para os coletores, pois normalmente esse processo é realizado através de UDP.

### **Formas de medição**

- Através de protocolos presentes nos roteadores, como Netflow, J-FLow e Sflow;
- Através do espelhamento de interfaces para gerar a contabilização, como Flow-Capture (flow-tools) e CoralReef.

### **Unidade de medição**

- Dependente da métrica.

## **Aplicabilidade**

- Segurança/Detecção de anomalias;
- Estudo de padrões de tráfego;
- Planejamento de redes;
- Engenharia de tráfego;
- Contabilização (Accounting);
- Bilhetagem (Billing).



## Disponibilidade

- É a medida percentual do tempo durante o qual um canal de comunicação, dispositivo de rede ou serviço está totalmente funcional em relação ao estado não funcional.
- É aplicada principalmente em acordos de níveis de serviços (SLA), onde paradas programadas podem ser consideradas para o cálculo da métrica.
- Pode ser obtida através do monitoramento de Variáveis SNMP, análise de comandos de CLI, monitoramento do estado do serviço e/ou ferramenta baseada em ICMP.

A disponibilidade é calculada pela medição do “uptime ou downtime” de um dispositivo da rede ou serviço usando medições passivas.

Paradas programadas (exemplo: dispositivos de rede ou serviços que são derrubados para manutenção) podem não serem consideradas no cálculo da disponibilidade, dependendo do Acordo de Nível de Serviço (SLA) firmado entre o provedor e o cliente.

Erros e incertezas na medição:

- A medição do estado operacional do dispositivo em intervalos de tempos muito grandes pode acarretar na falta de precisão dessa medição. Com SNMP o uso de mecanismos de trap (alerta) pode ser avaliada como opção para melhoria da precisão.

### Formas de medição

- Monitoramento de Variáveis SNMP;
- Análise de comandos de CLI;
- Monitoramento do estado do serviço;
- Ferramenta baseada em ICMP.

### Unidade de medição

- A disponibilidade ou índice de disponibilidade em geral é expresso em % de tempo em que o estado foi funcional em relação ao tempo total observado.

### Aplicabilidade

- Acordos de Nível de Serviço (SLA);
- Largura de banda utilizada;
- Fluxos;
- Disponibilidade.

## Ferramentas de diagnóstico/monitoração da rede

- Existem inúmeras ferramentas desenvolvidas para esse fim.
- As seguintes ferramentas são relevantes para o diagnóstico e monitoração das redes:
  - Ping.
  - Traceroute.
  - OWAMP.
  - IPERF.
  - NDT etc.

Existem inúmeras ferramentas que auxiliam o diagnóstico e desempenho da rede. Como Ping, Traceroute, OWAMP, IPERF, NDT etc. As principais ferramentas serão abordadas a seguir.

## Ping

- ▣ Uma das mais antigas e utilizadas ferramentas de diagnóstico.
- ▣ Baseada em mensagens ICMP (Echo Request/Echo Reply).
- ▣ Reporta:
  - ▣ RTT entre os pacotes enviados, sumarizando.
  - ▣ Tempo (Mínimo, Médio, Máximo e desvio padrão).
  - ▣ Pacotes Enviados/perdidos (número e percentual).

Ping é uma das mais antigas ferramentas que tem como intuito medir a alcançabilidade de um host e o atraso de ida-e-volta através do envio de mensagens ICMP do tipo echo Request/Echo Reply. Essa ferramenta está presente na maioria dos dispositivos de rede e Sistemas Operacionais, e permite mostrar o atraso e perdas de pacotes.

Quando existe um congestionamento na rede, na maioria dos casos observando a métrica de variação do atraso (jitter), quanto maior seu tempo, maior é o congestionamento na rede.

Um atraso alto e constante pode representar uma característica do meio de propagação, como por exemplo, o satélite que possui aproximadamente 500ms de ida-e-volta.

Cabe salientar que se dispararmos pings para equipamentos de rede, como roteadores e switches, podemos ter uma falsa impressão de congestionamento, pois estes não priorizam a resposta do ping, pois sua finalidade é rotear/comutar os pacotes. A melhor forma sempre é realizar medições com hosts finais ou máquinas dedicadas para medições.

## Traceroute

- ▣ Baseada no campo TTL do pacote IP.
- ▣ Incrementa o TTL em uma unidade para cada conjunto de pacotes.
- ▣ O roteador, ao receber o pacote, decrementa-o em uma unidade e, como seu valor chega a zero, envia uma mensagem ICMP (time exceeded in-transit).
- ▣ Reporta:
  - ▣ Caminho percorrido de um pacote até seu destino.
  - ▣ Reporta o atraso (RTT) em cada nó.

Traceroute é um utilitário que mostra o caminho que um pacote percorreu em uma rede IP até chegar ao destino. Essa ferramenta também pode ajudar a diagnosticar se existem pontos de congestionamento até o destino de final, pois reporta o atraso de cada nó.

Para realizar esse rastreamento do caminho por onde o pacote passou, o traceroute utiliza o campo *TTL* (*Time to Live*) do cabeçalho IP. O TTL é um mecanismo criado para evitar que loops de roteamento acabem fazendo com que os pacotes fiquem circulando para sempre na rede. Para contornar isso, o campo TTL é decrescido de 1 unidade a cada roteador por onde o pacote passa. Quando esse número chega a 0 (zero), uma mensagem ICMP time exceeded in-transit é enviada pelo roteador.



## OWAMP

- ▣ One-way Active Measurement Protocol.
- ▣ Baseado na RFC 4656.
- ▣ Protocolo que define as medições em um sentido.
  - ▣ One Way Delay (OWD).
- ▣ Ferramenta OWAMP:
  - ▣ Implementa o protocolo.
  - ▣ Também conhecida como One-Way Ping.
- ▣ Finalidade: realizar medições em um sentido.
- ▣ Requer sincronismo dos hosts via NTP.
  - ▣ Recomenda-se Stratum 1 para melhor acuracidade.

O One-way Active Measurement Protocol (OWAMP) é um protocolo definido na RFC 4656 que define o protocolo para a implementação de medições ativas em um sentido. Uma implementação desse protocolo que leva o mesmo nome do protocolo é a ferramenta OWAMP (também chamada de One-Way Ping).

Essa ferramenta tem como finalidade medir o atraso em um sentido, diferente da ferramenta ping, que mede o atraso de ida-e-volta.

Para realizar medições de atraso em um sentido, é imprescindível que os hosts estejam com seu relógio sincronizado, pois um carimbo de tempo é marcado na origem e em sua chegada, sendo a diferença do tempo de chegada e saída o atraso.

Caso o relógio do host A esteja com horário diferente de B, mesmo que esteja no mesmo segundo de tempo, haverá um erro na medição que poderá influenciar no atraso em um sentido. Normalmente os pontos de medição são sincronizados com o protocolo Network Time Protocol (NTP) e, para possuir uma melhor precisão, recomenda-se o uso de uma fonte externa de sincronismo ao NTP, como o GPS, conectado diretamente nos pontos de medição, capacitando o NTP a ser um servidor do tipo Stratum 1.

### OWAMP – Para que serve medições em um sentido

- ▣ Identificar sentido de congestionamento.
- ▣ Que geralmente ocorre primeiro em um sentido.
- ▣ Roteamento (assimétrico ou pode sofrer mudanças).
- ▣ Intervalos de Pooling SNMP mascaram grandes níveis de enfileiramento.
- ▣ Medições ativas podem identificar esses pequenos congestionamentos.

Detectar congestionamentos na rede de forma mais precisa que com as ferramentas que implementam RTT, pois os congestionamentos geralmente acontecem primeiramente em um sentido, já que a maioria dos canais de comunicação utilizados nas redes atuais são Full-DUPLEX.

Em um exemplo simples, geralmente na maioria das redes, o download dos usuários é saturado, enquanto o upload está livre. Isso às vezes é também notado em ligações VoIP, quando um dos locutores ouve perfeitamente a ligação e o outro sente que a ligação está picotada; com o OWAMP, esse gargalo poderia ser identificado nos sistemas finais.



### Roteamento (assimétrico ou existem mudanças)

Os intervalos de Pooling SNMP mascaram grandes níveis de enfileiramento que podem ser melhor mostrados por medições ativas.

Existem muitas implementações de One-Way delay (Surveyor, Ripe, MGEN etc.), mas são não interoperáveis, para isso o protocolo One-Way Delay (OWD) foi especificado na RFC.

A figura a seguir apresenta o funcionamento do OWAMP, no modo clássico, entre um cliente e servidor.

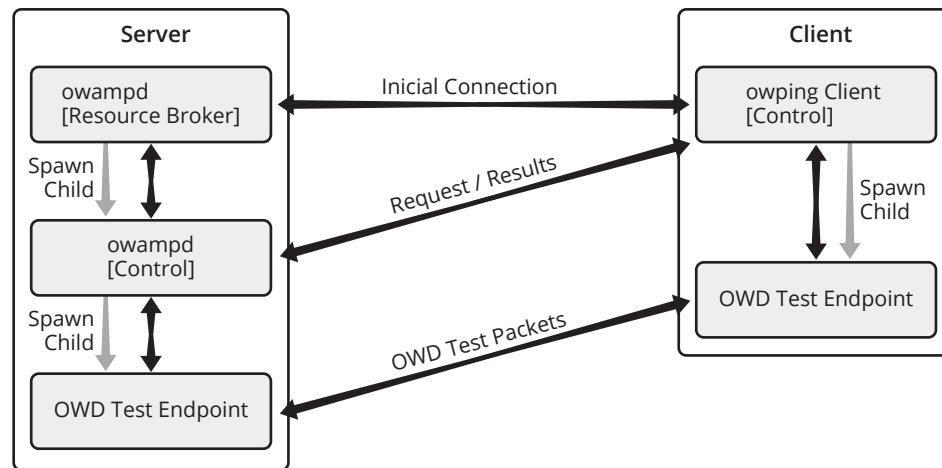


Figura 8.17  
OWAMP –  
Protocolo.

### OWAMP – Funcionalidades

- Suporte a AAA.
- Controle de recursos de rede.
  - Banda máxima para um teste.
  - Com ou sem autenticação.

Com o OWAMP, é possível implementar autenticação, autorização e contabilização (do termo em inglês AAA). Possibilita criar um arquivo com diversos parâmetros de configuração, como IPs autorizados para realizar testes, classes hierarquizadas de alocação de recursos (banda, testes, autenticação etc.), suporte a autenticação usuário/senha, chave criptográfica, IP ou sem autenticação, se realizará ou não armazenamento das medições locais e onde, entre outras inúmeras configurações.



```
$ owping owamp.pop-am.rnp.br

Approximately 13.8 seconds until results available
--- owping statistics from [owamp.pop-sc.rnp.br]:53874 to [owamp.pop-am.rnp.br]:50630 ---
SID: c8819c6acf40067165b5c233e151be3d
first: 2010-03-08T20:27:47.342
last: 2010-03-08T20:27:57.591
100 sent, 13 lost (13.000%), 0 duplicates
one-way delay min/median/max = 43.8/89.3/115 ms, (err=0.0763 ms)
one-way jitter = nan ms (P95-P50)
Hops = 7 (consistently)
no reordering

--- owping statistics from [owamp.pop-am.rnp.br]:53751 to [owamp.pop-sc.rnp.br]:64715 ---
SID:c8edc105cf4006717537c996cfb89310
first: 2010-03-08T20:27:47.163
last: 2010-03-08T20:27:57.213
100 sent, 0 lost (0.000%), 0 duplicates
one-way delay min/median/max = 36.1/40.6/94.2 ms, (err=0.0763 ms)
one-way jitter = 32.9 ms (P95-P50)
Hops = 7 (consistently)
no reordering
```

**Figura 8.18** O uso básico da ferramenta é bastante simples nos parâmetros padrões, com o cliente “owping” é especificado um endereço de ponto de medição owamp e a ferramenta executa 100 testes da origem para o destino e mais 100 do destino para a origem. Conforme o exemplo apresentado no slide ao lado.

Em negrito estão marcados os principais resultados obtidos com essa ferramenta, sendo pacotes enviados (sent), pacotes perdidos (lost) e seu percentual, e se houve ou não pacotes duplicados (duplicates).

Na próxima linha, apresenta o atraso em um sentido (one-way delay), sendo o tempo mínimo/mediana/máximo e entre parênteses o erro da medição, em milissegundos, baseado na soma dos erros fornecido pelo NTP em ambos os hosts. Além disso, observa-se duas medições, sendo a primeira os resultados da origem para o destino e a próxima do destino para origem.

Analizando os testes acima, podemos observar facilmente que existe perda de pacotes de SC → AC (13%) e não no sentido contrário e ainda que essas perdas estão sendo provavelmente causadas por causa de congestionamento de algum enlace no caminho SC → AC, pois o tempo da mediana referente ao atraso é de 89.3ms, muito superior ao tempo mínimo de 43.8ms. Outro indicativo de congestionamento é o One-Way Jitter, quanto mais variável indica oscilações no atraso. Dependendo do número elevado de perdas, a One-Way Jitter é desconsiderado e marcada como “nan”.

O One-Way Jitter, ou simplesmente variação de atraso em um sentido, é calculado pela ferramenta se utilizando os percentis 95 e 50 (P95 e P50). A ferramenta obtém esse valor através da subtração do P95 com o P50, variáveis que são mais explanadas a seguir.



## IPERF

- Realiza testes de vazão em TCP e UDP.
- Com ela, é possível medir a largura de banda máxima alcançável em TCP.
- Suporte a Ipv4, IPv6 e Multicast.
- Resultados são apresentados pelo servidor, com os dados que ela realmente recebeu.



O IPERF é uma ferramenta para mensurar desempenho da rede realizando testes de vazão TCP e UDP.

Com ela, é possível medir a largura de banda máxima alcançável em TCP, permitindo o tunnning de diversos parâmetros e características do UDP.

Os testes são de “memória” para “memória”, evitando que operações de I/O em disco influenciem nos resultados.

O IPERF é uma ferramenta que roda tanto como cliente quanto como servidor.

Possui suporte a Ipv4, IPv6 e Multicast.

Os resultados são apresentados pelo que o servidor realmente receber para medição em UDP e TCP.

O UDP envia pacotes a uma taxa solicitada sem garantia de entrega.

O TCP reconhece a recepção de todos os pacotes e retransmite qualquer pacote que falte.

### IPERF – Características TCP

- TCP:
  - O teste tenta utilizar toda a banda disponível até ocorrer um congestionamento, em caso de falha na entrega o TCP realiza a retransmissão.
  - Realiza a medição de largura de banda.
  - Reporta MSS e MTU.
  - Suporta o ajuste do tamanho das janelas do TCP.



Características específicas do TCP:

- O teste em TCP reconhece a recepção de todos os pacotes, no caso de falta, há retransmissão de pacotes;
- Realiza medição de largura de banda;
- Reporta o tamanho do MSS e MTU;
- Suporta o ajuste do tamanho das janelas do TCP;
- Pode atuar com Multithread, ou seja, cliente e servidor podem ter múltiplas conexões simultâneas.

### IPERF – Características UDP

- UDP:
  - O teste envia pacotes UDP na taxa especificada, sem garantia de entrega.
  - Usuário especifica a banda que quer enviar.
  - Medição de perda de pacotes, Jitter e pacotes fora de ordem.
  - Suporte a Multicast.



Características específicas UDP:

- O teste em UDP envia pacotes a uma taxa solicitada sem garantia de entrega;
- Clientes podem criar fluxos UDP para uma banda específica, exemplo: 10Mbps;
- Há a medição de perda de pacotes, variação do atraso (jitter) e pacotes fora de ordem;
- Suporte à Multicast;
- Pode atuar com Multithread, ou seja, cliente e servidor podem ter múltiplas conexões simultâneas.

```
(servidor) (ouvindo em TCP)

servidor$ iperf -s
....
```

```
(cliente)

cliente$ iperf -c 200.237.XXX.XXX -t 10
-----
Client connecting to 200.237.XXX.YYY, TCP port 5001
TCP window size: 16.0 KByte (default)
-----
[  3] local 200.237.XXX.XXX port 32908 connected with 200.237.XXX.YYY port 5001
[  3] 0.0-10.0 sec 1017 MBytes   853 Mbits/sec
```

**Figura 8.19**  
IPERF – Exemplo.

- Servidor ouvindo em TCP (Host remoto);
- Cliente disparando testes para o servidor (-c), com duração de 10 segundos (-t).

### Parâmetros de rede relacionados

- TCP (MSS) = MTU-40 Bytes.
- Tamanho da janela do TCP.
- MTU.
- Tabela de parâmetros de Sistemas Operacionais.

Existem diversos parâmetros relacionados com o desempenho da rede que podem ser utilizados para especificar melhor os testes realizados pela ferramenta. Destacam-se os seguintes parâmetros:

- **TCP(MSS) = MTU-40**: esse indica o tamanho dos pacotes utilizados pelo TCP;
- **Tamanho da janela TCP**: é o mais relevante na questão de desempenho;
- **MTU**: MTUs grandes reduzem o número de interrupções recebidas pelo sistema, incrementando a performance: Path MTU Discovery.

A seguir, há uma tabela que lista alguns parâmetros de diferentes Sistemas Operacionais que podem ser ajustados para aumentar as taxas de transferências.

Operating System	RFC Path MTU Discovery	RFC1323 Suport	Default maximum socket buffer size	Defalt TCP socket buffer size	Defalt UDP socket buffer size	EFC2018 SACK Support
FreeBSD	Yes	Yes	256kB	32Kb	40Kb	Yes
Linux 2.4 and 2.6	Yes	Yes	64kB	32Kb	32Kb(?)	Yes
Mac OS X	Yes	Yes	256kB	32Kb	42kb (receive)	Yes!
Sun Solaris 10	Yes	Yes	1 MB TCP, 256kB UDP	48Kb	8kB	Yes
Windows XP	Yes	Yes				Yes

### Cálculo do BDP para utilizar o IPERF

- Para melhorar os testes de vazão com TCP.
- BDP = Banda em Mbytes/s \* Atraso RTT (Segundos).
- RTT SC <-> PA ~ 60 ms
- Menor enlace no caminho a 100 Mbps / 8 =12,5MB/s
- BDP =  $12,5 \times 0,06 = 0,75$  Mbytes
- Janela TCP recomendada para vazão máxima = 750KB.
- Para medir no IPERF, parâmetro -W 750k

### IPERF::SC <- PA – Janela padrão 64K

```

RECEIVE START
iperf -B 200.237.193.1 -P 1 -s -f -m -p 5045 -w 65536 -t 10
-----
Server listening on TCP port 5045
Binding to local address 200.237.193.1
TCP window size: 65536 byte
-----
[ 14] local 200.237.193.1 port 5045 connected with 200.129.132.13 port 5045
[ 14] 0.0-10.0 sec 9969664 Bytes 7.960.334 bits/sec
[ 14] MSS size 1448 bytes (MTU 1500 bytes, ethernet)
RECEIVER END

```

### BWCTL::SC<-PA – Janela Calculada 750K

```

RECEIVE START
iperf -B 200.237.193.1 -P 1 -s -f -m -p 5046 -w 768000 -t 10
-----
Server listening on TCP port 5046
Binding to local address 200.237.193.1
TCP window size: 768000 byte
-----
[ 14] local 200.237.193.1 port 5046 connected with 200.129.132.13 port 5046
[ 14] 0.0-10.2 sec 29802496 Bytes 23.405.799 bits/sec
[ 14] MSS size 1448 bytes (MTU 1500 bytes, ethernet)
RECEIVER END

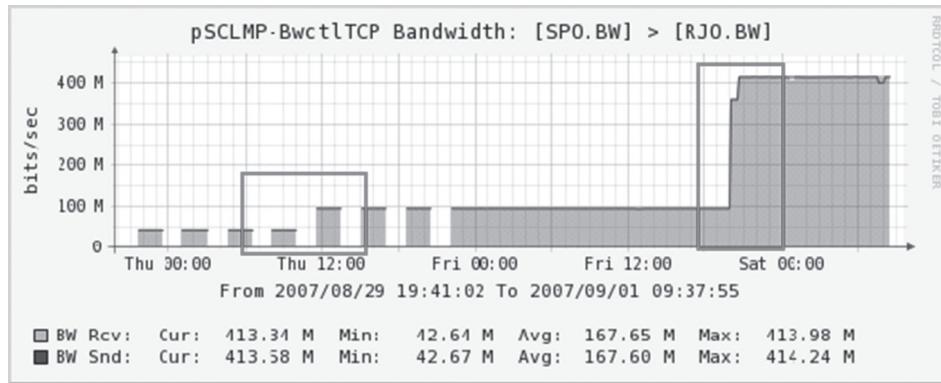
```

Tabela 8.4

Podem ser ajustados para aumentar as taxas de transferências.

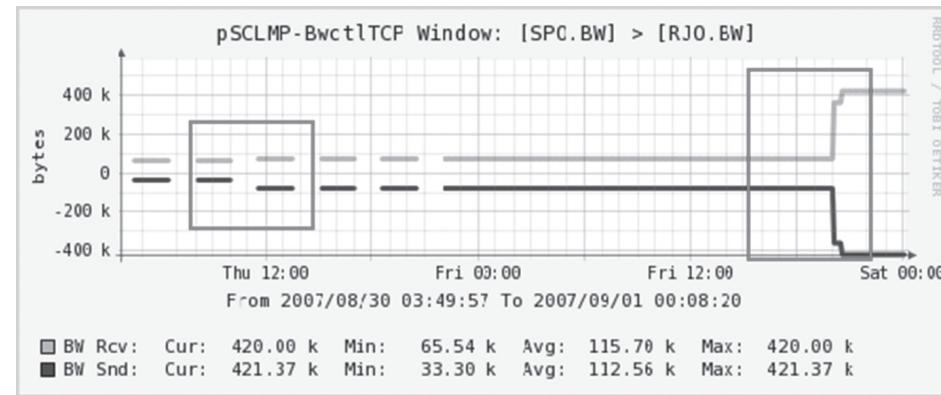
Figura 8.20  
IPERF: Exemplo 1: Uso do BDP na medição entre SC e PA.

O exemplo de medição mostra o emprego do cálculo do BDP para realizar o ajuste da janela; É possível observar uma ótima melhora na vazão, de 7.9Mbps para 23.4Mbps.



**Figura 8.21**  
IPERF: Exemplo 2  
(parte 1): Efeito do  
tamanho da janela  
TCP na vazão.

As figuras mostram o efeito do tamanho da janela TCP na vazão máxima alcançada.



**Figura 8.22**  
IPERF – Exemplo 2  
(parte 2).

Nesse caso, a janela TCP com a vazão correspondente foram:

(RTT ~ 6ms):

- Janela TCP de 64KB = Largura de banda de 42 Mbps;
- Janela TCP de 75KB = Largura de banda de 100 Mbps;
- Janela TCP de 420K = Largura de banda de 413Mbps.

### NDT

- Network Diagnostic Tool.
- Provê testes de diagnóstico de configuração e desempenho de rede.
- Baseado no kernel WEB100.
- O usuário final realiza a diagnóstico através de seu navegador WEB.
- Recomendado para testes de “última milha”.

O NDT é um aplicação cliente/servidor que provê testes de configuração e performance de rede para os usuários finais. Ela é composta por um cliente (linha de comando ou Applet Java) e um servidor (servidor páginas web e um engine de teste/análise). O servidor utilizar um kernel de Linux modificado (WEB100) para capturar estatísticas de fluxos TCP e retorna após os testes resultados multiníveis, permitindo aos usuários novatos ou avançados compreenderem melhor os resultados.

Uma das grandes vantagens é que ela pode ser utilizada pelo usuário final em qualquer lugar e em qualquer tempo, sem a necessidade de um software adicional, sendo necessário somente um navegador WEB com suporte a Java instalado. Ela foi projetada para identificar de forma rápida e fácil condições específicas de problemas normalmente encontrados próximos ao usuário que impactam o desempenho da rede, pois a grande maioria dos problemas de desempenho ocorrem na última milha próximos dos usuários em suas estações de trabalho; sendo assim, essa ferramenta é recomendada para testes de última milha.

Elá possibilita o usuário final ter uma visão do desempenho da rede, facilitando a ajuda na identificação de problemas de configuração de rede (falha na negociação do Duplex, por exemplo), desempenho (incluindo problemas na estação do usuário), fornecendo algumas evidências que podem ajudar usuários e administradores a isolar problemas.

É uma ferramenta de diagnóstico em tempo real, não utilizando dados históricos de medições. Elá funciona através de “assinaturas de rede”, desenvolvidas para identificar problemas típicos que acontecem na maioria das redes, além de realizar alguns aconselhamentos para melhorar o desempenho de uma conexão.

O modo de operação do NDT funciona através da execução de testes de ida e volta para obtenção de dados fim-a-fim, além da obtenção de dados de múltiplas variáveis do servidor. Após ser realizada a comparação do desempenho através de valores analíticos, é realizada a tradução de valores de rede em mensagens de texto, gerando um diagnóstico mais voltado a rede de campus, onde os usuários estão geralmente bem conectados no backbone da universidade e participando de uma rede acadêmica de alta capacidade de transmissão de dados.

### O que o NDT pode fazer

- Identificar se o Cliente, Servidor ou a Rede estão operando conforme esperado.
- Fornecer informações para ajuste da aplicação.
- Sugerir mudanças para melhorar o desempenho.
- Dizer ao usuário final que tem algo errado mesmo quando o administrador da rede diz “Tudo está normal, o problema deve ser sua máquina ou aplicação”.



O NDT pode detectar o canal mais lento em um caminho fim-a-fim, medindo o tempo de chegada entre os pacotes (através da libpcap) e conhecendo o tamanho de cada pacote. Ele pode calcular a taxa de transferência (velocidade) para cada par de pacotes enviados ou recebidos. Os resultados são quantificados segundo os padrões de tecnologias (Ethernet, FE, t3). O NDT não reporta velocidades fracionadas ou agregadas (LACP). Em implementações futuras identificará redes sem fio.

Outra informação interessante que o NDT pode reportar é a detecção de erro na negociação do modo Duplex, que é uma falha de negociação entre a placa de rede do PC e do switch. Ela acarreta alto impacto no desempenho, embora exista a conectividade básica. Não pode ser detectado por ping, traceroute e faz com que as aplicações de rede rodem extremamente lentas.

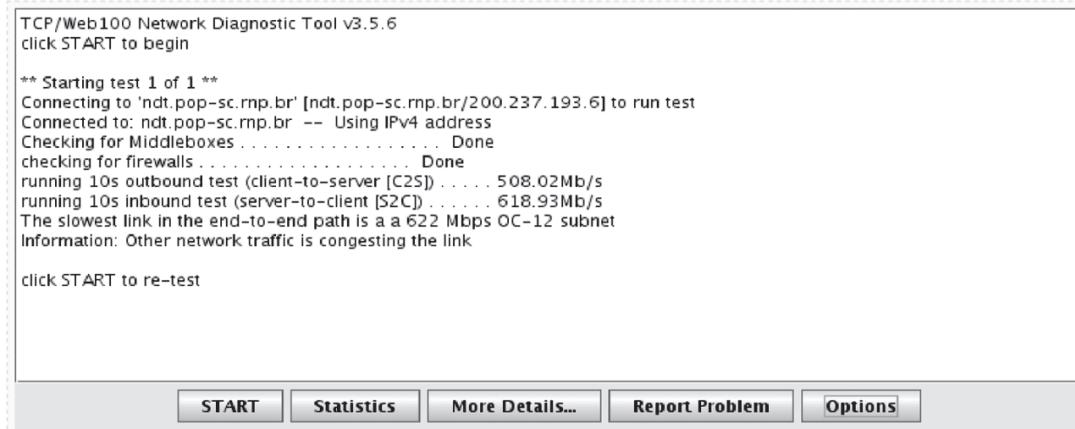
Além disso, pode ser útil na detecção de perdas ocorridas sem congestionamento, como falha na placa de rede ou Interface do Switch, problemas no cabo UTP, sujeira no conector óptico etc. Trabalhos preliminares indicam que é possível distinguir entre perdas ocorridas por congestionamento e ocorridas sem congestionamento.

Possibilita a detecção de conexão half-duplex em caminho fim-a-fim, identificando quando a vazão é limitada por operações half-duplex.

As infraestruturas de redes compartilhadas causam episódios de congestionamentos periódicos, possibilitando a detecção e relatório quando a vazão é limitada por tráfego cruzado ou pelo próprio tráfego gerado pelo NDT.

### O que o NDT não pode fazer

- Dizer exatamente onde está o problema na rede.
- Dizer como os outros servidores estão operando.
- Dizer como os outros clientes estão operando.



**Figura 8.23**  
NDT – Exemplo de uso.

A figura mostra um exemplo do applet Java executado pelo cliente reportando a banda enviada e recebida.

Tenta detectar o menor link no caminho – essa informação não é muito precisa.

Caso o usuário clique em estatísticas e mais detalhes, níveis de diagnóstico mais detalhados serão apresentados.

## perfSONAR

- É uma infraestrutura de monitoramento de desempenho de redes para resolver problemas de desempenho fim-a-fim.
- Arquitetura baseada em serviços.
- É definido em três contextos:
  - Um consórcio de organizações.
  - Um protocolo.
  - Pacotes de software interoperáveis.
- Parceiros:
  - RNP, internet2, GÉANT e Esnet.
- Principais Serviços:
  - MP, MA, LS, AS, entre outros.

O perfSONAR é uma infraestrutura de monitoramento de desempenho de redes, o qual foi projetado para facilitar a resolução de problemas de desempenho fim-a-fim. Contém um conjunto de serviços que entregam medições de desempenho em um ambiente federativo. Esses serviços formam uma camada intermediária entre as ferramentas de medições de

performance e as aplicações de visualização e/ou diagnóstico. Essa camada foi projetada para realizar e trocar ou exportar medições de desempenho entre as redes, usando protocolos bem definidos.

É uma arquitetura orientada a serviços. Isso significa que o conjunto de funções elementares têm sido isoladas e podem ser providas por diferentes entidades chamadas serviços. Todos esses serviços se comunicam entre si usando protocolos bem definidos. O perfSONAR é definido em três contextos: um consórcio de organizações, um protocolo e pacotes de software interoperáveis.

Os principais serviços disponibilizados pelo perfSONAR são: Measurement Point Service, Measurement Archive Service, Lookup Service, Authentication Service, Transformation Service, Resource Protector Service e Toplogy Service, serviços que são definidos pelo GFD.

Os parceiros envolvidos nessa iniciativa são contemplados nas redes da ESnet, GÉANT, internet2 e RNP.

## General Framework Design (GFD)

- É a especificação da arquitetura de monitoramento orientada a serviços.
- Ação conjunta entre a atividade GN2-JRA1 (Performance Measurement) e o internet2 piPEs (End-to-End Performance Initiative, Peformance Environment System).



O General Framework Design (GFD) é a especificação de uma arquitetura de monitoramento orientada a serviços. Ele foi definido por uma ação conjunta entre a atividade GN2-JRA1 (Performance Measurement) e o internet2 piPEs (End-to-End Performance Initiative, Peformance Environment System).

### Identificação e definição de serviços

É definido por:



- Serviços de infraestrutura: LS, TopS e AA.
  - Serviços produtores e consumidores de dados: MP, MA, RP e TS.
- Infraestrutura mínima é composta pelos seguintes serviços:
- MP, MA e LS (AA opcional porém é fortemente recomendado).

O GFD definiu um conjunto de serviços de monitoramento. Ele é definido por serviços produtores e consumidores de dados e por serviços de infraestrutura. No que se refere à infraestrutura, podem-se citar os seguintes serviços:

- Lookup Service (LS);
- Topology Service (TopS);
- Authentication Service (AA).

Para os serviços produtores e consumidores de dados, citamos:

- Measurement Point Service (MP);
- Measurement Archive Service (MA);
- Resource Protector Service (RP);
- Transformation Service (TS).

A infraestrutura mínima para formar uma arquitetura perfSONAR é composta por pelo menos os seguintes serviços: MP, MA e LS, tendo o AA como opcional porém fortemente recomendado.



## MP e MA

Principais funções do MP:

- Realização de medições de rede.
- Publicação dos dados de medições.
- Gerenciamento do serviço.

Principais funções do MA:

- Armazenar dados de medições de rede.
- Buscar e fornecer esses dados.
- Permitir o gerenciamento do serviço.

### **Measurement Point Service (MP)**

É o produtor efetivo dos dados (medições ativas ou passivas). Formalmente, esse serviço não armazena ou transforma os dados. Entre suas principais tarefas, se enquadram:

- Realização de medições de rede;
- Publicação dos dados de medições;
- Gerenciamento do serviço.

Diferentes tipos de funcionalidades de medição podem ser implementadas:

- Atraso unidirecional ou bidirecional, perdas, e jitter;
- Medições baseadas em fluxos;
- Vazão alcançável através de medição ativa de stress.

### **Measurement Archive Service (MA)**

É usado para armazenar e publicar dados históricos. Esse serviço é essencial na análise de tendências e é responsável em garantir alta disponibilidade dos dados. As suas principais atribuições são:

- Armazenar dados de medições de rede;
- Buscar e fornecer esses dados;
- Permitir o gerenciamento do serviço.

Nota-se que para diferentes tipos de dados existem melhores formas de armazenamento, podendo-se adequar melhor a diferentes tipos de banco de dados. Com isso, diferentes MAs podem ser implementados: RRD, BD Relacional, BD XML, entre outros.

## LS e AS

Principais funções do LS:

- Manter as informações sobre a infraestrutura atual existente.
- Manter informações sobre os recursos de cada serviço e publicar para os demais serviços.
- Permite que os serviços sejam visíveis a arquitetura.

Principais funções do AS:

- Autenticar clientes e serviços.

- Manipular informações de atributos de identidades específicas.
- Gerenciar relações de confiança em confederação com outros ASs.
- Permitir o gerenciamento do serviço.

### **Lookup Service (LS)**

Esse é o serviço que realiza a descoberta de outros serviços. Sua atribuição principal é registrar e manter atualizado as informações sobre a infraestrutura do perfSONAR. Essencialmente age como um diretório de serviços. Entre as principais atribuições, encontram-se:

- Manter as informações sobre a infraestrutura atual existente;
- Manter informações sobre os recursos de cada serviço e publicar para os demais serviços;
- Permite que os serviços sejam visíveis a arquitetura.

É parte essencial da arquitetura permitir a descentralização dos serviços. Deve dar informações suficientes para que o usuário saiba se um determinado serviço atende a sua necessidade.

### **Authentication Service (AS)**

O Authentication Service (AS) é o serviço de autenticação. É um serviço opcional na arquitetura do perfSONAR, porém é muito recomendado, especialmente por garantir a autenticidade das transações, entre os serviços.

As necessidades definidas no GFD são:

- Suportar clientes com múltiplas identidades;
- Suporte de autenticação por papéis;
- Formação de comunidades confederadas que aceitem autenticações umas das outras;
- Detalhes de confederação devem ser escondidos de outros serviços dentro de um domínio administrativo.

As tarefas que deve atender são:

- Autenticar clientes e serviços;
- Manipular informações de atributos de identidades específicas;
- Gerenciar relações de confiança em confederação com outros AS;
- Permitir o gerenciamento do serviço.

### **RP, TS e TopS**

Principais funções do RP:

- Mediar o consumo de recursos limitados.

Principais funções do TS:

- Utilizado para agregação, correlação, filtragem e tradução das informações geradas pelo perfSONAR.

Principais funções do TopS:

- Prover informações topológicas sobre as redes disponíveis ao arcabouço perfSONAR.



### **Resource Protector (RP)**

Esse serviço é usado para mediar o consumo de recursos limitados. Possui um componente de agendamento para tratar o consumo de recursos dependentes de tempo. Cada MP administra seus recursos localmente com um gerenciador de recursos interno, e esse gerenciador pode ser configurado para contatar um MP dependendo dos recursos necessários para uma medição.

### **Transformation Service (TS)**

O Transformation Service é o serviço de transformação. É utilizado para fazer o pipelining dos dados, entre os outros serviços do arcabouço. Pode ser usado para realizar qualquer tipo de operação sobre os dados. Fica entre os geradores e os consumidores dos dados.

Existem muitas funções em potencial, entre elas:

- Agregação;
- Correlação;
- Filtragem;
- Tradução.

As operações de transformação podem ser realizadas com dados de diversos produtores. Um exemplo específico de transformação é o Serviço de Topologia.

### **Topology Service (TopS)**

O Serviço de Topologia (TopS) é responsável em prover informações topológicas sobre as redes disponíveis ao arcabouço perfSONAR. É um serviço que transforma as informações coletadas de diversos MPs em informação de topologia através de algoritmos definidos.

As informações que esse serviço provê pode refletir múltiplas camadas de rede. Entender a topologia é essencial para o sistema de medição otimizar as operações.

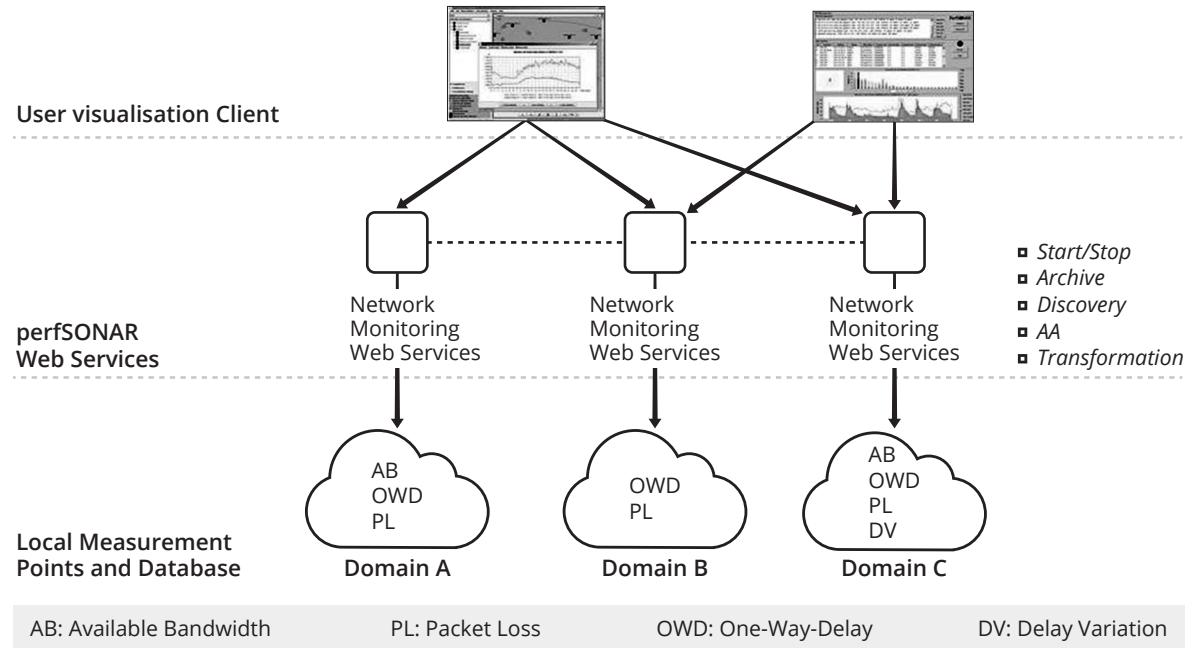
### **Definição da arquitetura/tecnologia**

- Através das premissas mínimas identificadas pelo GFD, foi escolhida a Arquitetura Orientada a Serviço (SOA) para o desenvolvimento do perfSONAR.
- Usam-se tecnologias XML.
- Implementa-se o SOA através de Serviços web.
- O arcabouço de serviços é definido por três camadas bem definidas:
  - Cliente.
  - Serviços.
  - Ferramentas.

A arquitetura do pacote de software é dividida em três camadas:

- **Camada de cliente:** essa é a camada dos clientes, as quais se encontram as ferramentas de visualização e análise dos usuários;
- **Camada de serviços:** essa é a camada dos serviços que compõe a infraestrutura dos serviços definidos pelo GFD;
- **Camada de ferramentas:** na camada de ferramentas se encontram os programas responsáveis em realizar as medições de fato, que são controladas pelos serviços.

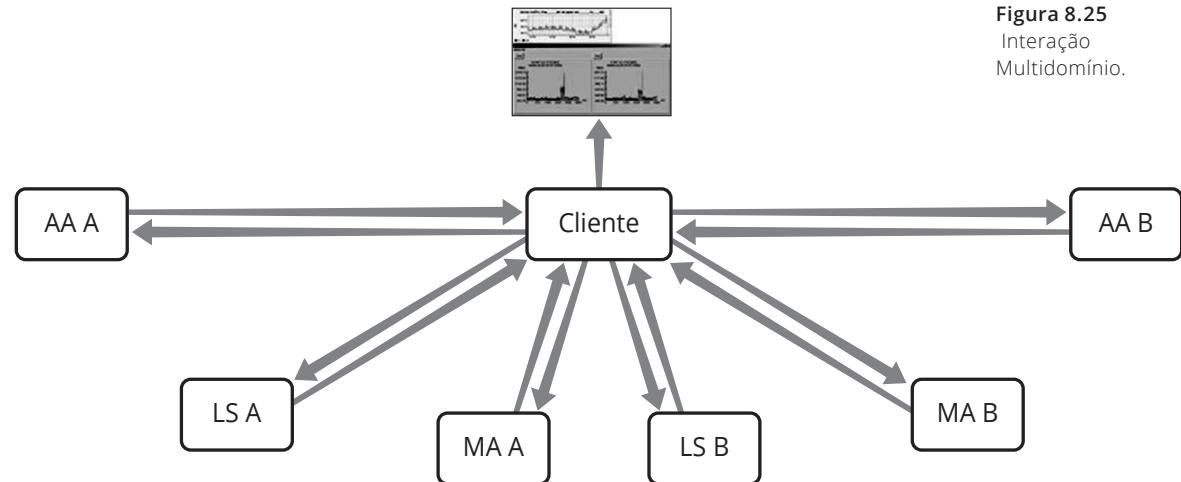




### Interação multidomínio

- É implícito a questão de se possuir múltiplos domínios, pois a infraestrutura é baseada em serviços.
- A interação multidomínio se dá através dos serviços de localização, para descobrir os componentes pertencentes à infraestrutura perfSONAR e/ou através das ferramentas dos clientes diretamente.

**Figura 8.24**  
Definição da arquitetura/ tecnologia.



**Figura 8.25**  
Interação Multidomínio.

No perfSONAR há implícita a questão de se possuir múltiplos domínios. Como a infraestrutura é baseada em serviços, a interação multidomínio se dá através dos serviços de localização, para descobrir os componentes pertencentes à infraestrutura perfSONAR e/ou através das ferramentas dos clientes diretamente.



## Serviços implementados na RNP

- Os componentes do perfSONAR implantados na RNP são: MP, MA, LS e AS.
- Ferramentas de visualização:
  - CACTISonar: para usuários administradores de rede que tenham necessidade de realizar testes regulares agendados.
  - ICE: para usuários finais que desejem realizar testes sob demanda.

Os componentes perfSONAR implantados na malha de medição da RNP são:

- **MPs**: o serviço definido para gerenciar os testes, realizando agendamentos e testes sob demanda, é o Command Line MP (CLMP);
- **MAs**: para realizar o armazenamento dos testes é utilizado o SQL-MA, com o intuito de armazenar dados históricos e o RRD-MA, que guarda informações das medições utilizando arquivos RRD;
- **LS**: o serviço de descoberta implantando é o gLS (escopo global) e hLS (escopo local);
- **AS**: o serviço de autenticação (AS) fornecido pelo perfSONAR foi implantado.

As ferramentas de visualização usadas pela RNP são o CACTISonar, para usuários administradores de rede que tenham necessidade de realizar testes regulares agendados, e o ICE, ferramenta para usuários finais que desejem realizar testes sob demanda.





# 9

## Montagem em laboratório de solução de gerência

objetivos

Aprender a definir requisitos de gerência; Conhecer o Sistema de Registro de Problemas (Trouble Ticket System); Montar uma solução de gerência.

conceitos

Requisitos de gerência; Sistema de Registro de Problemas (Trouble Ticket System); Solução de gerência.

### Introdução

- Definindo seus requisitos de gerência.
  - Identificando o seu estado atual.
  - Avaliando opções de plataformas de gerência.
- Sistema de Registro de Problemas (Trouble Ticket System).
- Montando uma solução de gerência.
  - Fully Automated Nagios (FAN).



Como visto em capítulos anteriores, existem sistemas com grau de integração bastante variável que permitem a operação correta de atividades de uma empresa através de suas várias áreas. Ou então existe a opção de várias plataformas de gerência com uma integração razoável. Cabe ao administrador da rede definir junto com seus pares e superiores qual é o nível de integração e qual é o nível de gerência esperada para a sua rede.

Em redes que contemplam um nível de gerência razoável, o sistema de gerenciamento de rede e serviços costuma encontrar-se alinhado com o funcionamento interno da própria empresa, refletindo com a maior proximidade possível os processos internos da área de TI. Para isso, deve haver adequação de sistemas de gerenciamento de rede, até porque a informação técnica de gerenciamento tem grande impacto no negócio da empresa.

Em vários casos, essas mesmas áreas estão envolvidas:

- Atendimento ao cliente: call center e help desk;
- Network Operation Center (NOC);
- Equipes de engenharia de redes, sistemas e P&D;



- Equipes de manutenção (eletricidade, ar-condicionado, fibras óticas etc.);
- Controle de qualidade do serviço (administração de contratos de TI);
- Vendedores, responsáveis por contas de clientes.

Cada uma dessas áreas utiliza as informações coletadas pelos sistemas de gerência como uma forma de coordenar suas atividades e obter informações sobre cada um dos sistemas que possui responsabilidade.

Entre as atividades esperadas pela gerência da rede, pode-se citar:

- Registro de pedidos e reclamações de serviços;
- Atendimento e manutenção (agendados ou não);
- Acompanhamento e registro da solução;
- Geração de relatórios de SLA (Service Level Agreement).

O processo de autoconhecimento da situação atual da gerência da sua rede é o ponto de partida em qualquer das situações, seguida da definição do escopo da gerência desejável para a sua rede. Um exercício simples para diagnosticar o seu nível atual de gerência é sugerido nas atividades desta sessão.

## Implementando Gerenciamento

- Identificar:
  - Restrições e necessidades – serviços.
  - Recursos de rede, desenhar a topologia da rede.
- Determinar etapas (abrangência e recursos).
- Política de gerência:
  - Indicadores de SLA.
  - Funções e executores.
- Definição de produtos de gerenciamento:
  - Requisitos do produto e do fornecedor.
  - Definição de roteiro de avaliação e seleção.
- Implementação.

O passo seguinte ao autoconhecimento da sua rede é a definição de até onde você pretende chegar com o desenvolvimento da gerência da sua rede. Essa questão, na maioria das vezes, envolve as esferas administrativas da empresa, já que o nível de integração e gerência desejável é diretamente proporcional aos recursos humanos e financeiros destinados a esse fim.

É necessário um planejamento dos processos e serviços que se deseja gerenciar dentro da empresa. Esse planejamento terá como resultado um projeto de implementação, que define as etapas, equipes e acompanhamento adequado, pois vai alterar vários processos internos junto a diversas áreas da empresa – novamente, diretamente relacionado ao grau de gerência definido anteriormente.

Uma maneira simples de realizar esse planejamento é o uso do modelo FCAPS como ponto de partida, agrupando as necessidades de gerenciamento dentro de suas cinco áreas: falha, configuração, contabilização, desempenho e segurança. Dessa forma são divididas as tarefas das equipes, e cada um dos executores responsáveis por cada etapa do projeto de implementação.



Outro ponto que também deve ser definido nesse momento é o SLA. O nível de serviço tem impacto direto na escolha de equipamentos, softwares e fornecedores de serviços a serem utilizados pela empresa, tendo impacto direto na solução de NMS a ser utilizada.

O passo seguinte é a escolha dos produtos a serem utilizados na solução de gerenciamento, de forma a contemplar todos os requisitos listados (incluindo aqui o financeiro). É importante definir o modelo de distribuição das aplicações, baseado nas características do ambiente a ser gerenciado, como a possibilidade de criar diferentes domínios de gerenciamento, comuns em redes maiores. Tópicos como protocolos a serem usados e a modelagem da informação de gerenciamento, com base no modelo adotado, também serão importantes.

Em muitos casos, será concebido um Network Operations Center (NOC): uma infraestrutura centralizada de apoio à atividade de gerência, composta de recursos de rede (sistemas e humanos) para uma melhor resposta da atividade de gerenciamento.

Pode ser necessário o desenvolvimento de novas aplicações de gerenciamento que atendam às necessidades locais. Muitas vezes são customizações ou scripts; em outras, sistemas inteiros complementam o conjunto de aplicações. O NMS a ser definido está diretamente relacionado a esses itens.

Outros tópicos não menos importantes são o planejamento do suporte e do treinamento da equipe.

Instalação e configuração dos produtos de gerenciamento:

- Identificação e contratação dos executores;
- Preparação da infraestrutura;
- Treinamento da equipe.



## Avaliação de Plataformas de Gerenciamento

- Fornecedor: localização, suporte, parcerias e integração.
- Arquiteturas de protocolos suportados.
- Plataformas de hardware e SO suportadas.
- Suporte a um ou mais padrões de gerência.
- Interface com programas aplicativos.
- Integração com outras plataformas.
- Documentação.



Conforme visto nas sessões anteriores, pontos como integração e modelo de gerência a ser utilizados são cruciais nessa definição. A maior integração de facilidades de modelagem de processos geralmente são características difíceis de obter em plataformas gratuitas.

Normalmente a ferramenta está diretamente relacionada ao conhecimento e disponibilidade de recursos humanos para a sua integração, ou então à aquisição de uma plataforma comercial com suporte do próprio fornecedor.

No caso de um NMS comercial, além do custo inicial, deve-se lembrar da necessidade de um contrato de suporte para que se tenham as atualizações necessárias do produto – uma plataforma com três anos sem atualização costumeiramente se torna obsoleta à configuração de novos serviços e equipamentos. No caso de um NMS freeware/open source, existe sempre o risco de o desenvolvimento ser descontinuado, então o tempo de vida que essa plataforma já possui e a regularidade das suas atualizações deve ser levada em conta.



O que avaliar em uma solução de gerência:

- Visão gráfica dos elementos gerenciados;
- Relacionamentos entre elementos de rede;
- Modelos adotados (agente-gerentes, objetos distribuídos etc.);
- Visualização de notificações de eventos e alarmes; graus de severidade para os eventos reportados;
- Logs de eventos;
- MIB Browser;
- Utilização de SGBD padrão;
- Importação e exportação de informações da MIB;
- Software livre: provedores de serviços disponíveis;
- Interface com outros programas aplicativos;
- APIs para desenvolvimento de aplicações de gerenciamento; linguagens oferecidas; Integrated Development Environment (IDE); ferramentas;
- Facilidade de operação e curva de aprendizagem da ferramenta;
- Integração entre as diversas gerências FCAPS e sistemas de ticket.

## Trouble Ticket Systemas (TTS)

Registra chamados de problemas que estejam ocorrendo. Informações constantes em um TTS:

- Hora e data do início do problema.
- Operador que está abrindo o registro.
- Descrição e seriedade do problema.
- Equipamentos envolvidos e seus detalhes.
- Destinatário (e responsável) do registro.
- Próxima ação e recomendações de ação baseada em ocorrências semelhantes.

Depois:

- Hora e data da resolução.
- Descrição da solução.
- Outros.

A utilização de um “Sistema de Registro de Problema” (Trouble Ticket System) auxilia o NOC no diagnóstico do problema e permite criar um Banco de Dados (BD) de experiências com problemas, viabilizando o uso de sistemas especialistas na solução. Muitos desses sistemas identifica já na abertura do problema possíveis soluções baseadas no sistema em questão e nas palavras chaves que descrevem questão a ser resolvida.

Os TTSs também agilizam o processo de controle da rede, porque permitem comunicação direta com os responsáveis pelo NOC [RFC 1297], seja pela abertura de muitos chamados semelhantes, seja pelo repasse direto de informações entre a fila de problemas do service desk diretamente para a equipe do NOC.

## Funções e características de um Sistema TTS

O “Registro de Problema” (o cadastro que é realizado a cada novo registro no sistema TTS) deve prover um histórico completo do problema de forma que qualquer operador possa tomar alguma iniciativa sem que para isso tenha de consultar outro operador:

- ▣ Deve permitir um melhor escalonamento de problemas atribuindo prioridades a estes. Os supervisores e operadores poderão tomar decisões acerca da necessidade ou não de mais pessoal pela carga corrente do “Centro de Operações de Rede”. Seria interessante permitir que a prioridade dos registros mudasse de acordo com a hora do dia ou em resposta a alarmes de tempo;
- ▣ Se o TTS for suficientemente integrado ao sistema de e-mail, então alguns registros podem ser despachados diretamente aos responsáveis;
- ▣ Deve-se atribuir um “timeout” para cada registro de problema. Caso o problema não seja resolvido em tempo, automaticamente é acionado um alarme. A fim de se evitar “postergação indefinida”, pode-se adotar um escalonamento baseado no tempo de espera, no tipo de rede e na severidade do problema;
- ▣ Caso a empresa opere em mais de um Centro de Operações de Rede, devem-se canalizar os registros ao grupo de engenheiros, operadores ou representantes de clientes responsáveis por aquela rede de onde provem o registro de problema;
- ▣ Os sistemas de ticket normalmente fornecem mecanismos para a obtenção de estatísticas, tais como: “Tempo Médio entre Falhas” (MTBF – Mean Time Between Failure), “Tempo Médio de Conserto” (MTBR Mean Time Between Repair) e “Tempo Médio para Falhas” (MTTF – Mean Time ToFailure). Uma coleta e análise apropriadas de tais estatísticas permitem que se tomem medidas preventivas a eventuais falhas em dispositivos do sistema;
- ▣ Do ponto de vista de gestão de equipes, os TTSSs permitem que se obtenham também informações sobre o tempo médio de resolução de problemas por equipe ou indivíduo da equipe, obtendo também informações da sua produtividade.

Uma opção no uso de TTS adotada em várias instituições é a permissão de abertura direta de chamados ao usuário final, diminuindo a burocracia na solução de problemas e permitindo linha direta entre o usuário e quem mantém os serviços dentro da empresa. Alguns TTSSs possuem integrados a eles sistemas de pergunta-resposta, como um wiki que exprime a resolução dos problemas mais comuns encontrados pelo usuário, ou simplesmente a comunicação de problemas já conhecidos e em tratamento, evitando que o usuário registre como um novo problema algo que já se encontra em tratamento.

Algumas opções em FS/OSS são:

- ▣ **Request Tracker (RT):** sistema utilizado por várias organizações para rastreamento de bugs, chamados de help desk, atendimento ao cliente, documentação de workflow, gerenciamento de mudanças e operações de rede. É um software bastante maduro e com atualizações constantes desde a sua primeira versão. (<http://bestpractical.com/rt>);
- ▣ **OTRS:** sistema de gerenciamento para uma ampla gama de processos de negócios. Baseado em um conjunto de funções construídas sobre um “ticket” (problem report), o OTRS permite o suporte tanto de TI como de qualquer outro departamento da empresa que preste serviços aos usuários (<http://otrs.org/>);
- ▣ **Ocomon:** projeto nacional criado em 2002, atualmente possui módulos para cadastro, acompanhamento, controle, consulta de ocorrências de suporte, além de um módulo de inventário não automatizado. Em 2004, foram adicionadas características de gerenciamento de SLAs (<http://ocomonphp.sourceforge.net/>);



- **Simpleticket:** sistema de controle de tickets, possui funcionalidades como: informações básicas sobre o cliente; criação de ticket; rastreamento da vida do ticket (aberto, fechado, em análise etc.); notificações por e-mail sobre status do ticket; etc. (<http://simpleticket.net/>);
- **Liberum:** fornece uma maneira simples e fácil de usar a interface web para gerenciar e controlar os problemas de suporte técnico, sendo necessário para seu funcionamento um servidor Windows configurado com o Internet Information Service (IIS). Seu foco é para pequenas empresas (<http://www.liberum.org/>).

Algumas opções comerciais:

- **BMC Remedy:** possivelmente o mais completo sistema de TTS existente no mercado. Entre seus pontos fortes está a integração com as melhores plataformas de NMS comerciais (<http://www.bmc.com/remedy/>).
- **Frontrange Heat Service Management:** oferece uma solução bastante completa para gerência de serviços, permitindo controlar requisições de usuários e definir um workflow para as requisições recebidas, conforme previsto pelo modelo ITIL (<http://www.frontrange.com/heat/products/service-management>).

## Fan (Fully Automated Nagios)

- O principal objetivo dessa distribuição é fornecer uma instalação customizada da solução Nagios, incluindo a maioria das ferramentas/plugins fornecidos pela Comunidade Nagios.
- As principais ferramentas integradas são:
  - Nagios: aplicativo de monitoramento central.
  - Nagios plugins: plugins para monitorar diferentes tipos de equipamentos/serviços.
  - Centreon: interface web para o Nagios.
  - NagVis: permite a criação de mapas para visualizar o sistema de monitoramento (geográfica, funcional, pelos serviços ...).
  - NDOUtils: armazena os dados do Nagios em um banco de dados MySQL.
  - NRPE: torna possível monitorar os servidores Windows (o daemon NRPE não é fornecido).
  - NaReTo (Nagios Reporting Tools): ferramenta de geração de relatório de disponibilidade.



FAN significa Fully Automated Nagios (“Nagios Totalmente Automatizado”). O principal objetivo dessa distribuição é fornecer uma instalação customizada da solução Nagios, incluindo a maioria das ferramentas/plugins fornecidos pela Comunidade Nagios.

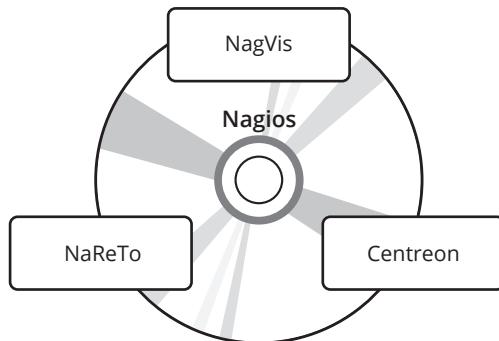
O FAN fornece uma imagem ISO baseado no CentOS. Todos os pacotes do CentOS permanecem disponíveis, de modo que você pode manter todas as vantagens do CentOS, além de possuir todas as ferramentas Nagios já instaladas e configuradas para você.

As ferramentas integradas:

- **Nagios:** aplicativo de monitoramento central;
- **Nagios plugins:** plugins para monitorar diferentes tipos de equipamentos/serviços;
- **Centreon:** interface web para o Nagios;
- **NagVis:** permite a criação de mapas para visualizar o sistema de monitoramento (geográfica, funcional e pelos serviços);
- **NDOUtils:** armazena os dados do Nagios em um banco de dados MySQL;

- **NRPE:** torna possível monitorar os servidores Windows (o daemon NRPE não é fornecido);
- **NaReTo (Nagios Reporting Tools):** ferramenta de geração de relatório de disponibilidade.

**Figura 9.1**  
Ferramentas integradas ao Nagios.



## Nagios

Conhecido como Netsaint, é um aplicativo desenvolvido com o objetivo de realizar o monitoramento de ativos de uma rede. Basicamente é dividido em três partes:

1. Uma engine central que agenda a realização de tarefas de monitoramento.
2. Uma interface web, o que dá uma visão geral do sistema de informação e as possíveis anomalias.
3. Um conjunto de plug-ins que podem ser configurados de acordo com as necessidades do administrador da rede.

Principais características:

- Monitoramento de serviços de rede (SMTP, POP3, HTTP, NNTP, ICMP e SNMP).
- Monitoramento de recursos de computadores ou equipamentos de rede (carga do processador, uso de disco, logs do sistema).
- Monitoração remota suportada através de túneis encriptados SSH ou SSL.
- Desenvolvimento simples de plugins (fácil customização). Os plug-ins podem ser desenvolvidos em diferentes linguagens de programação: shell script (bash, ksh ...), C, Perl, Python, Ruby, PHP, C # etc.
- Checagem paralelizada.

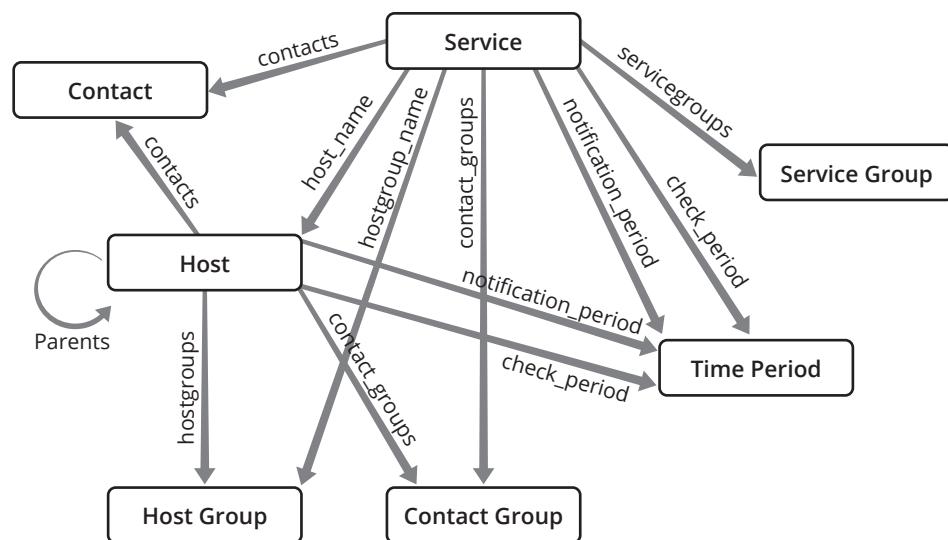
Capacidade de definir a rede hierarquicamente.

- Capacidade de notificar quando um serviço ou equipamento apresenta problemas e quando o problema é resolvido (via e-mail, pager, SMS ou qualquer outro meio definido pelo usuário por plugin).
- Capacidade de definir tratadores de eventos.
- Rotação automática de log.
- Excelente interface web para visualização do status atual da rede, notificações, histórico de problemas, arquivos de log etc.
- Cada teste retorna um estado especial:
  1. OK (tudo bem).
  2. WARNING (o limiar de alerta foi ultrapassado).
  3. CRITICAL (o serviço tem um problema).
  4. UNKNOWN (é impossível saber o estado do serviço).



Nagios (anteriormente conhecido como Netsaint), é um aplicativo GPL (General Public License) desenvolvido com o objetivo de realizar o monitoramento de ativos de uma rede. Ele monitora os hosts e serviços, e informa sobre o estado do sistema. Foi desenvolvido de forma modular, dividido em três partes:

1. Uma engine central, que agenda a realização de tarefas de monitoramento;
2. Uma interface web, o que dá uma visão geral do sistema de informação e as possíveis anomalias;
3. Um conjunto de plug-ins que podem ser administrador da rede para o monitoramento de cada serviço ou recurso disponível em todos os computadores ou dispositivos de rede do Sistema de Informação. Graças ao sistema de plug-ins, o Nagios pode ser expandido existindo vários com essas extensões para o software.



**Figura 9.2**  
Objetos disponíveis no Nagios e seus relacionamentos.

## Objetos disponíveis no Nagios e seus relacionamentos

A figura anterior representa os principais objetos disponíveis no Nagios e como estes estão associados entre si. Como pode ser observado, o fundamento central de um sistema de gerência de falhas é identificar problemas nos serviços da rede. Portanto, as primeiras definições para a implantação desse tipo de sistema é identificar quais serão os serviços (Service) e dispositivos (hosts) monitorados, e quem será avisado em caso de falhas (contact).

## Plugins do Nagios

Uma das muitas vantagens do Nagios é a sua excelente API para criação de plugins. Entre os principais plug-ins, destacamos:

- Nagios::Plug-in::SNMP (./check\_snmp\_procs, ./check\_snmp\_loads etc.).
- HTTP Scraping Plug-ins (./check\_http).
- Testing Telnet/SSH-like Interfaces (./check\_port).
- Monitoring LDAP (./check\_ldap\_replication.pl)
- Monitoring Databases (./check\_mysql, ./check\_postgres etc.).



### Saiba mais

O Nagios possui formas interessantes de agrupamento de Serviço (Service Group), Contatos (Contact Group) e dispositivos (Host Group), que permitem customizar e facilitar o desenho da solução como um todo.

Uma das muitas vantagens do Nagios é a sua excelente API para criação de plugins. A API de plugins do Nagios é aberta, fácil de usar e bem documentada. Você pode desenvolver seus plugins em qualquer linguagem que você quiser, permitindo monitorar tudo o que você considerar importante. Principais plug-ins:

- **Nagios::Plug-in::SNMP (./check\_snmp\_procs, ./check\_snmp\_loads, etc.):** permite o monitoramento, via SNMP, de informações como: uso de CPU, utilização de swap, uso de memória RAM e uso de partição de disco, entre outros;
- **HTTP Scraping Plug-ins (./check\_http):** define códigos de erro, realiza testes com robôs e monitora atividades referente ao protocolo HTTPe linguagem HTML;
- **Testing Telnet/SSH-like Interfaces (./check\_port):** verifica se os serviços Telnet e SSH estão ativos e disponíveis na máquina alvo. Esse plugin pode, também, ser utilizado para verificar se uma determinada porta TCP está aberta na máquina alvo;
- **Monitoring LDAP (./check\_ldap\_replication.pl):** permite avaliar a replicação, integração e escalabilidade de servidores LDAP;
- **Monitoring Databases (./check\_mysql, ./check\_postgres etc.):** conecta a um banco de dados SGBP e realiza um comando SQL com o objetivo de medir o desempenho e resposta.

### Como monitorar roteadores e switches utilizando os plugins do Nagios

Dispositivos de rede gerenciáveis oferecem grande variedade de informações através de SNMP, por isso, pode ser difícil decidir o que é importante para monitorar.

Para os exemplos a seguir, vamos assumir que:

**Tabela 9.1**  
Exemplos de monitoramento no Nagios.

- Todos os dispositivos gerenciados são do fabricante CISCO;
- A versão do protocolo SNMP é a 2;
- A string referente à comunidade pública está armazenada na variável “snmp\_community”.

Objetivo	MIBs necessárias	OIDs necessários	Exemplo de comando
Utilização de CPU	CISCO-PROCESS-MIB ENTITY-MIB	CISCO-PROCESS-MIB cpmCPUTotal5secRev: 1.3.6.1.4.1.9.9.109.1.1.1.6 cpmCPUTotal1minRev: 1.3.6.1.4.1.9.9.109.1.1.1.7 cpmCPUTotal5minRev: 1.3.6.1.4.1.9.9.109.1.1.1.8 cpmCPUTotalPhysicalIndex: 1.3.6.1.4.1.9.9.109.1.1.1.2 ENTITY-MIB entPhysicalName: 1.3.6.1.2.1.47.1.1.1.7	/check_snmp_cisco_cpu.pl --hostname rtr1.example.com --snmp-version 2c --rocommunity mycommunity -w 90 -c 95
Utilização de memória	CISCO-MEMORY-POOL-MIB	ciscoMemoryPoolName: 1.3.6.1.4.1.9.9.48.1.1.1.2 ciscoMemoryPoolUsed: 1.3.6.1.4.1.9.9.48.1.1.1.5 ciscoMemoryPoolFree: 1.3.6.1.4.1.9.9.48.1.1.1.6	./check_snmp_cisco_mem_pool.pl --hostname rtr1.example.com --snmp-version 2c --rocommunity mycommunity -w 90 -c 95



Objetivo	MIBs necessárias	OIDs necessários	Exemplo de comando
Controle de temperatura	ENTITY-MIB CISCO-ENTITY-SENSOR-MIB	ENTITY-MIB: entPhysicalDescr: 1.3.6.1.2.1.47.1.1.1.2 CISCO-ENTITY-SENSOR-MIB entSensorType: 1.3.6.1.4.1.9.9.91.1.1.1.1 entSensorScale: 1.3.6.1.4.1.9.9.91.1.1.1.2 entSensorValue: 1.3.6.1.4.1.9.9.91.1.1.1.4 entSensorStatus: 1.3.6.1.4.1.9.9.91.1.1.1.5	<pre>./check_snmp_cisco_temp.pl --hostname 192.168.3.1 --snmp-version 2c --rocommunity mycommunity --warning 30 --critical 36 --snmp-max-msg-size 50000 --sensor-regex inlet</pre>
Utilização de banda	IF-MIB ETHERLIKE-MIB	IF-MIB ifDescr: 1.3.6.1.2.1.2.2.1.2 ifSpeed: 1.3.6.1.2.1.2.2.1.5 ifOperStatus: 1.3.6.1.2.1.2.2.1.8 ifInOctets: 1.3.6.1.2.1.2.2.1.10 ifOutOctets: 1.3.6.1.2.1.2.2.1.16 ETHERLIKE-MIB: dot3StatsIndex: 1.3.6.1.2.1.10.7.2.1.1 dot3StatsDuplexStatus: 1.3.6.1.2.1.10.7.2.1.19	<pre>./check_snmp_if_bw_util.pl --snmp-version 2c --hostname rtr1.example.com --rocommunity mycommunity --warning 'in_util,gt,90:out_util,gt,90' --critical 'in_util,gt,98:out_util,gt,98' --interface FastEthernet0/1 --sleep-time 5 --max-speed 100m</pre>
Utilização de CPU	UCD-SNMP-MIB	Raw user ticks: .1.3.6.1.4.1.2021.11.50.0 Raw nice ticks: .1.3.6.1.4.1.2021.11.51.0 Raw system ticks: .1.3.6.1.4.1.2021.11.52.0 Raw idle ticks: .1.3.6.1.4.1.2021.11.53.0 Raw wait ticks: .1.3.6.1.4.1.2021.11.54.0 Raw kernel ticks: .1.3.6.1.4.1.2021.11.55.0 Raw interrupt ticks: .1.3.6.1.4.1.2021.11.56.0	<pre>./check_net_snmp_cpu.pl --hostname host.example.com --snmp-version 3 --auth-username user --auth-password password --auth-protocol md5 -c 'user,gt,80:system,gt,80' -w 'idle,gte,10'</pre>



Objetivo	MIBs necessárias	OIDs necessários	Exemplo de comando
Uso de partição	HOST-RESOURCES-MIB	hrFSMountPoint: 1.3.6.1.2.1.25.3.8.1.2 hrFSIndex: 1.3.6.1.2.1.25.3.8.1.1 hrFSStorageIndex: 1.3.6.1.2.1.25.3.8.1.7 -> link to hrStorageEntry for this device hrFSType: 1.3.6.1.2.1.25.3.8.1.4 -> FS type from hrFSTypes HR FS Types: 1.3.6.1.2.1.25.3.9 hrStorageDescr: 1.3.6.1.2.1.25.2.3.1.3.1 hrStorageAllocationUnits: 1.3.6.1.2.1.25.2.3.1.4 hrStorageSize: 1.3.6.1.2.1.25.2.3.1.5 hrStorageUsed: 1.3.6.1.2.1.25.2.3.1.6 hrStorageAllocationFailures: 1.3.6.1.2.1.25.2.3.1.7 hrStorageType: 1.3.6.1.2.1.25.2.1 - type of storage device; e.g., (hrStorageFixed) Disk, hrStorageRemovableDisk) Device Type Index: 1.3.6.1.2.1.25.3.2.1.2	./check_snmp_hr_storage.pl --hostname 192.168.3.1 --snmp-version 3 --auth-username my_user --auth-password my_pass -w 90 -c 95 -U % -P all
Carga	UCD-SNMP-MIB	1-minute load average: .1.3.6.1.4.1.2021.10.1.3.1 5-minute load average: .1.3.6.1.4.1.2021.10.1.3.2 15 minute load average: .1.3.6.1.4.1.2021.10.1.3.3	./check_net_snmp_load.pl -H hostname --snmp-version 3 --auth-username joesmith --auth-password mypassword -w 20:15:10 -c 40:30:20
Número de processos por status	HOST-RESOURCES MIB	hrSWRunName: 1.3.6.1.2.1.25.4.2.1.2 hrSWRunPath: 1.3.6.1.2.1.25.4.2.1.4 hrSWRunParameters: 1.3.6.1.2.1.25.4.2.1.5 hrSWRunStatus: 1.3.6.1.2.1.25.4.2.1.7	./check_snmp_procs.pl --hostname host1.example.com --snmp-version 3 --auth-username myuser --auth-password mypass --auth-protocol md5 ./check_snmp_procs.pl -mode count --match /bin/httpd:apache --match 'mysqld.+basedir:mysql' --critical apache,lt,1:apache,gt,150:mysql,lt,1:mysql,gt,20 --snmp-max-msg-size 50000 ou ./check_snmp_procs.pl --hostname host1.example.com --snmp-version 3 --auth-username myuser --auth-password mypass --auth-protocol md5 ./check_snmp_procs.pl --mode state --match 'mysqld.+basedir:mysql' --warning runnable,gt,30 --critical total,gt,100 --snmp-max-msg-size 50000

Para saber mais, acesse o site oficial do Nagios: <http://www.nagios.org/>. E conheça seus plug-ins: <http://nagiosplug.sourceforge.net/>.



## Centreon



- Tem como objetivo fornecer uma interface amigável, o que torna possível para um grande número de usuários (incluindo pessoas não técnicas) ver o estado do sistema, especialmente com gráficos.
- Suas principais características são:
  - Interface intuitiva e customizável multiusuário.
  - Gestão de todos os arquivos de configuração do Nagios (cgi, nagios.cfg ...).
  - Configuração do módulo de carga do Nagios.
  - Compatibilidade com Nagios 1.x, 2.x Nagios e Nagios 3.x.
  - A verificação da validade de configuração com o depurador Nagios.
  - Configuração de servidor de rede/hardware arquivos utilizando arquivos de identificação, que incluem todas as informações básicas sobre esses tipos de recursos.
  - Representações gráficas avançadas e personalizável.
  - Gestão dos direitos de acesso, incluindo os recursos, bem como páginas de interface.
  - Modular.
  - Um relatório de incidente completo.
  - Um sistema de cálculo em tempo real para avaliar a qualidade do serviço, que notifica o utilizador sempre que diminui a qualidade do serviço.
  - Um mapa em Java, que oferece uma versão simplificada do estado do sistema de informação (de propriedade da Companhia Merethis).

Centreon é um software de monitoramento de rede com base na ferramenta open source Nagios, tendo como objetivo fornecer uma interface amigável que torna possível para um grande número de usuários (incluindo pessoas não-técnicas) ver o estado do sistema, especialmente com gráficos.



Em julho de 2007, o software Oréon mudou de nome para se tornar Centreon.

## Nareto e NagVis



- Nareto: utiliza as informações Nagios para fornecer visões de alto nível para diferentes grupos de usuários.
- NagVis: módulo de visualização que torna possível a criação de pontos de vista funcionais de monitoramento. NagVis pode usar como plano de fundo um diagrama da rede, permitindo ao Nagios atualizar o diagrama em tempo real.

O Nagios Reporting Tools (NaReTo) utiliza as informações Nagios para fornecer visões de alto nível para diferentes grupos de usuários. É possível criar uma árvore de navegação com NaReTo: mediante a atribuição de direitos específicos para alguns nós da árvore, permitindo restringir a visão. Três tipos de visualização estão disponíveis atualmente: em tempo real, relatórios, História e Acompanhamento dos alertas.

NagVis é o módulo de visualização. Ele torna possível a criação de pontos de vista funcionais de monitoramento. NagVis pode utilizar como plano de fundo um diagrama da rede, permitindo ao Nagios atualizar o diagrama em tempo real.



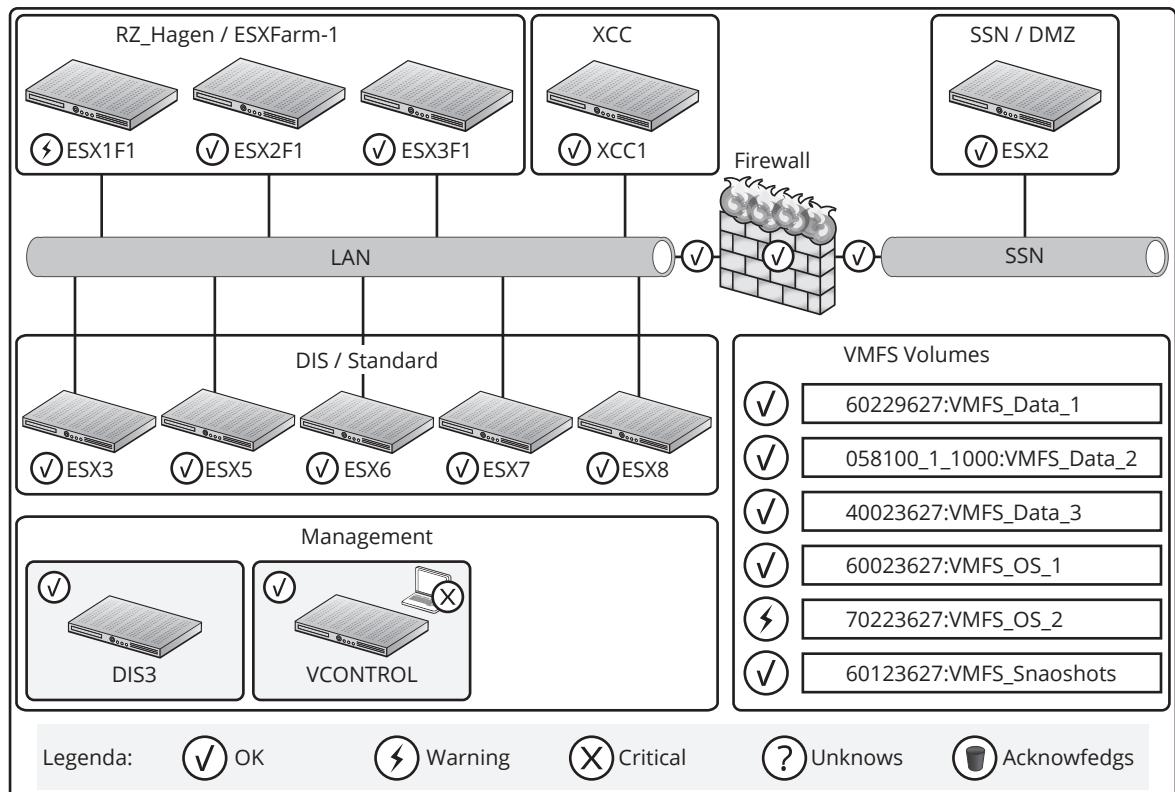


Figura 9.3

NagVis.

## Primeiras configurações

Antes da primeira linha de comando, é importante definir com precisão os requisitos. As seguintes perguntas devem ser feitas:

- Quais dispositivos serão monitorados?
- Quais serviços serão monitorados?
- Quem vai receber os e-mails ou comunicados de alerta?
- Quem vai usar essa plataforma e modificá-la?

Não existe um “método de milagre”, mas os conselhos a seguir podem ser úteis:

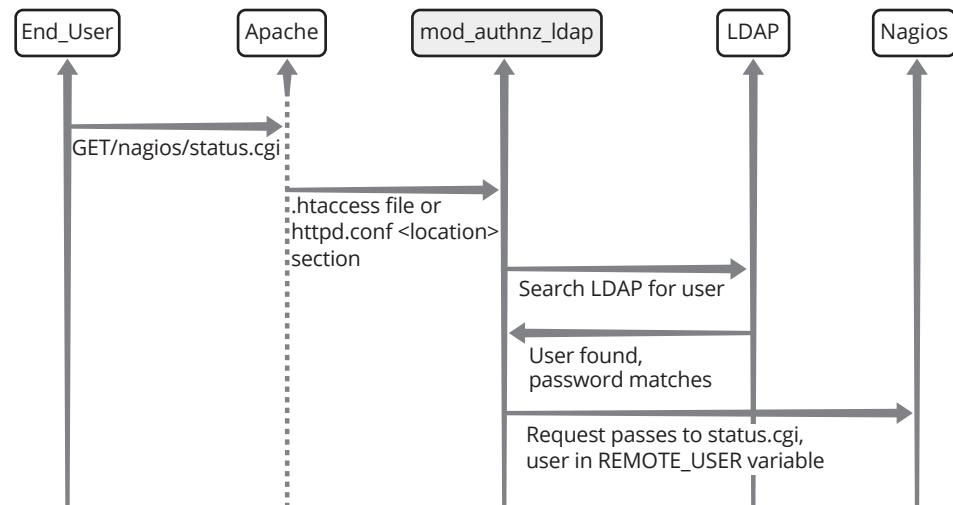
- Faça uma lista de todos os dispositivos a serem monitorados (nome e endereço IP).
- Identifique os serviços críticos e associe-os aos dispositivos.
- Defina uma política de comunicação de alerta (definição de contatos e grupos de contato).
- Crie um diagrama de rede que especifique a dependência entre os dispositivos.

## Integração Nagios com OpenLdap

- O Nagios não possui integração com o OpenLDAP.
- A única forma de utilizarmos um usuário do LDAP para gerenciar o servidor Nagios é configurar o servidor Apache para autenticar no servidor LDAP.

O Nagios não possui integração com o OpenLDAP. A única forma de utilizarmos um usuário do LDAP para gerenciar o servidor Nagios é configurar o servidor Apache para autenticar no servidor LDAP. Para tal, torna-se necessário instalar no servidor Apache o módulo de autenticação LDAP (mod\_authnz\_ldap) e configurar, no diretório do Nagios, o arquivo .htaccess seção <Location> o módulo mod\_authnz\_ldap.

No Nagios será necessário desativar qualquer mecanismo de autenticação, garantindo o acesso de qualquer objeto para qualquer usuário. A figura a seguir representa como seria esse processo de autenticação.



**Figura 9.4**  
Nagios e LDAP.

## Integração Nagios com Splunk

- Splunk é um produto comercial que atua como uma espécie de “Google” para os registros gerados pelo Nagios. O Nagios nos permite monitorar qualquer dispositivo ou serviço, enquanto o Splunk nos permite pesquisar praticamente qualquer formato de arquivo ou produção textual.
- Após instalar o Splunk, para integração, modifique o arquivo de configuração do Nagios (*cgi.cfg*), modificando as seguintes opções:

```
enable_splunk_integration=1
splunk_url=http://splunk.localhost.net:8000/
```

Splunk é um produto comercial que atua como uma espécie de “Google” para os registros gerados pelo Nagios. O Nagios nos permite monitorar qualquer dispositivo ou serviço, enquanto o Splunk nos permite pesquisar praticamente qualquer formato de arquivo ou produção textual.

Após instalar o Splunk, para integração, modifique o arquivo de configuração do Nagios (*cgi.cfg*), modificando as seguintes opções:

```
enable_splunk_integration=1
splunk_url=http://splunk.localhost.net:8000/
```

## Integração Nagios com o Cacti

- Quando integrado com o Nagios, MySQL e SNMPTT, o Cacti pode ser usado para criar vários relatórios de acompanhamento de tendências e gráficos.
- Uma das dificuldades de integração Cacti e Nagios é a autenticação. A utilização do módulo mod\_authnz\_ldap do Apache vai trazer os mesmos problemas já citados.
- Outra forma de integrar os dois programas é o uso do Cacinda add-on (<http://cacinda.sf.net/>).

Cacti ([www.cacti.net](http://www.cacti.net)) é uma aplicação open source que utiliza a ferramenta de dados round-robin (RRD Tool) para criar gráficos de desempenho e disponibilidade. Quando integrado com o Nagios, MySQL e SNMPTT, o Cacti pode ser usado para criar vários relatórios de acompanhamento de tendências e gráficos.

Uma das dificuldades de integração Cacti e Nagios é a autenticação. O Cacti suporta autenticação LDAP, enquanto a autenticação do Nagios não suporta tal mecanismo de autenticação. A utilização do módulo mod\_authnz\_ldap do Apache vai trazer os mesmos problemas já citados.

Outra forma de integrar os dois programas é o uso do Cacinda add-on (<http://cacinda.sf.net/>). O Cacinda é um programa feito em PHP, que permite que você configure painéis para dispositivos em sua rede por tipo de dispositivo e, em seguida, exiba informações a partir desses dispositivos no Nagios. Ele acessa diretamente o banco de dados do Cacti e permite a autenticação em ambos os sistemas (Cacti/Nagios).

## Integração Nagios com Puppet

- O Puppet (<http://www.reductivelabs.com/projects/puppet>) é um software open source utilizado para automatização das atividades de administração e gerenciamento.
- O Puppet e o Nagios formam uma boa equipe. Puppet ajuda a automatizar muitos tipos de tarefas de administração do sistema, enquanto o Nagios, por outro lado, é muito bom em fornecer uma visão de “fora para dentro” de aplicações, sistemas e redes.

O Puppet (<http://www.reductivelabs.com/projects/puppet>) é um software open source utilizado para automatização das atividades de administração e gerenciamento. As políticas do sistema são configuradas usando uma linguagem declarativa de configuração muito semelhante ao Ruby. Cada servidor da rede executa um agente Puppet, que recupera a sua configuração e a envia para o servidor central Puppet. Uma vez que um agente tem uma configuração válida, ele vai periodicamente aplicar as políticas e regras que recebe do servidor central no sistema gerenciado e pode também enviar os resultados dessa aplicação para o servidor central de forma a produzir relatórios.

O Puppet e o Nagios formam uma boa equipe. Puppet ajuda a automatizar muitos tipos de tarefas de administração do sistema enquanto o Nagios, por outro lado, é muito bom em fornecer uma visão de “fora para dentro” de aplicações, sistemas e redes. Um arranjo típico seria o Nagios produzindo relatórios de status de um serviço em um host gerenciado, enquanto o Puppet executa as tarefas de administração de sistemas, que normalmente seriam feitas com intervenção humana (figura a seguir).



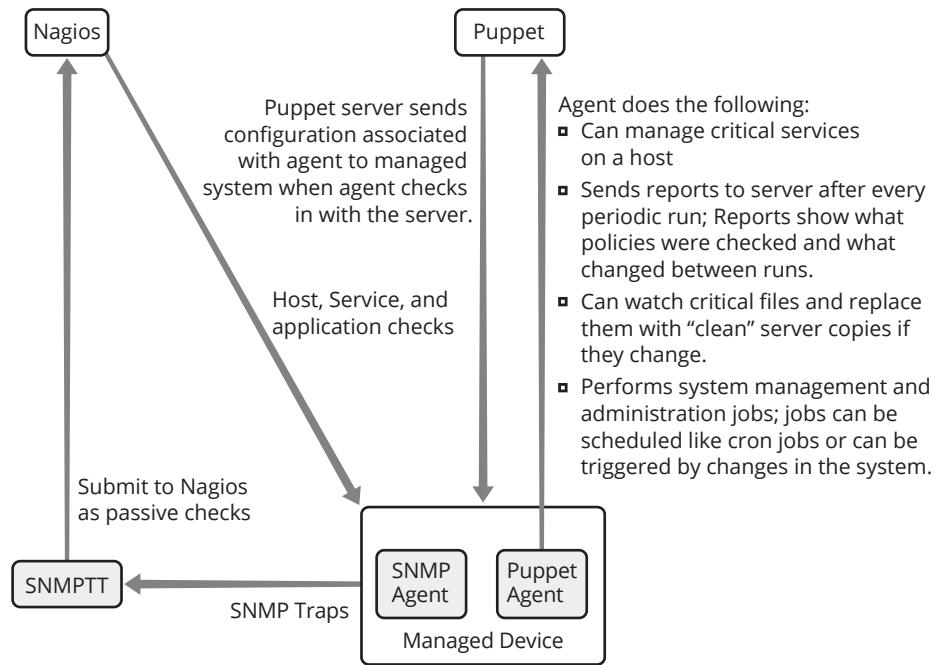


Figura 9.5  
Puppet e Nagios.

## Integração Nagios com TTS

- Uma funcionalidade interessante é permitir que o Nagios crie, automaticamente, tickets de problemas em seu sistema de chamados.
- A única recomendação é que essa funcionalidade deve ser avaliada com cuidado, com o objetivo de evitar falsos positivos.

Uma funcionalidade interessante é permitir que o Nagios crie, automaticamente, tickets de problemas em seu sistema de chamados. A única recomendação é que essa funcionalidade deve ser avaliada com cuidado, com o objetivo de evitar falsos positivos.

O Nagios possui um grande número de macros que permite que os scripts possam reunir informações sobre o host, serviços etc. As macros podem ser empregadas por manipuladores de eventos e executar scripts para envio de e-mail, de SMS ou inserir um chamado no seu sistema de trouble ticketing, por exemplo.

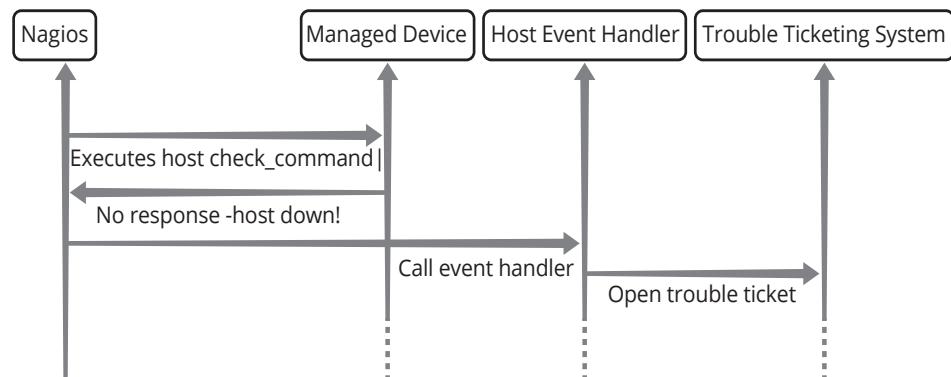


Figura 9.6  
Integração Nagios com TTS.



# 10

## Tópicos avançados em gerenciamento

objetivos

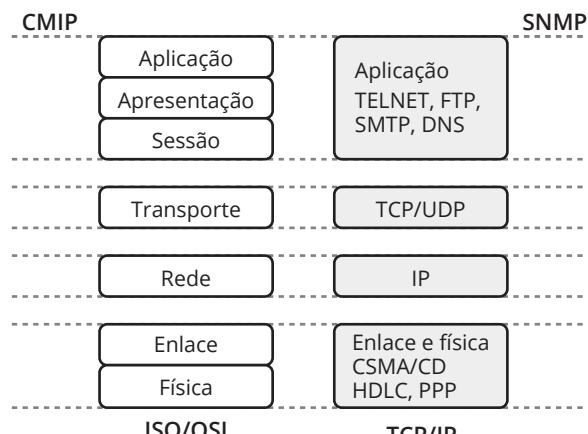
Entender o gerenciamento de rede no contexto ISO/OSI e TCP/IP. Conhecer a gerência da Internet do Futuro. Conhecer as novas abordagens em gerenciamento. Entender os Sistemas Especialistas.

conceitos

A Internet das Coisas. Gerenciando a Internet do Futuro. Novas abordagens em gerenciamento. Sistemas especialistas. Gerenciamento de capacidade.

Os dois principais arcabouços sobre os quais foram construídas as soluções de gerência de redes, o modelo baseado na arquitetura ISO/OSI e o modelo baseado na arquitetura da internet (TCP/IP), evoluíram de forma bem diferente.

O modelo de gerência baseado no modelo ISO/OSI, que usava o protocolo de aplicação Common Management Information Protocol (CMIP), apoiado pela arquitetura ISO/OSI, e todas as camadas nela previstas, como mostra a figura 10.1, não prosperou e foi sendo gradativamente abandonado pelo mercado, mesmo tendo sido oficialmente o adotado pelo International Telecommunication Unit (ITU) para uso no âmbito das administrações de telecomunicações. A complexidade do software para implementar essa solução, o excesso de overhead nos protocolos utilizados, bem como o custo dos produtos disponíveis no mercado, foram elementos decisivos para o abandono das soluções de gerência de rede baseadas nesse modelo.



**Figura 10.1**  
Gerenciamento de rede no contexto ISO/OSI e TCP/IP.



Por outro lado, a solução de gerência de rede baseada na arquitetura TCP/IP e que tinha no próprio nome do protocolo de gerência, Simple Network Management Protocol (SNMP) a ideia de uma solução simples e leve, teve continuidade e foi disseminada amplamente. Atualmente, pode-se afirmar que a arquitetura SNMP de gerenciamento é um padrão de fato, apesar da arquitetura OSI de gerenciamento oferecer uma estrutura mais robusta que SNMP, permitindo a execução de tarefas mais sofisticadas.

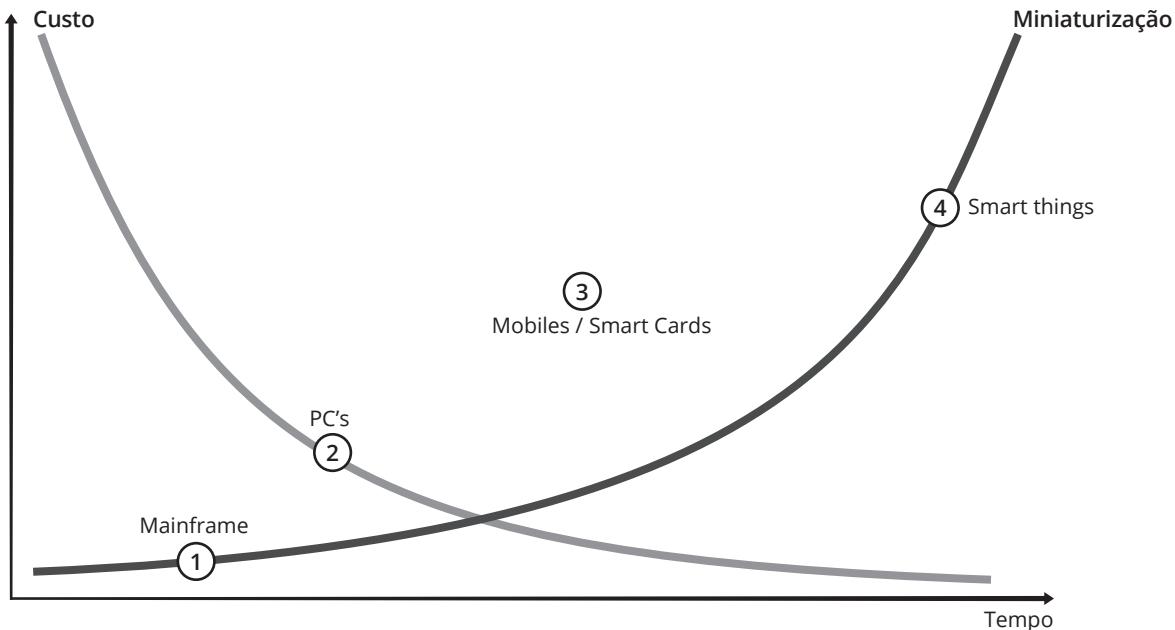
Apesar da disseminação da solução de gerência de rede, são bem conhecidas as limitações do protocolo SNMP, especialmente em suas versões iniciais, que não usam mecanismos fortes de segurança para proteção contra acesso não autorizado. Como os dados de controle de acesso (community name) são transmitidos sem criptografia nas versões SNMP v1 e v2, seu uso tem sido limitado a funções de monitoração. Gerenciamento de configuração tem sido realizado mediante o uso de soluções envolvendo acesso direto aos recursos da rede (com SSH) e alguma linguagem de comando (CLI – Command Line Interface) específica de famílias de produtos produzidos por fornecedores. Diversas soluções e estratégias de evolução foram propostas para a internet em si e para a gerência da futura internet. Tais soluções têm sido classificadas como evolutivas ou “clean state”. No primeiro grupo temos soluções que consideram a rede atualmente existente e as propostas indicam adaptações nos protocolos para atender às novas necessidades de redes com maior velocidade, mobilidade e heterogeneidade de dispositivos. O grupo “clean state” é mais radical e propõe uma arquitetura totalmente nova para a internet e soluções de gerência que separam a parte de comando da parte de dados.

As novas soluções têm sido demandadas especialmente porque o contexto de gerência de rede está sendo ampliado muito em relação ao que inicialmente estava previsto, que incluía apenas roteadores e equipamentos de rede de modo geral, e informações básicas sobre os servidores e computadores conectados à internet. Atualmente, a quantidade de equipamentos que começa a ser interligada está crescendo. A proliferação de dispositivos interconectáveis em rede, com baixo custo e tamanho cada vez menor, levou à ampliação no espectro de soluções para serviços de Tecnologia de Informação (TI) bem diferenciados daquelas que foram usadas nos primórdios da computação e das redes.

Estimativas do Silicom Labs (2014) indicam que em 2015 mais pessoas vão acessar a internet a partir de dispositivos móveis do que de computadores convencionais. Em 2015, 14 bilhões de dispositivos e aparelhos estarão conectados à internet móvel, incluindo não apenas dispositivos do tipo smartphone e tablet, mas também máquinas de lavar, carros e roupas, que serão conectados. A figura 10.2 salienta essa tendência de redução no custo dos dispositivos interligados à internet em função de miniaturização de componentes, o que enseja um crescimento em seu uso, bem como ampliação na quantidade e variedade de equipamentos interligados.

A operação e gestão da internet considerando o contexto da Internet das Coisas vai ser uma tarefa importante na operação do backbone da rede. Para migrar para a Internet das Coisas a partir dos ambientes atuais, a tarefa de integrar esse novo conjunto mais heterogênea de dispositivos é complexo, mas importante. Nesse tipo de rede, a qualidade da comunicação entre objetos ou coisas precisa contemplar novas soluções e ser melhorada.





**Figura 10.2**  
Evolução dos equipamentos interligados à internet.

A gerência de rede precisa contemplar soluções para atender as necessidades de gerenciamento também da Internet das Coisas, pois esse segmento é parte integrante da Internet do Futuro. A rede para a Internet das Coisas pode ser definida como uma infraestrutura de rede global dinâmica baseada em protocolos de comunicação padrão, com recursos de autoconfiguráveis e interoperáveis onde “coisas” físicas e virtuais têm identidades, atributos físicos, personalidades virtuais e usam interfaces inteligentes estando perfeitamente integrados na rede de informação. As características diferenciadas desses dispositivos demanda um atendimento diferenciado da rede para que eles possam ser atendidos, a despeito das limitações que os caracterizam e que serão analisadas a seguir.

## A Internet das Coisas

### RFID

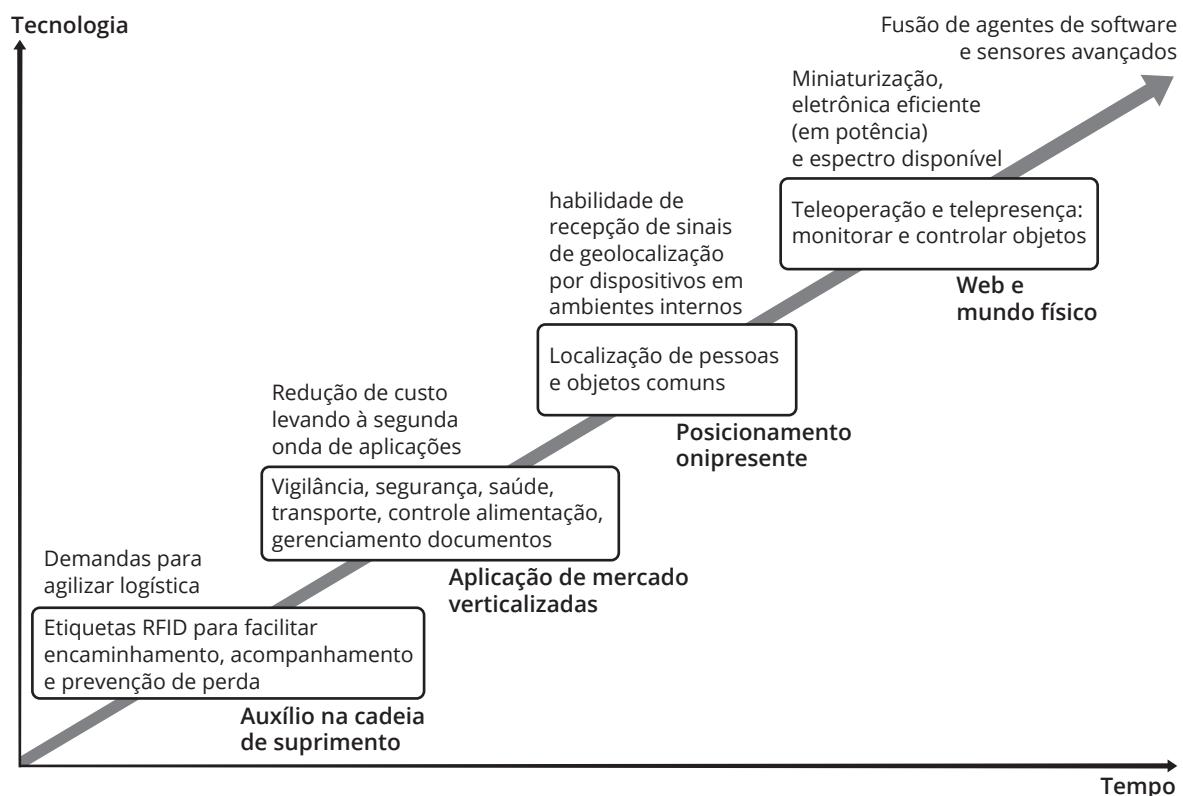
Identificação por radiofrequência ou RFID (“Radio-Frequency IDentification”) é um método de identificação automática através de sinais de rádio, recuperando e armazenando dados remotamente através de dispositivos denominados etiquetas RFID.

O termo Internet das Coisas começou a ser usado por volta de 2000 pela comunidade de desenvolvimento de **RFID**, e referia à possibilidade de descobrir informação sobre objetos vasculhando um endereço na internet que corresponderia a um particular RFID (SRI 2014).

A partir desse início, mais limitado, a expressão “Internet das Coisas”, também conhecida pela sigla IoT (Internet of Things), passou a se referir à ideia geral de coisas, especialmente objetos do dia a dia que podem ser lidos, reconhecidos, localizados, endereçados e/ou controlados via internet, seja por RFID, rede local sem fio, rede de longa distância ou outros meios. Os objetos do dia a dia incluem não apenas dispositivos eletrônicos e nem tampouco apenas produtos de mais alto desenvolvimento tecnológico, tais como veículos e equipamentos, mas considera coisas nas quais não se pensaria em termos de serem dispositivos eletrônicos, tais como:

- Alimento, vestuário e abrigo;
- Materiais, partes e montagens;
- Bens e itens de conforto;
- Limites de locais e monumentos;
- Toda a miscelânea de itens de comércio e cultura.





A figura 10.3 mostra a evolução do uso da Internet das Coisas no escopo de serviços de TI com tecnologias cada vez mais baseadas em dispositivos de pequeno porte e software altamente sofisticado. Um exemplo de novas aplicações é a derivada da “wearable computing”. Uma tendência atual de uso de novos tipos de equipamentos que as pessoas “usam” está sendo designada como Bring Your Own Wearables (BYOW) traz para o contexto das empresas tecnologia de “vestir”, tais como monitores de aptidão física e saúde, relógios inteligentes ligados a telefones celulares, além de óculos com funcionalidades avançadas de captura e exibição de dados, tal como o Google Glass. Isso vai demandar das empresas uma reavaliação das políticas existentes para acomodar esses novos dispositivos, tendo em vista potenciais ameaças à segurança, por exemplo.

Na visão do Internet Engineering Task Force (IETF), o novo paradigma inerente à Internet das Coisas implica em muitas visões por parte de um órgão de padronização. Usualmente, o IETF concentra seus esforços da definição de novos objetos com funções de comunicação, sensoriamento e ação que possam interoperar via IP (Lee 2013). Mas embora sensores sejam baseados em IP ou dispositivos com restrições (memória e recursos de CPU muito limitados) possam usar protocolos de camada de aplicação para fazer o monitoramento e gestão de recursos simples, soluções adaptadas que contemplam essas limitações precisam ser delineadas (Ersue 2014). O IETF está investigando uma nova estrutura de arquitetura para suportar escalabilidade e interoperabilidade para a Internet das Coisas. As metas para alcançar esse objetivo envolvem identificar vários problemas nos protocolos existentes e encontrar possíveis soluções para resolver esses problemas. Uma lista de possíveis problemas é apresentada a seguir:

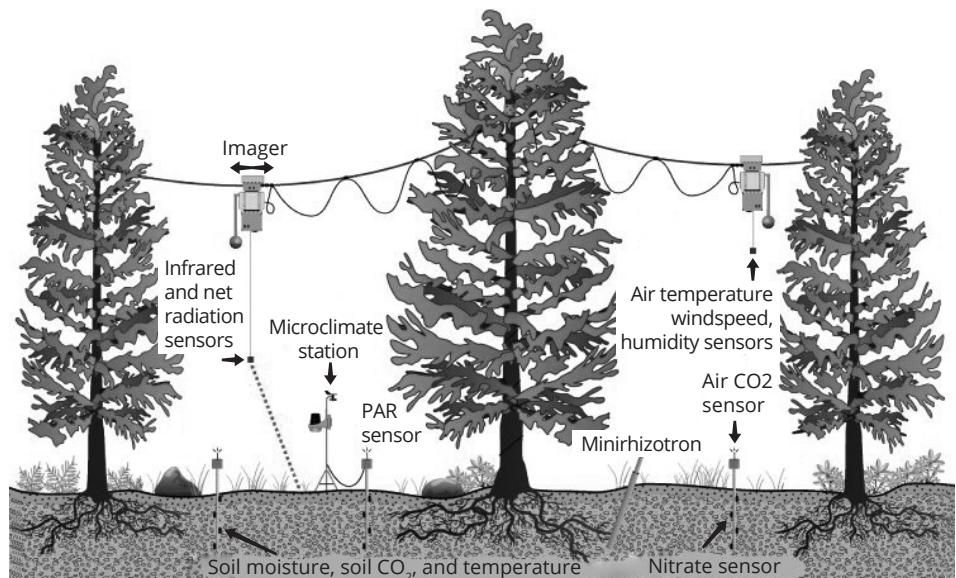
- **Aspectos gerais:** a Internet das Coisas tem como característica inerente uma quantidade de dispositivos pelo menos uma ordem de grandeza maior do que a internet atual, não opera com pessoas diretamente comandando seu funcionamento, possui mobilidade inerente, opera em modo desconectado e sem supervisão direta;

**Figura 10.3**  
Evolução de uso da Internet das Coisas.



- ▣ **Aplicações:** cada dispositivo pode ser usado por múltiplas aplicações e serviços, com diferentes características;
- ▣ **Rede:** é necessária uma tecnologia de comunicação comum que suporte todas as aplicações ou serviços, bem como equipamentos heterogêneos;
- ▣ **Aspecto de camada de enlace:** há vários tipos de interface de rede com cobertura e velocidade diferenciados. Esses ambientes têm características de operação para baixo consumo de energia e redes com perda, tais como Bluetooth ou IEEE 802.15.4;
- ▣ **Aspectos dos objetos inteligentes:** os objetos inteligentes interconectados são heterogêneos e têm tamanho, mobilidade, potência, conectividade e protocolos diferentes. Podem operar também em modo federado com sensores e atuadores interagindo diretamente em escala residencial ou em escopo mais amplo, tais como no contexto de cidades inteligentes.

Os dispositivos da Internet das Coisas são tipicamente dispositivos de uso específico, com CPU, memória e recursos de energia limitados. Esses dispositivos limitados (sensores, objetos inteligentes ou dispositivos inteligentes) podem ser conectados a uma rede. Essa rede dos dispositivos limitados pode ser em si limitada ou prejudicada por canais não confiáveis ou com perdas, tecnologias sem fio com largura de banda limitada e uma topologia dinâmica, necessitando o serviço de um gateway ou proxy para se conectar à internet tal como exemplificado na figura 10.4. Em outras situações, os dispositivos limitados podem ser conectados a uma rede não limitada usando pilhas de protocolos normais. Dispositivos limitados podem ser responsáveis pela coleta de informações em diversas situações, incluindo os ecossistemas naturais, edifícios e fábricas, e enviar as informações para um ou mais servidores.



**Figura 10.4**  
Sensores em  
ecossistemas.

A gerência da rede é caracterizada por monitorar o status da rede, detecção de falhas e inferir as suas causas. Envolve a definição de parâmetros de rede e contempla realizar ações para eliminar falhas, manter a operação normal, melhorar a eficiência da rede e desempenho do aplicativo. A aplicação tradicional de gerenciamento de rede recolhe periodicamente as informações a partir de um conjunto de elementos que são necessários para gerenciar, processar os dados e apresentar resultados aos usuários de gerenciamento de rede. Dispositivos limitados, no entanto, muitas vezes têm poder limitado, faixa de transmissão baixa e podem não ser confiáveis. Eles também podem precisar operar em ambientes hostis com requisitos de segurança avançadas ou podem necessitar serem usados em ambientes agressivos por um longo tempo sem supervisão.





Devido a essas limitações, a gestão de uma rede com dispositivos restritos oferece vários tipos de desafios em comparação com a gestão de uma rede IP tradicional.

Além dos requisitos de gestão impostas pelos diferentes casos de uso, as tecnologias de acesso usadas por dispositivos limitados pode impor restrições e exigências sobre o sistema de gerenciamento de rede e protocolo passíveis de uso.

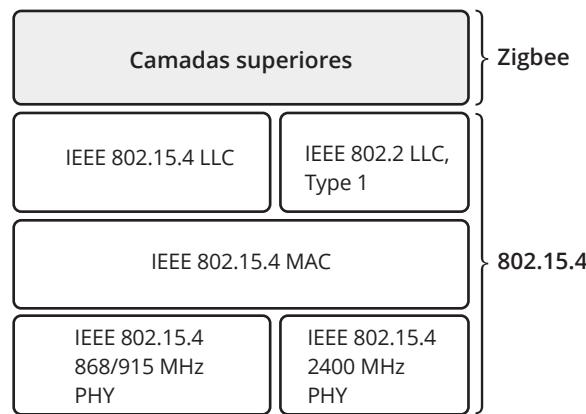
É possível que algumas redes de dispositivos limitados possam utilizar tecnologias de acesso não restritas ou tradicionais de acesso à rede, como por exemplo, redes de área local com plena capacidade. Em tais cenários, a limitação dos dispositivos apresenta restrições especiais de gestão e requisitos em vez da tecnologia de acesso utilizada.

No entanto, em outras situações, tecnologias de acesso móvel ou limitado podem ser usadas para acesso à rede, ocasionando limitações nos requisitos de gestão devido às tecnologias de acesso usadas.

Devido a restrições de recursos, dispositivos embarcados implantados como sensores e atuadores em diversos casos usam tecnologias de acesso sem fio de baixa potência e baixa velocidade, tais como as soluções previstas em diversos padrões:

#### IEEE 802.15.4

Para conectividade da rede, IEEE 802.15.4 é um padrão que especifica a camada física e efetua o controle de acesso para redes sem fio pessoais de baixas taxas de transmissão. O padrão IEEE 802.15.4 pretende oferecer os fundamentos para as camadas inferiores em uma rede do tipo de área pessoal e sem fio (WPAN-Wireless Personal Area Network), que tem foco no baixo custo, a comunicação de baixa velocidade onipresente entre os dispositivos em contraste com outros, mais o usuário final abordagens orientadas, tais como Wi-Fi. A estrutura básica pressupõe uma distância média de 10 metros para comunicações com uma taxa de transferência de 250 kbit/s. A figura seguinte ilustra os padrões que integram a definição a solução IEEE 802.15.4



**Figura 10.5**  
Padrão IEEE  
802.15.4.

O termo ZigBee designa um conjunto de especificações para a comunicação sem fio entre dispositivos eletrônicos, com ênfase na baixa potência de operação, na baixa taxa de transmissão de dados e no baixo custo de implantação. Tal conjunto de especificações define camadas do modelo OSI subsequentes àquelas estabelecidas pelo padrão IEEE 802.15.4 (IEEE 2003).

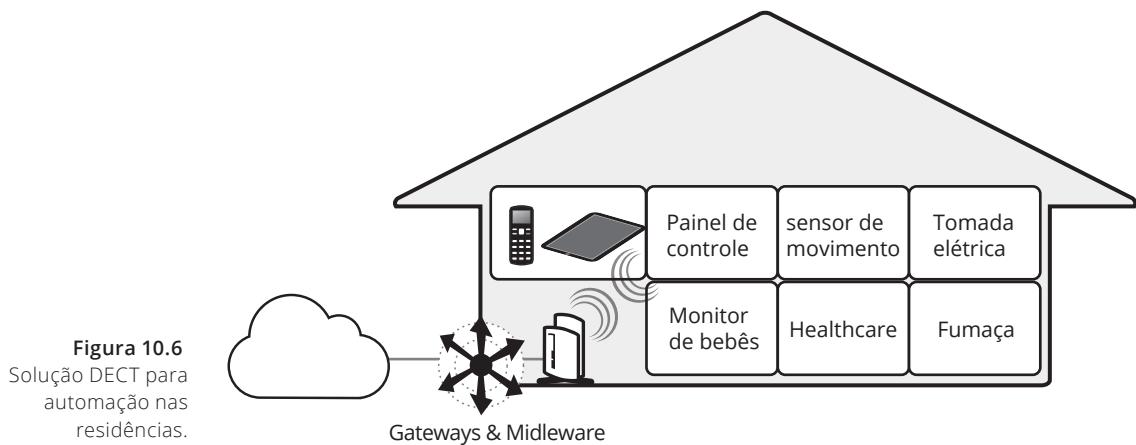
Foi pensada para inteligar pequenas unidades de coleta de dados e controle recorrendo a sinais de radiofrequência não licenciados. A tecnologia utilizada é comparável às redes



Wi-Fi e Bluetooth, e diferencia-se dessas por desenvolver menor consumo, por um alcance reduzido (cerca de 100 metros) e a comunicação entre duas unidades poder ser repetida sucessivamente pelas unidades existentes na rede até atingir o destino final. Todos os pontos da rede podem funcionar como retransmissores de informação. Uma malha (Mesh) de unidades ZigBee pode realizar-se em uma extensão doméstica ou industrial sem necessidade de utilizar ligações eléctricas entre elas.

### DECT ULE

Digital Enhanced Cordless Telecommunications (DECT) foi definido em 1987 como um padrão para comunicação com telefones sem fio. A última versão é DECT Ultra Low Energy (ULE) e popularizou-se como uma tecnologia para automação e segurança nas residências. Suas características incluem baixo custo, baixo consumo de energia, longo alcance (pode alcançar a casa toda com uma topologia estrela), resistente a interferências, estável permitindo transmissão de voz e vídeo.



**Figura 10.6**  
Solução DECT para  
automação nas  
residências.

A automação das residências pode envolver a interligação de dispositivos tais como:

- **Smart Plugs:** proporcionam monitoração inteligente e controle para aparelhos conectados às tomadas de energia;
- **Indicação e alerta de consumo:** proporciona ao usuário na casa a indicação do consumo de energia monitorado;
- **Controle de luz:** proporciona ao usuário a possibilidade de controlar a iluminação na casa com controle remoto;
- **Controle de aparelhos domésticos:** proporciona ao usuário a possibilidade de controlar aparelhos domésticos com controle remoto, incluindo temporização e desligamento em períodos de pico, quando o custo da energia é maior;
- **Controle ambiental:** Termostatos, AVAC (aquecimento, ventilação e ar-condicionado) e persianas remotamente controláveis.

### BT- LE: Bluetooth low energy, Bluetooth LE ou BLE

É uma tecnologia de rede de área pessoal sem fio, desenvolvida e comercializada pelo Grupo de Interesse Especial Bluetooth, destinado a novas aplicações nas indústrias de saúde, de aptidão, de segurança e de entretenimento doméstico. Comparado ao Bluetooth clássico, BLE se destina a operar com consumo de energia e custo consideravelmente reduzido, mantendo alcance de comunicação similar. Bluetooth LE usa as mesmas frequências de rádio de 2,4 GHz, como o Bluetooth clássico.



Na área da saúde, esse padrão é usado para interligar dispositivos tais como:

- Dispositivos médicos de medição de temperatura;
- Monitores de glicose no sangue;
- Monitores de pressão sanguínea;
- Na área de esportes e aptidão, pode permitir interligar dispositivos como:
  - Medidos de ritmo cardíaco;
  - Sensores ligados a bicicletas ou aparelhos de exercícios, para medir cadência e velocidade;
  - Velocidade e cadência do perfil de corrida;
  - Perfil e potência ao pedalar;
  - Perfil de localização e navegação.

Para sensoriamentos de proximidade, essa tecnologia pode ser usada em dispositivos de:

- Localização;
- Detecção de proximidade.



**Figura 10.7**  
Exemplos de uso  
de dispositivos  
operando com  
BT-LE.

Em todos esses cenários, é importante para o sistema de gerência de rede estar ciente das restrições impostas por essas tecnologias de acesso, para gerenciar com eficiência esses dispositivos limitados.

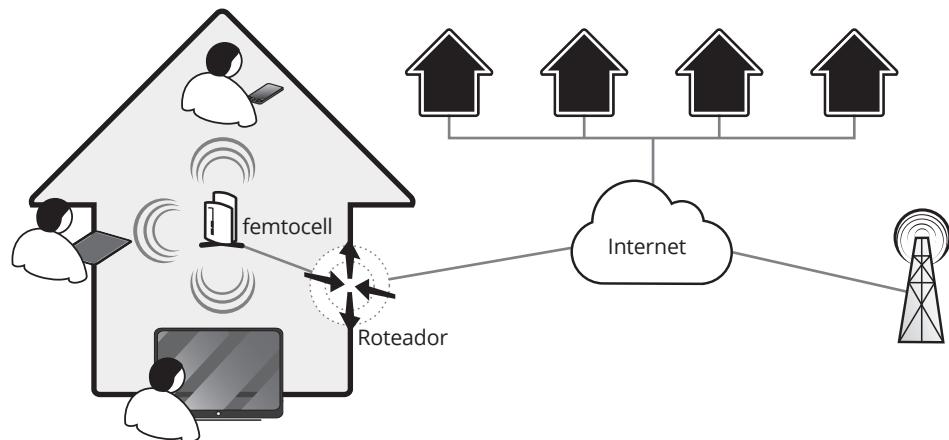
Os dispositivos que utilizam essas tecnologias de acesso podem funcionar através de um gateway que realiza a tradução dessas tecnologias de acesso para protocolos de internet mais tradicionais, tal como ilustrado na figura 10.6. A abordagem hierárquica para o gerenciamento de dispositivos em tal situação pode ser útil, sendo o dispositivo de gateway encarregado de gerenciar os dispositivos conectados a ele, enquanto que um sistema de gerência de nível mais alto realiza operações de gestão apenas para o gateway.

Serviços baseados em comunicação máquina a máquina (M2M) também estão sendo cada vez mais usados pelos prestadores de serviços móveis, na medida em que vários dispositivos, eletrodomésticos, medidores de serviços públicos, carros, câmeras de vigilância de vídeo e monitores de saúde podem ser conectados com tecnologias de banda larga. Diferentes aplicações, por exemplo, em um aparelho em casa ou na rede do carro, podem usar

Bluetooth, Wi-Fi ou Zigbee localmente e se conectar a um módulo celular agindo como um gateway entre o ambiente restrito e da rede celular móvel, tal como ilustrado na figura 10.7. O gateway pode ser responsável pela filtragem e agregação dos dados recebidos a partir do dispositivo, uma vez que as informações enviadas pelo dispositivo podem ser na maior parte redundantes.

Um gateway pode fornecer diferentes opções para a conectividade de redes móveis e dispositivos com restrições:

- Um telefone inteligente com 3G/4G e WLAN rádio pode usar BT- LE para se conectar com os dispositivos em uma rede de área local;
  - Um femtocell (pequenas estações rádio base ou ERBs, desenvolvidas para operar dentro de residências) pode ser combinado com uma funcionalidade de gateway doméstico atuando como estação base celular de baixa potência, conectando dispositivos inteligentes com o servidor de aplicativos de uma operadora de telefonia móvel.



**Figura 10.8**  
Rede usando  
tecnologia  
femtocell.

A solução femtocell envolve operação em baixa potência, nas frequências utilizadas pelas operadoras de celular, e a conexão com a rede da operadora é feita através de conexão banda larga existente na residência (ADSL, Cabo etc.).

- Um módulo celular incorporado com rádio LTE pode conectar os dispositivos na rede de um carro com o servidor que está executando o serviço de telemática. LTE é sigla de Long Term Evolution (em português, “Evolução a Longo Prazo”), cujo significado se refere a uma tecnologia de telefonia móvel também conhecida como 4G (quarta geração). LTE é um padrão de redes celulares que permite banda larga móvel com velocidades de conexão de até 100 Mbps, possibilitando maior abrangência de comunicações de voz e transferência de dados.



**Figura 10.9**  
Carro com serviços proporcionados pela tecnologia LTE (4G).

- Um gateway M2M ligado à rede do operador móvel apoiando diversas tecnologias de conectividade da Internet das Coisas, incluindo **ZigBee** e CoAP sobre 6LoWPAN (IPv6 Over Low Power Wireless Personal Network e é o nome do grupo de desenvolvimento da IETF, que cria e mantém as especificações para uso do IPv6 nas redes IEEE 802.15.4).

Constrained Application Protocol (CoAP) é um protocolo de software destinado a ser utilizado em dispositivos eletrônicos muito simples e que permite a sua comunicação de forma interativa através da internet. É particularmente direcionado para pequenos sensores de baixa potência, interruptores, válvulas e componentes semelhantes, que precisam ser controlados ou supervisionados remotamente, por meio de redes de internet padrão. CoAP é um protocolo de camada de aplicação que se destina para uso em dispositivos de internet com recursos limitados, como nós de RSSF. CoAP é projetado para facilitar a tradução para HTTP visando a integração simplificada com a web e, ao mesmo tempo, atender aos requisitos especializados, como suporte a multicast, muito baixo em cima, e simplicidade.

O comum a todos os cenários descritos é que eles são incorporados em um serviço e ligados a uma rede de provedor de serviços móveis. Geralmente há uma implantação hierárquica na topologia e gerenciamento ao invés de serem os dispositivos gerenciados diretamente, pois a quantidade de dispositivos para gerenciar é elevada (por exemplo, muitos milhares). Em geral, a rede é constituída por múltiplos tipos de dispositivos. Como tal, a entidade gestora tem de estar preparado para gerenciar dispositivos com capacidades diversas, utilizando diferentes protocolos de comunicação ou de gestão. No caso de os dispositivos que são conectados diretamente a um gateway, eles provavelmente são geridos por uma entidade de gestão integrada com o gateway, que em si é parte do Sistema de Gerenciamento de Rede. Telefones inteligentes ou módulos embarcados conectados a um gateway podem ser responsáveis por gerenciar os dispositivos a eles conectados. A configuração inicial e subsequente desse tipo de dispositivo utiliza principalmente a autoconfiguração e é acionada pelo próprio dispositivo.

#### **ZigBee**

Padrão de rede sem fio para arquitetura em malha de baixo custo e baixa potência, usualmente implementado em chips com rádios integrados e micro-controladores com memória flash entre 60 KB e 256 KB, operando nas faixas de rádio industriais, científicas e médicas: 868 MHz na Europa, 915 MHz nos EUA e Austrália, e 2,4 GHz na maioria das jurisdições em todo o mundo.

## **Gerenciando a Internet do Futuro**

Um modelo de transporte seguro para o SNMP foi proposto como uma alternativa para os modelos de segurança existentes no SNMP V1 e V2, baseados em comunidades e para o modelo de segurança baseado em usuário.

A internet desempenha um papel central na sociedade e em consequência surgiram muitos novos serviços de trabalho, negócios, educação e entretenimento.

Nos últimos anos, o equilíbrio, a evolução e as relações entre os vários requisitos de rede mudaram significativamente, criando a necessidade de novos sistemas de rede. Essas alterações incluem diferentes novos equipamentos e dispositivos móveis conectados à rede pública de telecomunicações, número significativo de centros de dados, soluções de computação em nuvem e grande número de diferentes sensores, atuadores e outras “coisas” conectadas na rede, operando com comunicação Machine-to-Machine (M2M) e provendo serviços envolvendo Internet das Coisas.

A comunidade de pesquisa, bem como a indústria, tem dedicado esforço contínuo para investigar tecnologias e sistemas para as redes futuras. Várias tecnologias, tais como virtualização de redes, redes centradas em informação, gerenciamento autônomo e conectividade aberta tem sido consideradas.

Diversas organizações e grupos se esforçam para levar padrões para esse novo cenário. A International Telecommunication Union Telecommunication Standardization Sector (ITU-T) iniciou um esforço de padronização da futura internet visando as redes para o período de 2015-2020. O resultado da análise refletiu na Recomendação ITU-T Y.3001 do ITU (Matsabura 2013). Ela inclui diversas tecnologias candidatas a serem usadas como parte da futura rede, tais como tecnologia de virtualização de rede nos moldes das soluções Software Defined Networks (SDN), que já começam a ser usadas.

No que concerne especificamente ao gerenciamento da futura internet, Festor, Pras e Stilles (2010) destacaram as seguintes necessidades:

- ▣ **Mecanismos de Gestão para a futura internet:** embora uma série de futuras arquiteturas de internet estivessem sendo debatidas, o conjunto de princípios de gestão relacionados não haviam sido abordadas;
- ▣ **Falha, Configuração e Operação de Segurança na Internet do Futuro:** o design, modelagem e avaliação de algoritmos em três áreas funcionais têm de lidar com sistemas de larga escala;
- ▣ **Gestão autônoma intra e interdomínio na Internet do Futuro:** para redes fixas, o grau de gerenciamento automatizado de QoS interdomínio, a autogestão de redes ópticas usando MPLS, o gerenciamento de falhas automatizado intra e interdomínio, isto é, a recuperação de serviço e sua resiliência precisam ser determinados;
- ▣ **Gestão de Rede e Serviços econômica na internet do Futuro:** em apoio a uma abordagem tecnicamente viável de operar uma rede para numerosos serviços comerciais, de forma eficiente, é necessário a gerenciamento de rede e serviço baseados em considerações tecnológicas e econômicas.

Existem diversos problemas no gerenciamento de rede para o contexto da Internet das Coisas, tal como identificado por Lee (2013):

- ▣ **Identificação:** os códigos de identificação usados nos dispositivos são diferentes e derivados de seu uso. Embora o esquema de endereçamento do IPv6 possa contemplar, há a necessidade de esquemas de identificação diferenciados, pois alguns itens, tais como livros, medicamentos e roupas podem não requerer identificação global – alguns objetos têm existência temporária e desaparecem após algum tempo;



- ▣ **Serviços de nomes:** serviços de suporte a nomes na internet, tal como Domain Name Services (DNS) são indispensáveis para o funcionamento da internet. De modo similar, são necessários serviços de tradução que convertam nomes apropriados para as “coisas” e que podem seguir um esquema de identificação heterogêneo de um particular espaço de nomeamento em diferentes redes. Essa compatibilização de esquemas de nomeamento usados na internet e no contexto da IoT é um dos aspectos mais importantes a serem resolvidos. Esquemas de endereçamento IP e não IP vão precisar conviver. IPv6 vai ser especialmente necessário para acomodar a quantidade de dispositivos a serem endereçados;
- ▣ **Segurança, privacidade, autoridade:** a perda de segurança e privacidade na comunicação e serviços que tratam com dados pessoais é um dos problemas que crescem muito atualmente. A necessidade de assegurar confiabilidade e sigilo na comunicação colide com a limitada capacidade de tratar algoritmos mais complexos de autenticação, autorização e controle de uso (AAA – Authentication, Authorization, Accounting);
- ▣ **Detecção de presença:** refere-se a mecanismos que aceitem, armazenem e distribuam informação relativa à presença e proximidade de pessoas e dispositivos. Como a mobilidade é uma característica intrínseca, protocolos que contemplam as mudanças derivadas da mobilidade são relevantes e necessários;
- ▣ **Descoberta e pesquisa:** cada objeto pode ser uma fonte de informação que deve poder ser armazenada e descoberta para que possa ser usada pelas pessoas;
- ▣ **Autonomia:** autoconfiguração é necessária nesse contexto e os dispositivos devem poder estabelecer automaticamente os parâmetros necessários para sua conectividade de forma fácil e automatizada. Os mecanismos de autoconfiguração do IPv6 são úteis nesse sentido. Objetos com recursos limitados podem ter menor capacidade de adaptação e autoconfiguração;
- ▣ **Energia:** dadas as limitações de energia dos dispositivos, é importante que os mecanismos de comunicação sejam eficientes. Modos ativo e suspenso podem ser usados alternativamente para economizar energia. Face à variedade de volume de tráfego transmitido/recebido, esse tipo de solução é apropriado e necessário para prolongar a operação do dispositivo;
- ▣ **Serviços web:** o acesso aos objetos pode ser realizado via web e, nesse sentido, catalogação de objetos e associação com URLs devem ser realizadas.

## Novas abordagens em gerenciamento

- ▣ Sistemas especialistas (AI).
- ▣ Objetos distribuídos.
- ▣ Gerenciamento web-based.
- ▣ Distributed Management Task Force (DMTF).



Tudo indica que SNMP terá um futuro promissor por muito tempo. Apesar disso, a crescente diversidade de contextos e orçamentos busca alternativas para gerenciamento, especialmente tendo em vista a quantidade de objetos gerenciados que continuamente são definidos.

O uso de SNMP em dispositivos limitados foi analisado por Schenwalder (2013), que demonstrou ser possível implementar uma pilha SNMP (Contiki-SNMP) em dispositivos limitados.

Essa implementação do SNMP oferece mecanismos básicos e mesmo mais complexos:

- Mensagens SNMP com até 484-bytes de comprimento;
- Operações Get, GetNext e Set;
- SNMPv1 e SNMPv3;
- Modelos de segurança USM (User Security Model do SNMPv3) sem VACM;
- API para definir e implementar objetos.

Os módulos de MIB testados foram:

- SNMPv2-MIB { SNMP entity information}
- IF-MIB { network interface information}
- ENTITY-SENSOR-MIB { temperature sensor readings}

## Sistemas especialistas

### DISMAN – Distributed Management

- Distributed Management Expression MIB (RFC 2982).
- Event MIB (RFC 2981).
- Notification Log MIB (RFC 3014).
- Definitions of Managed Objects for the Delegation of Management Scripts (RFC 3165).
- Definitions of Managed Objects for Scheduling Management Operations (RFC 3231).
- Alarm MIB (RFC 3877).
- Alarm Reporting Control MIB (RFC 3878).
- Definitions of Managed Objects for Remote Ping, Traceroute and Lookup Operations (RFC 4560).

O gerenciamento de redes distribuído é reconhecido como uma necessidade atualmente. Uma aplicação gerente é uma boa candidata a ser distribuída.

O DISMAN é um esforço do IETF para o desenvolvimento de objetos gerenciados padronizados para gerenciamento distribuído. Prevê a distribuição das atribuições da aplicação gerente: funções de gerenciamento. Inicialmente, usará como mecanismo de comunicação o framework SNMP. Seus objetos serão criados para serem consistentes com SNMP.

Definirá objetos para tratar itens como: agendamento de operações de gerenciamento, notificações, alarmes, eventos etc.

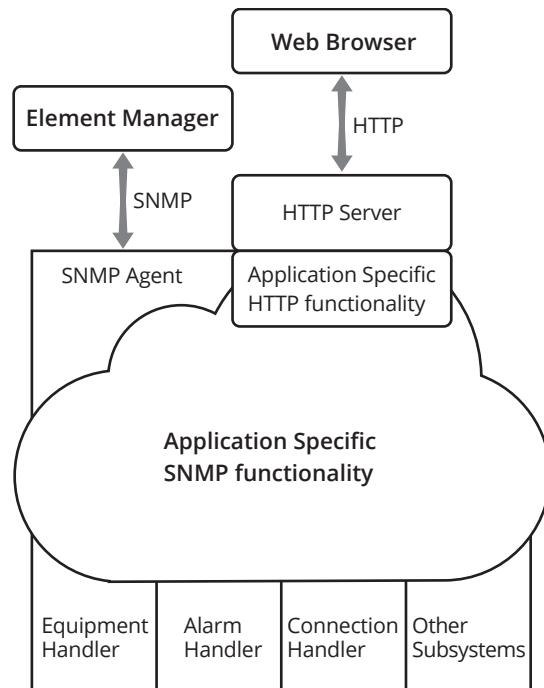
### Gerenciamento web-based

- HTTP/HTML.
- CGI.
- XML.
- SSL.
- SOAP.
- LDAP.

Aplicações web em servidores disponibilizariam a informação e as operações de gerenciamento com o uso de tecnologias presentes na internet.

Entre as vantagens das ferramentas de gerência baseadas em web, encontram-se:

- A possibilidade de monitorar e controlar os elementos da rede usando qualquer navegador em qualquer nó da rede. Antes, quando falávamos de aplicações de gerência standalone, os gerentes de rede só podiam usar a ferramenta em máquinas onde ela estivesse instalada e devidamente configurada;
- A interface gráfica da web já é bem conhecida, e as operações realizadas em um navegador também, não sendo necessários gastos com treinamento de pessoal;
- Usar a web para distribuir as informações sobre a operação da rede tem se mostrado uma tarefa eficaz. Por exemplo, em um determinado endereço da intranet poderíamos disponibilizar informações para os usuários sobre o estado da rede e atualizações que precisam ser realizadas. Isso evitaria, entre outras perturbações, ligações excessivas ao help desk;
- Além disso, não precisamos usar ferramentas diferentes para gerenciar os diversos elementos da infraestrutura de TI, inclusive os serviços.



**Figura 10.10**  
Gerenciamento  
Web-based.

## DMTF – Distributed Management Task Force

- Common Information Model (CIM).
- Web-Based Enterprise Management (WBEM).
- Distributed Management Interface (DMI).
- Directory Enabled Networks (DEN).

Esforço de um consórcio de empresas para criar um padrão de gerenciamento desktop.  
Evoluiu de gerenciamento de desktops para gerenciamento Web-based dentro de empresas.



Usa como modelo de informação o Common Information Model (CIM), especificação orientada a objetos para compartilhamento de informações de gerenciamento, além de Managed Object Format (MOF) para descrever objetos. É usado também por outras iniciativas (modelo unificado).

Outras iniciativas do DMTF:

- **Distributed Management Interface (DMI)**: framework para gerenciamento de sistemas desktop, notebooks, PCs e servidores;
- **Web-Based Enterprise Management (WBEM)**: iniciativa que usa tecnologias web (browsers, XML, SOAP, WSDL etc.) para gerenciamento de sistemas. Serviu de origem para o desenvolvimento do CIM. O objetivo principal do WBEM é alcançar a gerência unificada de todos os sistemas e redes de uma organização;
- **Directory Enabled Networks (DEN)**: formatos padrão para diretórios interoperáveis, onde administradores podem usar os serviços de diretórios para gerenciar serviços de rede (como um diretório LDAP).

## Gerenciamento de capacidade

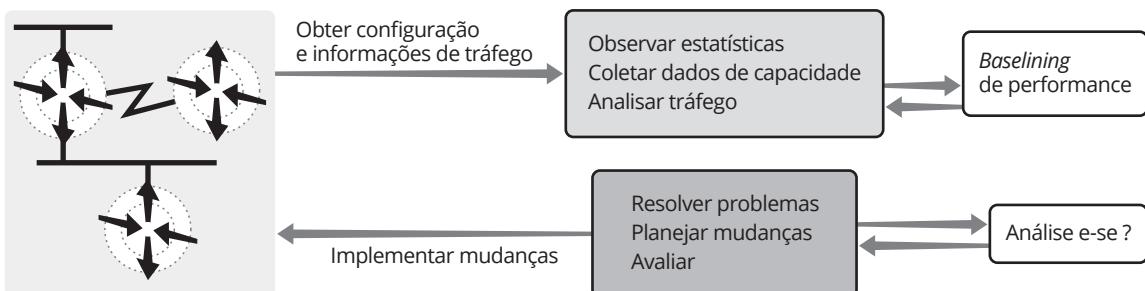
Gerenciamento de capacidade (capacity planning) é o processo de determinação dos recursos de rede necessários para evitar um impacto de desempenho ou disponibilidade sobre aplicações críticas.

Costuma-se dizer que gerenciamento de performance é a prática de gerenciar tempo de resposta, consistência e qualidade da rede para serviços, individualmente e de modo global.

Problemas de performance normalmente são relacionados com capacidade. Dois aspectos envolvem a capacidade de uma rede:

- O chamado “data plane” é a porção da capacidade gasta efetivamente com os dados no percurso pela rede. São os dados de usuário;
- O chamado “control plane”, por outro lado, é a porção relativa aos mecanismos de rede que a mantém em operação. Pode-se incluir aí o tráfego adicional gerado por protocolos de gerência, spanning-tree, protocolos de roteamento, “keep alives”, além de CPU, memória e buffering;
- Análise “what-if: a forma como as mudanças propostas poderiam alterar o uso da rede;
- “Baselining” e análise de tendências: permitem que se planeje melhorias em uma rede antes que problemas de capacidade causem piora significativa na performance ou até a queda do serviço. Trata-se de comparar medições de utilização no decorrer do tempo, de forma a tornar mais claro o comportamento de indicadores que refletem a resposta da rede. É necessária uma grande quantidade de informações para que isso seja realizado em uma rede grande.

**Figura 10.11**  
Gerenciamento de capacidade.





# Bibliografia

- ▣ BACKMAN, Dan. Basking in Glory-SNMPv3.  
<http://www.networkcomputing.com/915/915f1.html>
- ▣ BALLEW, Scott. Managing IP Networks. O'Reilly, 1997.
- ▣ BLUM, Rick. IT Operations Centers (ITOCs),  
<http://www.ins.com/resources/surveys/2008>
- ▣ BRISA (Nome de conjunto de 19 autores). Gerenciamento de Redes: Uma abordagem de sistemas abertos, Makron Book, 1993.
- ▣ CASE, J.; FEDOR, M.; SCHOFFSTALL, M.; DAVIN, J. Simple Network Management Protocol: RFC 1157. [S.I.]: Internet Engineering Task Force, Network Working Group, 1990.
- ▣ CISCO. Capacity and Performance Management: Best Practices White Paper. Cisco Systems. 2009-2010  
<http://www.cisco.com/image/gif/paws/20769/performwp.pdf>
- ▣ CISCO. Change Management: Best Practices White Paper. Cisco Systems. 2008.  
[http://www.cisco.com/en/US/technologies/collateral/tk869/tk769/white\\_paper\\_c11-458050.pdf](http://www.cisco.com/en/US/technologies/collateral/tk869/tk769/white_paper_c11-458050.pdf)
- ▣ CISCO. Network Management System: Best Practices White Paper. Cisco Systems. 2007.  
[http://www.cisco.com/image/gif/paws/15114/NMS\\_bestpractice.pdf](http://www.cisco.com/image/gif/paws/15114/NMS_bestpractice.pdf)
- ▣ CHIU, Dah. SUDAMA, Ram. Network Monitoring Explained. Ellis Horwood Limited. 1992.
- ▣ CLEMM, Alexander. Network Management Fundamentals.. Cisco Press. Morgan – Kaufman Publishers. 2007.
- ▣ CHAPPEL, Laura. Wireshark – Network Analysis. Protocol Analysis Institute. 2010.
- ▣ COELHO, Josiane et all. How much management is management enough? Providing Monitoring Processes with Online Adaptation and Learning Capability. IFIP/IEEE International Symposium on Integrated Network Management. New York. 2009.



- ▣ COMER, Douglas. Automated Network Management System – Current and Future Capabilities. Prentice-Hall, 2007.
- ▣ FARRIEL, Adrian et all. Network Management Know It All. Morgan – Kaufman Publishers. 2009.
- ▣ FEIT, Sidnie. SNMP – A Guide to Network Management. McGraw-Hill, 1995.
- ▣ FIOREZE, T., GRANVILLE, L.Z., ALMEIDA, M.J., TAROUCO, L.R. Comparing web services with SNMP in a management by delegation environment. In: IFIP/IEEE INTERNATIONAL SYMPOSIUM, 9, 2005. INTEGRATED NETWORK MANAGEMENT, IM, 2005, Nice, France.
- ▣ GASPARY, L et all. Charcaterization and Measurement of Enterprise Network Traffic with RMON2. In ternational Workshop on Distributed Systems, Operatiions & Management. DSOM. 1999 Zurich, Switerland. Springer-Verlag.
- ▣ GRANVILLE, Lisandro. Gerenciamento integrado de QoS em redes de computadores. PGCC-UFRGS. 2001.
- ▣ GUNTHER, Neil. The Practical Performance Analyst. Mc Graw-Holl. 1998.
- ▣ HARNEDY, Sean. Total SNMP. Prentice-Hall, 2<sup>a</sup> edição, 1998.
- ▣ HARNEDY, Sean. Total SNMP. Prentice-Hall, 2<sup>a</sup> edição, 1998.
- ▣ HELD, Gilbert. LAN testing and troubleshooting. John Wiley & Sons. 1996.
- ▣ INTERNET2. Performance Initiative.  
Disponível em <http://www.internet2.edu/performance/index.html>.
- ▣ Jakobson, Van ET all. TCPDUMP. Lawrence Berkeley National Labs. <http://ee.lbl.gov/>. acessado em novembro de 2012.
- ▣ JOHNSON, D. NOC Integrated Trouble Ticket System: Functional Specification Wishlist. RFC 1297. IAB 1992.
- ▣ NLANR DAST Iperf. Disponível em: <<http://iperf.sourceforge.net/>>  
Acesso em: 19 de novembro de 2012.
- ▣ LEINWAND, Allan. CONROY, Karen. Network Management – A practical Perspective. Addison Wesley. 2nd edition. 1996.
- ▣ LOPES, Raquel; SAUVÉ, Jacques; NICOLLETTI, Pedro. Melhores Práticas para Gerência de Redes de Computadores. Campus, 2003.
- ▣ MCCANNE, Steven ET all. Packet Capture Library (Libcap). Lawrence Berkeley National Labs. <http://ee.lbl.gov/>. acessado em novembro de 2012.
- ▣ MCCLOGHRIE, K.; PERKINS, D.; SCHOPENWAELDER, J.; CASE, J.; ROSE, M.; WALDBUSSER, S. Structure of Management Information Version 2 (SMIV2): RFC 2578. [S.I.]: Internet Engineering Task Force, Network Working Group, 1999.
- ▣ MELCHIORS, Cristina. TAROUCO, Liane. Fault Management in Computer Networks Using Case Based Reasoning: DUMBO System. Lecture Notes on Artiticial Intelligence. Springer-Verlag 1999 p 510-524.



- ▣ MILLER, Mark A. Troubleshooting Internetworks: Tools, Techniques and Protocols. San Mateo, California: M&T Books, 1991.
- ▣ MILLER, Mark. Managing Internetworks with SNMP. 2nd edition. M & T Books. 1997.
- ▣ NASSER, Dan. Network Optimization and TroubleShooting: Achieve Maximum Network Performance. NRP, 1994.
- ▣ NET-SNMP. <http://www.net-snmp.org/>
- ▣ PERKIN, David. RMON: Remote Monitoring Monitoring of SNMP – Managed LANS. Prentice-Hall. 1999
- ▣ PRAS, A. Network Management Architectures. 1995. Tese (Doutorado) – University Twente, Enschede, Netherlands.
- ▣ PRESUHN, R.; CASE, J.; McCLOGHRIE, K.; ROSE, Marschal.; WALDBUSSER, S. Management Information Base (MIB) for the Simple Network Management Protocol (SNMP): RFC 3418, [S.I.]: Internet Engineering Task Force, Network Working Group, 2002.
- ▣ ROSE, Marshal. A Convention for Defining Traps for use with the SNMP: RFC 1215. [S.I.]: Internet Engineering Task Force, Network Working Group, 1991.
- ▣ ROSE, Marschal.; McCLOGHRIE, Keith. Concise MIB Definitions: RFC 1212. [S.I.]: Internet Engineering Task Force, Network Working Group, 1991.
- ▣ ROSE, Marschal.; McCLOGHRIE, Keith. How to Manage your Network Using SNMP. Prentice Hall. 1995.
- ▣ SANDERS, Cris. Practical Packet Analysis. 2nd edition. No Starch Press. 2011.
- ▣ SANTOS, Rafael et all. DOS SANTOS, C. R. P. et al. On using mashups for composing network management applications. IEEE Communications Magazine, Piscataway, NJ, USA, v.48, p.112–122, December 2010.
- ▣ SCHONWALDER, J. et al. Future Internet = content + services + management. Communications Magazine, IEEE, [S.I.], v.47, n.7, p.27 –33, 7 2009.
- ▣ SCHONWALDER, J.; QUITTEK, J.; KAPPLER, C. Building distributed management applications with the IETF Script MIB. Selected Areas in Communications, IEEE Journal on, [S.I.], v.18, n.5, p.702–714, 2000.
- ▣ Service Level Management: Best Practices White Paper. Cisco Systems. [http://www.cisco.com/en/US/tech/tk869/tk769/tech\\_white\\_papers\\_list.html](http://www.cisco.com/en/US/tech/tk869/tk769/tech_white_papers_list.html)
- ▣ SNMP White Paper. <http://www snmp com/snmpv3/v3white.html>
- ▣ STALLINGS, William. SNMP, SNMPv2, SNMPv3 and RMON 1 and 2. Addison-Wesley, 3<sup>a</sup> edição, 1999.
- ▣ STALLINGS, W. SNMP, SNMPv2 and CMIP – The practical guide to network management standards. Reading: Addison-Wesley, 1993.
- ▣ SWISHER, Valerie, HARRIS, David. MARNEY-PETIX, V.C. Mastering Network Managemetn – Self – Paced Learning Series. Numidia-Press. 1967.
- ▣ TANENBAUM, Andrew S. Redes de Computadores. 3<sup>a</sup> edição. Prentice Hall, 1997.



- ▣ TAROUCO, Liane M.R. Inteligência Artificial Aplicada ao Gerenciamento de Redes de Computadores. São Paulo: USP-Escola Politécnica, 1990.
- ▣ VERNA, Dinesh. Policy-Based Networking. New Riders. 2001.



**Mauro Tapajós** é mestre em Engenharia Elétrica pela Universidade de Brasília.

**Liane Tarouco** é mestre em Ciências da Computação pela Universidade Federal do Rio Grande do Sul e doutora em Engenharia Elétrica/Sistema Digitais pela Universidade de São Paulo.

**Leandro Bertholdo** é mestre em Ciências da Computação pela Universidade Federal do Rio Grande do Sul.

**Francisco Marcelo Marques de Lima** é Mestre em Engenharia Elétrica pela Universidade de Brasília e Mestre em Liderança pela Universidade de Santo Amaro.

**Vanner Vasconcellos** é especialista em Redes de Computadores pela UFPA.

LIVRO DE APOIO AO CURSO

O curso desenvolve competências para analisar as necessidades da gerência de redes, entendendo a estrutura de uma solução de gerência e o suporte adequado a ela. O conhecimento adquirido sobre as principais ferramentas do mercado permitirá ao aluno montar em laboratório uma solução integrada de gerenciamento de redes. Serão estudados os conceitos básicos de Gerência de Redes, as áreas funcionais de gerenciamento (modelo OSI), o modelo ITIL, os protocolos de gerenciamento mais utilizados na prática, as MIBs padronizadas, os Sistemas de Gerenciamento de Redes (NMS), ferramentas livres de gerenciamento, Service Level Agreements (SLA), mecanismos de QoS, as principais plataformas de gerenciamento e as ferramentas de diagnóstico e monitoração de redes.