**UNIVERSIDADE FEDERAL DO CEARÁ**
**DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA**
**CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA**

# Gerenciamento SNMP com Autenticação Remota: Aplicações em UPSs

***AUTOR: JARBAS ARYEL NUNES DA SILVEIRA***

***ORIENTADOR: HELANO DE SOUSA CASTRO***

*Dissertação submetida à coordenação do curso de pós-graduação em Engenharia de Teleinformática, como requisito parcial para a obtenção do grau de* ***Mestre em Engenharia de Teleinformática****.*

**FORTALEZA –CE**

**MAIO - 2006**

# Agradecimentos

A Deus, sem o qual nada seria possível;

Aos meus pais, que sempre me apoiaram em minha educação;

Ao meu orientador, Helano de Sousa Castro, pela preciosa orientação neste trabalho;

Ao meu irmão Ricardo Jardel Nunes da Silveira, pelos valiosos conselhos;

Aos companheiros de laboratório da Microsol, Alonso Marinho e Andre Rodrigues Pinheiro, pelo grande ajuda prestada;

Ao Diretor da Microsol, Valdelírio Soares, pela grande ajuda e oportunidade.

# Notas do Autor

Mencionaremos aqui alguns dos padrões adotados neste, para melhor compreensão e entendimentos dos textos e figuras.

1. Todos os termos provenientes da língua inglesa foram escritos em *Itálico*. Os nomes específicos da descrição dos protocolos utilizados foram mantidos em inglês.

2. Os termos particulares da aplicação aqui proposta foram escritos em **Negrito**.

3. Foram utilizadas “aspas” para citações e termos em português próprios da aplicação.

# Resumo

Devido ao crescente uso de computadores e a forte dependência dos mesmos em várias aplicações, as UPS (Uninterruptible Power Supplies) vêm adquirindo cada vez mais importância. Estes dispositivos encontram-se muitas vezes dentro de um contexto de um sistema distribuído, o que fortalece a necessidade de gerenciamento dos mesmos. Devido à grande utilização das UPS em computadores, e devido ao fato destes, por sua vez, estarem inseridos em um ambiente de redes TCP/IP, torna-se bastante viável a idéia de se usar um sistema de gerenciamento que seja compatível com esse ambiente, e que faça uso de suas potencialidades. No presente trabalho é apresentado a implementação de um sistema de gerenciamento SNMP (Simple Network Management Protocol) para UPS que possibilita o gerenciamento de um número arbitrário de UPS, e que permite a autenticação remota para gerentes, facilitando o uso para um maior número de dispositivos. Detalha-se a implementação do sistema bem como os testes realizados. No final do trabalho é apresentada uma proposta para trabalhos futuros na área.

# Abstract

The reliability of a computer system is based not only on the computer itself, but also on its related power system. For that reason, more and more critical systems are counting on UPSs (Uninterruptible Power Supplies) to account for a continuous supply of power. Although the task of managing an UPS for a standalone system be quite straightforward, when it comes to computer distributed systems that task has to follow an equally distributed approach. Depending on the strategy used to manage that distributed UPS network, this task may be very complex. However, keeping in mind that most computer systems in a distributed architecture make use of TCP/IP to be logically connected, one could profit from the facilities provided by this protocol and design a management strategy that makes use of it. In fact, TCP/IP is not the only suite of protocols used to interconnect computer systems. SNMP (Single Network Management Protocol) is also frequently used as a way of managing (hardware and/or software) devices associated with computers. This dissertation proposes and implements a way of managing distributed UPSs by using SNMP, taking advantage of all the facilities furnished by the protocol. This works describes how the use of SNMP makes it possible to manage an arbitrarily number of UPSs remotely. This makes the design of UPS management systems a relatively easy task, and contributes to introduce a high degree of flexibility on the task of managing distributed UPSs. Finally, it is described how the system and tests were implemented as well as proposals for future work is presented.

# Lista de Símbolos

**ANS.1** – Abstract Syntax Notation One

**CMIP** – Commom Management Internet Protocol

**CMOT** – CMIP Over TCP/IP

**CCITT** – Comité Consultatif International Téléphonique et Télégraphique

**HTTP** – Hyper Text Transfer Protocol

**IETF** – The Internet Engineering Task Force

**IAB** – Internet Activities Board

**IP** – Internet Protocol

**LAN** – Local Area Network

**MTBF** – Mean Time Between Failure

**NIST** – US National Institute for Standards and Technology

**OSI** - Open Systems Interconnection

**PSTN** – Public Switched Telephony Network

**SMI** – Structure of Management Information

**SNMP** – Simple Network Management Protocol

**TCP/IP** – Transmission Control Protocol, Internet Protocol

**UDP** – User Datagram Protocol

**UPS** – Uninterruptible Power Supply

**TCP** – Transmission Control Protocol

**WAN** – Wide Area Nework

# Sumário

Agradecimentos ..... iii
Notas do Autor ..... iv
Resumo ..... v
Abstract ..... vi
Lista de Símbolos ..... vii
Sumário ..... viii
Lista de Figuras ..... xi
Lista de Tabelas ..... xiii
Capítulo 1 ..... 14
1. Introdução a Gerenciamento de UPSs ..... 14
1.1 – Introdução ..... 14
Capítulo 2 ..... 19
2. UPSs e Gerenciamento ..... 19
2.1 - As UPS e suas aplicações ..... 19
2.2 - Principais soluções para problemas de má qualidade de fornecimento de energia elétrica ..... 22
2.3 – Principais Topologias de UPSs ..... 24
2.4 – UPSs e gerenciamento ..... 30
2.5 – Comparação com soluções comerciais de gerenciamento existentes ..... 33
Capítulo 3 ..... 35
3. O Protocolo SNMP ..... 35
3.1 – Histórico ..... 35
3.2 - Conceitos do Protocolo SNMP ..... 39
3.3 – As Informações de Gerenciamento ..... 45
3.4 – A Base de informação de Gerenciamento: MIB ..... 46
3.4.1 – A Estrutura da MIB ..... 48
3.4.2 – A Sintaxe da MIB ..... 52
3.5 – As mensagens SNMP ..... 55
3.6 – Aplicações ..... 57
3.7 – Paradigmas em SNMP ..... 59

3.8 – Versões do Protocolo SNMP ..... 63
3.9 – Considerações Finais ..... 66
Capítulo 4 ..... 68
4. Gerenciamento SNMP com Autenticação Remota: Aplicações em UPSs ... 68
4.1 – Introdução ..... 68
4.2 – Soluções Estudadas ..... 69
4.3 – Justificativa ..... 74
4.4 – A Implementação do Sistema ..... 76
4.4.1 – A MIB Utilizada ..... 78
4.4.1.1 - Descrição da MIB ..... 78
4.4.1.2 – Grupo de Identificação ..... 79
4.4.1.3 - Grupo de Bateria ..... 80
4.4.1.4 – Grupo de Entrada ..... 81
4.4.1.5 - Grupo de Saída ..... 82
4.4.1.6 - Grupo de Bypass ..... 84
4.4.1.7 - Grupo de Alarmes ..... 85
4.4.1.8 - Grupo de Testes ..... 87
4.4.1.9 - Grupo de Controle ..... 88
4.4.1.10 - Grupo de Configuração ..... 89
4.4.2 – A MIB Complementar ..... 91
4.4.2.1 – Descrição da MIB Complementar ..... 92
4.4.2.2 – Grupos Implementados ..... 94
4.4.3 – Funcionalidades do Sistema ..... 96
4.4.3.1 - O Agente e suas funções ..... 98
4.4.3.2 - O Gerente e suas funções ..... 100
4.4.3.3 - O Autenticador de Gerentes ..... 102
4.4.4 – Operações do Sistema ..... 104
4.4.4.1 – A leitura de recursos do objeto gerenciado ..... 104
4.4.4.2 – A escrita nos recursos do objeto gerenciado ..... 105
4.4.4.3 – Os alarmes do objeto gerenciado ..... 106
4.5 – Testes e Validação ..... 107
4.5.1 – Operação de leitura ..... 107

4.5.2 – Operação de Escrita ..... 109
4.5.3 – Alarmes ..... 110
4.5.4 – Métrica de quantidade de estações a serem gerenciadas ..... 111
4.6 – Considerações Finais ..... 113
Capítulo 5 ..... 114
5. Conclusões e trabalhos futuros ..... 114
5.1 – Conclusão ..... 114
5.2 - Trabalhos futuros ..... 117
Referências ..... 119
Apêndice ..... 129
Apêndice A – Mensagens SNMP ..... 129
A.1 – Estrutura geral de uma mensagem SNMP ..... 129
A.2 – A Mensagem SNMP Get-Request ..... 133
A.3 – A Mensagem SNMP Get-NextRequest ..... 136
A.4 – A Mensagem SNMP Get-Response ..... 138
A.5 – A Mensagem SNMP Set-Request ..... 141
A.6 – A Mensagem SNMP Trap ..... 144
Anexos ..... 148
Artigo Publicado ..... 148

# Lista de Figuras

1. Introdução a Gerenciamento de UPSs ..... 14
Figura 1.1: típica rede de computadores WAN (Wide Area Network). ..... 15
2. UPSs e Gerenciamento ..... 19
Figura 2.1: as diversas topologias de UPSs, quanto ao seu funcionamento. ..... 24
Figura 2.2: topologia de UPS Standby. ..... 26
Figura 2.3: topologia de UPS Interativa. ..... 27
Figura 2.4: topologia de UPS Online. ..... 28
3. O Protocolo SNMP ..... 35
Figura 3.1: o SNMP incluído dentro do conjunto de protocolos TCP/IP. ..... 36
Figura 3.2: o funcionamento do PING. ..... 37
Figura 3.3: elementos em um sistema de gerenciamento SNMP. ..... 40
Figura 3.4: fluxo de uma mensagem SNMP em um agente. Adaptada de (COMER, 1994). ..... 42
Figura 3.5: relação entre as linguagens utilizadas em SNMP e mensagens de gerenciamento utilizadas neste protocolo. ..... 46
Figura 3.6: estrutura hierárquica da MIB. ..... 49
Figura 3.7: objetos da MIB-II. ..... 51
Figura 3.8: fluxo por Estímulo Gerente – Agente (Leitura). ..... 55
Figura 3.9: fluxo por Estímulo Gerente – Agente (Escrita). ..... 56
Figura 3.10: fluxo por Estímulo interno do Agente (Alarme). ..... 56
Figura 3.11: fases de aquisição de dados em uma gerente SNMP. ..... 61
Figura 3.12: versões do protocolo SNMP. ..... 65
4. Gerenciamento SNMP com Autenticação Remota: Aplicações em UPSs ..... 68
Figura 4.1: proposta de solução de gerenciamento proprietária. ..... 70
Figura 4.2: implementação com duas redes. ..... 72
Figura 4.3: gerenciamento via navegador. ..... 73
Figura 4.4: gerenciamento via SNMP. ..... 74
Figura 4.5: conceito de comunidade em SNMP. ..... 75
Figura 4.6: ambiente de gerenciamento completo. ..... 76
Figura 4.7: hierarquia da MIB para UPS. ..... 79

Figura 4.8: MIB complementar - Módulo privativo GSAR. ..... 92
Figura 4.9: MIB complementar - Módulo privativo GSAR. ..... 93
Figura 4.10: Estrutura da MIB complementar. ..... 93
Figura 4.11: MIB complementar - Módulo identidade em ASN.1 ..... 94
Figura 4.12: modelo de definição de um grupo em ASN.1. ..... 95
Figura 4.13: estrutura da MIB no sistema implementado. ..... 97
Figura 4.14: camadas funcionais do agente SNMP. ..... 99
Figura 4.15: camadas funcionais do gerente SNMP. ..... 101
Figura 4.16: gerente SNMP Corporativo. ..... 102
Figura 4.17: a funcionalidade do Autenticador. ..... 103
Figura 4.18: mensagem SNMP Get-Request. ..... 108
Figura 4.19: uma mensagem SNMP Get-Response. ..... 109
Figura 4.20: uma mensagem SNMP Set-Request. ..... 110
Figura 4.21: uma mensagem SNMP Trap. ..... 111
5. Conclusões e trabalhos futuros ..... 114
Figura 5.1: gerenciamento SNMP com acesso via HTTP ..... 118
Apêndice ..... 129
Figura A.1: estrutura geral de uma mensagem SNMPv1. ..... 129
Figura A.2: cabeçalho de PDU Get ou Set. ..... 131
Figura A.3: a PDU Get-Request. ..... 133
Figura A.4: operação de Set-Request simples. ..... 141
Figura A.5: operação de Set-Request Robusta. ..... 142
Figura A.6: cabeçalho de PDU Trap. ..... 145
Anexos ..... 148

# Lista de Tabelas

1. Introdução a Gerenciamento de UPSs ..... 14
2. UPSs e Gerenciamento ..... 19
3. O Protocolo SNMP ..... 35
4. Gerenciamento SNMP com Autenticação Remota: Aplicações em UPSs ... 68
Tabela 4.1: módulo de Identificação – MIB UPS RFC 1628. ..... 80
Tabela 4.2: módulo de Baterias – MIB UPS RFC 1628. ..... 81
Tabela 4.3: módulo de Entrada – MIB UPS RFC 1628. ..... 82
Tabela 4.4: módulo de Saída – MIB UPS RFC 1628. ..... 83
Tabela 4.5: módulo de Bypass – MIB UPS RFC 1628. ..... 84
Tabela 4.6: módulo de Alarmes – MIB UPS RFC 1628. ..... 85
Tabela 4.7: alarmes – MIB UPS RFC 1628. ..... 86
Tabela 4.8: testes definidos – MIB UPS RFC 1628. ..... 87
Tabela 4.9: módulo de testes – MIB UPS RFC 1628. ..... 88
Tabela 4.10: módulo de controle – MIB UPS RFC 1628. ..... 89
Tabela 4.11: módulo de configuração – MIB UPS RFC 1628. ..... 90
5. Conclusões e trabalhos futuros ..... 114
Apêndice ..... 129
Tabela A.1: valores de erros em mensagens SNMP. ..... 132
Tabela A.2: PDU Get-Request codificada. ..... 135
Tabela A.3: PDU Get-NextRequest codificada. ..... 137
Tabela A.4: mensagem Get-Request com respectiva Get-Response. ..... 139
Tabela A.5: PDU Get-Response codificada. ..... 140
Tabela A.6: PDU Set-Response codificada. ..... 143
Tabela A.7: tipos Padrões de Mensagens Trap. ..... 146
Tabela A.8: PDU Trap Cold Start. ..... 147
Anexos ..... 148

# Capítulo 1

# 1. Introdução a Gerenciamento de UPSs

## 1.1 – Introdução

Os sistemas de computação são hoje largamente utilizados em todos os segmentos da sociedade, desde uma simples compra em uma loja de roupas até numa transação bancária. Sendo os computadores tão importantes nos dias atuais, foram criados vários mecanismos de segurança para garantir o bom funcionamento dos mesmos. Um destes mecanismos refere-se à energia elétrica que garante o bom funcionamento de sistemas computacionais (MYERSON, 2002). Dentre as opções de garantia ininterrupta de energia, podem-se citar as UPSs (*Uninterruptible Power Supply*), que são dispositivos que garantem a qualidade de energia elétrica[1] fornecida para as cargas por ela alimentadas, mesmo numa falha da rede elétrica convencional, fornecendo energia elétrica para suas cargas, no tempo disponível de sua autonomia, normalmente provida por baterias.

Em computação distribuída, as informações trocadas pelos computadores encontram-se normalmente em máquinas geograficamente distribuídas, cada uma com seu próprio suprimento de energia fornecido, por exemplo, por uma UPS. Como a disponibilidade das informações está relacionada com o estado operacional destes suprimentos, é importante monitorar o estado de tais equipamentos. Em outras palavras, é desejável se realizar o gerenciamento de UPSs, quando estas são os equipamentos que suprem energia para o sistema. Este é o cenário desta dissertação.

[1] O termo Qualidade de Energia, citado nesta dissertação, refere-se à garantia dos valores de freqüência e tensão elétrica dentro do limite de operação aceito pelas cargas, não sendo alusivo à definição mais completa de Qualidade de Energia.

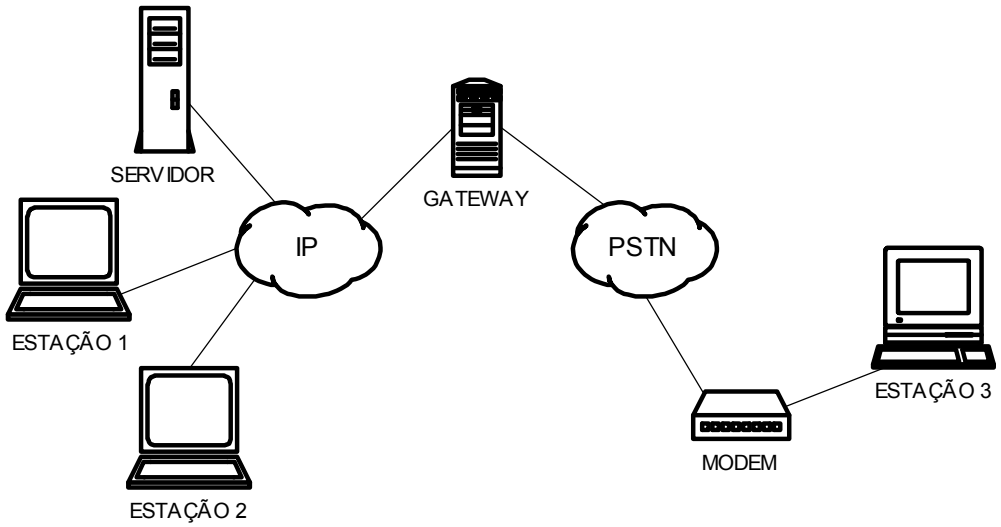


Figura 1.1: típica rede de computadores WAN (Wide Area Network).

Por outro lado, a grande complexidade dos sistemas computacionais hoje implementados, envolvendo terminadores de rede, redes comutadoras de pacotes (PSTNs), *gateways* e outros elementos (Figura 1), torna não trivial a tarefa de se introduzir mais um ponto a ser gerenciado neste sistema, caso o mesmo na seja compatível com os protocolos de comunicação do meio (GILADI, 2004). A dificuldade é ainda maior quando se leva em consideração alguns aspectos como: interfaces de gerenciamento, tráfego de dados na rede, meios de transmissão utilizados, operação do sistema, dentre outros.

As UPSs não são fundamentais somente em ambientes computacionais. O uso de uma garantia de alimentação ininterrupta pode ser vista em várias aplicações, como um simples eletrodoméstico em casas inteligentes ou em aplicações críticas como equipamentos de sustentação da vida, controle de tráfego de trens e sistemas de controle de tráfego aéreo.

A utilização de UPSs é também fundamental em situações em que se deseja garantir integridade e segurança de dados. Em muitos sistemas, o desligamento do suprimento de energia sem uma prévia solicitação, pode destruir equipamentos e dados, sendo estes últimos muitas vezes irrecuperáveis. Embora a maioria destes sistemas já implementem

internamente técnicas de tolerância a falhas e rotinas de recuperação de dados, eles podem tornar-se intolerantes a esse tipo de falha, ao seu desligamento ou à falta brusca de alimentação. Em operações de sistemas bancários, por exemplo, em que a perda de dados pode gerar imensos prejuízos, torna este problema ainda mais complexo. Podem-se citar também aplicações em aeroportos com grande fluxo de partidas e aterrissagens, em que a necessidade de controle integrado do sistema é ainda maior.

Em resumo, pode-se dizer que as UPSs desempenham um papel fundamental no funcionamento de sistemas simples ou complexos, estando presente, na maioria das vezes, de forma distribuída e em locais diferentes. Esta característica de funcionamento exige um sistema de gerenciamento eficiente, que seja compatível com as soluções de gerenciamento já existentes, não acarretando em mais complexidade, nem custo adicional para meios de transmissão. Dentro desta temática, define-se o foco principal do problema da dissertação: desenvolver um sistema de gerenciamento para UPS que seja compatível com os protocolos de gerenciamento de redes de computadores hoje existentes, e possibilite gerenciar um número arbitrário de UPSs.

Podem-se definir sistemas computacionais distribuídos como sendo uma coleção de computadores independentes que se apresenta ao usuário como um sistema único e consistente (TANENBAUM, 2001). Para sintetizar este exemplo vamos descrever o processo genérico de um saque em um caixa eletrônico. Inicialmente é feita uma coleta de informações do usuário através de algum identificador. As mesmas são encaminhadas para um outro servidor, onde esses dados serão analisados e comparados em um banco de dados. Este mesmo banco pode estar distribuído em diversos computadores que, muitas vezes, podem estar separados por quilômetros de distância, e a informação pode trafegar pelos mais diversos roteadores. É fácil perceber que o sistema tornar-se-ia comprometido em caso de uma falha em um dos elementos de rede. Portanto vemos a importância de um bom gerenciamento das fontes de alimentação de cada um destes sistemas.

As redes de computadores permitem compartilhar recursos, resultando na otimização dos mesmos, bem como na redução de custos na implementação do sistema. Em contrapartida, elas necessitam que todos os seus elementos funcionem corretamente, sendo isso alcançado através, entre outros mecanismos, de um sistema de gerenciamento de redes. Dentro desta problemática se define a necessidade de implementação de um sistema de gerenciamento para UPSs, que são elementos vitais para o bom funcionamento de um sistema distribuído.

Nesta dissertação, apresentaremos uma solução para problemas de gerenciamento de UPSs que consiste na proposta e implementação de um sistema que permita gerenciar um número arbitrário de UPSs, sendo compatível com as ferramentas de gerenciamento de redes de computadores hoje existentes. Além disso, propomos e implementamos também uma ferramenta gerencial, que possibilita, através de uma autenticação remota, construir uma rede de gerenciamento, facilitando a sua aplicação em um maior número de UPSs. Este trabalho teve como alvo um pacote de software comercializado atualmente por uma empresa de UPSs.

A solução do problema aqui descrito está explanada nesta dissertação em 5 capítulos. No Capítulo 2 iremos dissertar sobre a utilização das UPSs, os tipos e topologias existentes, e o surgimento das UPSs inteligentes. Descreveremos os paradigmas da utilização de sistemas de gerenciamento em UPSs e os tipos de soluções existentes.

No Capítulo 3 definiremos o protocolo de comunicação utilizado, bem como um breve histórico do seu surgimento, suas vantagens e desvantagens. Descreveremos também vários tipos de aplicações, e os tipos de problemas existentes em sua implementação.

No Capítulo 4 descreveremos a implementação da solução aqui proposta. Apresentaremos as soluções estudadas, enumerando suas vantagens e desvantagens. Descrevemos também a arquitetura utilizada, os gerentes e suas interfaces, funcionalidades e recursos disponibilizados. Dissertaremos sobre os recursos de autenticação remota que possibilitaram o uso desta solução para utilização em gerência de redes. Por fim, faremos uma validação da utilização do sistema utilizando uma ferramenta de análise de protocolos de redes, com o intuito de verificar a compatibilidade da solução de gerenciamento. Faremos também o cálculo de uma métrica que possibilita uma sucinta análise de desempenho da ferramenta de gerenciamento.

No Capítulo 5 concluiremos, enumerando os objetivos alcançados com a pesquisa, destacando suas vantagens e desvantagens. Faremos um paralelo com o estado da arte em gerenciamento de redes para situarmos a solução proposta dentro do atual contexto. Relataremos também algumas futuras soluções e melhorias a serem implementadas em pesquisas futuras.

# Capítulo 2

# 2. UPSs e Gerenciamento

## 2.1 - As UPS e suas aplicações

Como citado anteriormente, pela dependência causada na utilização de sistemas computacionais, o usuário anseia por, cada vez mais, proteções contra falhas nestes sistemas. Este problema pode ser formulado tendo em vista três situações: proteção de dados, execução de serviços e proteção de equipamentos. Na primeira, o usuário deseja proteger dados, sejam eles armazenados há muito tempo ou simplesmente processados há alguns minutos. Como exemplo, podemos citar a simples edição de um documento em uma planilha ou os dados de um cliente cadastrado, em que sua perda demandasse tempo para refazer uma planilha ou, no caso do cliente, talvez fosse impossível contatá-lo novamente. Na segunda situação, temos a garantia de execução de programas, que muitas vezes podem ser a garantia de vidas ou a simples operacionalidade de um atendimento. Como exemplo, podemos citar o funcionamento de uma loja de vendas, em que o faturamento é feito de maneira automática utilizando impressoras fiscais, o que é muito comum nos dias de hoje. A paralisação de um desses sistemas pode representar a impossibilidade de faturamento em uma atividade comercial simples. Na terceira situação, temos a proteção do próprio equipamento, que está intimamente ligada aos custos do mesmo. Muitos dispositivos não suportam os surtos de tensão em uma rede elétrica de má qualidade. Isso pode refletir em custo de manutenção do equipamento e, indiretamente, em custos de paralisação e de perdas de dados.

De todas as proteções elétricas disponíveis, a UPS é uma das mais eficientes (BALACHANDRA, 2000). Isto se deve ao fato da UPS dispor de energia armazenada para suprir o equipamento quando da ocorrência de um

surto de tensão na alimentação, proveniente da rede elétrica. Podemos dizer que a principal função de uma UPS é manter energia de forma ininterrupta e de qualidade, para o tipo de carga a qual a mesma foi projetada.

Diante do fato apresentado, algumas variáveis devem ser consideradas, como por exemplo, o tempo de autonomia. A autonomia de uma UPS se entende como sendo o tempo de fornecimento de energia para a alimentação da carga, quando da ausência de energia elétrica fornecida externamente. Esta função é executada transformando a energia armazenada, por exemplo, nas baterias internas ou externas, em uma tensão de saída adequada à carga para a qual foi projetada. Dependendo da previsão de falhas, e do comportamento das mesmas ao fornecer energia elétrica, da potência fornecida às cargas e ainda levando em consideração o MTBF (Mean Time Between Failures) da rede elétrica, define-se a quantidade de ampéres-hora das baterias (LINZ, 2001).

Em sistemas *Tolerantes a Falhas*, devemos prevenir a ocorrência de falhas que não sejam antecipadas. Podemos entender como falha não antecipada aquela que não consegue ser percebida pelo sistema para que se tome uma prévia decisão, sem que o sistema seja avisado com antecedência do acontecimento do mesmo (CASTRO, 1992). Como exemplo, podemos citar o fim de autonomia das baterias em uma UPS.

Sabemos que, em sua grande maioria, os sistemas exigem procedimentos de desligamento complexos. Podemos citar, como exemplo, uma central de atendimento que faz uso de servidores de dados, em que vários atendentes utilizam informações provenientes desses servidores. Os atendentes necessitam de avisos que sinalizem a iminente paralisação dos serviços. Em outras aplicações, podem acontecer danos maiores, como casos de controle de máquinas que exigem inter-travamento mecânico, sendo necessário estabelecer seus atuadores em uma posição de falha segura, evitando que na ausência do funcionamento ocorram situações que acarretem

maior prejuízo.

O fim da autonomia das baterias exige o desligamento do sistema, de forma segura, sendo importante informar aos sub-sistemas a iminência deste fato. É neste contexto que se definem UPSs inteligentes como sendo aquelas que enviam às cargas sinalização dos principais eventos ocorridos. Esta sinalização, na maioria dos fabricantes, se faz de duas maneiras: através de comunicação serial (RS232 ou USB) ou interface de sinais on/off. Este nível de informações evoluiu de tal forma que, hoje, temos também parâmetros de tensão, corrente elétrica, freqüência e várias outras variáveis pertinentes à energia elétrica, no sentido de medir sua qualidade e fazer estatísticas do fornecimento da mesma (SMITH, 1995).

Modernos programas de monitoração de UPSs permitem o armazenamento periódico de dados, sendo possível a geração de gráficos que possibilitam a visualização da qualidade da energia de entrada e saída das UPSs. É possível ainda realizar cálculos estatísticos, bem como prever falhas, dimensionar parâmetros para melhoria do sistema, por exemplo, através da visualização de dados coletados por um sistema de monitoração em uma determinada rede elétrica. Pode-se ainda dimensionar um banco de baterias que venha prover a autonomia necessária para suprir o intervalo das falhas nesta rede elétrica.

Pelos motivos citados, pode-se observar uma ampla utilização de UPSs hoje em várias aplicações. Esta tendência pode ser entendida pela necessidade de uma energia de qualidade para que os sistemas possam funcionar corretamente. Alguns dispositivos se utilizam de modernas tecnologias, como sistemas inteligentes adaptativos, para obterem maior rendimento e, para isso, utilizam processadores de sinais que efetuam cálculos matemáticos complexos para obterem o desempenho a que se propõem, exigindo assim uma energia confiável e de qualidade. Além disso, o alto valor físico agregado que as aplicações adquirem no dia a dia das pessoas. Uma

simples central de telefones pode paralisar uma empresa, caso a mesma fique indisponível por poucas horas. Todos estes fatores fazem da UPS uma peça importante no funcionamento de vários sistemas.

## 2.2 - Principais soluções para problemas de má qualidade de fornecimento de energia elétrica

A alta disponibilidade de fornecimento de energia elétrica é um requisito muito importante em equipamentos que se utilizam da tecnologia da informação, bem como dos dispositivos inteligentes utilizados em controles industriais e outras aplicações. Apesar do projeto desses equipamentos poder incorporar mecanismos de tolerância a falhas internamente, eles podem encontrar-se ainda vulneráveis a alguma falha no suprimento de energia elétrica.

A alimentação elétrica dos computadores é considerada umas das maiores vulnerabilidades nas redes de computadores corporativas (MYERSON, 2002). Podemos citar como principais problemas na rede elétrica: problemas de transientes, que são anormalidades de amplitude ou frequência na rede elétrica; afundamentos de tensão, caracterizados por quedas de tensão temporárias no fornecimento de energia elétrica; variações de frequência e falhas na rede, caracterizadas por afundamento abaixo de um determinado nível de tensão (SOLTER, 2002). As causas de problemas na rede elétrica podem ter diversos motivos. Dentre eles podemos mencionar a ocorrência de distúrbios na transmissão de energia elétrica, como quedas de cabos causadas por raios e fadiga mecânica nas conexões elétricas.

Os efeitos de distúrbios nos equipamentos podem ser observados de diversas maneiras, desde a diminuição de sua vida útil, o que é muito comum em equipamentos eletromecânicos, até a destruição de partes que não suportam esses surtos de tensão, provocando sua danificação completa.

Essa dependência dos sistemas de uma boa qualidade de energia

motivou, ao longo do tempo, a concepção de várias soluções com o intuito de neutralizar problemas relacionados a esta dependência. Em muitos casos, as soluções para este problema já vêm embutidas no próprio equipamento, na forma de proteções. Embora tais proteções internas dos equipamentos tenham evoluído de maneira significativa, na maioria das vezes, elas não são suficientes. Este fato deve-se tanto pelo aspecto do custo, que poderia se tornar muito elevado, como pelo espaço e integração dessas soluções.

Uma solução bastante utilizada, e das mais simples, é o uso de filtros. Estes têm por principal função a proteção de equipamentos pela extração, de componentes de alta frequência causados por interferência elétricas ou magnéticas. Os filtros não funcionam nos surtos de tensão, como afundamentos e sobretensões.

Os transformadores isoladores, soluções também muito utilizadas em proteção de equipamentos, têm um funcionamento similar ao de um filtro, rejeitando altas freqüências, e isolando a presença de possíveis distúrbios e interferências advindas de um mau aterramento. Pode-se também citar o uso de reguladores de tensão, que protegem os equipamentos contra variações de tensão, desde que estas variações estejam dentro do seu intervalo de atuação, que é o intervalo máximo de variação de tensão na entrada que o regulador consegue garantir um nível mínimo de variação na tensão de saída. Eles são muito úteis quando o nível da rede elétrica oscila entre valores fora dos quais os equipamentos podem funcionar.

Finalmente, uma solução bastante completa, as UPSs, que podem conter todas as outras soluções aqui já relatadas, pois existem modelos de UPSs que já vêm incorporadas com filtros de linha, transformadores isoladores e estabilizadores de tensão, protegendo os equipamentos contra variações de frequência, de amplitude, presença de harmônicos e sub-tensões ou sobre-tensões.

## 2.3 – Principais Topologias de UPSs

O avanço da tecnologia proporcionou um avanço equivalente das UPSs, no sentido de busca de uma melhor qualidade de energia e mais recursos de monitoração, bem como uma de redução de custos. Esta evolução foi influenciada pelos tipos de carga a que se destinam as UPSs, ou seja, o tipo de aplicação. Na Figura 2.1 é apresentada uma visão das diversas topologias de UPS, referentes ao quesito funcionamento.

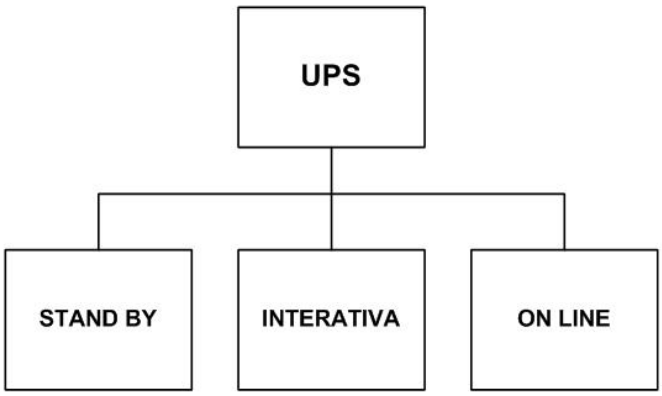


Figura 2.1: as diversas topologias de UPSs, quanto ao seu funcionamento.

Muitas vezes, são cometidos erros ao designar qual a topologia da UPS a ser utilizada. Este fato normalmente acontece devido à competição entre os fabricantes, que desejam vender seu produto, divulgando termos criativos, como QUASE ON LINE ou SEMI ON LINE (SOLTER, 2002). Para evitar erros deste tipo, nos referenciamos à norma internacional de UPSs (IEC 62040-3, 1999) para apresentarmos as topologias existentes.

Para compreendermos a diferença entre as diversas topologias de UPS, precisamos introduzir o conceito de tempo de transferência. A UPS tem dois modos de funcionamento bem distintos. No primeiro, chamado de modo rede, a mesma está fornecendo energia para a carga, mas também está recebendo energia da rede elétrica, o que permite que as baterias permaneçam carregadas, mantendo-as assim em um nível constante a sua disponibilidade

de uso em sua total autonomia. Além disso, a UPS também usa a mesma energia da entrada para alimentar suas cargas. No momento em que ocorre a falha no fornecimento da rede elétrica, a UPS passa a funcionar no segundo modo de funcionamento, chamado de modo inversor. Neste caso, a UPS passa a usar a energia das baterias para fornecer alimentação à carga.

No instante da falha do fornecimento de energia, devido à diferença no modo de funcionamento da UPS, pode haver um reflexo no padrão de energia que é fornecida à carga nesse instante. Essa mudança pode ser caracterizada pelo intervalo de tempo que a tensão elétrica leva para voltar às condições estabelecidas. Esse tempo é denominado tempo de transferência.

A norma internacional de UPS (IEC 62040-3, 1999) define uma UPS do tipo Stand-by no modo rede como aquela que tem suas tensão e freqüência de saída dependentes da tensão e frequência da alimentação de entrada. Nesta definição, podemos observar o fato de que, no modo rede, a saída da UPS tem o mesmo valor da alimentação de entrada e, no caso de alteração da alimentação de entrada para um comportamento além dos limites especificados, a UPS assume o modo inversor. Observa-se que esta topologia apresenta tempo de transferência não nulo e também que não é realizada a estabilização da tensão de saída, podendo a saída passar ou não por um transformador de tensão. Na Figura 2.2 é apresentado um diagrama de blocos desta topologia. O bloco de controle monitora a tensão de entrada, verificando se a mesma encontra-se dentro dos limites de tensão e freqüência especificados. Caso estes limites sejam extrapolados, o bloco de controle realiza a comutação entre o modo rede e inversor. O carregador, na presença de rede elétrica, fornece carga às baterias, para ser utilizada em modo inversor.

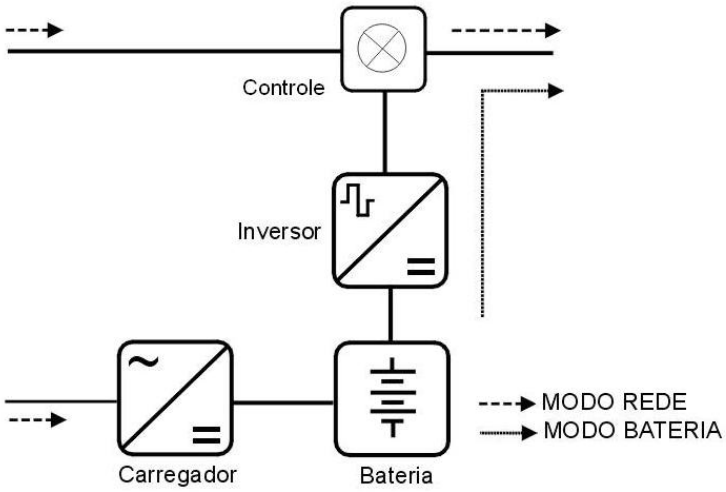


Figura 2.2: topologia de UPS Standby.

Na topologia de UPS chamada interativa, o funcionamento é caracterizado por, no modo rede, a tensão de saída ser independente da alimentação de entrada, enquanto que a frequência da tensão de saída é dependente (IEC 62040-3, 1999).

Desta definição, conforme o diagrama de blocos apresentado na Figura 2.3, observa-se que a entrada, no modo rede, é direcionada à saída através de um estabilizador que, dentro de uma faixa especificada em norma, realiza a estabilização da saída. A esta faixa especificada chamamos de faixa de regulação, que especifica dentro de uma faixa de variação na entrada, um valor máximo de variação na saída. O bloco de controle monitora a tensão de entrada para garantir a estabilização, quando dentro da faixa de regulação, comandando também a comutação do modo rede para o modo inversor, onde a saída é fornecida pelo bloco inversor, recebendo energia das baterias. Devido à saída estar conectada à entrada através de um estabilizador, não havendo controle da freqüência da tensão de saída, diz-se que a freqüência de saída é dependente da freqüência de entrada nas UPSs do tipo interativa.

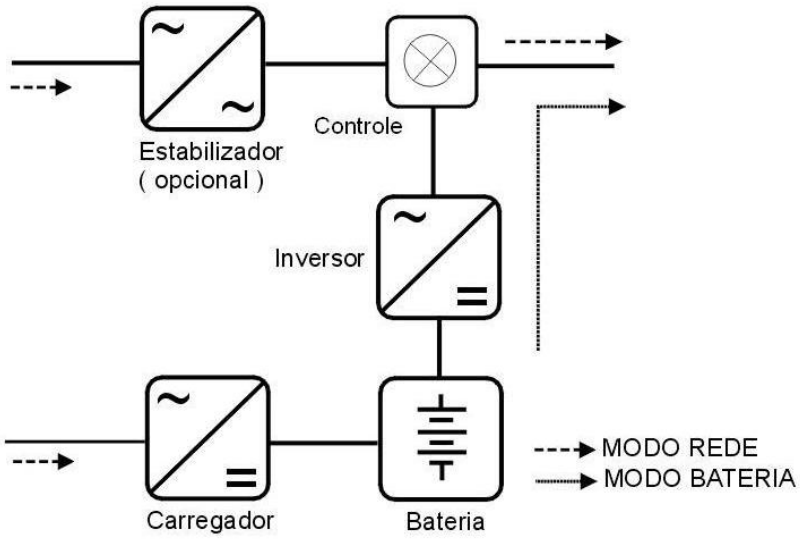


Figura 2.3: topologia de UPS Interativa.

A topologia de UPS do tipo on-line se caracteriza pelo fato de, no modo rede, a tensão e a freqüência de saída serem independentes da tensão e da freqüência da alimentação de entrada (IEC 62040-3, 1999). Como pode ser visto no diagrama de blocos apresentado na Figura 2.4, o inversor assume as cargas durante todo o tempo de funcionamento da UPS, independente da UPS encontrar-se ou não no modo rede. A carga é continuamente alimentada pelo conjunto retificador / inversor, operando em sistema de dupla conversão, isto é, conversão CA para CC e CC para CA. Quando as características da rede CA estiverem fora das faixas operacionais pré-estabelecidas da UPS, esta entra no modo bateria, onde o conjunto bateria / inversor continua a alimentar a carga pelo tempo de duração da energia armazenada na bateria, ou até o retorno da rede CA a sua faixa especificada, o que ocorrer primeiro. Nesta topologia o tempo de transferência é zero.

As UPSs ON-LINE normalmente são as mais eficientes em termos de proteção, no entanto são as que apresentam o maior custo (LINZ, 2001). Este fato acontece devido ao tipo de tecnologia empregada em sua construção, que por ser mais cara, só se justifica em potências mais altas, na maioria das vezes acima de 5 KVAs. Nesta categoria, a forma de onda elétrica de saída é

do tipo senoidal. Este fato é bastante importante no caso de se ter cargas do tipo linear, caso em que não é aconselhável utilizar forma de onda trapezoidal, que é a forma de onda característica das UPSs do tipo Interativa.

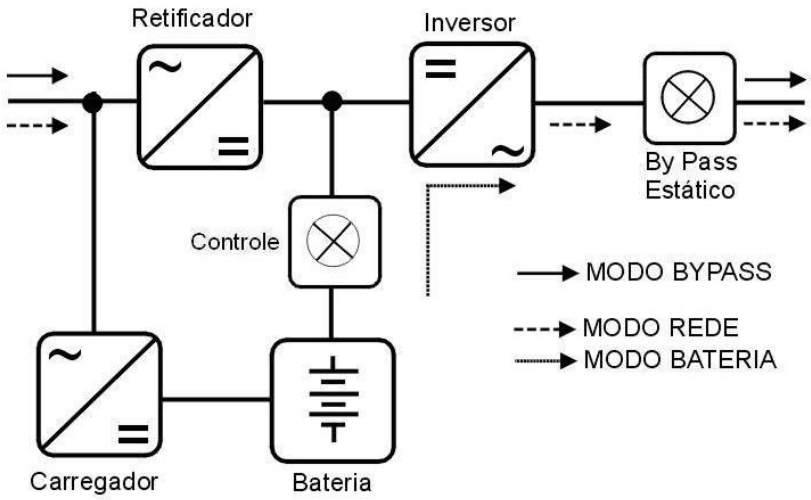


Figura 2.4: topologia de UPS Online.

Na categoria de UPSs ON LINE, a forma de onda de saída sempre passa por duas conversões, independente do modo de funcionamento. Para se entender a necessidade do uso de duas conversões, outros conceitos são requeridos. Um deles é o rendimento, que é a relação entre a energia de saída e a energia de entrada de um sistema. Este parâmetro torna-se bastante importante, pois nas conversões gasta-se muita energia, o que provoca uma queda acentuada no rendimento do sistema. Como estas conversões são sempre utilizadas, independente do modo de funcionamento, é necessário se ter um alto rendimento, pois do contrário o sistema se tornaria pouco viável do ponto de vista de usabilidade.

Todos os fatores acima contribuem ainda mais para a elevação dos custos desta categoria de UPSs. Outro fator a ser considerado é a confiabilidade desses sistemas. Como as potências nessa categoria normalmente são mais altas, pelo menos acima de 5 KVAs, a UPS se torna um ponto de falha que pode afetar um grande número de usuários, caso este sistema não seja redundante. Isso exige uma maior confiabilidade do

equipamento, que terá uma missão mais crítica a ser cumprida.

Existem ainda vários aspectos a serem considerados na análise de evolução das UPSs, principalmente no quesito qualidade de energia. Um desses é a distorção harmônica total da forma de onda de saída (KASSIK, 2000). Este fator de análise, o DTH (Distortion Total Harmonics) representa o quanto a forma de onda de saída se aproxima de uma função seno, que é a forma de onda desejada. Este fator merece uma maior atenção nas UPSs ON LINE, mas não tem sentido quando falamos em UPSs OFF LINE, pois as suas formas de onda são quadradas ou trapezoidais (SOLTER, 2002).

Outro fator relevante é a correção do fator de potência. Este por sua vez, representa a relação entre a potência útil e a potência aparente do sistema. O controle do fator de potência é exigido pelas concessionárias de energia elétrica, ficando o cliente passível de multas, caso o seu valor não esteja acima de um valor determinado (KASSIK, 2000).

Um componente fundamental das UPSs são as baterias. Elas são responsáveis pela energia armazenada que será utilizada na ocorrência de uma falha, e a quantidade de energia armazenada representará o tempo de autonomia do sistema. Nesse caso, é importante salientar características como a troca desse banco de baterias com o sistema em funcionamento. Esta característica de UPSs é conhecida como HOT-SWAP. Este fato permite que seja feita manutenção na UPS sem ser preciso desligá-la, permitindo que a carga continue sendo alimentada de maneira ininterrupta (LINZ, 2001).

Outra característica que pode estar presente em uma UPS é o BY-PASS. O mesmo pode ser definido como a comutação da alimentação da carga da UPS para a rede elétrica. Esta operação pode ser automática ou manual. Na primeira opção, o sistema dispõe de um dispositivo vigia que monitora o bom funcionamento da UPS. No caso da falha da UPS, uma comutação automática é feita e um alarme é acionado, e neste caso a carga

continua sendo alimentada com a energia da rede elétrica. Na segunda opção, é necessário a presença de um operador para realizar a comutação. O BY-PASS pode ainda ser do tipo SINCRONIZADO ou NÃO-SINCRONIZADO. Está característica diz respeito ao momento da comutação da energia fornecida à carga. No primeiro caso o instante de comutação acontece sempre no final de um semiciclo positivo para o início de um semiciclo negativo, ou vice-versa. Esta característica permite que a carga não tenha mau funcionamento no instante da comutação. No segundo caso, o instante da ocorrência da comutação é aleatório, e dependendo da sensibilidade da carga, pode levar à ocorrência de uma falha.

As UPSs também podem variar conforme a configuração de entrada e de saída de energia. Podemos ter UPSs do tipo monofásica – monofásica, trifásica - monofásica e trifásica – trifásica. O primeiro tipo é mais comum para UPSs até 10 KVA. Os outros casos se aplicam às UPSs de maior potência. No caso de uma UPS trifásica – trifásica, o custo se torna bastante elevado. Este fator é importante, sobretudo no tipo de carga a ser utilizada, no caso da UPS fornecer energia a uma carga que necessite alimentação trifásica.

## 2.4 – UPSs e gerenciamento

A crescente modernidade de equipamentos eletrônicos, e a fluência e rapidez dos modernos meios de comunicação hoje existentes, têm mudado os conceitos de alguns equipamentos, em especial as UPSs. Como já salientado anteriormente, os sistemas de energia ininterruptos têm por objetivo fornecer uma energia de qualidade, dentro das especificações exigidas. A incorporação da tecnologia relacionada com esses avanços nos permite projetar um elemento computacional gerenciador que monitore o funcionamento do conjunto de UPSs, avaliando parâmetros tais como qualidade de energia de entrada, análise do funcionamento, dentre outros.

Sabemos que alguns sistemas hoje dependem do bom

funcionamento de UPSs para funcionarem corretamente. O Windows XP, sistema operacional para servidores de rede da Microsoft, já vem inclusive com uma interface nativa de comunicação de sinais ON/OFF para comunicação com algumas UPSs compatíveis. Este tipo de comunicação ON/OFF, chamada também de interface de “contatos secos”, foi muito utilizada pelas UPSs até a década de 80, sendo uma interface de comunicação genérica para muitos equipamentos eletrônicos (SMITH, 1995).

Para classificarmos um sistema para gerenciamento de UPSs precisamos entender alguns fatores. Em um ambiente de rede de computadores iremos encontrar vários sistemas que deverão coexistir de maneira harmônica. Dentre esses sistemas podemos citar: Hubs, bridges, roteadores e switchs. Todos esses equipamentos também evoluíram muito em suas funcionalidades, inclusive no seu potencial de gerenciamento. Podemos executar várias funções através de suas ferramentas de gerenciamento, como realizar amostragens, dados estatísticos e até mesmo provocar uma reinicialização do sistema remotamente. Além disso, devemos considerar ainda que todos estes equipamentos utilizados em de redes de computadores há muito tempo já vêm se utilizando de ferramentas de gerenciamento, resultando em uma ambiente estável e seguro (GILADI, 2004). É de se esperar que as UPSs, em sua grande maioria, presente nesses ambientes, sigam a mesma tendência. É dentro desse contexto que hoje se encontra o estado da arte em gerenciamento de UPSs.

Podemos ressaltar algumas características importantes no ambiente de gerenciamento de redes de computadores. A primeira a considerar é o fator segurança. Os ambientes de rede foram enormemente beneficiados com os avanços nas pesquisas relacionadas com confiabilidade nesses sistemas. No entanto, falhas devem ser encaradas como eventos possíveis de ocorrer e, portanto, técnicas de tolerância a falhas devem ser incorporadas no projeto desses sistemas. Uma outra característica salutar é a compatibilidade com as ferramentas já existentes (HONG, 2001). Imagine um administrador de rede ter

numerosos sistemas de gerenciamento, cada um em uma plataforma diferente, tendo que ter treinamento e manutenção para cada um deles. Certamente esse não seria um bom sistema de gerenciamento. Esta característica de compatibilidade não envolve somente os meios físicos de comunicação, mas também os sistemas operacionais sobre os quais os aplicativos de gerenciamento irão funcionar.

Para isso, foi pensado um sistema que utilizasse um protocolo comum de comunicação e gerenciamento que atendesse a uma ampla gama de peculiaridades, e que fosse um padrão em gerenciamento de plataformas. O protocolo escolhido foi o SNMP (Simple Network Management Protocol), utilizado em plataformas gerenciáveis como equipamentos de rede, HUBs, roteadores, bridges e outros (CHATZIMISIOS, 2004).

Este protocolo não apresenta compatibilidade somente nos meios de comunicação. Vários equipamentos de rede disponibilizam ferramentas completas de gerenciamento SNMP, onde as UPSs compatíveis com esse protocolo podem ser adicionadas à paleta de gerenciamento e compor mais um item gerenciável. Através desse protocolo, dependendo da implementação de cada fabricante, podemos monitorar uma ampla gama de dados importantes, bem como executar comandos nas UPSs remotas, como por exemplo comandos programados para ligar e desligar a saída da UPS em um determinado dia e hora pré-determinados.

Ao conceber um sistema de gerenciamento para UPSs, devemos considerar outros fatores relevantes, além do protocolo de gerenciamento a ser utilizado, embora a partir deste já se garantam muitas características e funcionalidades. Um dos mais importantes fatores na concepção deste sistema de gerenciamento é o fator topologia da rede de dados. Este dá praticidade e funcionalidade, permitindo ao administrador manipular dados de monitoração e comandos que sejam gerenciáveis. Pode-se incluir também, neste mesmo quesito de concepção de sistemas de gerenciamento de UPSs, a interface de

gerenciamento que é utilizada no sistema, pois através dela conseguimos mais praticidade e rapidez na execução de determinadas tarefas.

Outro fator de bastante relevância é o custo computacional de implementação deste protocolo. Devemos considerar que as UPSs são ambientes do tipo embarcados, onde mais restrições são impostas ao sistema, como poder de processamento e quantidade de memória disponível. Por esse lado (custo computacional da entidade gerenciada), a UPS torna-se ainda mais relevante. É também de fundamental importância que na implementação desse sistema de gerenciamento todas as operações necessárias ao seu uso estejam incorporadas. Como exemplo podemos citar a monitoração do estado das UPSs e a execução de comandos para as mesmas. Ao conceber o sistema de gerenciamento todos estes fatores foram levados em consideração.

Neste capítulo, um estudo das várias topologias foi realizado, onde ressaltamos a presença de fatores que foram fundamentais para atingirmos os objetivos de desempenho e funcionalidade desejados na nesta pesquisa.

## 2.5 – Comparação com soluções comerciais de gerenciamento existentes

Esta seção tem por objetivo de situar quais as atuais soluções de gerenciamento de UPSs existentes atualmente no mercado, e de situar, comparativamente, a solução proposta e implementada neste trabalho.

Para isso, foram coletados dados de três principais fabricantes nacionais de UPSs, através do seu site, e explanamos em forma de um quadro comparativo, conforme descrito no Quadro 2.1, suas principais características.

Quadro 2.1: Quadro comparativo de soluções comerciais em gerenciamento de UPSs

| Características | Sistema proposto | SMS (SNMP Power View) | Engetron (Power Sups Plus) | CP Eletrônica (SNMP View) |
|---|---|---|---|---|
| Compatível com protocolo de gerenciamento SNMP V1 e MIB RFC1628 | SIM | SIM | SIM | SIM |
| Interface gráfica para visualização de número arbitrário de UPSs | SIM | NÃO | NÃO | SIM |
| Compatibilidade com sistema operacional | Windows | Todos que suportem Browser com HTML e JAVA | Windows, Linux e Unix | Windows |
| Fornece dados adicionais à MIB de gerenciamento de UPS RFC 1628 | SIM | NÃO | NÃO | NÃO |
| Suporta múltiplos níveis de gerenciamento através de autenticação remota | SIM | NÃO | NÃO | NÃO |

Conforme podemos ver no quadro comparativo, o sistema aqui proposto apresenta principais vantagens quanto a inserção de dados adicionais de gerenciamento à MIB RFC 1628, permitindo ao usuário a utilização de recursos adicionais ora presentes na UPS. Além disso apresenta vantagem também no recurso de suporte a autenticação remota, permitindo ao usuário obter vários níveis de gerenciamento com árvores de gerenciamento pré-definidas. Como desvantagem podemos citar a compatibilidade com outros sistemas operacionais.

# Capítulo 3

# 3. O Protocolo SNMP

## 3.1 – Histórico

O termo SNMP (Simple Network Management Procotocol) define um protocolo de gerenciamento de redes que atualmente é adotado como padrão de gerenciamento para a Internet. Na realidade, o termo SNMP é usado não só para definir esse protocolo, mas para referir-se a uma coleção de especificações de gerenciamento de rede que, além do próprio protocolo, inclui a definição de especificações de gerenciamento de rede, definição de estrutura de dados e os conceitos associados (STALLINGS, 1999).

O surgimento, bem como a evolução do protocolo SNMP, tem grande relação com o surgimento do conjunto de protocolos TCP/IP. Na Figura 3.1 é apresentado um diagrama de blocos mostrando a relação entre os protocolos. O conjunto de protocolos TCP/IP, atualmente o protocolo mais utilizado para troca de informações entre computadores, e o mesmo em uso na atual rede mundial de computadores (Internet) datam seus primeiros estudos do ano de 1969, quando o Departamento de Defesa dos Estados Unidos, através da ARPA (Advanced Research Projects Agency), iniciou o desenvolvimento da primeira rede de comutação de pacotes, a ARPANET. Esta rede tinha o objetivo de estudar tecnologias relacionadas com o compartilhamento de recursos entre computadores, bem como suprir as necessidades de troca de informações dos usuários do Departamento de Defesa dos Estados Unidos. A rede cresceu rapidamente e passou a ser usada por centenas de computadores, executando os mais diversos tipos de sistemas operacionais.

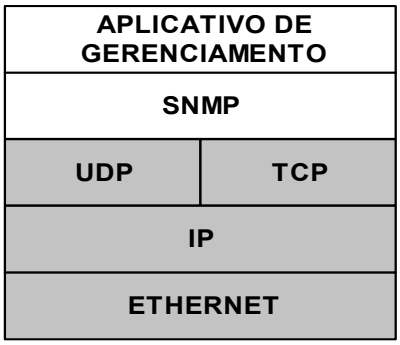


Figura 3.1: o SNMP incluído dentro do conjunto de protocolos TCP/IP.

O TCP/IP nasceu em um ambiente de pesquisadores e programadores, portanto, inicialmente todos os seus usuários utilizavam as próprias máquinas do trabalho cotidiano para fazerem seus estudos e evoluções no protocolo. Como conseqüência, problemas de gerenciamento passavam despercebidos, pois todos os seus usuários eram especialistas e bastante conhecedores dos detalhes de operação e manutenção do seu ambiente de rede. Portanto, pode ser entendido de maneira bastante natural que, no final de 1970, ainda não existia um protocolo de gerenciamento para TCP/IP.

O único protocolo usado para gerenciamento era o ICMP (Internet Control Message Protocol). O ICMP permitia uma troca de mensagens de tamanho de dados variável entre dois dispositivos de rede, sendo usado assim para diagnosticar a presença de computadores na rede, bem como verificação de taxas de transmissão de dados, e tempo de latência na troca de mensagens entre os computadores, embora de maneira bastante superficial (LIU, 2003). O ICMP hoje continua presente na maioria dos protocolos que utilizam o IP (Internet Protocol). Uma das aplicações mais típicas do ICMP é o comando PING, onde podemos verificar desde endereços físicos até endereços de rede. No PING, como mostrado na Figura 3.2, um pacote de dados é enviado através de um ponto de rede para um destinatário, o qual deve devolver o mesmo pacote de dados, sendo assim possível observar características importantes como o tempo de latência na transmissão dessas mensagens.

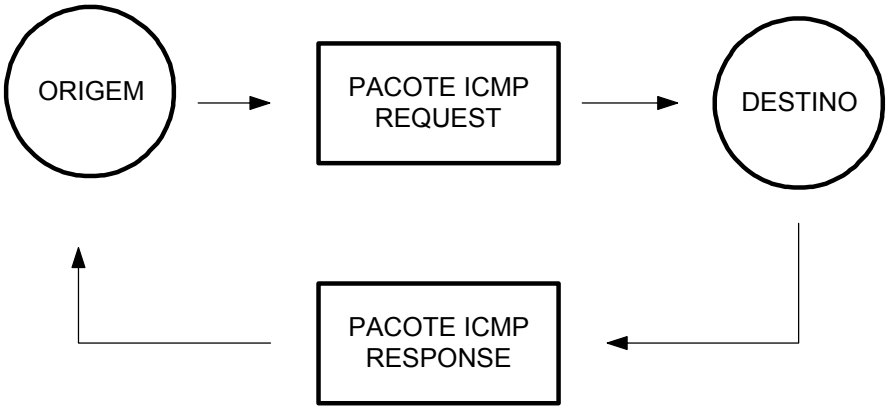


Figura 3.2: o funcionamento do PING.

Somente nos começo dos anos 80, as pesquisas sobre ferramentas de gerenciamento mais poderosas no ambiente TCP/IP foram iniciadas, porque até então, com algumas soluções complementares, o conjunto de ferramentas do ICMP era capaz de resolver a maioria dos problemas existentes. Como o número de computadores aumentou vertiginosamente com o uso da Internet, o problema da necessidade de gerenciamento tornou-se ainda maior. Isso se deveu, sobretudo, ao crescimento muito rápido do número de sub-redes, o que exigia um sistema de gerenciamento realmente capaz de solucionar os problemas já existentes, e os que ainda estariam por vir. Todos os grupos responsáveis por gerenciamento de seções da Internet usavam diferentes ferramentas e políticas de gerenciamento. No fim da década de 80, existiam vários grupos independentes de desenvolvedores pesquisando modelos para gerenciamento de redes que pudessem ser usados para padronizar o conceito de como a Internet deveria ser gerenciada.

O primeiro modelo que surgiu foi o HEMS (High Level Entity Management System). Embora tenha provado ser um bem aceito em redes experimentais, este protocolo nunca viu seu uso e distribuição ativa na Internet. Hoje seu uso é limitado a redes experimentais e está definido nas RFCs 1021, 1022, 1024 e 1076 (PARTRIDGE, 1987).

Em 1987, outro modelo foi proposto pelo grupo OSI (Open System

Interconnection) da ISO (International Standards Organization). O modelo proposto pelo OSI usava o CMIP (Common Management Information Protocol) como estrutura de gerenciamento de rede para gerenciar a Internet. O CMIP já era usado internamente nas redes OSI, e a proposta envolvia a criação do CMOT (CMIP over TCP/IP) como o protocolo de gerenciamento para ser usado na Internet. Por várias razões, o CMOT não obteve sucesso para ser usado como protocolo de gerenciamento, e já em 1989 era pouco usado (WARRIER, 1990).

Em março de 1987, um terceiro grupo de desenvolvedores de protocolos de rede iniciou um trabalho em SGMP (Simple Gateway Monitoring Protocol). Este protocolo tinha um projeto e implementação muito simples, tendo como objetivo gerenciamento de gateways. Em agosto do mesmo ano, o SGMP foi portado para diferentes plataformas de operação. O seu uso cresceu por parte de outros grupos e fabricantes na Internet. Em novembro de 1987 o projeto básico foi apresentado na RFC 1028 (WARRIER, 1987).

Em fevereiro de 1988, o então IAB (Internet Activities Board) convocou um comitê entre seus consultores para determinar qual desses três protocolos de gerenciamento de rede deveriam ser usados na Internet. A decisão tomada neste comitê era que o CMOT, embora não estivesse pronto para distribuição, era a melhor escolha para um protocolo de gerenciamento de rede. O SGMP, devido a sua grande aceitação e ampla distribuição na comunidade da Internet, foi escolhido como protocolo temporário para gerenciamento da Internet, que eventualmente seria substituído pelo CMOT. O HEMS, devido a sua falta de aceitação pela comunidade Internet foi descartado como protocolo de gerenciamento a ser usado na Internet.

Para permitir uma transição tranquila dos sistemas SGMP para CMOT, foi definida uma estrutura comum de gerenciamento que deveria ser usada pelos dois protocolos. Esta estrutura foi chamada de SNMP (Simple Network Management Protocol), executada por um grupo de trabalho

comandado por Marshall T. Rose. Em agosto de 1988 a Internet-Standard Network Management Framework foi criada com o objetivo de construir uma primeira definição da estrutura de gerenciamento da Internet. Em abril de 1989, o SNMP foi promovido pelo IAB ao nível de "recomendado" para gerenciamento de redes TCP/IP, conforme descrito na RFC 1098. Nesta época existiam vários problemas e incompatibilidades crescendo entre CMOT e SNMP em muitos detalhes de gerenciamento de rede. O SNMP, nesta época, já era estável e amplamente aceito como padrão, diferentemente do CMOT.

Em junho de 1989, a IAB convocou uma reunião entre seus consultores novamente, mas desta vez descartou a idéia de uma estrutura de gerenciamento comum, e permitiu que os pesquisadores envolvidos com CMOT e SNMP desenvolvessem suas pesquisas independentemente. Os grupos de SNMP começaram a trabalhar em sua padronização em agosto de 1989 e no início de dezembro alcançaram um consenso, sendo esse protocolo recomendado como padrão em maio de 1990. Este acontecimento cimentou firmemente o SNMP como protocolo de gerenciamento de redes recomendado para uso na Internet e em redes TCP/IP. Em março de 1991, documentos definindo o formato da MIB (Mangement Information Base) e Traps (alarmes assíncronos) eram publicados nas RFCs 1212 e 1215. A definição de MIB foi revisada, publicada na RFC 1213, para criar o que até hoje é conhecido como SNMP versão 1 (ROSE, 1991).

## 3.2 - Conceitos do Protocolo SNMP

O protocolo SNMP (Simple Network Management Protocol) é definido como um conjunto de regras de troca de informações entre dispositivos com o objetivo de gerenciamento. Uma coleção de estações de gerenciamento executa aplicações que, através desta troca de informações, monitoram e controlam os elementos de rede. Dentre estes se encontram roteadores, switchs, UPSs e outros dispositivos pertencentes a uma rede (CASE, 1990).

Como mostrado na Figura 3.3, o modelo de gerenciamento de rede utilizado inclui os seguintes elementos chave: gerente, agente, base de informação de gerenciamento e protocolo de gerenciamento de rede.

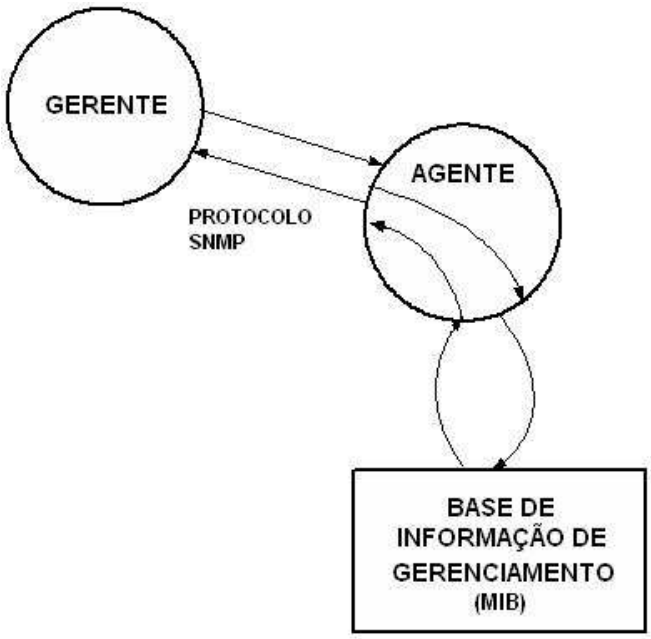


Figura 3.3: elementos em um sistema de gerenciamento SNMP.

O gerente, ou estação de gerenciamento, tem por objetivo realizar pedidos de informações através da interface homem-máquina de gerenciamento, ou efetuar modificações na base de dados do gerenciamento; ou seja, tem a função de ser um cliente do agente SNMP. Por esse motivo muitas vezes é citado como cliente em muitas literaturas (STALLINGS, 1999).

O agente por sua vez tem a missão de responder aos pedidos de requisições do gerente, sendo por esta razão muitas vezes citado como servidor SNMP. Na ocorrência de alterações pré-definidas de dados na base de informação, o agente envia pacotes assíncronos (não solicitados) ao gerente do sistema, que tem o propósito de sinalizar a ocorrência de alarmes no agente.

Cada agente presente na rede dispõe de vários recursos a serem gerenciados. Podemos representar esses recursos como objetos, e a coleção desses objetos é denominada de MIB (Management Information Base). A função da MIB é representar um conjunto de objetos, presentes no agente, conhecidos e acessados pelo gerente. Estes objetos são padronizados através do sistema em uma classe particular (por exemplo, um conjunto de objetos que são usados num mesmo tipo de dispositivo). Uma estação de gerenciamento executa uma função de monitoração para adquirir os valores de objetos da MIB. Além disso, uma estação de gerenciamento pode estabelecer uma ação para ser realizada em um agente, através da mudança de algum objeto dessa base de dados.

O protocolo SNMP possibilita a localização e a correção de problemas em uma rede TCP/IP. O princípio de funcionamento do protocolo é baseado no paradigma busca – armazenamento. Os gerentes invocam agentes em computadores de uma rede, usualmente em uma estação de trabalho, e através de mensagens SNMP especificam se o agente deve ler ou escrever valores em uma variável da MIB. Devido o protocolo não incluir outras operações, todos os comandos devem ser realizados através do paradigma busca - armazenamento.

O protocolo SNMP define a sintaxe e valor de suas mensagens como ASN.1 (Abstract Syntax Notation One). Como na maioria dos protocolos usados nas redes TCP/IP, as mensagens SNMP não têm tamanho fixo e não podem ser definidas como estruturas fixas.

Um servidor SNMP (Agente) deve aceitar um pedido de requisição, executar a operação especificada e retornar uma reposta. Na Figura 3.4, adaptada de (COMER, 1994), é ilustrado o fluxo de uma mensagem através de um agente SNMP.

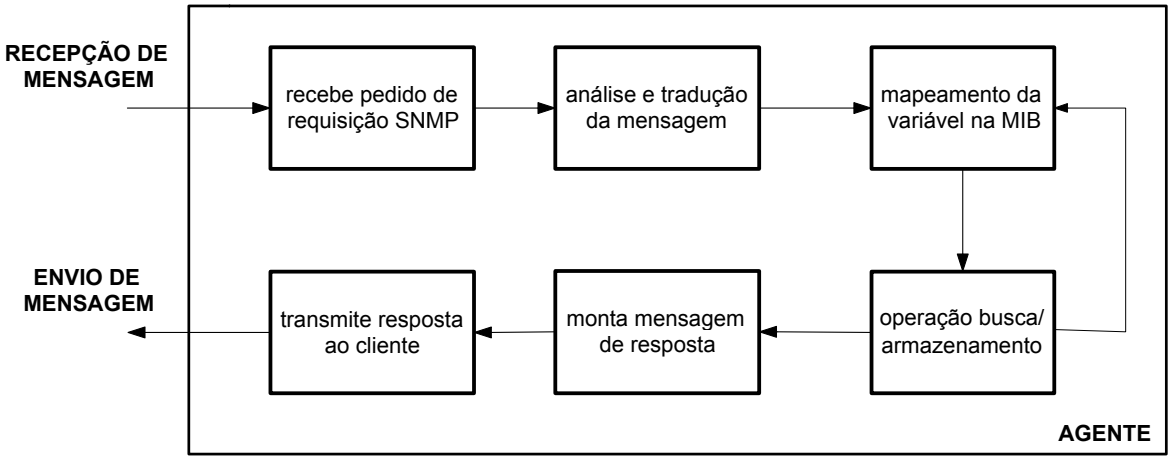


Figura 3.4: fluxo de uma mensagem SNMP em um agente. Adaptada de (COMER, 1994).

Como mostrado na Figura 3.4, o agente primeiro analisa e decodifica a mensagem SNMP. Em seguida ele localiza a variável especificada na MIB e executa a operação de busca ou armazenamento conforme o tipo da mensagem. Para operações de busca, a área de dados é substituída na mensagem SNMP pelo valor da variável que está sendo buscada. Se a mensagem tem múltiplas variáveis, o terceiro e quarto passos são repetidos para cada uma dessas variáveis. Finalmente, uma vez que todas as operações foram realizadas, o agente monta a mensagem de resposta e retorna a mensagem para o solicitante.

Um gerente interage com um agente de acordo com as regras estabelecidas pelo framework de gerenciamento. O Internet Standard Network Management Framework foi definido segundo a filosofia SNMP, que se baseia no axioma de Marshall Rose: “O impacto do gerenciamento de rede adicionado para gerenciar os nós deve ser mínimo, refletindo o menor denominador comum. Cada nó é visto como tendo algumas variáveis. Pela leitura dessas variáveis, o nó é monitorado. Alterando os valores dessas variáveis, o nó é controlado” (BRISA, 1993). O resultado deste axioma é que os agentes SNMP são simples e executam operações elementares, como estabelecer e obter

valores das variáveis. O programa que analisa, manipula, combina ou aplica algum algoritmo sobre os dados, deve residir no gerente.

Através de três classes de comandos, o protocolo SNMP possibilita a troca de informações para a execução do gerenciamento de qualquer agente em uma rede TCP/IP. São estes os conjuntos:

- get: habilita o gerente a adquirir valores de objetos no agente;
- set: habilita o gerente a mudar os valores de objetos no agente;
- trap: habilita o agente a notificar o gerente de eventos importantes.

O SNMP foi projetado para ser um protocolo que atua em nível de aplicação, como parte do conjunto de protocolos TCP/IP. Ele foi planejado para operar sobre o protocolo User Datagram Protocol (UDP). Para uma estação de gerenciamento independente, o gerente pode acessar a estrutura das informações de gerenciamento presentes na estação e fornecer uma interface para gerenciar a rede. Os processos gerentes alcançam o gerenciamento de rede pelo uso do SNMP, o qual é implementado utilizando os datagramas UDP, IP e demais protocolos dependentes (por exemplo Ethernet, FDDI e X.25).

Quando uma ação de gerenciamento deve ser efetuada por um gerente SNMP, três tipos de mensagens são emitidas pelo gerente: Get-Request, Get-NextRequest e Set-Request. A primeira consiste de duas variações da função Get. Estas três mensagens são respondidas pelo agente na forma de uma mensagem Get-Response, a qual é passada para uma aplicação de gerenciamento. Adicionalmente, um agente pode emitir uma mensagem de Trap em resposta a um evento que afeta a MIB e os recursos por ela gerenciados.

O fato do protocolo SNMP ser projetado para trabalhar com mensagens UDP, o qual é um protocolo sem conexão, torna o SNMP por si só

sem conexão. Assim, as conexões entre uma estação de gerenciamento e seus agentes não são mantidas continuamente. Ao invés disso, cada troca é uma operação separada, entre uma estação de gerenciamento e um agente (STALLINGS, 1999).

O protocolo SNMP minimiza explicitamente o número e a complexidade das funções de gerenciamento realizadas pelos agentes de gerenciamento. A meta é ser atrativo em pelo menos quatro itens (RFC 1157):

- O custo de desenvolvimento do software do agente necessário para dar suporte ao protocolo deve ser reduzido;
- O grau das funções de gerenciamento que são suportadas remotamente deve ser maior, admitindo assim o uso dos recursos da Internet nas tarefas de gerenciamento.
- O grau das funções de gerenciamento que são suportadas remotamente deve ser aumentado, impondo assim o mínimo possível de restrições na forma e sofisticação das ferramentas de gerenciamento;
- Os conjuntos simplificados de funções de gerenciamento são entendidos facilmente e usados por desenvolvedores de ferramentas de gerenciamento de rede.

Uma segunda meta do protocolo é que o paradigma funcional de monitoração e controle seja suficientemente extensível para acomodar aspectos adicionais, possivelmente não antecipados, de operação e gerenciamento de rede.

Uma terceira meta é que a arquitetura seja, tanto quanto possível, independente da arquitetura e mecanismos de máquinas e gateways particulares.

## 3.3 – As Informações de Gerenciamento

Como já definido anteriormente, o gerenciamento SNMP nada mais é do que operações de busca e armazenamento em uma estrutura de dados, que representa as informações de gerenciamento. Portanto, essas informações de gerenciamento são a base da interoperabilidade do protocolo. Para isso, o SNMP define muito bem a estrutura e o formato dessas informações. O SNMP utiliza três linguagens para transportar essas informações de gerenciamento, que serão descritas em seguida (MURRAY, 1998).

A SMI (Structure Management Information) especifica o formato usado para definir objetos gerenciados que podem ser acessados via protocolo SNMP (SCHONWALDER, 2005). Estes dados são representados de forma bem estruturada. É através da SMI que são definidas regras para existência e criação de objetos gerenciados, bem como sua estrutura dentro da MIB. A SMI existe em duas versões: SMIv1 e SMIv2. A SMIv1 é descrita nas RFCs 1155, 1212 e 1215, enquanto a SMIv2 é descrita nas RFC 1442, 1443, e 1444, e é compatível com SMIv1 (A única exceção é o tipo Counter64 definido na SMIv2).

A notação descrita na especificação ASN.1 (Abstract Syntax Notation One) é usada para definir o formato das mensagens SNMP e dos objetos gerenciados pela MIB. A ASN.1 é uma linguagem criada pela ISO para descrição de dados e para ser independente de qualquer implementação (MURRAY, 1998). A sua notação é um padrão internacional, independente de fabricantes e de plataforma de implementação, bem como independente também das estruturas de dados das linguagens utilizadas, sendo uma forma de descrição de alto nível de abstração (LARMOUTH, 1999). Esta linguagem permite que qualquer dado seja representado em uma forma textual, sendo usada como um modelo. A descrição completa da linguagem ASN.1 é originalmente descrita na especificação CCITT X.208. A SMI limita o uso a somente um subconjunto de especificações de ASN.1 para ser usado em

SNMP. A sintaxe será melhor descrita quando descrevermos a estrutura da MIB.

A terceira linguagem usada em SNMP, a BER (Basic Encoding Rules), se refere às regras de codificação das mensagens SNMP. A BER é usada para codificar as mensagens SNMP para serem transmitidas através de uma rede TCP/IP. Podemos relacionar a ASN.1 e a BER ao código fonte de um programa e ao seu código de máquina, respectivamente (MURRAY, 1998). A ASN.1 é uma notação legível, que quando traduzida, usando as regras da BER, são passíveis de serem transmitidas através de uma rede. Antes de um nó de rede transmitir uma mensagem SNMP, ele deve converter essa mensagem em um formato de representação binária, que conterá o conteúdo da informação de gerenciamento a ser transmitida. Esta conversão é feita usando as regras da BER. As regras de codificação serão melhor definidas quando descrevermos as mensagens SNMP. Na Figura 3.5 é mostrado o relacionamento entre as três linguagens SNMP.

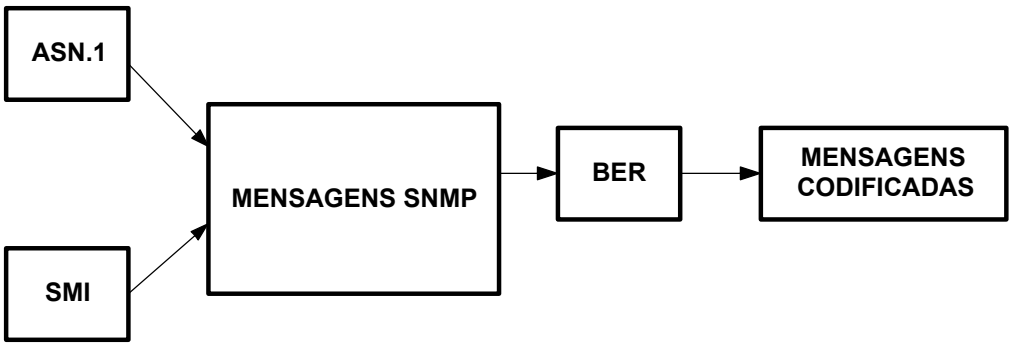


Figura 3.5: relação entre as linguagens utilizadas em SNMP e mensagens de gerenciamento utilizadas neste protocolo.

## 3.4 – A Base de informação de Gerenciamento: MIB

A MIB (Management Information Base) representa as informações que um agente ou um gerente SNMP pode tratar. Esse formato de representação define um conjunto de variáveis que um agente pode acessar. A MIB é caracterizada principalmente pela sua estrutura, na forma de organização dos itens e em como identificá-los (BRISA, 1993). Como

descrevemos anteriormente, essa notação é definida pela SMI. Dentro desse contexto, os recursos a serem gerenciados são representados, e são frequentemente chamados de objetos gerenciados. Cada objeto gerenciado pode ser visto como recurso que um agente permite o seu gerenciamento. Portanto, cada sistema, uma estação de trabalho, um servidor de dados ou um switch, mantém uma base de dados particular relacionada aos seus recursos de gerenciamento disponíveis.

A MIB, no intuito de servir às necessidades de um sistema de gerenciamento, deve ter pelo menos dois objetivos (STALLINGS, 1999):

- A representação de um recurso de gerenciamento particular de cada sistema através de um objeto gerenciado, que deve ser o mesmo para cada sistema. Esta necessidade refere-se à unicidade da MIB para um determinado tipo de aplicação;
- Uma estrutura comum de representação deve ser usada para manter a interoperabilidade. Esta necessidade refere-se à definição de estrutura da MIB.

Devido à estrutura definida na SMI, a MIB só pode armazenar variáveis do tipo escalar ou variáveis vetoriais de uma dimensão. A MIB não suporta criação de dados mais complexos, visando manter a simplicidade implementada em todo o protocolo SNMP. Essa filosofia é totalmente contrária à usada no gerenciamento OSI, a qual provê estrutura e dados complexos (STALLINGS, 1999). A simplicidade presente em SNMP não impede que o resultado de alguma computação seja armazenado internamente nos dados de gerenciamento do sistema. Em alguns casos podemos computar dados nos valores da MIB. Como exemplo podemos citar o armazenamento de uma variável que represente o tempo que um processo esteja ativo. Muitos sistemas simplesmente gravam o momento em que o sistema foi iniciado, e computam o tempo que ele tem estado operacional pela subtração do horário corrente do tempo que o sistema foi iniciado. Assim, um gerente SNMP pode simular a

variável MIB que contém o tempo desde que o sistema foi iniciado. Isto permite que o resultado da computação iniciada seja armazenado em uma variável da MIB. Podemos resumir como em (COMER, 1994):

> *A MIB define conceitos de variáveis que nem sempre correspondem diretamente à estrutura de dados que o nó usa. Sistemas de Gerenciamento SNMP podem executar computação para simular alguns conceitos de variáveis, mas o objeto remoto permanecerá inconsciente da computação realizada.*

Ou seja, a única computação presente no nó deve ser a busca e o armazenamento das informações. Qualquer outro processamento de dados necessário deverá ser feito no gerente SNMP.

### 3.4.1 – A Estrutura da MIB

Todos os objetos gerenciados em SNMP são estruturados em forma de uma árvore hierárquica. Os ramos dessa árvore é que são realmente os objetos gerenciados, sendo cada um deles um recurso de gerenciamento presente no dispositivo. Associado com cada tipo de objeto em uma MIB está um identificador de tipo ASN.1 do tipo OBJECT IDENTIFIER. O identificador serve para nomear o objeto. Como exposto na Figura 3.6, no topo desta estrutura hierárquica existem três nós: ISO, ITU e ISO/ITU. A raiz desta hierarquia não tem nome, sendo especificados somente os filhos deste nó. A ISO alocou uma sub-árvore desses nós para o uso de outras organizações de padrão nacional ou internacional, e o U.S. National Institute for Standards and Technology (NIST) alocou uma sub-árvore para o Departamento de Defesa Norte Americano. O IAB também solicitou ao Departamento de Defesa Norte Americano a alocação de uma sub-árvore. Abiaxo do nó Internet foram criados quatro sub-árvores: diretório, gerenciamento, experimental e privativo (COMER, 1994).

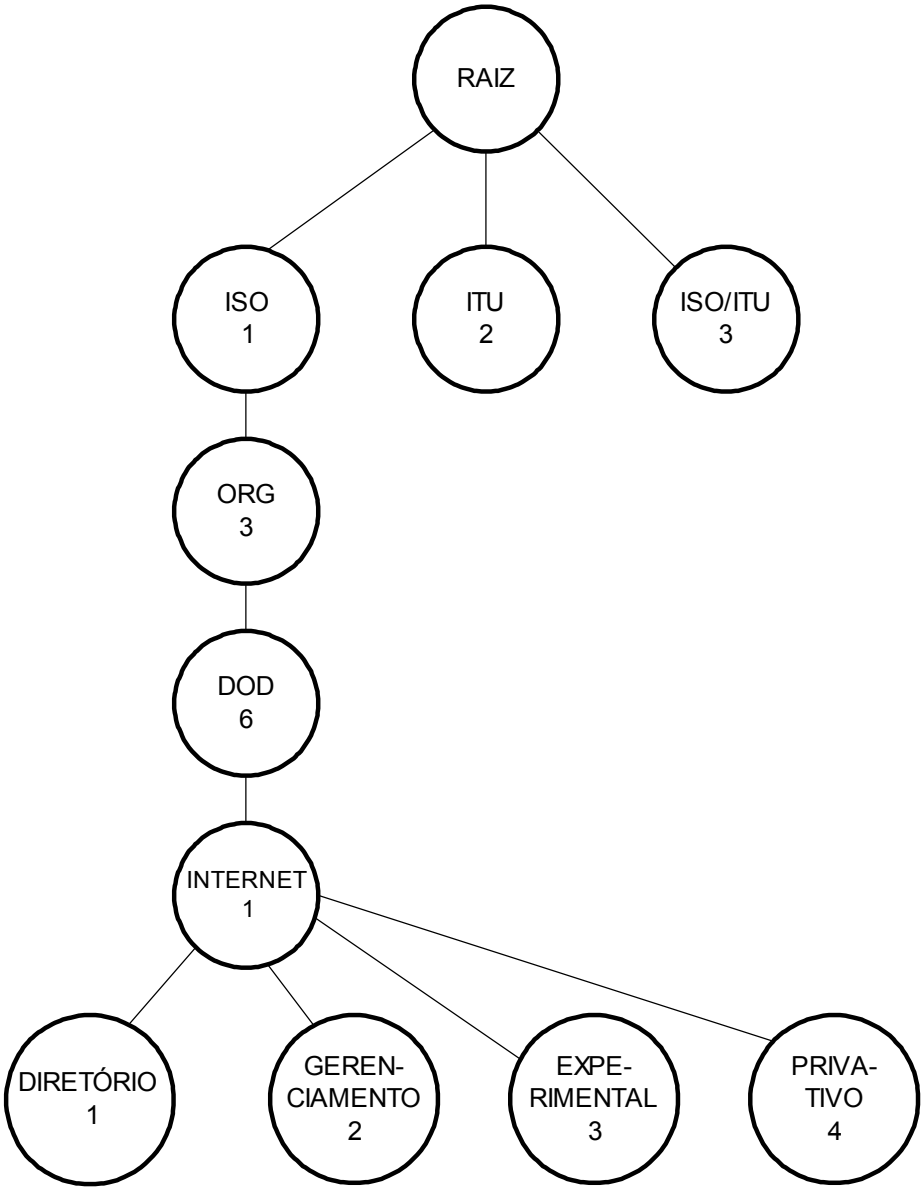


Figura 3.6: estrutura hierárquica da MIB.

Os nós definidos abaixo do nó Internet têm as seguintes funções:

- Diretório: reservado para o uso futuro com OSI (X.500);
- Gerenciamento: usado para os objetos definidos nos documentos aprovados pelo IAB;
- Experimental: usado para identificar objetos usados em

experimentos da Internet;

- Privativo: usado para identificar objetos unilateralmente, definidos apenas no gerente ou no agente.

O nó de gerenciamento contém as definições para gerenciamento que foram aprovadas pela IAB. Atualmente duas versões foram desenvolvidas: mib-1 e mib-2. A segunda MIB é uma extensão da primeira. Ambas podem ser usadas com o mesmo objeto identificador na sub-árvore desde que somente uma das MIBs esteja presente na configuração.

Objetos adicionais podem ser definidos para a MIB de três maneiras (STALLINGS, 1999):

1. A estrutura mib-2 pode ser expandida ou substituída por uma revisão completamente nova (presumivelmente mib-3);
2. Uma MIB experimental pode ser construída para uma aplicação particular. Tais objetos podem posteriormente serem movidos para o nó Gerenciamento.
3. Extensões privadas podem ser adicionadas à sub-árvore *private*. Uma delas é documentada na RFC 1227, em MUX MIB.

O nó Privativo tem somente um nó, o *Enterprises*. Este nó é usado para permitir aos fabricantes aumentarem o gerenciamento de dispositivos e dividir as informações com outros usuários, e fabricantes que necessitam partilhar informações com outros sistemas.

Abaixo do nó gerenciamento foi criado o nó MIB-II, que agrupa as outras categorias de gerenciamento, como mostrado na Figura 3.7.

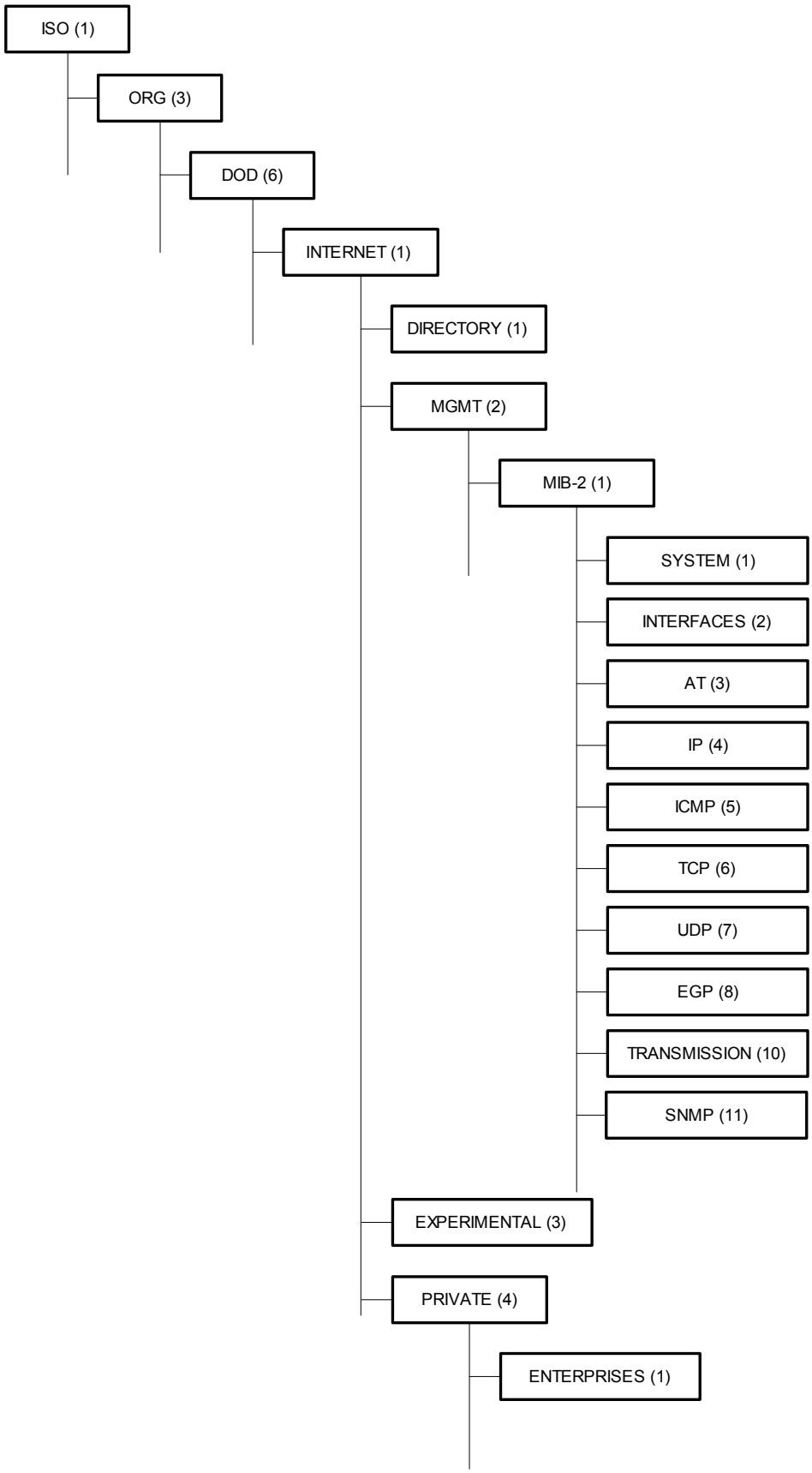


Figura 3.7: objetos da MIB-II.

Para representar um objeto gerenciado usamos uma representação de nomes que carrega o número de nó na estrutura de gerenciamento da SMI. Para identificarmos o nó IP, que na árvore tem número 4, devemos usar como representação textual. Por exemplo:

*iso.org.dod.internet.mgmt.mib.ip*

Se quisermos representar uma variável, de nome *ipInReceives*, que está abaixo do nó IP, deveremos textualmente representá-la assim:

*iso.org.dod.internet.mgmt.mib.ip.ipInReceives*

Finalmente, se desejarmos usar a representação numérica, utiliza-se os números atribuídos a cada nó da rede. Supondo que *ipInReceives* no exemplo acima possui ID = 3, tem-se, numericamente:

1 . 3 . 6 . 1 . 2 . 1 . 4 . 3

A esta representação de um grupo ou de uma variável na MIB, chamamos de OBJETO IDENTIFICADOR ou simplesmente de OID da variável ou do grupo.

### 3.4.2 – A Sintaxe da MIB

Cada objeto dentro da MIB SNMP é definido de maneira formal. A definição especifica o tipo de dado deste objeto, seu direito de acesso e limites de valores, bem como sua relação com outros objetos dentro da MIB. Esta definição formal é feita na ASN.1.

Cada objeto gerenciado é definido em um módulo da MIB usando a macro OBJECT-TYPE como definido em SNMPV1. Uma macro ASN.1 é muito similar a uma classe que descreve a forma e atributos do objeto gerenciado. O objeto pode ser escalar, quando tem somente um valor; ou pode

ser do tipo vetorial, e ter nenhum ou mais valores. Uma definição típica de uma variável escalar da MIB é o objeto sysContact definido na MIB-II, como descrito na RFC 1157:

*SysContact OBJECT-TYPE*
*SYNTAX* *DisplayString* *(SIZE (0..255))*
*ACCESS* *read-write*
*STATUS* *mandatory*
*DESCRIPTION*
"A identificação textual da pessoa de contato para o nó gerenciado bem como a maneira para contatar esta pessoa".

O nome " *sysContact* " é o nome ou objeto descritor da variável e o *OBJECT-TYPE* é a macro. Os atributos do objeto são descritos por quatro cláusulas definidas pelo *OBJETC TYPE* e definidas na RFC 1155. *SYNTAX* indica o tipo de dado do objeto. *ACCESS* descreve o direito de acesso do objeto. Os modos de acesso definidos em SNMPv1 são *read-only*, *read-write*, *write-only*, ou *not-acessible*.

O *STATUS* descreve o estado atual de validade do objeto. Os possíveis valores são:

*Mandatory:* Objetos que são do tipo obrigatório e que devem ser implementados por um agente;
*Optional:* Objetos que são do tipo opcional, assim podem ou não ser implementados por um agente. (Na comunidade SNMP se desencoraja o uso do tipo "*Optional* " no valor STATUS);
*Deprecated:* Objetos deste tipo estão em transição para o STATUS de obsoleto. Eles foram substituídos por novos objetos, mas os dados continuam contidos em um objeto do tipo " *Deprecated* " podendo muitas vezes serem ainda válidos;
Obsolete: Um objeto obsoleto não é mais suportado pela MIB e não

deve ser implementado por um agente. Não é necessário que um objeto seja substituído por outro objeto antes de ser transformado em obsoleto; Além disso, não é necessário que um objeto seja do tipo “Deprecated“ para que o mesmo venha a ser do tipo “Obsolete“.

A *DESCRIPTION* é uma cadeia de caracteres do tipo alfanuméricos que descreve o propósito do objeto e como o mesmo deve ser usado. O intervalo de valores que podem ser associados com o objeto deve ser descrito, a menos que não seja significante para o mesmo objeto na MIB. A documentação para os objetos descritos na MIB pode ser somente encontrada no campo DESCRIPTION desses objetos. Para definir esses objetos são usados também tipos de dados independentes de aplicações, que são de uso geral. Dentro da classe universal ASN.1 somente os seguintes tipos de dados são permitidos para serem usados para definir objetos da MIB:

- Integer (UNIVERSAL 2);
- OctetString (UNIVERSAL 4);
- Null (UNIVERSAL 5);
- Object Identifier (UNIVERSAL 6);
- Sequence, Sequence-of (UNIVERSAL 16).

Os quatro primeiros são tipos primitivos, que são os blocos de construção básicos de outros tipos de objetos. O *Object Identifier* é o identificador único de um objeto, consistindo de uma seqüência de inteiros. Esta seqüência, lida da esquerda para a direita, define a localização de um objeto na estrutura da MIB. O último identificador consiste do tipo construtor *sequence* e *sequence-of*. Estes tipos são usados para construção de tabelas.

## 3.5 – As mensagens SNMP

As mensagens em SNMP representam a comunicação entre o agente e o gerente SNMP. Toda fluxo de dados entre dois sistemas usando o protocolo SNMP é realizada no formato dessas mensagens. Em uma rede usando o conjunto de protocolos TCP/IP, esta mensagem SNMP é encapsulada em um datagrama UDP.

As mensagens em SNMP são chamadas de PDU (Protocol DataUnit). Existem três grupos de mensagens em SNMP. As mensagens que têm o objetivo de ler variáveis são as do grupo Get. As mensagens em que o objetivo é de escrever em variáveis são representadas pela PDU Set; e as mensagens que notificam alertas, que são as do grupo Trap. Estas mensagens podem ter diferentes estímulos, como as mensagens em que o gerente SNMP envia para o agente, que são estimuladas pelo gerente a fim de ler dados no agente, como representada na Figura 3.8,

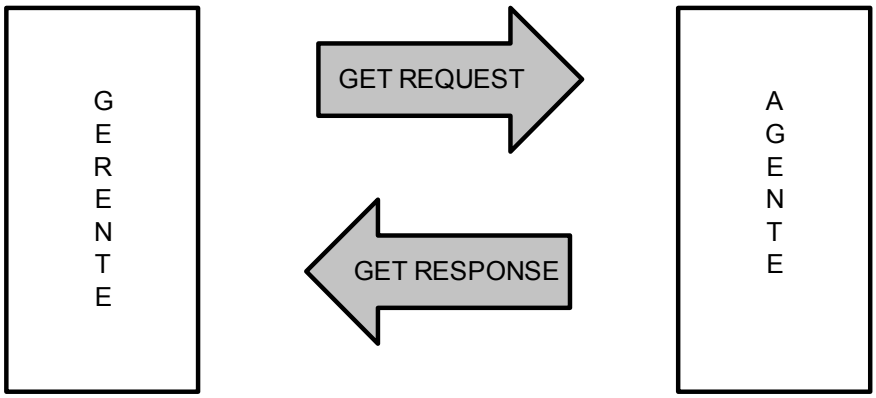


Figura 3.8: fluxo por Estímulo Gerente – Agente (Leitura).

ou as mensagens que o agente envia para o gerente, em resposta a um pedido recebido deste último, como mostrado na Figura 3.9;

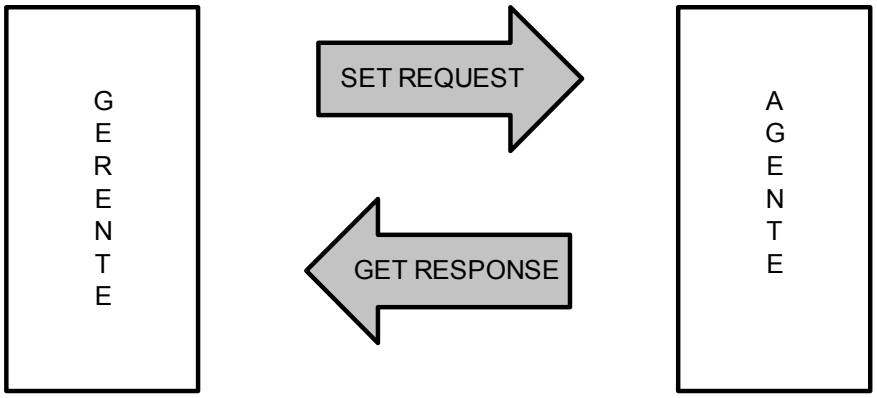


Figura 3.9: fluxo por Estímulo Gerente – Agente (Escrita).

ou ainda as mensagens que o agente envia para o gerente SNMP por um estímulo interno, que são os alertas, os quais são despertados por eventos previamente determinados, como mostrado na Figura 3.10.

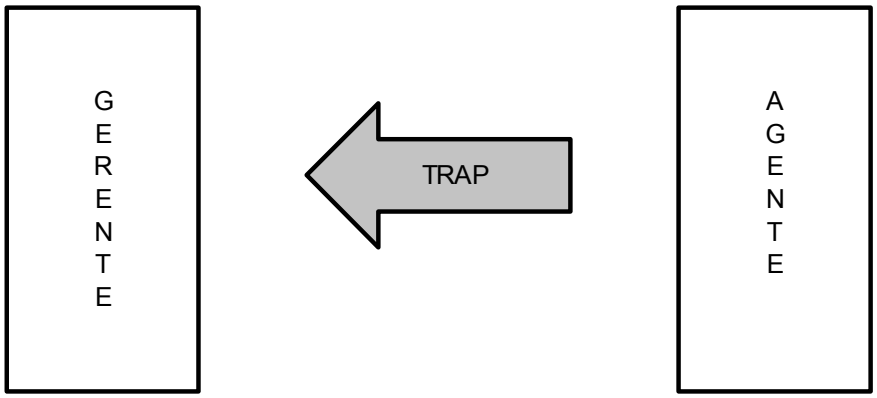


Figura 3.10: fluxo por Estímulo interno do Agente (Alarme).

Os demais detalhes sobre as mensagens SNMP, como sua estrutura geral, bem como a estrutura particular de cada classe de mensagem está descrito no apêndice desta dissertação.

## 3.6 – Aplicações

O protocolo SNMP, desde seu surgimento, sempre foi associado com gerenciamento de redes TCP/IP, em ambientes de redes locais. Atualmente, os fabricantes de equipamentos de rede disponibilizam interfaces para gerenciamento SNMP em praticamente todos os seus produtos, como roteadores, bridges e hubs, entre outros. O protocolo SNMP é, de longe, o protocolo de gerenciamento mais utilizado em corporações e aplicações de gerenciamento de Internet (CHEIKHROUHOU, 2002). No entanto, estas características não limitaram a atuação do SNMP a este conjunto de aplicações. O protocolo se difundiu em muitas outras áreas, como implementações em ambientes industriais, dentre outros.

Um exemplo do uso do protocolo SNMP, que é descrito em (CURY, 2000), consiste da integração com o SIS, um sistema integrado de gerência, o qual foi projetado para gerenciar plantas de redes de telecomunicações. O protocolo SNMP foi usado como interface entre o sistema de gerência e os objetos gerenciados, os quais internamente já dispunham de agentes SNMP implementados. As PDUs implementadas foram as do tipo Get e Trap, e foram usadas para monitoração dos dados da MIB dos objetos gerenciados. O sistema foi implementado em uma rede ATM (Asynchronous Transfer Mode), sendo uma rede de alta velocidade, e teve como aplicação a BH2 (Rede Metropolitana de Alta Velocidade de Belo Horizonte). A MIB utilizada para gerenciamento de redes ATM está descrita na RFC 1695, e possui objetos com informações específicas para o gerenciamento destas redes.

É comum aplicações SNMP serem utilizadas para monitorar o desempenho de redes TCP/IP. Em (APOSTOLOPOULOS, 1995), é proposta uma estrutura de MIB especializada para avaliação de recursos em uma rede TCP/IP. Nesta aplicação, o processo de gerenciamento efetua uma varredura nos objetos gerenciados, armazenando os dados para análise estatística com o objetivo de prever a disponibilidade dos recursos nesta rede. Os dados foram

coletados dos agentes via protocolo SNMP usando as primitivas Get-Request, Get-Response, Set-Request e Trap.

Algumas implementações são também direcionadas a gerenciamento de falhas em redes locais. Como foi proposto em (DUARTE, 2001), um sistema de gerenciamento de falhas para redes locais usando o protocolo SNMP. Nesta aplicação foi utilizado um SNMP replicado, onde através de uma camada adicional de software entre o gerente e o agente, foi possível introduzir tolerância a falhas nos dados da MIB dos agentes falhos. Para isso, um agente replicado no sistema, recebia e enviava dados para o gerente, e por sua vez replicando os dados trocados entre as duas entidades SNMP. Através desta implementação se tornou possível ter acesso aos dados da MIB dos agentes falhos da rede.

A grande aceitação do protocolo SNMP, e a inexistência de protocolos de gerenciamento para aplicações em tempo real embarcadas, motivaram a pesquisa e implementação de SNMP neste tipo de sistemas, como descrito em (LOU, 1999). Nesta implementação, foi utilizado o protocolo SNMP para coletar dados de temperatura em um sistema de tempo real. Foram implementadas duas MIBs, uma para conter os recursos do dispositivo gerenciado e outra para controlar a freqüência de amostragem e envio desses dados para o gerente SNMP. Neste envio de dados, um Get-Response contínuo foi implementando, sendo ativado pela MIB auxiliar do sistema. Esta aplicação mostra a alta aceitação do protocolo em ambientes diferentes de redes locais e Internet.

No intuito de introduzir a estabilidade e contribuir para uma boa definição de um protocolo de gerenciamento para as redes industriais, foi proposto um modelo de gerenciamento para aplicações industriais utilizando SNMP (CARDOSO, 1998). Neste sistema, a rede industrial, composta de sensores analógicos e digitais, se interligava à rede de gerenciamento através de um proxy, sendo neste implementada uma MIB que continha os dados dos

objetos gerenciados na rede industrial. No sistema foi garantido o modelo de informação através da MIB presente no proxy, bem como os modelos administrativos, mantendo uma comunidade de acesso dentro da rede industrial. A noção de simplicidade do SNMP foi mantida pela manutenção das primitivas Get, Set e Trap utilizadas. O trabalho demonstrou a boa aceitabilidade do protocolo em gerenciamento de redes industriais. Em (LEE, 2004), foi apresentado um trabalho de projeto e implementação do protocolo SNMP para gerenciamento de redes industriais, onde a coleta de dados dos sensores e atuadores foram feitas através de um CLP (Controlador Lógico Programável), permitindo uma ampla compatibilidade com ambientes industriais.

Devido a grande aceitação do protocolo SNMP em ambientes de rede, e pelo seu crescimento em quase todas as aplicações de gerenciamento, este protocolo também foi utilizado em (GILADI, 2004), onde um sistema de gerenciamento para automação residencial foi implementado. Nesta aplicação, foi utilizado um gateway, desenvolvido em um computador pessoal, para interligar a rede de comunicação residencial, construída sob uma rede do tipo EIB (European Installation Bus), e a rede de gerenciamento do sistema.

## 3.7 – Paradigmas em SNMP

Desde a definição do protocolo SNMP, entre suas várias aplicações e implementações, vários paradigmas foram criados. A maneira de projetar o sistema de gerenciamento e a experiência obtida com os problemas encontrados em SNMP, criaram algumas boas práticas no uso deste protocolo. Este item do capítulo tem por objetivo relatar alguma dessas particularidades do protocolo.

Em um sistema de gerenciamento, é muito comum que uma estação de gerenciamento seja responsável por gerenciar um número muito grande de objetos gerenciados. Neste caso, pode tornar-se impraticável para a estação

de gerenciamento varrer todos os elementos gerenciados num tempo especificado (CHUN, 2002). A estratégia utilizada é, no momento da inicialização, ou talvez em intervalos não muito freqüentes (como por exemplo uma vez ao dia), a estação de gerenciamento varrer todos os agentes conhecidos para obter informações importantes, como características de interface e talvez algumas estatísticas importantes para o sistema. Para o restante dos objetos a serem gerenciados, estabelece-se uma varredura de dados menor, para não sobrecarregar o gerente. Nessa estratégia, cada agente é responsável por notificar à estação de gerenciamento algum evento previamente determinado através de mensagens SNMP do tipo Trap. Este fato pode levar o gerente a mudar de atitude, aumentando assim a varredura dos dados de um determinado agente, quando este estiver em um estado de alerta (quando uma determinada situação de alarme houver ocorrido), uma vez que a estação de gerenciamento é alertada para esta condição de exceção, mudando a ação a ser tomada com relação àquele objeto gerenciado.

Uma maneira de calcular o tempo de varredura de dados pelo gerente em um nó gerenciado é proposta em (STALLINGS, 1999). Essa métrica é utilizada para efetuar uma estimativa de quantos objetos gerenciados, em uma dada situação, um gerente SNMP pode gerenciar. Para simplificar o problema, vamos considerar que o gerente possa monitorar somente um agente por vez. Esse cálculo é aproximado, pois depende do tempo de resposta da rede, e qual o tempo de processamento que o gerente irá usar para manipular os dados de gerenciamento. A relação proposta é:

$$N \leq \frac{T}{\Delta},$$

em que,

N = Número de agentes possíveis de serem gerenciados

T = Intervalo de busca de dados desejado entre cada agente

Δ = Tempo médio para adquirir um dado do agente.

O tempo médio para adquirir um dado do agente, representado na

equação pelo valor de Δ, depende do tamanho da mensagem solicitada, do tempo de latência da rede, e o tempo que o agente leva para buscar os dados na MIB e montar a mensagem de resposta (LIU, 2003).

Para exemplificar o cálculo citado, vamos imaginar uma aplicação hipotética, onde o intervalo de busca desejado entre cada agente seja de 60 segundos. O tempo médio total para adquirir cada pacote de dados gerenciados será considerado de 12ms. Na Figura 3.11, temos uma visão mais detalhada dos tempos envolvidos em uma aquisição de dados de um objeto gerenciado.

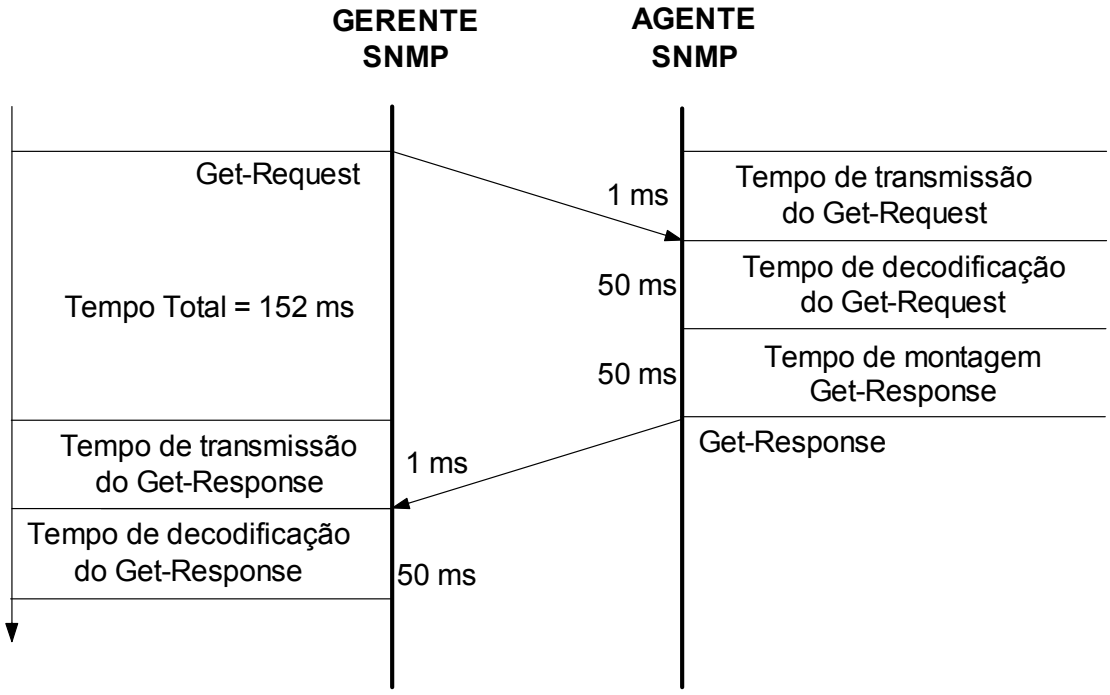


Figura 3.11: fases de aquisição de dados em uma gerente SNMP.

Portanto, considerando os valores dados anteriormente, podemos calcular o número máximo de agentes gerenciáveis dentro das condições estabelecidas como:

$$N \leq \frac{60}{0{,}152} \approx 394 \ .$$

Logo, poderemos gerenciar um número máximo de 394 objetos gerenciáveis.

Esta problemática do tempo de busca dos dados nos objetos gerenciados torna-se bastante importante em sistemas onde existe a necessidade de intensa coleta de dados em seus objetos gerenciados. Para otimizar este tempo e o modo de busca de dados, foi proposto em (CHEIKHROUHOU, 2002), uma implementação de uma camada de software específica para esta função. Esta camada de busca de dados propõe otimizar o tempo e o número de mensagens SNMP transmitidas. Este fato pode ser bastante importante em redes onde um pequeno incremento do trafego de dados seja crucial para o sistema. Nesse trabalho foi concluído que a camada implementada ofereceu melhoras na busca de dados entre os agentes SNMP, mas a limitação do tamanho das mensagens tendeu a aumentar o tráfego, pois para cada Get-Request enviado era necessário um Get-Response, aumentando assim o tráfego de dados na rede.

Outra questão, envolvendo ainda o tráfego de dados na rede, diz respeito às mensagens SNMP Traps. É cada vez mais comum a implementação de traps adicionais por parte dos fabricantes de elementos de rede. Este fato aumenta a quantidade de Traps recebidos pelo gerente SNMP. Nesse caso, quando esse gerente é responsável por um grande número de agentes, esta quantidade de mensagens pode tornar-se bastante grande, aumentando o tráfego de dados, além do processamento necessário para o gerente tratar estas mensagens. Neste sentido, foi proposto em (AUGUSTO, 2001) um discriminador de repasse de eventos em ambientes SNMP. Neste trabalho foi proposta a inclusão de mais uma estação de gerenciamento para tratamentos das mensagens SNMP Traps. Com isto foi possível melhorar o gerenciamento dos alarmes em diferentes gerentes, reduzindo o tráfego em certos pontos da rede além de aliviar o processamento dos gerentes.

## 3.8 – Versões do Protocolo SNMP

Embora o protocolo SNMPv1 tenha sido um protocolo muito bem definido através da sua estrutura na SMI (Structure of Management Information, RFC 1155), e apresente uma simplicidade e atomicidade em suas operações, mostrando-se, com isso, de fácil implementação, algumas falhas foram detectadas. A principal delas refere-se à segurança e política de acesso. A simples implementação do recurso das "comunidades" não forneceu o suporte necessário à segurança desejada nas grandes redes e corporações (KIM, 2002).

Após uma análise do protocolo SNMPv1, constatou-se a necessidade de maior segurança no protocolo, e de algumas outras funcionalidades de busca e armazenamento de dados. Com isso ficou bastante clara a necessidade de uma revisão na especificação de gerenciamentos em SNMP. Como descrito em (STALLINGS, 1999), em julho de 1992, foi proposto um conjunto de RFCs acerca de uma nova versão de SNMP, conhecida como "SNMP Seguro". Nesta versão, somente eram tratadas as questões de segurança, não contemplando outras necessidades de funcionalidade já presentes em SNMP. Portanto, também em julho de 1992 foi proposta, em forma de documentos individuais, uma versão do protocolo, chamada SMP, que trazia as melhorias necessárias de funcionalidade e visava incorporar as questões de segurança tratadas na versão do "SNMP Seguro". Por sua vez, em outubro de 1992, foi iniciado oficialmente o trabalho em SNMPv2, onde um grupo iria trabalhar com segurança, e outro com os aspectos de funcionalidade. O trabalho conjunto dos grupos terminou em março de 1993. Após alguns anos de experiência, o IETF pediu uma revisão na especificação do protocolo SNMPv2. Nesta revisão o quesito segurança foi retirado, permanecendo ainda como única opção para segurança o campo "comunidade", presente em SNMPv1. A especificação de funcionalidade foi aprovada com modestas mudanças, e a versão ficou conhecida como SNMPv2C ou "Community-based SNMPv2".

A não abordagem dos aspectos de segurança em SNMPv2 desencorajou fabricantes e vendedores a migrarem seus sistemas, permanecendo assim o protocolo SNMPv1 como um dos mais usados. As mudanças funcionais presentes em SNMPv2 restringiram-se a adicionar algumas características na SMI, permitindo definir alguns tipos diferentes de variáveis. Outra mudança importante implementada em SNMPv2 foi a comunicação entre gerentes através da mensagem InformRequest. Uma outra nova mensagem também adicionada foi a PDU GetBulkRequest, permitindo o agente trocar longos blocos de dados de maneira mais eficiente (BIVENS, 2004).

Devido à necessidade de mais segurança ainda presente no desenvolvimento do protocolo SNMPv2, após a revisão pelo IETF que resultou na retirada dos quesitos de segurança, dois grupos de trabalho propuseram duas versões, o protocolo SNMPv2u e SNMPv2*. Estas duas vertentes serviram como ponto de entrada para um novo grupo de estudos do IETF em SNMPv3, que em janeiro de 1998 produziram um conjunto de propostas publicadas nas RFCs 2271-2275. Este conjunto de documentos define uma estrutura de informação para incorporar os quesitos de segurança às outras versões do protocolo, SNMPv1 e SNMPv2. Uma nova especificação de SNMPv3, até onde temos conhecimento, ainda hoje é esperada, pois nas RFCs 2271-2275 são descritas características de segurança, mas eles não definem nenhum formato ou estrutura de novas mensagens (STALLINGS, 1999).

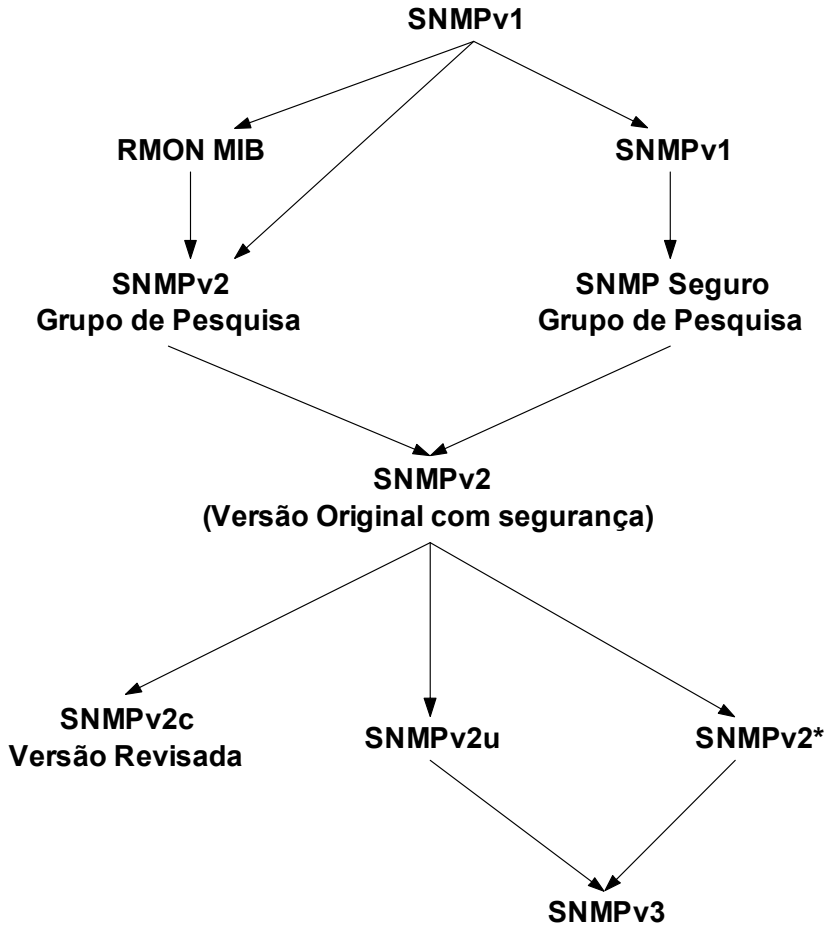


Figura 3.12: versões do protocolo SNMP.

Na Figura 3.12 são apontados alguns marcos na história do surgimento e implantação do protocolo SNMP, como proposto em (STALLINGS, 1999). Logo após o surgimento do protocolo SNMP, foi observada uma necessidade de uma melhor estrutura de gerenciamento para grandes redes de computadores, que foi implementada no conjunto de especificações RMON, a qual, por sua vez, estabelecia conjuntos de objetos gerenciados, facilitando assim o gerenciamento de redes. Estas especificações encontram-se definidas na RFC 1757. Todas as outras mudanças em SNMP todas resultaram da necessidade de segurança do protocolo, que só vieram a eclodir com o SNMPv3.

Existem controvérsias sobre o desempenho dos métodos de segurança implantados no protocolo SNMPv3, no tocante à capacidade de utilização do tráfego de dados (FIANG, 2002). Como descrito em (HIA, 2001) foram implementadas duas versões de SNMP seguro, uma usando o protocolo SNMPv2C, encapsulado em IPSec (IP over security), e a outra usando o protocolo SNMPv3. Foi concluído que o SNMPv3 consumiu 24% a mais da capacidade de tráfego da rede quando comparado com a versão SNMPv2C do protocolo sobre IPSec. Este fato foi atribuído às regras de codificação utilizadas para codificar as mensagens SNMP.

## 3.9 – Considerações Finais

Como explanado durante todo o capítulo, o SNMP é um protocolo largamente utilizado em aplicações de redes TCP/IP. O seu grande sucesso deve-se à sua simplicidade de construção e sua boa definição, através da SMI, mostrando-se uma boa estrutura de dados. Podemos citar como vantagem do protocolo a sua simplicidade, tanto em sua operação como em sua implementação, resultando num mínimo de recursos necessários para o funcionamento do protocolo. Outra característica positiva é a sua extensibilidade através da criação de MIBs particulares, permitindo aos fabricantes desenvolverem novos recursos de gerenciamento para seus equipamentos (SCHONWALDER, 2005).

O SNMP também mostra robustez no seu uso ponto a ponto, onde o fluxo de dados sai do gerente para o agente diretamente, sem sofrer interferências de nós intermediários. A capacidade de manipular variáveis da MIB também se torna uma excelente característica em SNMP permitindo maior poder de gerenciamento. Como desvantagens do protocolo podemos citar a falta de segurança em suas operações, estando as mesmas vulneráveis, podendo o protocolo ter sua segurança facilmente “quebrada” através da monitoração das mensagens.

Outra desvantagem do protocolo é a necessidade de implantação de operações seguras no gerente, pois o mesmo opera sobre um protocolo sem conexão, o UDP, o que pode acarretar em perda de dados nas trocas de mensagens, devendo essa função ser monitorada pelo gerente SNMP. Outra desvantagem diz respeito ao baixo desempenho do protocolo na transmissão de dados complexos, como tabelas, arquivos e informações visuais, em um curto espaço de tempo (AMIRTHALINGAM, 1995). Embora na segunda versão do protocolo, uma PDU específica para este fim tenha sido implementada, a mesma ainda apresenta baixo desempenho na pesquisa de dados aleatórios em uma tabela. Alguns autores apresentam métodos que melhoram este desempenho, no intuito de solucionar este problema, utilizando somente as PDUs existentes no protocolo SNMP, como em (BREITGAND, 2001).

Podemos concluir que o SNMP é atualmente o protocolo de mais eficiência e sucesso no uso de gerenciamento de redes, e que não existem dúvidas que a demanda futura por este protocolo irá crescer fortemente entre os fabricantes de equipamentos de rede, e os que por ventura usarem desta estrutura para interligar dispositivos.

# Capítulo 4

# 4. Gerenciamento SNMP com Autenticação Remota: Aplicações em UPSs

## 4.1 – Introdução

Como já discutido anteriormente, existe cada vez mais, uma forte dependência de sistemas computacionais, que de alguma forma processam alguma informação para auxílio em nossas atividades. Esses equipamentos necessitam de energia elétrica para executarem suas funções e, na maioria das vezes, desta falta pode resultar em perda de funcionalidade, que pode, por sua vez, resultar em prejuízos. Estes prejuízos que resultam não só da sua falta momentânea, por alguma falha temporária na rede elétrica, mas também pela destruição total ou parcial de dispositivos conectados na rede elétrica.

A proteção mais indicada para equipamentos que necessitam de energia elétrica para funcionar são as UPSs, por possuírem várias proteções que asseguram a qualidade de energia, além de manterem sua constância, sendo esta sua principal função (MOLINA, 1999). Temos observado também que as UPSs têm adquirido outras funções importantes, além de proteger suas cargas e permitir sua operação na falta de energia. Estas funcionalidades estão associadas principalmente às atividades de coleta de informações. As UPSs mais modernas realizam internamente análise de funcionamento da rede elétrica, dados de rendimento elétrico, relatório de eventos, dentre outros (BALACHANDRA, 2000).

Devido ao grande número de aplicações em UPSs estarem voltadas para aplicações computacionais, e estas por sua vez se desenvolverem em

ambientes de aplicações distribuídas, as UPSs também têm o seu funcionamento dependente de aplicações distribuídas. As UPSs, por conviverem nesse ambiente computacional, e apresentarem cada vez mais inteligência embarcada, são cada vez mais requisitadas para executarem funções de gerenciamento. Como as UPSs são responsáveis por manterem aplicações críticas em funcionamento, elas por sua vez tornam-se fundamentais, e sua monitoração passa a ser uma peça importante neste cenário.

Neste capítulo iremos apresentar a implementação de um sistema de gerenciamento para aplicações em UPSs. Inicialmente iremos enumerar e analisar várias soluções estudadas, realçando suas vantagens e desvantagens, tanto no aspecto funcional como na sua implementação. Em seguida iremos apresentar qual a proposta escolhida, e quais os aspectos decisivos para a sua escolha. Após isso, iremos descrever a solução implementada, bem como aspectos de funcionalidade, operações disponíveis, interface com usuário, dentre outros. Iremos descrever também quais os testes realizados e apresentar os resultados encontrados. Por fim, faremos as considerações finais relativas à aplicação implementada.

## 4.2 – Soluções Estudadas

Para melhor entender as soluções aqui propostas para implementar um sistema de gerenciamento para UPSs, devemos descrever qual o provável cenário no qual este sistema se propõe a operar. A aplicação irá funcionar em um ambiente onde estará disponível uma rede de computadores, do tipo local ou metropolitana, onde a comunicação de dados seja baseada no conjunto de protocolos TCP/IP.

A primeira solução proposta consistiu no uso de um protocolo proprietário para comunicação entre as UPSs e a estação de gerenciamento. Nessa proposta, o meio físico para tráfego dos dados foi compartilhado com a

rede de dados do ambiente. Para facilitar a implementação de interfaces do sistema, a comunicação entre a rede de dados e a UPS foi feita através de um computador, o qual pode realizar além desta, outras funções. Na Figura 4.1 são mostrados os elementos presentes nesta proposta. A estação de gerenciamento monitora as UPSs na rede via um protocolo proprietário, através de uma conexão TCP, com um aplicativo que é executado no computador, que por sua vez se comunica com a UPS via protocolo RS-232. Esta solução apresenta algumas vantagens, pois devido ao protocolo ser proprietário, o mesmo pode ser otimizado em seu desempenho, resultando em um tráfego mínimo de informações de controle na rede.

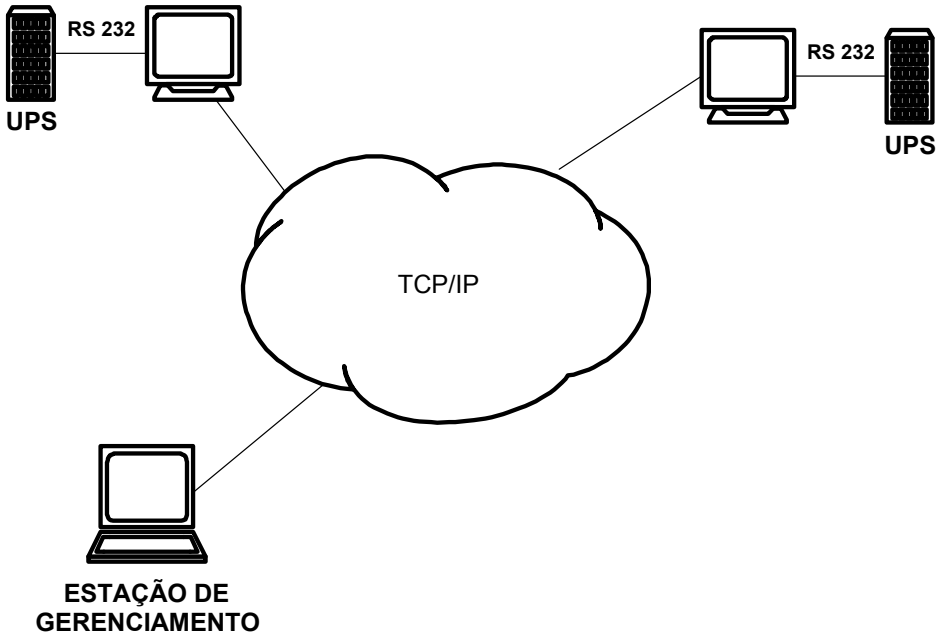


Figura 4.1: proposta de solução de gerenciamento proprietária.

Da mesma forma, a solução representaria também um menor esforço no desenvolvimento, pois não precisaria se adequar a nenhuma norma ou padrão. As desvantagens dessa implementação ocorrem principalmente devido à particularidade da implementação, que representa um investimento em uma solução muito particular. Nessa situação, caso haja alguma expansão

de UPSs no sistema, e com a troca do fornecedor, certamente seria necessário um novo investimento em uma nova ferramenta de gerenciamento bem como treinamento de pessoal. Consideramos esta solução como ineficiente para o sistema de gerenciamento de UPSs pretendido.

Uma segunda solução proposta foi a utilização de duas redes distintas, conforme mostrado na figura 4.2. Uma rede de dados, na qual trafegam os dados da rede da aplicação já existente, e uma segunda rede, que somente trafegariam os dados de comunicação entre as UPSs. Na primeira rede teríamos a estação de gerenciamento em comunicação com o elemento de ligação entre as redes, que garante a existência de um padrão de gerenciamento, como proposto em (CARDOSO, 1998).

A comunicação entre a estação de gerenciamento e o elemento de ligação entre as redes seria realizada através de um protocolo padrão de gerenciamento, como o SNMP. Já na segunda rede, teríamos um outro meio físico de rede, como por exemplo, o padrão RS485, em que é possível interligar vários dispositivos em rede através de um par de fios. Esta solução apresenta a vantagem de não precisar compartilhar os computadores com aplicativos de gerenciamento, pois as UPSs estariam em rede via um meio físico particular. Deste modo, uma falha no computador não impediria a continuidade de gerenciamento da UPS. As desvantagens dessa solução residem na dificuldade de se implantar uma nova rede, resultando em investimento para implantação e manutenção, além de recurso extra para o elemento de ligação das redes. Consideramos, por esses motivos, esta solução desvantajosa para o sistema de gerenciamento de UPSs.

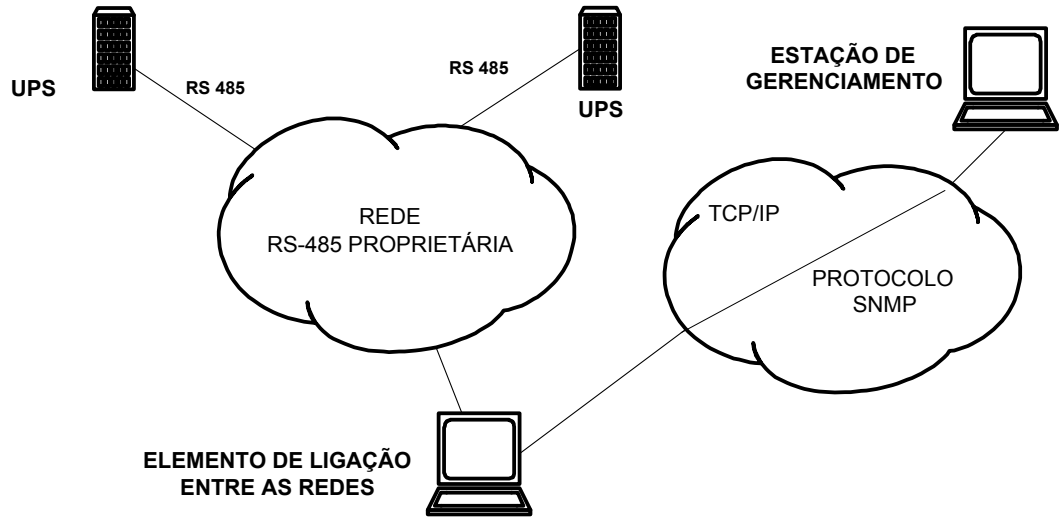


Figura 4.2: implementação com duas redes.

Uma terceira solução proposta seria utilizar um navegador padrão de Internet e usar o protocolo HTTP para monitoração das UPSs, como proposto em (SARKINEN, 1998), em que foi implementada uma solução de gerenciamento para equipamentos de energia. Nesta solução, como mostrado na Figura 4.3, temos mais uma vez a UPS disponibilizando os dados para gerenciamento através de um computador, onde será executado um aplicativo que, através de conexão, troca informações via RS-232 com a UPS e então disponibiliza via um servidor WEB opções de gerenciamento. Nesta solução, temos como vantagem a não necessidade de um software proprietário na estação de gerenciamento, já que os navegadores representam hoje um aplicativo básico em qualquer sistema operacional (HONG, 2001). Observamos como desvantagem a inviabilidade do gerenciamento simultâneo de várias UPSs em um ambiente integrado, visto que para isso seria necessário o uso de vários navegadores simultâneos o que dificultaria a operação. Esta solução, pelos motivos apresentados, também não foi avaliada como satisfatória para o sistema de gerenciamento de UPS.

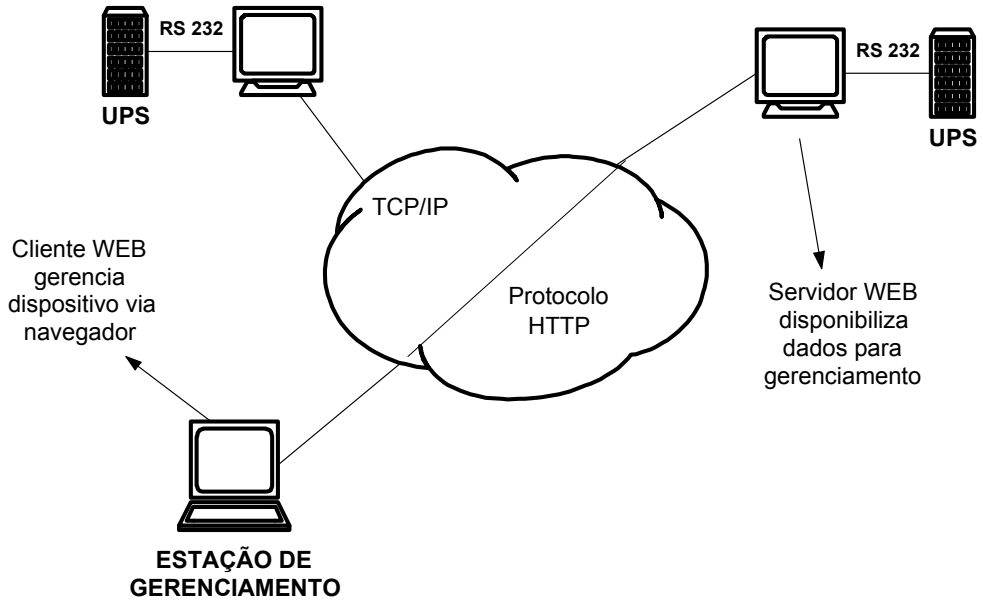


Figura 4.3: gerenciamento via navegador.

A quarta solução analisada para implementação de um sistema de gerenciamento para UPSs foi baseada no protocolo SNMP. Nesta solução, teríamos um agente em computador, que ligado via RS-232 a cada UPS para coletar os dados da mesma, disponibilizaria os dados na MIB do agente. Assim, as UPSs entrariam na rede através dos computadores. Como mostrado na Figura 4.4, nesta solução teríamos uma única rede, que seria a mesma em que trafegam os dados da rede de computadores. O gerente SNMP poderia gerenciar as UPSs através das mensagens SNMP disponíveis neste protocolo. Esta solução tem a vantagem de aproveitar o mesmo meio físico, além de proporcionar um gerenciamento centralizado, permitindo gerenciar um número arbitrário de UPSs. A desvantagem desta solução se dá pelo compartilhamento do agente SNMP na mesma plataforma pertencente à rede de dados com os computadores pessoais da rede. Esta foi a solução escolhida para ser implementada, para gerenciamento de UPS.

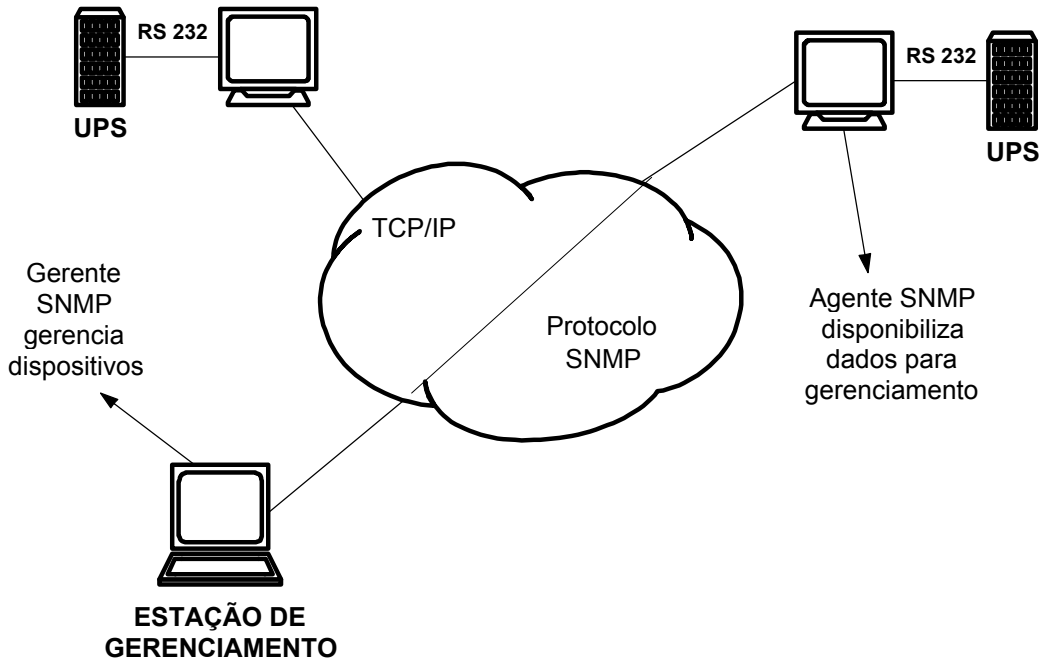


Figura 4.4: gerenciamento via SNMP.

## 4.3 – Justificativa

A solução escolhida, mostrada na Figura 4.4, apresenta a vantagem de usar um protocolo simples de implementar e consequentemente necessidade de menos recursos para o seu correto funcionamento. Uma outra vantagem da solução é a sua compatibilidade com o sistema já existente, pois, em redes TCP/IP, é muito comum o uso de ferramentas de gerenciamento compatíveis com SNMP. Caso haja alguma mudança no fornecedor de UPS, a planta de gerenciamento pode ser aproveitada como sistema de gerenciamento, ou facilmente incorporada ao novo sistema adquirido. Além disso, a solução garante interoperabilidade entre outros sistemas, sendo possível gerenciar em um mesmo ambiente todos os elementos de rede, sendo a UPS parte deles.

Outra grande vantagem é a possibilidade de centralização do gerenciamento, sendo possível gerenciar todas as UPSs a partir de uma só estação. Esta vantagem se reflete na forma descentralizada e na flexibilidade de gerenciamento, permitindo que um número arbitrário de UPSs possa ser gerenciado por uma estação, e outra porção das mesmas por outra estação de gerenciamento. Esta funcionalidade reflete muito bem o conceito de comunidade em SNMP, como pode ser visto na Figura 4.5.

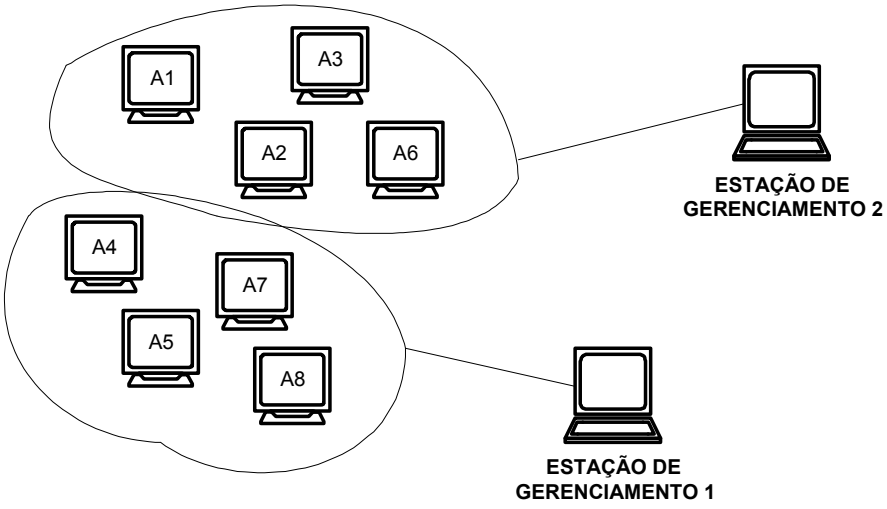


Figura 4.5: conceito de comunidade em SNMP.

A solução escolhida apresenta como desvantagem o fato do aplicativo do gerente estar sendo processado em compartilhamento com o computador em uso na rede, pois, no caso de uma falha em um dos computadores o agente torna-se inacessível. Esta problemática só existe quando não for possível disponibilizar um agente SNMP interno na própria UPS, o que é possível desenvolver, mas com o custo adicional de embarcar o hardware e processamento de uma conexão Ethernet. Esta solução será discutida ao longo do capítulo e será referenciada como agente embarcado, enquanto o agente que será processado em compartilhamento com o computador pessoal será chamado simplesmente de agente. Para o gerente SNMP, este problema torna-se transparente, pois os dois tipos de agentes devem disponibilizar os dados para gerenciamento.

A seguir, iremos descrever a implementação da solução de gerenciamento SNMP para UPS. A estrutura das informações de gerenciamento será relatada, bem como sua descrição. Em seguida serão discutidas quais as funções e operações de cada elemento de gerenciamento.

## 4.4 – A Implementação do Sistema

Como já salientado, este trabalho descreve a implementação de um sistema de gerenciamento SNMP para um número arbitrário de UPSs. Partindo da arquitetura de gerenciamento SNMP, em que são definidos um gerente e um agente, este sistema define, além desses elementos, uma terceira figura denominada **Autenticador de Gerentes**. Este tem a função de validar a identidade dos gerentes, após o que ele libera as informações (contidas em uma base de dados) necessárias para eles efetuarem suas funções. Na Figura 4.6 são mostrados os elementos presentes no sistema de gerenciamento proposto.

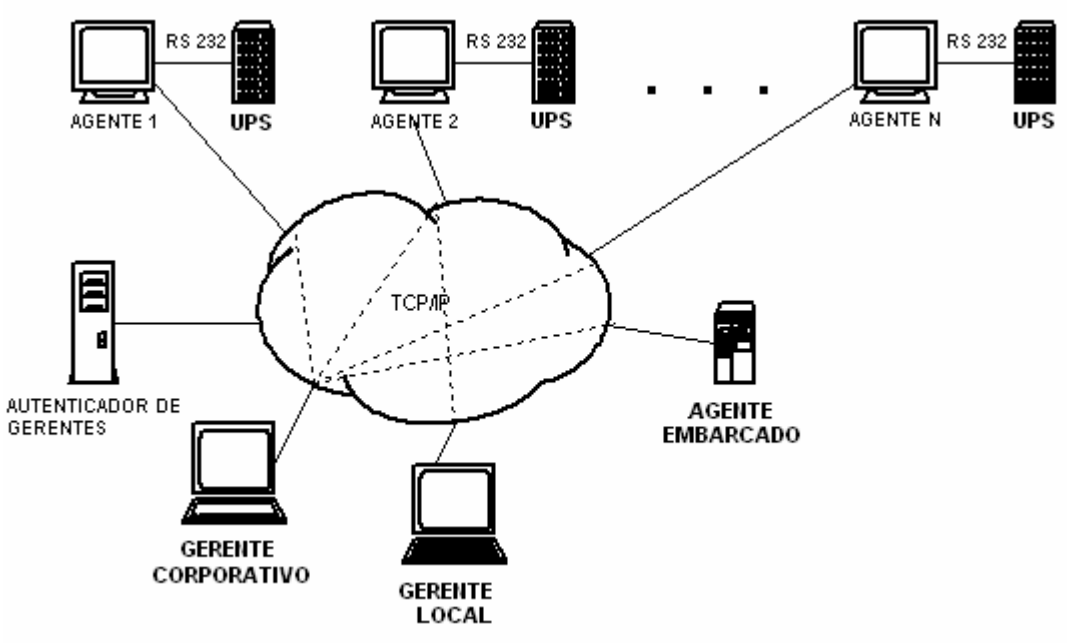


Figura 4.6: ambiente de gerenciamento completo.

O sistema é composto de dois tipos de gerentes SNMP: o gerente corporativo e o gerente local. O gerente corporativo permite que sejam visualizados, em um ambiente hierárquico, todas as UPSs presentes na comunidade de gerenciamento. O gerente local permite que o usuário gerencie uma UPS, visualizando através de interface gráfica o estado do agente gerenciado. Em ambos os gerentes, o protocolo utilizado é o SNMP. O gerente local é usado também para que o usuário de um computador pessoal monitore a UPS conectada ao seu próprio computador. Na Figura 4.6 observamos um agente embarcado compondo a comunidade de gerenciamento. Este agente participa da comunidade de maneira transparente para o ambiente de gerenciamento.

A UPS utilizada para desenvolver o sistema foi uma UPS de dupla conversão de 10 KVA, com correção ativa do fator de potência, alimentação de entrada monofásica, com bypass automático sincronizado, como descrito em seu manual (MANUAL DO RHINO, 2003). É importante salientar que muitas opções disponíveis na MIB RFC 1628 não foram implementadas na aplicação devido a UPS não dispor dos recursos, enquanto outras opções foram adicionadas para utilizar os recursos disponíveis na UPS usada, conforme descrito no seção seguinte.

### 4.4.1 – A MIB Utilizada

Como descrito no Capítulo 3, a MIB é a base de dados que é comum ao gerente e ao agente, e que deve ser usada como referência para troca de dados entre eles. A MIB recomendada para utilização em UPS está contida na RFC 1628, descrita em (CASE, 1994). Esta MIB descreve alguns valores que não são usados no sistema aqui proposto, em alguns casos por não serem aplicáveis ao tipo de topologia de UPS empregada, ou pela UPS não suportar o recurso em questão. Por outro lado, foram acrescentados alguns valores que não estão descritos na RFC 1628, devido à mesma não suportar alguns valores em uso. A MIB utilizada, que tem como base a RFC 1628, é descrita a seguir.

#### 4.4.1.1 - Descrição da MIB

A MIB RFC1628 foi desenvolvida por um grupo de trabalho do IETF composto pesquisadores e representantes de vários fabricantes mundiais de UPS. A MIB RFC 1628 é composta de 9 módulos de gerenciamento. Cada módulo é destinado a um conjunto de informações de origem semelhante. A MIB tem origem na estrutura de gerenciamento descendendo do nó MIB, cujo posicionamento na hierarquia de gerenciamento é descrita na figura 4.7. O nó UPS tem numeração 33, portanto terá a seguinte OID:

1(iso).3(org).6(dod).1(internet).2(mgmt).1(mib).33(ups).

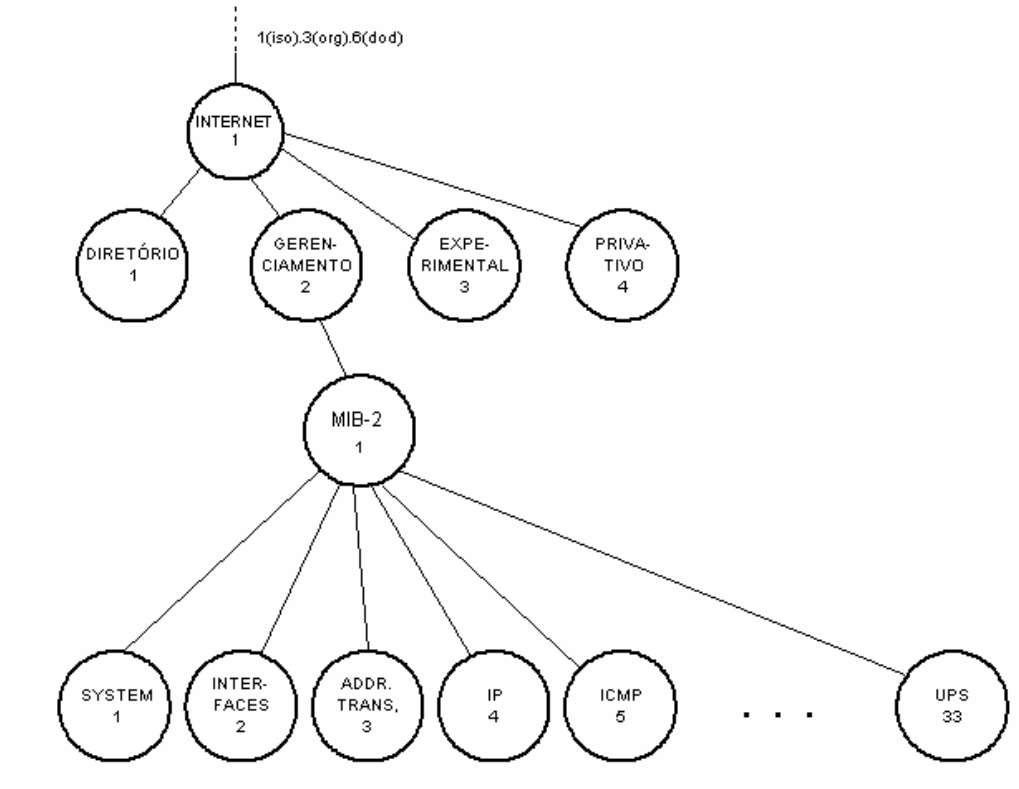


Figura 4.7: hierarquia da MIB para UPS.

A seguir, descreveremos um dos nove módulos da MIB RFC 1628. A descrição completa dos módulos da MIB para UPSs esta descrita na RFC 1628. Uma literatura complementar pode ser encontrada em [SILVEIRA, 2004].

## 4.4.1.2 – Grupo de Identificação

O grupo de identificação reúne variáveis relacionadas com a identificação do elemento gerenciado. Os dados compreendidos neste grupo, exceto *UpsIdentName* e *UpsIdentAttachedDevices*, são configurados na inicialização do agente, e permanecem inalterados. Na Tabela 4.1 são descritos os objetos presentes no módulo de identificação. Neste módulo, todos os valores foram implementados. Os objetos *upsIdentUPSSoftwareVersion* e *upsIdentAgentSoftwareVersion* são iguais quando a UPS não tem o recurso de informação da versão interna de firmware. Este fato acontece quando a

plataforma da UPS não oferece recursos para este nível de informações.

Tabela 4.1: módulo de Identificação – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsIdentManufacturer* | O nome do fabricante da UPS. |
| *upsIdentModel* | O modelo da UPS. |
| *upsIdentUPSSoftwareVersion* | A versão do firmware da UPS. Em algumas implementações pode ter o mesmo valor do objeto *UpsIdentAgentSoftwareVersion.* |
| *upsIdentAgentSoftwareVersion* | A versão do software do agente. Em algumas implementações pode ter o mesmo valor do objeto *UpsIdentUPSSoftwareVersion.* |
| *upsIdentName* | Seqüência de caracteres que indica o nome da UPS. Pode ser configurado pelo administrador de rede. |
| *upsIdentAttachedDevices* | Seqüência de caracteres que indica quais as cargas conectadas à saída da UPS. Pode ser configurado pelo administrador de rede. |

### 4.4.1.3 - Grupo de Bateria

O grupo de bateria traz informações referentes ao acumulador de energia da UPS. As informações compreendidas neste grupo estão detalhadas na Tabela 4.2. No objeto *upsBatteryStatus* o valor de “Bateria Baixa” , que por definição da RFC 1628 recebe o valor numérico 3, é o valor no qual o objeto *upsEstimatedMinutesRemaining* tem um valor menor do que o do objeto *upsConfigLowBattTime*, que se encontra no módulo de configuração da MIB. Este recurso torna flexível o uso do alarme de bateria baixa, para que o gerente possa ser avisado quando as baterias estiverem com autonomia desejada pelo administrador do sistema. A este grupo, acrescentamos um objeto não disponível para informar a tensão nominal das baterias que a UPS utiliza, sendo possível assim identificar o número de baterias utilizadas, que no caso

de uma manutenção torna-se um dado bastante importante. Este objeto foi adicionado na aplicação em questão, e será visto com mais detalhes na Seção 4.4.2.

Tabela 4.2: módulo de Baterias – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsBatteryStatus* | Indica a capacidade remanescente das baterias. Pode ter quatro valores: Desconhecido (1), Normal (2), Baixa (3), Esgotada (4). |
| *upsSecondsOnBattery* | Tempo passado a partir do início da utilização da energia das baterias pela UPS para alimentar a carga em sua saída. |
| *upsEstimatedMinutesRemaining* | Tempo (em minutos) de autonomia restante das baterias. |
| *upsEstimatedChargeRemaining* | Percentual restante para carga plena das baterias. |
| *upsBatteryVoltage* | Magnitude em 0,1 Volts DC da voltagem das baterias. |
| *upsBatteryCurrent* | Magnitude em 0,1 Amp DC da corrente das baterias. |
| *upsBatteryTemperature* | Magnitude em graus centígrados da temperatura nas baterias ou em ambiente próximo. |

### 4.4.1.4 – Grupo de Entrada

O grupo de entrada relata informações referentes à rede elétrica que alimenta a UPS. Os objetos deste grupo, originários da RFC 1628, estão descritos na Tabela 4.3. Neste módulo, dispomos de uma tabela (em sua forma original) que tem o número de linhas definido pelo objeto *upsInputNumLines*,

sendo possível assim armazenar os dados referentes à cada fase, no caso da alimentação da UPS ser monofásica ou trifásica. Às informações deste grupo, adicionamos um objeto para indicar o valor do fator de potência de entrada da UPS (não representado na tabela). Este valor constitui um importante dado de especificação de UPS, sendo de extrema importância para manter o nível de qualidade de energia do sistema (SARAIVA, 2003).

Tabela 4.3: módulo de Entrada – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsInputLineBads* | Conta o número de falhas na rede elétrica. |
| *upsInputNumLines* | Indica o número de entradas (alimentação pela rede elétrica) que a UPS dispõe. Indica também o número de linhas da Tabela de entrada. |
| *upsInputLineIndex* | Indica o ponto de entrada da tabela. |
| *upsInputFrequency* | Magnitude da freqüência de entrada em 0,1 Hz. |
| *upsInputVoltage* | Magnitude da voltagem de entrada em Volts RMS. |
| *upsInputCurrent* | Magnitude da corrente de entrada em 0,1 Amp RMS. |
| *upsInputTruePower* | Magnitude da potência de entrada em Watts. |

### 4.4.1.5 - Grupo de Saída

O grupo de saída descreve valores referentes à saída da UPS. Estes objetos, constantes da RFC 1628, estão descritos na Tabela 4.4. Neste módulo, dispomos de uma tabela que permite que todos os seus valores sejam gerenciados, caso a UPS tenha várias saídas, como, por exemplo, uma saída trifásica. Isto se torna possível através da variável *upsOutputNumLines*, que

representa o número de saídas da UPS. Incluímos neste grupo uma variável (não representado na tabela) para indicar o fator de potência de saída, embora este valor possa ser retirado da relação dos valores de corrente, tensão e potência de saída. Isto traria uma particularidade para o gerente, obrigando que o cálculo fosse feito no mesmo, e como na UPS utilizada para testes já se dispunha desta variável calculada internamente, a melhor opção foi incluir esta variável.

Tabela 4.4: módulo de Saída – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsOutputSource* | Indica o estado da tensão de saída da UPS. Pode ter os seguintes valores: outro(1), nenhum(2), normal(3), bypass(4), baterias(5), booster(6), reduzida(7). |
| *upsOutputFrequency* | Magnitude da freqüência de saída em 0,1 Hz. |
| *upsOutputNumLines* | Indica o número de saídas que a UPS dispõe. Indica também o número de linhas da tabela de saída. |
| *upsOutputLineIndex* | Indica o ponto de entrada da tabela. |
| *upsOutputVoltage* | Magnitude da tensão de saída em Volts RMS. |
| *upsOutputCurrent* | Magnitude da corrente de saída em 0,1 Amp RMS. |
| *upsOutputPower* | Magnitude da potência de saída em 0,1 Watts. |
| *upsOutputPercentLoad* | Percentual da potência máxima de saída em utilização. |

### 4.4.1.6 - Grupo de Bypass

O grupo de Bypass traz informações referentes ao ramo de Bypass da UPS. Esta funcionalidade é importante pela necessidade de se transferir a alimentação da carga da UPS para o ramo de Bypass, normalmente proveniente da rede elétrica. Esta operação, normalmente é efetuada em caso de manutenção da UPS, ou na ocorrência de uma falha na alimentação das cargas, ou ainda num caso de sobrecarga (IEC 62040-3, 1999). Os objetos deste grupo estão descritos na Tabela 4.5. Este grupo dispõe também de uma tabela que permite que vários ramos de Bypass sejam monitorados, caso a UPS disponha desse recurso.

Tabela 4.5: módulo de Bypass – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsBypassFrequency* | Magnitude da freqüência do ramo de Bypass em 0,1 Hz. |
| *upsBypassNumLines* | Indica o número de ramos de Bypass que a UPS dispõe. Indica também o número de linhas da tabela de Bypass. |
| *upsBypassLineIndex* | Indica o ponto de entrada da tabela. |
| *upsBypassVoltage* | Magnitude da tensão do ramo de Bypass em Volts RMS. |
| *upsBypassCurrent* | Magnitude da corrente no ramo de Bypass em 0,1 Amp RMS |
| *upsBypassPower* | Magnitude da potência no ramo de Bypass em Watts. |

### 4.4.1.7 - Grupo de Alarmes

Este grupo de alarmes contém informações sobre os tipos de alarmes que o agente irá enviar para os gerentes. Estes alarmes são referentes aos traps específicos implementados por cada fabricante. A RFC 1628 recomenda um total de 24 alarmes, dispostos em forma de tabelas, para que eles possam também ser acessados por busca contínua, necessidade principalmente observada quando da inicialização de um gerente, quando os agentes já estão ativos (STALLINGS, 1999). Este gerente precisa fazer um busca em todos os agentes ativos para conhecer todos os alarmes que estão acionados. Na Tabela 4.6 são descritas as variáveis que identificam estes alarmes, e serão usadas na mensagem Trap para identificar qual alarme foi ativado ou desativado no evento em questão. Este grupo dispõe de uma tabela de três linhas para que os alarmes possam ser acessados. Para cada item da tabela, que representa um evento referente a um alarme, dispomos de três informações: *upsAlarmId*, *upsAlarmDescr* e *upsAlarmTime*. Na Tabela 4.7 temos a descrição dos alarmes implementados na RFC 1628.

Tabela 4.6: módulo de Alarmes – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsAlarmsPresent* | O número presente de alarmes em condição de ativos. |
| *upsAlarmTable* | Lista de alarmes ativos. A cada alarme ativado mais uma linha é adicionada. |
| *upsAlarmEntry* | Entrada da tabela para o acesso aos alarmes. |
| *upsAlarmId* | Indica condição de alarme presente. |
| *upsAlarmDescr* | Uma referência ao objeto descritor do alarme. |
| *upsAlarmTime* | Valor do relógio do sistema |

Tabela 4.7: alarmes – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsAlarmBatteryBad* | Indica que uma ou mais baterias necessitam ser substituídas. |
| *upsAlarmOnBattery* | Indica que a UPS está utilizando a energia das baterias para alimentar as cargas. |
| *upsAlarmLowBattery* | Indica que a autonomia restante das baterias é menor que o valor configurado em *upsConfigLowBattTime.* |
| *upsAlarmDepletedBattery* | Indica que as baterias encontram-se descarregadas. |
| *upsAlarmTempBad* | Indica que a temperatura das baterias esta acima do valor tolerado. |
| *upsAlarmInputBad* | Indica falha na rede elétrica. A mesma está fora das condições de aceitação. |
| *upsAlarmOutputBad* | Indica Falha na saída da UPS por um motivo diferente de sobrecarga. |
| *upsAlarmOutputOverload* | Indica sobrecarga na saída da UPS, o que ocasionou o desligamento da mesma. |
| *upsAlarmOnBypass* | Indica que a UPS tem a saída comutada para Bypass. |
| *upsAlarmBypassBad* | Indica que o Bypass falhou. |
| *upsAlarmOutputOffAsRequested* | Indica que a saída da UPS foi desligada através de um comando. |
| *upsAlarmUpsOffAsRequested* | Indica que toda a UPS foi desligada através de comando. |
| *upsAlarmChargerFailed* | Indica falha no carregador de baterias da UPS. |
| *upsAlarmUpsOutputOff* | Indica que a saída da UPS está desligada |
| *upsAlarmUpsSystemOff* | Indica que a UPS está desligada. |
| *upsAlarmFanFailure* | Indica falha no sistema de ventilação da UPS. |
| *upsAlarmFuseFailure* | Indica falha em dos fusíveis da UPS (entrada, carga de bateria, saída). |
| *upsAlarmGeneralFault* | Indica a ocorrência de uma falha geral na UPS. |
| *upsAlarmDiagnosticTestFailed* | Indica que o último teste realizado na UPS detectou uma falha. |
| *upsAlarmCommunicationsLost* | Indica uma falha na comunicação entre a UPS e o agente SNMP. |
| *upsAlarmAwaitingPower* | Indica que a saída da UPS está desligada, provavelmente por baterias esgotadas e a UPS está esperando retorno de alimentação na entrada. |
| *upsAlarmShutdownPending* | Indica que o contador upsShutdownAfterDelay está sendo decrementado. |
| *upsAlarmShutdownImminent* | Indica que a UPS irá ser desligada em menos de 5 segundos. Este desligamento irá ocorrer, ou por baterias esgotadas, ou por desligamento programado. |
| *upsAlarmTestInProgress* | Indica que a UPS tem algum teste em progresso. |

### 4.4.1.8 - Grupo de Testes

No grupo de Testes estão disponíveis variáveis que possibilitam a realização de testes na UPS. O gerente pode comandar o início e o fim de testes, bem como monitorar dados durante a realização de testes. Na Tabela 4.8 temos as variáveis que identificam os tipos de testes disponíveis. Na Tabela 4.9 estão descritas as variáveis para manipular a execução dos testes. Na MIB RFC1628 estão implementados três tipos de testes:

- *UpsTestGeneralSystems*, em que é possível realizar um teste geral no sistema, e que é particular de cada fabricante, não sendo a sua implementação padronizada;
- *UpsTestQuickBatteryTest*, que é realizado através de uma rápida descarga na bateria, é possível avaliar se as mesmas ainda estão em condições de operar corretamente;
- *UpsTestDeepBatteryCalibration*, em que o agente faz uma calibração do tempo de autonomia da bateria considerando a sua vida útil, que após o teste, a bateria precisa ser recarregada.

Na implementação aqui descrita somente o upsTestQuickBatteryTest foi implementado, pois a UPS utilizada não dispunha dos recursos necessários para implementação dos outros testes.

Tabela 4.8: testes definidos – MIB UPS RFC 1628.

| | |
|---|---|
| *upsTestGeneralSystems* | Teste geral da UPS. Este teste pode mudar dependendo do fabricante. |
| *upsTestQuickBatteryTest* | Teste rápido de baterias. Determina se é necessário substituição das mesmas. |
| *upsTestDeepBatteryCalibration* | Teste profundo de baterias. As baterias são descarregadas completamente para efeito de calibração do tempo de autonomia. |

Tabela 4.9: módulo de testes – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsTestId* | É usado para indicar qual o teste que será realizado na UPS. Este identificador deve estar presente na lista de testes definida. |
| *upsTestSpinLock* | É usado para controle de acesso aos testes disponíveis na UPS. |
| *upsTestResultsSummary* | Indica o resultado do último teste realizado. Pode ter os seguintes valores: Pronto(1), Pronto com observação(2), Erro(3), Abortado(4), Em progresso(5), Não iniciado(6). |
| *upsTestResultsDetail* | Indica informação sobre resultado do último teste. O valor (0) indica nenhuma informação disponível. |
| *upsTestStartTime* | Valor do relógio do sistema no início do teste. |
| *upsTestElapsedTime* | Indica o tempo passado desde o início do teste. |

### 4.4.1.9 - Grupo de Controle

O grupo de Controle define variáveis para serem utilizadas no *shutdown* (desligamento) da UPS. As variáveis deste grupo estão descritas na Tabela 4.10. As funções de *shutdown* disponíveis na RFC 1628 permitem que a UPS seja ligada ou desligada após um tempo determinado. É muito comum que os administradores de rede se utilizem de um recurso conhecido como **Comandos de Shutdown Programados**. Através deste recurso, é possível que a UPS seja ligada ou desligada periodicamente em períodos previamente programados. Esta funcionalidade foi adicionada à MIB do sistema. Estes recursos serão tratados em detalhe no item 4.4.2 .

Tabela 4.10: módulo de controle – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsShutdownType* | Indica qual a natureza do desligamento a ser realizado na UPS. Se o valor for (1) somente a saída será desligada. Se o valor for (2) toda a UPS será desligada. |
| *upsShutdownAfterDelay* | Indica um valor para contagem em segundos para início do processo de desligamento da UPS, especificado em upsShutdownType. |
| *upsStartupAfterDelay* | Indica um valor para contagem em segundos para início do processo de saída ligada da UPS. |
| *upsRebootWithDuration* | Indica que a UPS fará um desligamento, somente retornando após o tempo especificado em segundos para ser novamente ligada. |
| *upsAutoRestart* | Indica que a UPS irá ser ligada novamente após um desligamento por bateria baixa ou desligamento com retardo por falha na alimentação de entrada. |

### 4.4.1.10 - Grupo de Configuração

O grupo de Configuração permite ao usuário configurar alguns parâmetros importantes no funcionamento da UPS, como tensão de entrada a ser utilizada, tensão de saída, e freqüência. Caso algum destes valores seja configurado com um valor não aceito pela UPS, o agente deve responder com uma mensagem de erro do tipo *BadValue* (em SNMPv1). Os objetos deste grupo estão descritos na Tabela 4.11. Para esta implementação, adicionamos variáveis para configuração dos limites de freqüência de entrada e de saída aceitáveis para a UPS, não disponíveis originalmente na MIB RFC 1628. Adicionamos também uma opção de funcionamento da UPS usada para desenvolvimento, o desligamento por ausência de consumo. No objeto descrito

na MIB complementar, o *upsConfigDapac*, é possível que a UPS desligue sua saída caso um limite mínimo de consumo não seja percebido em sua saída (MANUAL DO SOLIS, 2005). Neste grupo, somente foi implementado o objeto *upsConfigLowBattTime,* para que o alarme de bateria baixa seja detectado no ponto escolhido de autonomia restante. As outras variáveis não foram utilizadas pelo fato da UPS usada para testes não dispor dos recursos.

Tabela 4.11: módulo de configuração – MIB UPS RFC 1628.

| Objeto | Descrição |
|---|---|
| *upsConfigInputVoltage* | Indica o valor nominal em Volts RMS da tensão de entrada. |
| *upsConfigInputFreq* | Indica o valor nominal em 0,1 Hz da freqüência nominal de entrada. |
| *upsConfigOutputVoltage* | Indica o valor nominal em Volts RMS da tensão de saída. |
| *upsConfigOutputFreq* | Indica o valor nominal em 0,1 Hz da freqüência de saída. |
| *upsConfigOutputVA* | Indica o valor nominal em VA da potência de saída. |
| *upsConfigOutputPower* | Indica o valor nominal em Watts da potência de saída. |
| *upsConfigLowBattTime* | Indica o valor de autonomia que será considerado bateria baixa. |
| *upsConfigAudibleStatus* | Indica qual o estado audível dos alarmes. desabilitado(1), habilitado (2), ou mudo(3). No estado de mudo(3), o alarme ativo será temporariamente desabilitado. |
| *upsConfigLowVoltageTransferPoint* | O valor mínimo da alimentação de entrada para que a UPS alimente a carga pela baterias. |
| *upsConfigHighVoltageTransferPoint* | O valor máximo da alimentação de entrada para que a UPS alimente a carga pela baterias. |

## 4.4.2 – A MIB Complementar

Como já citado no item anterior, foram adicionados vários objetos gerenciados à MIB RFC 1628, na implementação aqui descrita. Este fato foi necessário devido à diferença existente entre as diversas UPSs, embora a RFC para UPSs, contida na RFC 1628, tenha sido concebida tendo em vista uma vasta faixa de fabricantes, contando inclusive com a presença de vários especialistas na área. A necessidade de complementar a MIB das UPSs se deve também à constante evolução em que se encontram as mesmas, oferecendo cada vez mais recursos ao seus usuários, estando as UPSs inseridas em um mercado bastante competitivo, existindo muitos fabricantes em todo o mundo.

A necessidade dos usuários, como exposto acima, levou à inclusão de objetos gerenciados, que se realiza de maneira muito natural, pois, o protocolo SNMP já oferece um excelente suporte à extensão dos dados de gerenciamento de um agente. Como descrito nas metas do protocolo na RFC 1157 (CASE, 1990), em que é descrito o protocolo, a arquitetura de gerenciamento deve ser flexível e independente da arquitetura e dos mecanismos particulares do ambiente de rede, em que o mesmo foi desenvolvido.

Segundo a RFC 1155 (ROSE, 1990), onde está descrita a estrutura das informações de gerenciamento, a sub-árvore usada para definir objetos privativos é a *private*(4) como mostrada na Figura 4.8. Estes objetos unilaterais se referem a aplicações de empresas e instituições particulares que desejem utilizar objetos adicionais à estrutura de gerenciamento em SNMP. Para isto, é necessário que a IAB delegue um número autorizado para identificar o fabricante e garantir que ele seja único. Para efeitos de implementação, utilizamos uma sub-árvore fictícia de nome GSAR (**Gerenciamento com Autenticação Remota**) de número 150. A seguir, descreveremos a estrutura utilizada para a implementação.

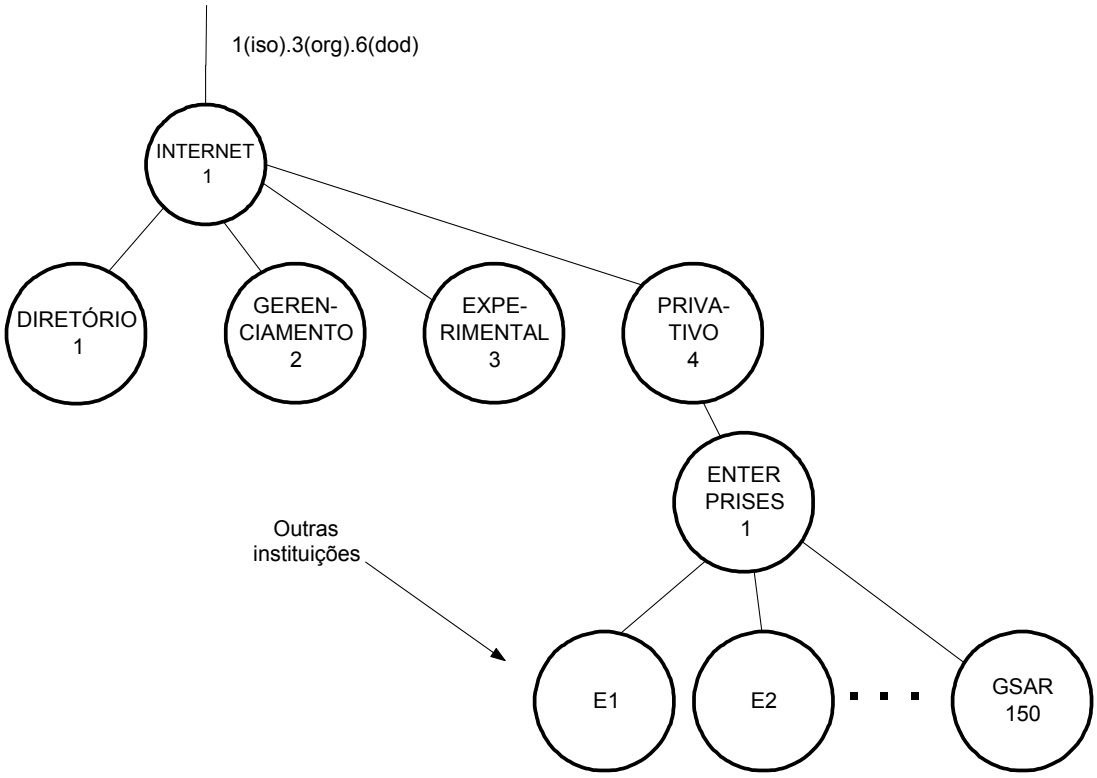


Figura 4.8: MIB complementar - Módulo privativo GSAR.

## 4.4.2.1 – Descrição da MIB Complementar

Na implementação da sub-árvore GSAR, utilizamos uma estrutura semelhante à que deveria ser utilizada por uma entidade particular em um caso real. Como descrito na Figura 4.9, abaixo do nó GSAR devemos criar uma outra sub-árvore UPS. Isto deve ser feito, pois, caso seja realizada outra implementação de um outro tipo de dispositivo, como por exemplo um retificador, o mesmo terá espaço para ser criado. Este cuidado deve ser tomado para garantir a organização e a estrutura hierárquicas da MIB. Abaixo do nó UPS criamos o nó VERS1 (indicando a versão 1 desta MIB) para ser usado em futuras evoluções do trabalho.

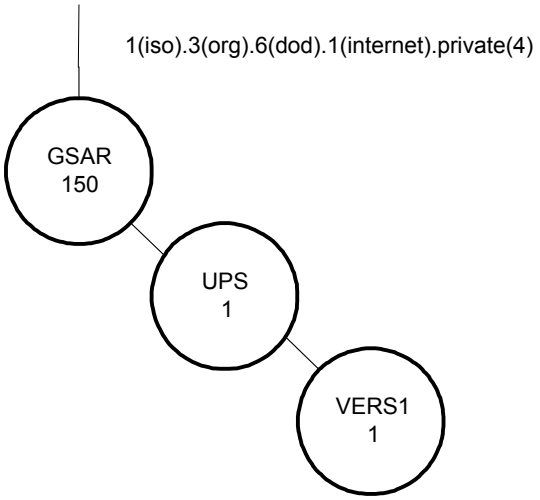


Figura 4.9: MIB complementar - Módulo privativo GSAR.

A estrutura da MIB complementar usa uma estrutura semelhante à da MIB RFC 1628, utilizando os mesmos grupos. Portanto, os objetos da MIB complementar terão o seguinte OID:

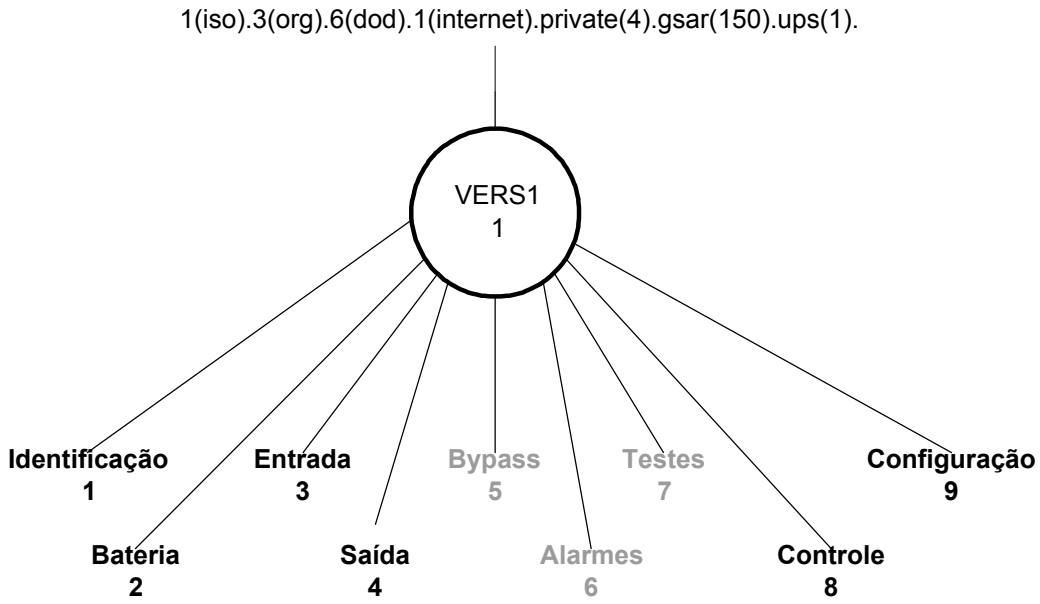


Figura 4.10: Estrutura da MIB complementar.

Para cada grupo implementado foi necessário definir formalmente, em ASN.1, os objetos adicionados, criando assim a definição formal da MIB complementar. Na Figura 4.10, temos em destaque os grupos implementados. Na Figura 4.11 temos a descrição formal em ASN.1 da MIB complementar. A descrição de cada módulo será vista a seguir.

```
upsMIBcomp MODULE-IDENTITY
    LAST-UPDATED "19072003"
    ORGANIZATION "GSAR – Mestrado – DETI - UFC"
    CONTACT-INFO
        "       Jarbas Silveira
                E-mail: jarbas@deti.ufc.br "
    DESCRIPTION
    "Este é um complemento da MIB RFC 1628 para o trabalho:
     Gerenciador SNMP com autenticação remota: aplicações em UPS "
   ::= { private 150 }
```

Figura 4.11: MIB complementar - Módulo identidade em ASN.1.

### 4.4.2.2 – Grupos Implementados

Como já afirmado, para cada grupo implementado, foi feita uma descrição formal em ASN.1. Nesta descrição estão presentes informações sobre a sintaxe do objeto, informando qual o tipo, acesso e descrição do mesmo. Na Figura 4.12 temos um modelo da descrição dos módulos.

```
-- Comentário sobre o grupo definido
Nome do Grupo        OBJECT IDENTIFIER::= { upsMIBCompObjects 1 }
Nome da Variável   OBJECT-TYPE
   SYNTAX     DisplayString (SIZE (0..31))
   MAX-ACCESS read-only
   STATUS     current
   DESCRIPTION
         " Descrever a variável implementada"
  ::= { upsIdent 1 }
```

Figura 4.12: modelo de definição de um grupo em ASN.1.

No grupo de identificação foi adicionada uma variável para indicar o número de série do produto. Este é um item muito importante em caso de manutenção, pois hoje a maioria dos fabricantes implanta informações de rastreamento, que tornam possível identificar se algum lote do produto teve algum material danificado na sua produção. Esta variável é a *upsIdentSerialNumber*.

Foi incluída uma variável no grupo de bateria, a *upsBatteryNominalVoltage*, para indicar qual a tensão nominal das baterias. Este dado é importante pois, em UPSs de maiores potências, normalmente acima de 5 KVA, utiliza-se um número elevado de baterias e, em caso de uma manutenção, é importante conhecer sua tensão nominal para se saber qual o número de elementos a utilizar.

A variável *upsInputPowerFactor* foi adicionada ao grupo de entrada, para indicar o fator de potência de entrada. Este dado usualmente está disponível em UPSs de maior potência, e indica em quanto a UPS está mantendo o fator de potência de entrada. Este dado é muito importante, pois um grande diferencial em UPSs hoje observado está relacionado à correção

ativa do fator de potência, que visa manter o fator de potência unitário, independente da carga (SANTIAGO, 2003; LEE, 1999).

No grupo de saída foi adicionada a variável *upsOutputPowerFactor* para medir o fator de potência de saída. Esta variável poderia ser calculada pelo gerente, pois, o mesmo já recebe os valores de potência em Watts e potência em VA. Escolheu-se incluir essa variável porque a UPS utilizada para testes já dispunha internamente desse valor.

Foram incluídas variáveis, no grupo de Shutdown, para uso na execução dos **Comandos de Shutdown Programados**, como descrito em (Manual do Solis, 2005). Para permitir gerenciar esta funcionalidade da UPS, incluímos a variável *upsWeekDay*, em que é possível programar qual a freqüência semanal de execução de comandos. As variáveis *upsTurnOnTime* e *upsTurnOffTime* indicam a hora de ligar e a hora de desligar a UPS, respectivamente. Foi incluída também uma variável de relógio interno da UPS, a *upsInternalTime*, para conter o relógio do sistema.

Ao grupo de configuração, foram adicionadas variáveis para habilitar, ou não, o recurso de desligamento por ausência de consumo, a *upsConfigDapac*, como descrito em (Manual do Solis, 2005). Outro grupo importante de variáveis adicionadas foi a *upsInputLowLimitFrequency* e a *upsInputLimitFrequency*, que são os valores mínimo e máximo, respectivamente, de aceitação de freqüência da alimentação de entrada. Estes valores são importante sobretudo quando do emprego de UPSs do tipo interativa, utilizadas com um grupo gerador (IEC 62040, 1999).

### 4.4.3 – Funcionalidades do Sistema

O sistema de gerenciamento aqui proposto, para efeitos de testes e validação, utilizou a implementação do agente SNMP compartilhando o computador presente na rede, como pode ser visto na Figura 4.13. Esta

solução divide a MIB em duas partes: uma no computador e a outra na UPS. O Agente, o Gerente e o Autenticador foram desenvolvidos em linguagem Delphi 3.0, utilizando um computador pessoal Pentium III 1GHZ, com 256 MB de memória RAM.

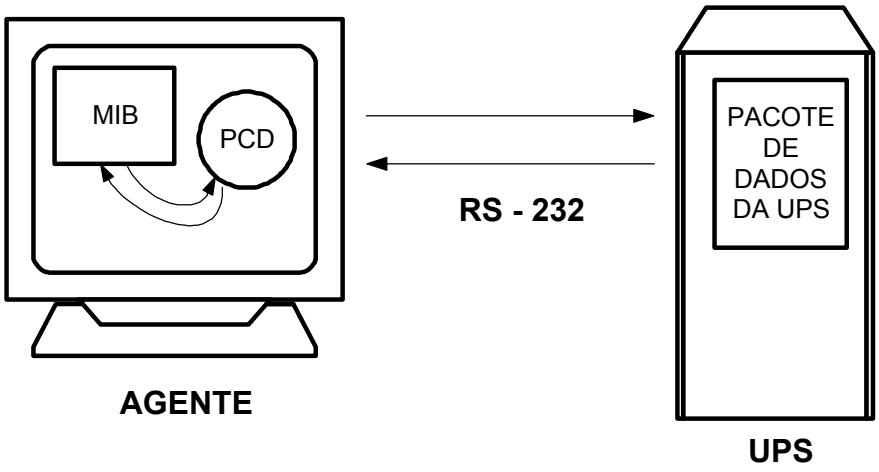


Figura 4.13: estrutura da MIB no sistema implementado.

A UPS utilizada não dispõe de recursos computacionais para ter uma MIB embarcada, devido à mesma não dispor de uma plataforma projetada para isso, pois, neste caso, normalmente precisaríamos de um hardware com mais poder de processamento (J. MA, 2005). Este fato obriga que a MIB, no formato em que a mesma deve ser implementada, esteja no computador, como visto na Figura 4.13. A UPS dispõe, internamente, de um pacote de dados em que são armazenadas as informações referentes ao seu funcionamento, como tensão de entrada, tensão de saída, dentre outros. A transmissão e recepção de pacotes de dados é realizada através de um programa residente no computador interagindo através de uma interface RS-232, através de um protocolo proprietário. Já existem protocolos abertos para a troca de mensagens entre as UPSs e softwares monitores, como o protocolo SEC (SEC Protocol, 1998) ou o PHOENIXTEC (PHOENIXTEC Protocol, 2001). Estes protocolos não foram implementados pelo fato da UPS utilizada não dispor de suporte para os mesmos. Esta funcionalidade pode ser adicionada ao sistema em trabalhos futuros, tornando assim a plataforma mais genérica.

Os dados recebidos pelo processo, chamado de PCD (**Processo de Coleta de Dados**), são então decodificados e armazenados na MIB, agora implementada no computador, para depois serem realizadas na mesma as operações de busca e armazenamento disponíveis em SNMP. Quando uma operação de armazenamento é feita, o mesmo processo PCD também é responsável por atualizar os dados na MIB que reside no computador, e realizar as operações na UPS.

Além das funções de agente SNMP, o software que reside no computador realiza outras funções inerentes a um software monitor de UPS e relativos à plataforma em que o mesmo se encontra.

### 4.4.3.1 - O Agente e suas funções

Devido às funções realizadas pelas UPSs inteligentes, o agente, além de desempenhar funções de recebimento, interpretação e resposta de pacotes SNMP, também executa funções essenciais para o funcionamento do sistema. Podemos citar algumas dessas funções, divididas aqui em 4 classes:

**Funções essenciais:** são funções que garantem a segurança com relação às aplicações que estão sendo executadas pelos computadores alimentados pela referida UPS. Dentre essas funções estão o encerramento do sistema operacional e o salvamento dos arquivos em alteração;

**Funções de sinalização:** são funções usadas principalmente para sinalizar aos usuários alterações sobre os dados referentes ao funcionamento da UPS em questão. Essas funções são: envio de mensagens instantâneas do estado da UPS na rede local e envio de mensagens do estado da UPS por correio eletrônico;

**Funções de Autoteste:** são funções que têm por objetivo realizar autotestes periódicos na UPS, com o objetivo de prever futuros defeitos que impedirão o bom funcionamento do sistema (DUARTE, 2002).

Dentre estas funções podemos exemplificar o autoteste de baterias. Embora estas funções possam ser efetuadas pelo gerente, elas são periodicamente executadas pelo agente para garantir sua execução mesmo na ausência do gerente;

**Funções de comunicação:** são funções referentes às primitivas de comunicação SNMP. Através destas funções o usuário, tem acesso aos dados do equipamento em monitoração.

Dentre os objetivos de implementação do agente aqui descrito, um dos mais importantes foi o de generalizar a plataforma de construção, para que o mesmo pudesse ser re-utilizado em outras implementações futuras de sistemas de gerenciamento de UPSs (dessa forma, outras funções podem ser incorporadas, facilitando a atualização de protocolos de comunicação) (KIM, 2002). Dentro desta expectativa, o agente foi dividido em camadas, visando uma melhor divisão de suas funções. Na Figura 4.14 temos uma descrição das camadas presentes no agente.

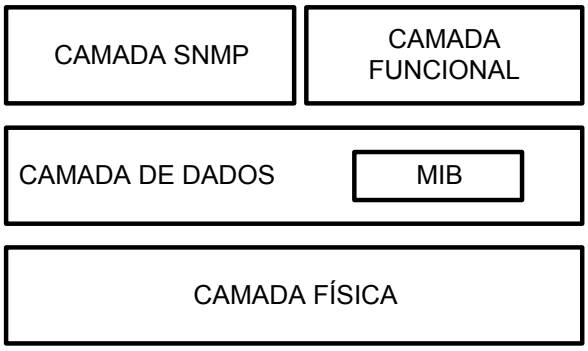


Figura 4.14: camadas funcionais do agente SNMP.

A camada física tem as funções de acesso à interface de dados da UPS, como por exemplo uma conexão RS-232 ou Ethernet. A camada física tem também por objetivo a correta recepção e/ou transmissão de dados, sendo responsável por uma retransmissão ou uma correção de dados, no caso da recepção de um pacote errado. Após a recepção desses dados, os mesmos são passados à camada imediatamente superior, a camada de dados. Nesta

camada, os dados são encapsulados no formato da MIB, para então serem enviados à camada SNMP, que tem a função de comunicação dos pacotes SNMP.

Estes dados também são enviados à camada funcional, que analisa esses dados, e dependendo deles, pode tomar decisões, que equivalem às funções essenciais ou de sinalização, conforme comentado anteriormente no tópico das classes de funções do agente. Esta implementação em camadas é bastante útil no caso de uma atualização de alguma funcionalidade do agente. Por exemplo, a adequação deste agente a uma UPS que tenha comunicação através de uma interface RS-485. Neste caso, somente a camada física seria mudada. O mesmo se aplica no caso do agente suportar o SNMPv3, sendo mudada somente a camada SNMP.

### 4.4.3.2 - O Gerente e suas funções

O gerente, na aplicação presente, pode assumir duas funções em aplicativos distintos, ambas com o mesmo objetivo de monitorar os objetos gerenciados. Isto decorre da necessidade do usuário do computador, em que reside o agente SNMP, e também da UPS conectada ao seu computador. Este gerenciamento, chamado neste trabalho de **Gerenciamento Local**, apresenta características diferentes das estações de gerenciamento SNMP, motivo pelo qual foram separados em duas aplicações distintas, embora obedecendo a um mesmo protocolo de comunicação, o SNMP. No gerente local, o usuário tem o objetivo de verificar dados relativos à UPS conectada ao seu computador, além de receber mensagens e alarmes somente referentes a ela. Por outro lado, nas estações de gerenciamento SNMP convencionais, vários agentes precisam ser monitorados, necessitando de uma interface que tenha uma estrutura capaz de gerenciar um número arbitrário de agentes, necessitando portanto de suporte a essas funções (BIVENS, 2004).

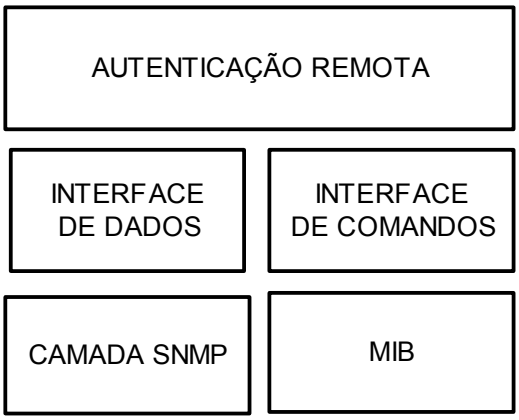


Figura 4.15: camadas funcionais do gerente SNMP.

Para a estação de gerenciamento, que pode comandar diversos agentes, neste trabalho chamado de **Gerente Corporativo**, foram implementadas algumas funcionalidades para dar suporte a mais de um agente gerenciados. Na Figura 4.15 apresentamos um diagrama de blocos funcional desse software. Nele podemos visualizar a presença das camadas de interface de dados e comandos, que permite ao usuário a visualização dos dados dos objetos gerenciados ou a alteração desses mesmos, respectivamente. Esta interface primou pela hierarquia de monitoração dos agentes. Neste caso, podemos agrupar os agentes em grupos e sub-grupos, podendo existir vários níveis de organização para que eles sejam melhor visualizados, como mostrado na Figura 4.16. Quando os alarmes são recebidos pelo gerente todo o ramo referente à sub-árvore pertinente é visualizado da presença de um alarme.


Figura 4.16: gerente SNMP Corporativo.

### 4.4.3.3 - O Autenticador de Gerentes

A outra funcionalidade desta implementação primou pela expansão da funcionalidade das comunidades em SNMP. Como já discutido no Capítulo 3, é possível segregar determinado número de agentes em menores porções, denominadas **Comunidades**. No trabalho aqui descrito, propomos que uma só **Comunidade** seja usada, e que a função de dividir esses agentes esteja embutida no próprio gerente. Assim, cada usuário, através de uma identificação, recebe uma árvore de gerenciamento contendo somente os agentes que lhes são permitidos gerenciar, além de ser possível dar permissões a cada usuário, podendo ainda executar comandos ou somente visualização de dados. Para isto, criamos um elemento chamado **Autenticador de Gerentes**, que representa um aplicativo com uma base de dados em que é possível autenticar cada gerente e liberar as permissões pertinentes a cada um deles.

Na Figura 4.17 é ilustrada esta funcionalidade que, como dissemos, realiza a função de centralizar as permissões de gerenciamento através do conceito de comunidade em gerentes SNMP. Entretanto, esta funcionalidade da implementação, não corrige a deficiência de segurança existente nas versões do protocolo SNMPv1 e SNMPv2, pois, o acesso aos agentes ainda se torna vulnerável, como descrito em (HIA, 2001). Para isto dispomos também da opção, pelo administrador de rede, de inibir a utilização das mensagens de Set-Request, no sentido de evitar ataques prejudiciais ao sistema (FIANG, 2002).

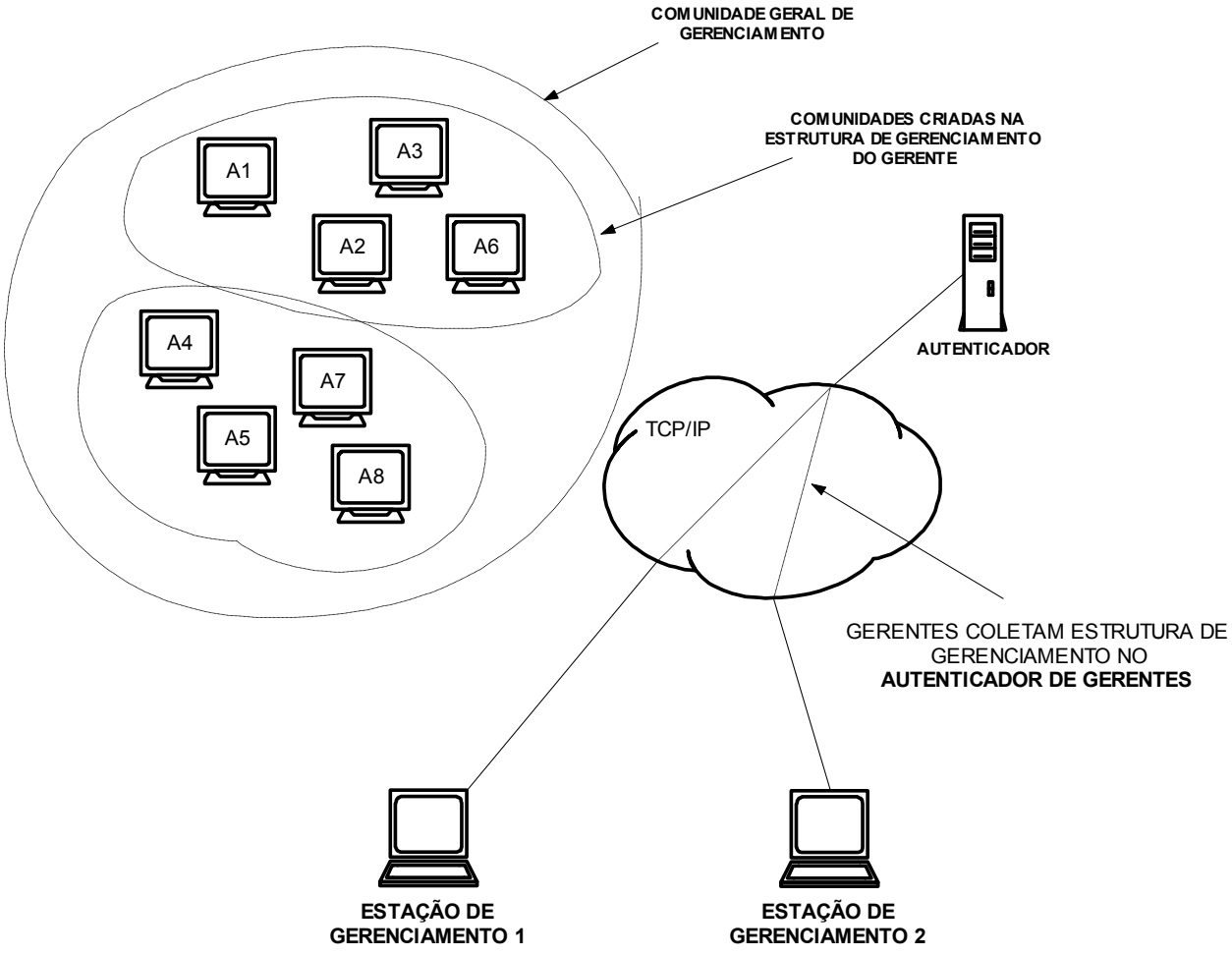


Figura 4.17: a funcionalidade do Autenticador.

### 4.4.4 – Operações do Sistema

As operações do sistema de gerenciamento SNMP aqui descritas, serão restritas ao uso do protocolo SNMP, pois o restante se refere às operações do sistema, e incorpora muitas particularidades do sistema operacional da plataforma de implementação. Portanto, iremos descrever as operações básicas do SNMP, que são as operações de busca, armazenamento e os alarmes, representados em forma de mensagens ou conjunto de mensagens.

### 4.4.4.1 – A leitura de recursos do objeto gerenciado

A operação da leitura dos recursos em um objeto gerenciado é realizada através da leitura de variáveis da MIB. Esta operação de leitura se divide em duas fases. A primeira com origem no gerente e destino no agente, através do envio da mensagem Get-Request. A segunda fase tem origem no agente e como destino o gerente que solicitou o pedido de leitura.

O ciclo de leitura representa uma ação contínua por parte do gerente e, dependendo do número de agentes, pode comprometer o seu funcionamento, necessitando de um poder computacional proporcional ao número de agentes. Para minimizar este problema, as operações de leitura no gerente foram implementadas conforme descrito em (STALLINGS, 1999).

Na inicialização do gerente, todos os agentes presentes em sua árvore de gerenciamento são solicitados para responder com todas as informações disponíveis na estrutura de gerenciamento. Este ciclo inicial serve para identificar quais os agentes presentes na rede. Após este ciclo inicial, somente o agente que está destacado na tela de gerenciamento é questionado sobre seus recursos, enquanto os outros só serão inquiridos em intervalos regulares de quinze minutos, quando uma nova leitura em todos os recursos é executada. Caso algum alarme, de algum agente, seja enviado, este recebe

destaque (atenção especial), passando a ser questionado continuamente. No item 4.5 é mostrado como estas operações foram validadas através de testes, e é feita uma análise sobre quantos agentes o gerente em estudo poderá gerenciar.

### 4.4.4.2 – A escrita nos recursos do objeto gerenciado

A operação de escrita nos recursos do objeto gerenciado corresponde a uma operação de armazenamento de variáveis em SNMP. Na maioria das vezes, esta operação de armazenamento corresponde à execução de um comando, ou seja, muda-se uma variável na MIB do objeto gerenciado para que uma operação seja realizada no agente. Outra possibilidade seria o armazenamento de variáveis para mudar o acesso a outras variáveis, ou ainda para que limites de aceitação de parâmetros sejam alterados. Como exemplo podemos citar a variável *upsConfigLowBattTime*, que controla qual o ponto de descarga da bateria é aceita como bateria baixa. Este limite tem influência no envio de alarmes.

Para a implementação aqui proposta, utilizamos a operação de Set-Request robusta como proposta em (MURRAY, 1998). Nesta operação, devemos inicialmente nos certificar do valor da variável, para depois escrevermos na mesma e, em seguida, verificar através de outro Get-Request se a operação foi realmente realizada. Esta estratégia é importante, pois muitas vezes, dependendo da maneira como é implementada a comunicação entre o software agente e o dispositivo de comunicação, a segurança desta operação pode não estar garantida em UPSs. Para tornar o nosso gerente SNMP independente das falhas do agente, esta redundância na certificação da escrita de variáveis torna-se bastante desejável.

A operação de escrita em SNMP, devido à falta de segurança, pode tornar-se proibitiva (LEE, 2004). Este fato se deve, muitas vezes, ao nível crítico dessas operações, como no caso de uma UPS, em que é possível,

através da escrita em uma variável, executar comandos como desligar sua saída e levar todas as cargas ao desligamento. Na implementação aqui proposta, as operações foram implementadas, mas elas podem ser desabilitadas no próprio agente, via configuração de inicialização do sistema. A operação de escrita só é autorizada para os usuários com nível máximo de acesso, que é ativada no Autenticador de gerentes.

### 4.4.4.3 – Os alarmes do objeto gerenciado

Os alarmes representam uma forma de comunicação assíncrona entre o agente e o gerente. Isso porque o estímulo desse evento parte do agente, embora a condição de atuação possa ser configurada pelo gerente. Nesta proposta, os alarmes têm uma importância muito grande para o funcionamento do sistema. Como um grande número de agentes pode estar ativo no sistema, o gerente realiza leituras contínuas somente nos agentes em destaque, ou que se encontram em situação de alerta. E essa mesma situação de alerta é habilitada via envio de mensagens.

Outra função importante dos alarmes é na inicialização dos agentes. Quando o gerente é inicializado, todos os agentes são questionados sobre sua existência. Os que não respondem somente serão questionados após um tempo definido no gerente. Enquanto esta temporização não é completada, caso um agente seja iniciado, um Trap, chamado *ColdStart*, é enviado ao gerente. Assim, o agente é ativado antes do momento de leitura, sinalizando ao gerente sua presença na rede. Este Trap também é necessário para que o agente seja reconfigurado, pois, em muitos sistemas, as configurações presentes no agente são armazenadas em memória não volátil. Isto resulta na perda dos dados, necessitando assim que esses dados sejam novamente carregados.

Todos os alarmes são armazenados em um banco de dados, sendo possível ao usuário do software de gerência SNMP construir relatórios para análise de falhas da rede elétrica em que as UPSs estão instaladas.

## 4.5 – Testes e Validação

Os testes realizados aqui têm o objetivo de demonstrar a compatibilidade das ferramentas desenvolvidas com o protocolo SNMP, bem como o desempenho da ferramenta para gerenciar UPSs. Realizamos testes nas três operações de SNMP:

- a leitura de dados, através de pares de comandos Get-Request e Get-Response;
- a escrita de dados através de pares de comandos Set-Request e Get-Response;
- Envio de alarme através de mensagens SNMP Trap.

Por último, foi feita uma análise de desempenho, considerando a realização das operações aqui descritas, para estimarmos qual o número de agentes que o sistema é capaz de gerenciar.

Para fazermos a validação das mensagens SNMP utilizamos um software de análise de protocolo. Este software, o "EtherReal", é capaz de identificar qual a versão do protocolo utilizado, bem como qual o tipo de mensagem SNMP, além de oferecer um excelente ambiente gráfico de visualização do pacote de dados (ETHEREAL, 2005).

### 4.5.1 – Operação de leitura

Para validar uma operação de leitura, realizamos uma operação de leitura na variável com o seguinte OID:

*1 . 3 . 6 . 1 . 2 . 1 . 3 . 3 . 1 . 1 . 7*

Como pode ser visto na Figura 4.18, através da ferramenta de análise de protocolo, o pacote foi identificado como sendo um pacote do tipo SNMP de versão 1, mensagem tipo Get-Request. O Identificador de requisição

não foi utilizado, sendo enviado como nulo. No pacote também vemos o OID da variável requisitada. Podemos ver também a mensagem codificada, em que é possível identificar os campos representando as informações decodificadas.

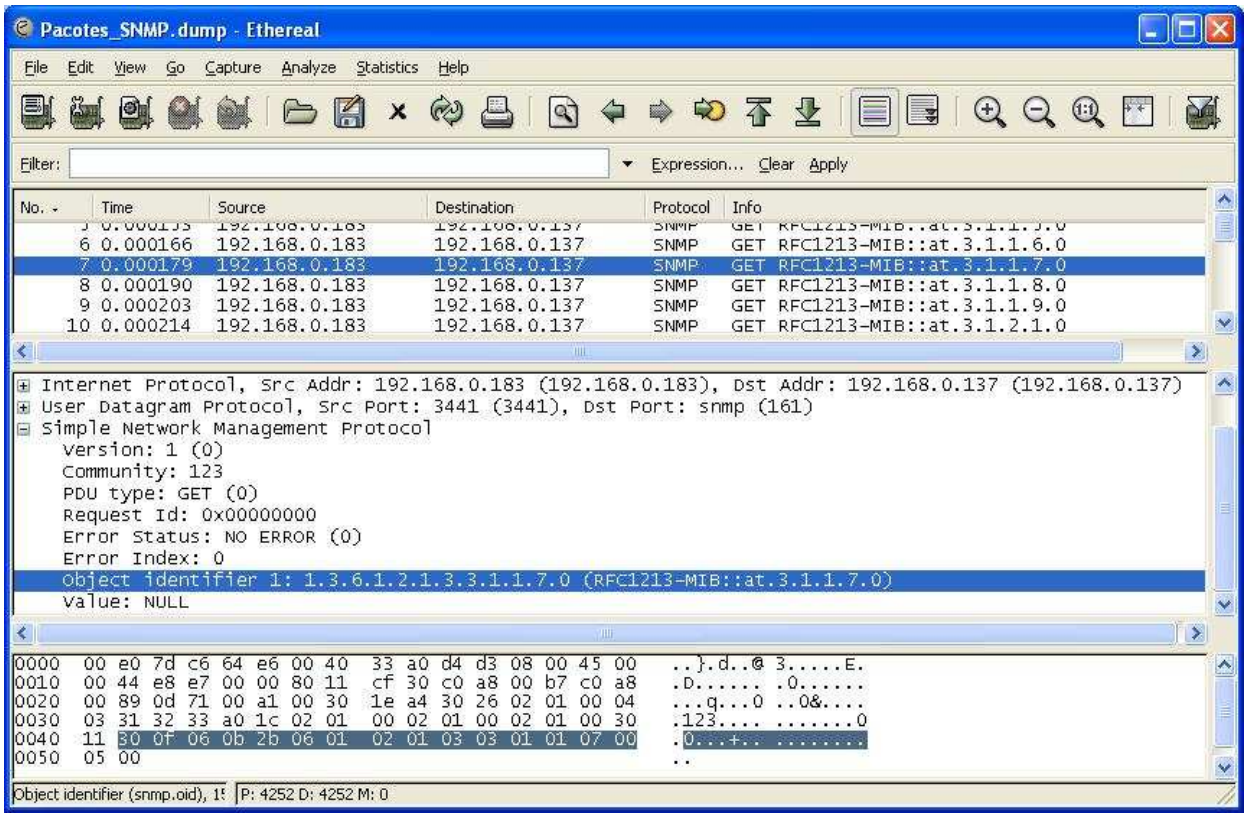


Figura 4.18: mensagem SNMP Get-Request.

Uma mensagem de resposta do agente foi capturada, como mostrada na Figura 4.19. Nesta mensagem foi identificado o tipo da mensagem, como sendo Get-Response. Mais uma vez o campo Identificador do Requisitante não foi utilizado. Nenhuma operação de erro também foi verificada na mensagem. O OID da variável também foi recebido corretamente, sendo semelhante ao OID da mensagem de Get-Request. Por fim, o valor da variável foi recebido, sendo do tipo STRING, apresentando o valor “123456”.

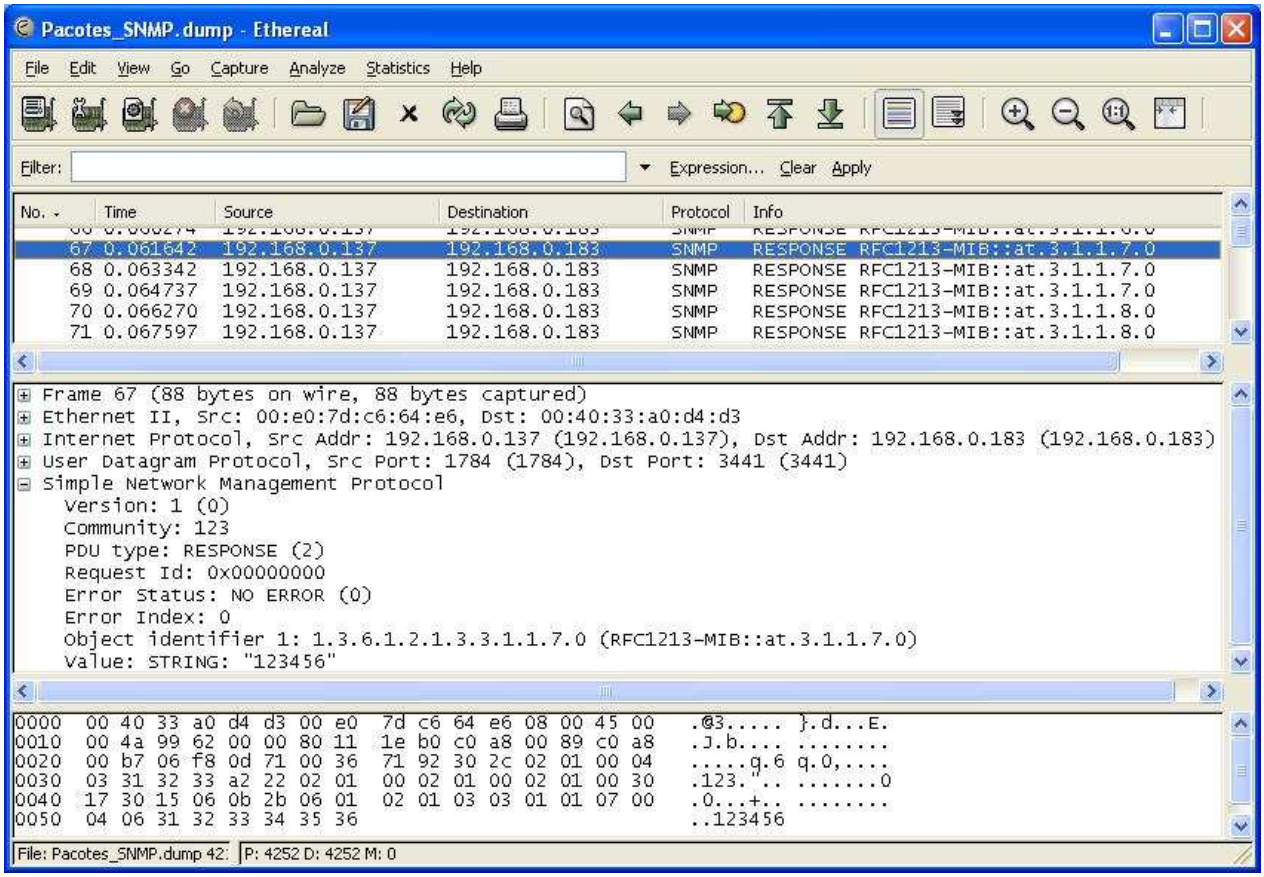


Figura 4.19: uma mensagem SNMP Get-Response.

## 4.5.2 – Operação de Escrita

Para validarmos a operação de escrita realizamos uma operação de escrita na variável do seguinte OID:

*1 . 3 . 6 . 1 . 2 . 1 . 3. 3 . 1 . 8 . 9*

Como pode ser visto na Figura 4.20, o pacote foi identificado corretamente como sendo do tipo Set-Request. Nesta operação, o campo de identificação do requisitante também não foi utilizado. Em seguida vemos o OID da variável em questão com o seu valor de escrita.

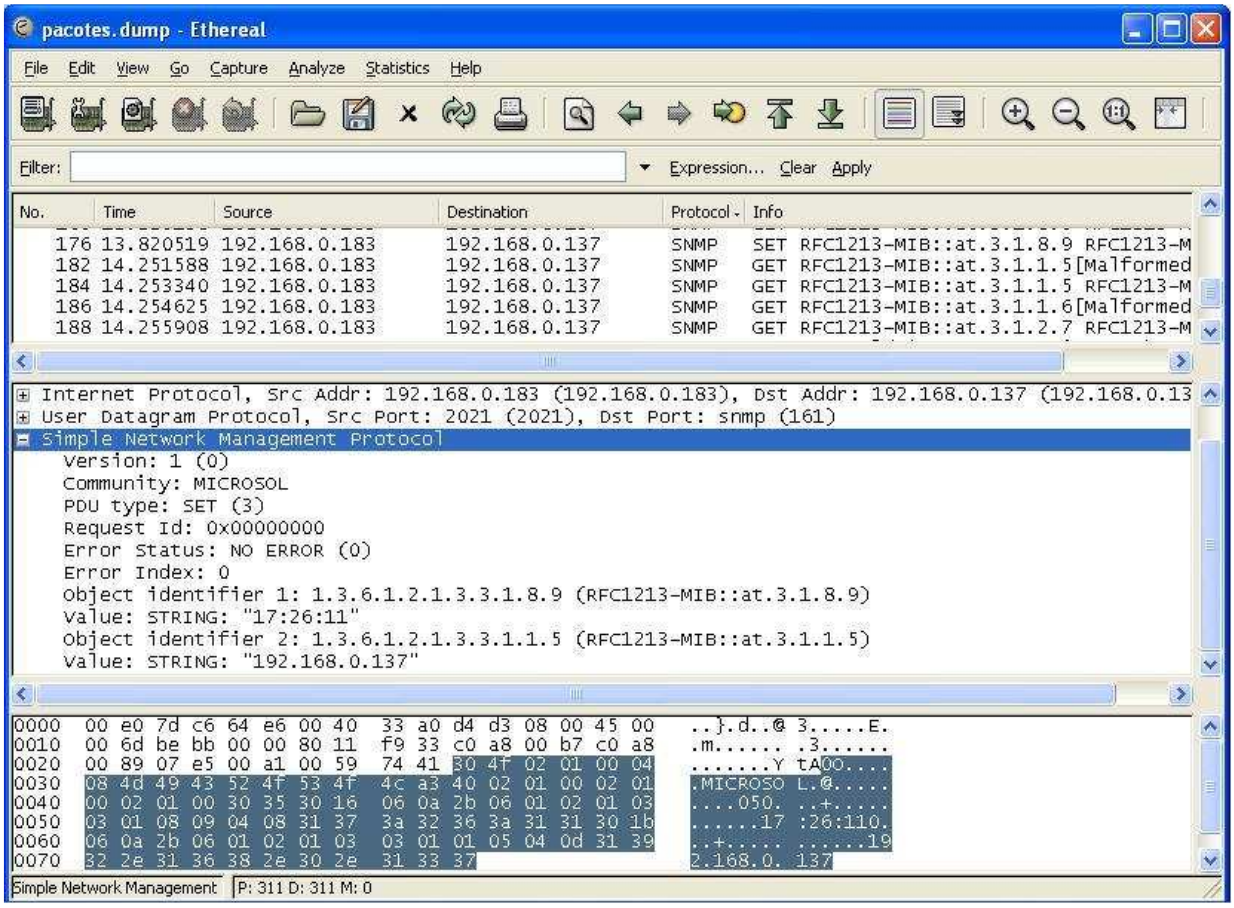


Figura 4.20: uma mensagem SNMP Set-Request.

## 4.5.3 – Alarmes

Para validarmos a recepção de uma mensagem de alerta em SNMP, provocamos a ocorrência de um alarme. O mesmo agente enviou uma mensagem de Trap. Como pode ser visto na Figura 4.21, a mensagem foi identificada como sendo uma mensagem do tipo Trap e também como sendo uma mensagem de Trap do tipo Específico, ou seja, apresenta valor 6. O campo seguinte, de Tipo de Trap específico, apresenta o valor que identifica o Trap específico, sendo de valor 3, como conferido.

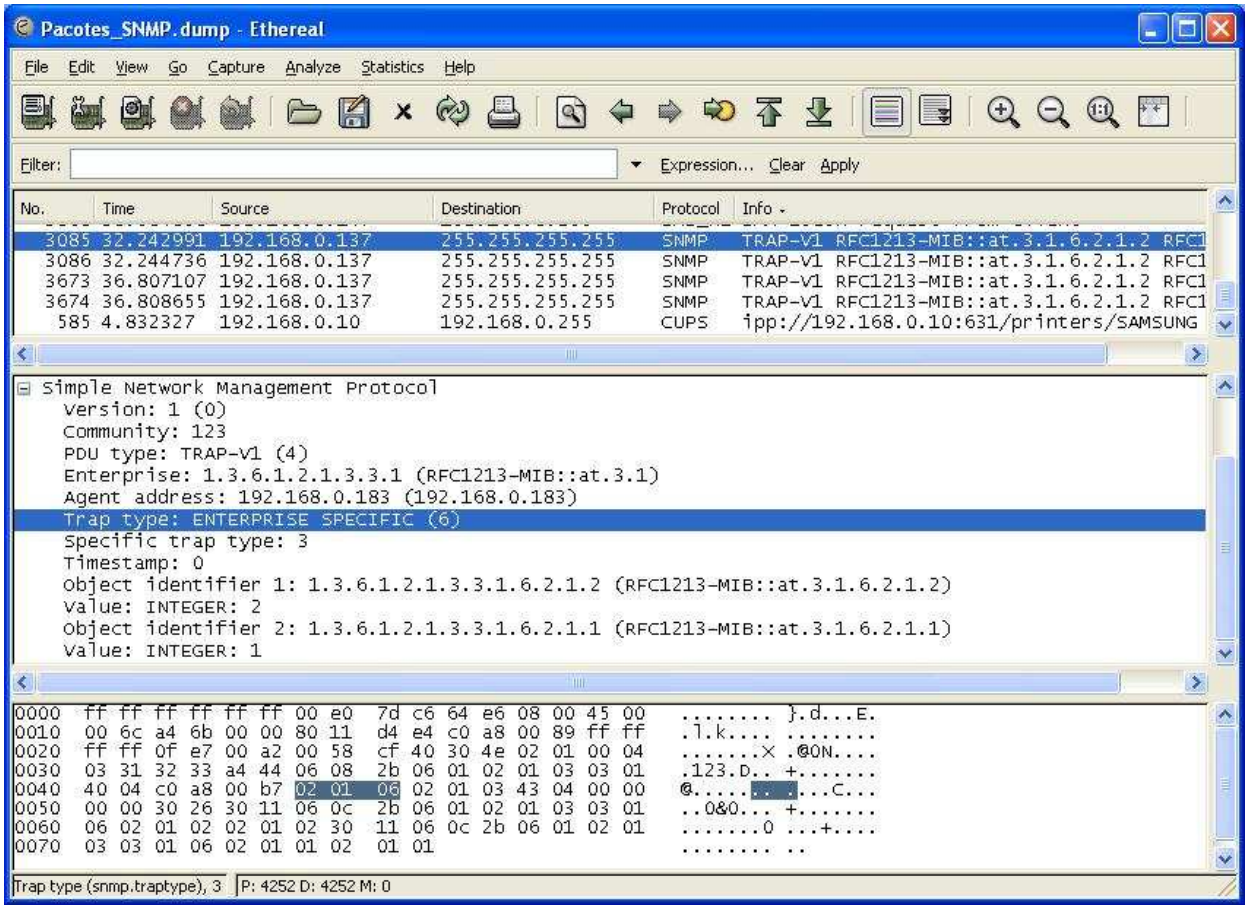


Figura 4.21: uma mensagem SNMP Trap.

## 4.5.4 – Métrica de quantidade de estações a serem gerenciadas

Conforme discutido no Capítulo 3, adotou-se uma métrica que nos informa qual a quantidade de estações que o sistema pode gerenciar. Esta métrica não pode ser adotada como medida de desempenho, uma vez que para que isto seja feito, vários fatores deveriam ser considerados, como capacidade de processamento da estação de gerenciamento, o tráfego nas redes e sub-redes presentes no sistema e o nível de congestionamento na rede, dentre outros fatores (LIU, 2003).

O seguinte exemplo é fornecido em (STALLINGS, 1999 *apud* Bem-Artzi, Chandna, e Warrier 1990). O exemplo consiste de uma simples LAN, em que cada dispositivo deve ser monitorado a cada 15 minutos, uma freqüência

típica em ambientes de monitoração de redes. No exemplo, foi assumido um tempo típico de processamento de pacotes de rede de 50 ms e um delay de rede 1 ms, totalizando um tempo aproximado de 202 ms, portanto, o número de estações a serem gerenciados será:

$$N \leq \frac{15 \times 60}{0,202}, N \leq 4450$$

Para o cálculo do número de estações que o nosso sistema proposto pode gerenciar, utilizamos a medida de tempo disponível no analisador de protocolo utilizado. Podemos observar na Figura 4.18, em que foi executada uma leitura de uma variável do grupo de identificação. Considerando-se que as principais variáveis da MIB RFC 1628 que deveriam ser lidas na aplicação implementada, sejam o valor da Tensão de Entrada, Tensão de Bateria, Tensão de Saída, e Potência de Saída. Podemos concluir que, tipicamente, após o quarto ciclo de leitura de variáveis, teremos o tempo para a leitura de um ciclo completo das variáveis essenciais para o gerenciamento de uma UPS. Como visto nos pacotes SNMP capturados, obteve-se um tempo médio de 1800 ms, entre os pacotes de get-request e get-response. Portanto, utilizando o cálculo sugerido em (STALLINGS, 1999) e como explanado no Capítulo 3, teremos o seguinte cálculo:

$$N \leq \frac{15 \times 60}{1,800}, N \leq 500$$

Portanto, considerando um tempo típico de varredura de 15 minutos entre cada dispositivo gerenciado, estando a estação de gerenciamento dedicada somente para este fim, poderemos monitorar um número de até 500 UPS no sistema de gerenciamento proposto. Este número pode variar, dependendo de como os pacotes são montados, qual a seqüência de dados a ser lida, bem como o poder de processamento das estações e o tráfego da rede.

## 4.6 – Considerações Finais

A aplicação aqui proposta teve como objetivo verificar a funcionalidade do protocolo SNMP no uso de gerência de UPS como um elemento de rede. Implementamos também o recurso do **Autenticador de Gerentes**, que estendeu o conceito das "Comunidades" em SNMP. Para isso, utilizamos um agente SNMP embarcado em computador pessoal que se comunicava com a UPS via RS-232.

Nesta implementação, foi possível ressaltar as principais operações do protocolo SNMP, sendo possível efetuar monitoração das UPS, através das mensagens Get-Request e Get-Response; controlar as UPS através das mensagens Set-Request e receber alarmes das UPS através das mensagens Trap. Através destas operações do protocolo SNMP, foi possível validar a compatibilidade do protocolo através de uma ferramenta de análise de protocolo de redes. Foi possível testar a capacidade de extensão da estrutura de dados da MIB, pois, a UPS utilizada continha alguns dados ainda não dispostos da MIB RFC 1628, possibilitando assim inserção de novos dados.

Através da topologia utilizada foi possível também testar a funcionalidade do **Autenticador de Gerentes**, observando assim suas facilidades na utilização de múltiplos gerentes para gerenciar um maior número de UPSs. Por último, foi possível realizar uma métrica para cálculo da quantidade de UPSs que o sistema de gerenciamento poderia gerenciar. Acreditamos que a implementação e os testes realizados foram de muita importância para a avaliação da aplicação e compatibilidade do sistema aqui proposto.

# Capítulo 5

# 5. Conclusões e trabalhos futuros

## 5.1 – Conclusão

A implementação aqui proposta, descrita no Capítulo 4, bem como o estudo sobre o protocolo de gerenciamento utilizado, nos permitiu enumerar algumas vantagens e desvantagens sobre a utilização de **Gerenciamento SNMP com Autenticação Remota**, aplicado a UPSs. Como principais vantagens da utilização do protocolo, podemos enumerar:

- Compatibilidade com outras ferramentas de gerenciamento, transformando-se numa grande vantagem para os usuários de UPS, pois, o sistema permite utilizar as ferramentas já existentes. Além disso ele garante a utilização destas UPS com outras compatíveis com o protocolo de gerenciamento SNMP, e permite eles compartilharem o mesmo sistema de gerenciamento;

- O sistema proposto permitiu gerenciar, num mesmo ambiente, múltiplas UPS, constituindo-se de um ambiente integrado de gerenciamento;

- Flexibilidade nas ferramentas de gerenciamento, sendo possível gerenciar localmente uma UPS ou gerenciar um número arbitrário de UPS, dando flexibilidade quanto à centralização ou não do sistema de gerenciamento;

- O sistema proposto oferece as operações necessárias para efetuar monitoração e controle dos dispositivos gerenciados;

- A base de dados, embora simples, constituiu-se de boa aceitação, sendo flexível nos vários tipos de dados, inclusive na inclusão de novos recursos de gerenciamento;

- Foi possível estender o conceito das comunidades em SNMP. Através do Autenticador de Gerentes, possibilitando numa facilidade a mais para os administradores de rede controlarem o acesso às ferramentas de gerenciamento, bem como melhor selecionar os diversos ambientes de gerenciamento em uma situação de um maior número de objetos gerenciados.

Como principais desvantagens do protocolo, podemos enumerar:

- A segurança, característica essa não suportada pelo protocolo SNMP, torna a ferramenta de gerenciamento incompleta, pois para corrigir este problema, os administradores limitam algumas operações de controle, como ligamento e desligamento das UPSs, para evitarem assim maiores riscos ao sistema em caso de um ataque (CHATZIMISIOS, 2004);

- Necessidade de maior processamento no gerente, caso haja uma necessidade de gerenciar um maior número de UPSs. Esta desvantagem pode ser observada no cálculo do número de UPSs que o sistema pode gerenciar, mostrado no Capítulo 4. Observamos uma significativa diminuição do número de objetos que podem ser gerenciados, devido ao baixo poder de processamento do gerente. Esta situação pode ser contornada, aumentando-se o poder computacional do gerente ou ainda através de uma gerência distribuída, utilizando–se de vários gerentes;

- Por último observamos uma baixa portabilidade na ferramenta de gerenciamento, que no caso do administrador de rede não se encontrar na sala de gerência de redes, fica impossibilitado de efetuar qualquer operação de gerenciamento, pois, torna-se necessário a utilização das ferramentas de gerenciamento.

Analisando as vantagens e desvantagens do sistema de **Gerenciamento SNMP com autenticação remota: aplicações em UPSs**, concluímos que o mesmo adequou-se aos objetivos da pesquisa, mostrando-se um sistema compatível com protocolos de gerenciamento já existentes, capaz de gerenciar um número arbitrário de UPS. Podemos ressaltar como méritos do sistema aqui desenvolvido a extrema flexibilidade da estrutura das informações de gerenciamento, bem como a simplicidade das operações de gerenciamento. A seguir, proporemos algumas sugestões de trabalhos futuros visando contornar as desvantagens aqui encontradas.

## 5.2 - Trabalhos futuros

Como discutido no Capítulo 4, onde várias soluções foram estudadas para a implementação desta pesquisa, algumas soluções apresentaram vantagens, enquanto outras apresentaram desvantagens. A implementação descrita neste trabalho relevou como principais pontos a possibilidade de gerenciar vários elementos de rede, a compatibilidade com outras ferramentas de gerenciamento, bem como o poder de gerenciamento obtido através da utilização do protocolo SNMP. Uma característica bastante importante em uma ferramenta de gerenciamento é a sua portabilidade. Esta mesma ganhou bastante funcionalidade com a Internet, pois com advento dos navegadores, um aplicativo pode estar presente em qualquer computador que esteja ligado à rede mundial de computadores (J. MA, 2005).

Portanto, seria uma grande contribuição se acrescentássemos à plataforma de gerenciamento aqui desenvolvida, uma característica de portabilidade. Para isso propomos uma implementação do estudo aqui descrito, utilizando uma solução com utilização da portabilidade do HTTP unida ao poder de gerenciamento que oferece SNMP, como foi apresentado em (HONG, 2001), onde é proposto um sistema de gerenciamento SNMP para aplicações embarcadas, com integração do sistema de gerenciamento através da WEB (navegador).

Esta solução seria possível se desenvolvêssemos o sistema de gerenciamento SNMP para UPSs em um gerente, onde nele estaria presente toda a rede de gerenciamento do sistema, e dele partiriam todos os pedidos de dados de gerenciamento, bem como nele chegariam todos os dados de alarme que fossem provenientes dos agentes SNMP das comunidades a ele ligados. Neste gerente seria disponibilizada uma interface de gerenciamento que poderia ser acessada pelos administradores de rede, representando a interface homem-máquina de gerenciamento do sistema. Através desta interface seria possível realizar as ações de gerenciamento, visualizar os dados dos objetos

gerenciados, bem como receber os alarmes dos agentes (KIM, 2002). Na Figura 5.1 são mostrados os elementos presentes nesta proposta. É importante salientar, que como todo o processamento de gerência seria feito neste computador, o mesmo deveria apresentar um alto poder computacional, pois caso contrário tornar-se-ia um causador de baixo desempenho do sistema.

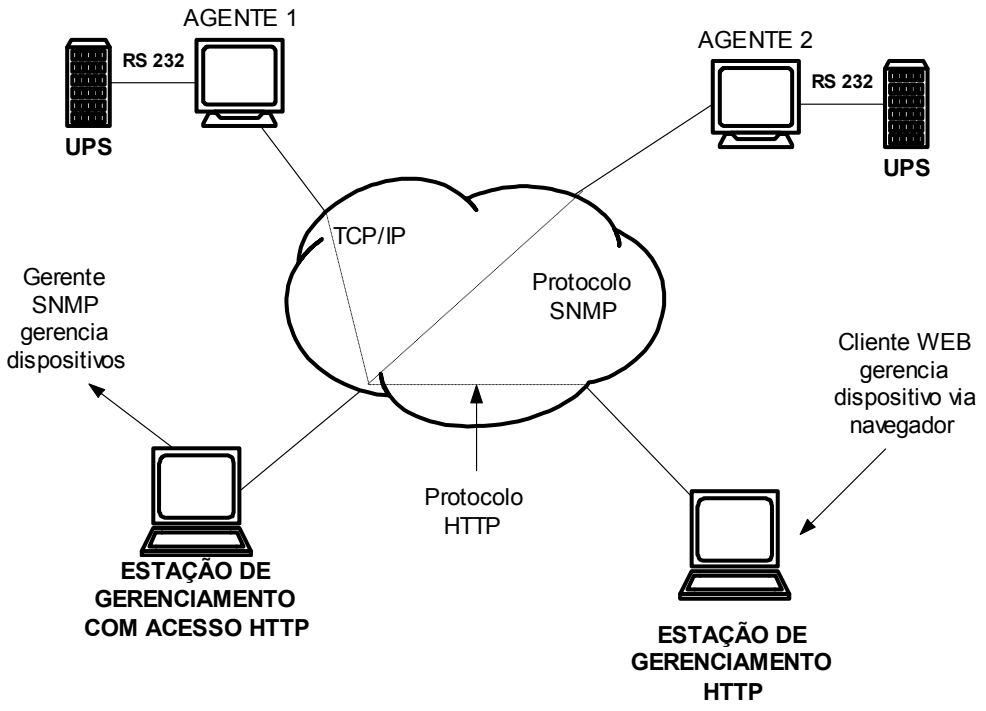


Figura 5.1: gerenciamento SNMP com acesso via HTTP

Através desta implementação, poderíamos manter as mesmas funcionalidades do sistema desenvolvido em SNMP, disponibilizando os agentes para participarem da comunidade de gerenciamento de algum sistema de gerência de rede já existente, além de manter a base de dados com as mesmas características já observadas em SNMP. Além disso, poderíamos também disponibilizar a vantagem do gerenciamento portátil através da interface HTTP.

# Referências

(AICKLEN, 1995) - AICKLEN, G. H; MAIN, P. M. *Remote Control of Diverse Network Elements Using SNMP*, IEEE: 1995.

(AMIRTHALINGAM, 1995) - AMIRTHALINGAM, K; MOORHEAD, Robert J. SNMP – *An Overview of its Merits and Demerits*. IEEE, 1995.

(APOSTOLOPOULUS, 1995) - APOSTOLOPOULUS, Theodore K; DASKALOU, Victoria C. *A Model for SNMP Based Performance Management Services*. IEEE: 1995.

(AUGUSTO, 2001) - AUGUSTO, V; Elizabeth, S. *Discriminador de Repasse de Eventos em Ambientes SNMP*. UFSC: Departamento de Informática e Estatística, 2001.

(BALACHANDRA, 2000) – Balachandra, John C.; Bialek, Thomas; Gravely, Michael; Weaver, Eugene. *Test Site and Methodologies for Testing and Comparing Energy Storage Systems for UPS, Load Management and Power Quality Applications.* IEEE: 2000.

(BENTLEY, 1993) - BENTLEY, Peter. *UPS and downs*. Standby Power Suplement. Fiskars Power Systems, Novembro de 1993.

(BIVENS, 2004) - Bivens, Alan; Gupta, Rashim; McLean, Ingo; Szymanski, Boleslaw; White, Jerome. *Scalability and performance of an agent-based network management middleware*. John Wiley & Sons Ltd. INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT: 2004.

(BREITGAND, 2001) – Breitgand; Shavitt, Y. Raz. *SNMP GetPrev - an efficient way to browse large MIB tables*. IEEE: 2001.

(BRISA, 1993) - BRISA. *Gerenciamento de Redes Uma Abordagem de Sistemas Abertos*. São Paulo – SP: MAKRON Books do Brasil, 1993.

(CARDOSO, 1998) - CARDOSO, Paulo F; MONTEIRO, João L. *SNMP and Industrial Networks*. IEEE: 1998.

(CASE, 1989) - CASE, J; FEDOR, M; SCHOFFSTALL, M; DAVIN, C. *A Simple Network Management Protocol (SNMP),* RFC 1098, April 1989.
< http://www.ietf.org/rfc/rfc1098.txt> Último acesso em 25/04/2006.

(CASE, 1990) - CASE, J; FEDOR, M; SCHOFFSTALL, M; at al. *A Simple Network Management Protocol (SNMP)*, RFC 1157: USA, 1990.
< http://www.ietf.org/rfc/rfc1157.txt> Último acesso em 25/04/2006.

(CASE, 1994) - CASE, J. *UPS Management Information Base*, RFC 1628: USA, 1994.
< http://www.ietf.org/rfc/rfc1628.txt> Último acesso em 25/04/2006.

(CASTRO, 1992) - CASTRO, Helano de Sousa. *Fault Tolerance Through Reconfigurability: Applications in Space Instrumentation.* University of Sussex, SUSSEX, Inglaterra, 1992.

(CHATZIMISIOS, 2004) – Chatzimisios, Periklis. *Security issues and vulnerabilities of the SNMP protocol*. IEEE: 2004.

(CHEIKHROUHOU, 2002) - CHEIKHROUHOU, M; LABETOULLE, J. *An Efficient Polling Layer for SNMP*. IEEE, 2002.

(CHUN, 2002) – Chun, Jae-Kyu; Cho, Ki-Yong; Cho, Seok-Hyung; Lee, Young-Woo; Kim, Young-Woo. *Network Management Based On PC Communication Platform With SNMP AND Mobile Agents*. IEEE: 2002.

(COMER, 1991) - COMER, E. D. *Internetworking with TCP/IP Vol I: Principles, Protocols, and Architecture*. Second Edition. USA: Prentice-Hall International Inc, 1991.

(COMER, 1994) - COMER, E. D. *Internetworking with TCP/IP Vol II: Design, Implementation, and Internals.* Second Edition. USA: Prentice-Hall International Inc, 1994.

(CURY, 2000) - CURY, Raphael S; SILVA, Ana P. R; LOUREIRO, Antonio A. F; NOGUEIRA, José M. S. *Gerência de Desempenho e Configuração de uma Rede de Alta Velocidade através do Sistema Integrado de Supervisão*. UFMG – DCC: 2000.

(DUARTE, 2001) - DUARTE, Elias P. Jr; SANTOS, Aldri L. *Network Fault Management Bades on SNMP Agent Groups*. IEEE: 2001.

(DUARTE, 2002) – Duarte, Elias Procopio Jr.; De Bona, Luis Carlos Erpen. *A Dependable SNMP-based Tool for Distributed Network Management*. IEEE: 2002.

(ETHEREAL, 2005) – Ethereal, Network Protocol Analyzer. Versão 0.10.11(C), 2005.<http://www.ethereal.com/download.html> Último acesso em 25/04/2006.

(FERNANDEZ, 2001) - FERNANDEZ, Marcial P; DELFINO, Gardel M; PEDROZA, Aloysio de Castro. *Protocolo para Gerenciamento Hierárquico de Redes de Computadores e de Telecomunicações*. UFRJ – COPPE/PEE: 2001.

(FIANG, 2002) - Jiang, Guofei. Multiple Vulnerabilities in SNMP. Institute for Security Technology Studies (ISTS), Dartmouth College: 2002.

(GILADI, 2004) – Giladi, Ran. *SNMP for home automation*. John Wiley & Sons Ltd. INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT: 4 de maio de 2004.

(HIA, 2001) - HIA, H. Erik; MIDKIFF, Scott. *Securing SNMP Across Backbone Networks*. USA: Virginia Polytechnic Institute, 2001.

(HONG, 2001) – Hong, Liu; Dong, Bai; DingWei. *The integration of SNMP and Web in embedded devices*. IEEE: 2001.

(IEC 62040-3, 1999) - IEC 62040-3, *Uninterruptible Power Systems*. INTERNATIONAL STANDARD. Geneve, Switzerland: 1999.

(J. MA, 2005) - J. Ma, Kevin; Bartos, Radim. *Performance Impact of Web Service Migration in Embedded Environments*. IEEE: 2005.

(KASSIK, 2000) - KASSIK, E.V. *Harmônicas em sistemas industriais de baixa tensão*. UFSC, DEE-CT, INEP, Florianópolis, Janeiro de 2000.

(KIM, 2002) - Kim, Myung-Sup; Choi, Mi-Joung; Hong, James W. *A load cluster management system using SNMP and web*. John Wiley & Sons Ltd. INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT: 22 de maio de 2002.

(KONOPKA, 1995) - KONOPKA, R; TROMMER M. *A Multilayer-Architecture for SNMP-Based, Distributed and Hierarchical Management of Local Area Networks*. IEEE: 1995.

(LARMOUTH, 1999) - LARMOUTH, John. *ASN.1 Complete.* Open Systems Solutions Edition: 1999.

(LEE, 2004) – Lee, Jin-Shyan; Hsu, Pau-Lo. *Design and Implementation of the SNMP Agents for Remote Monitoring and Control via UML and Petri Nets*

(LI, 1995) - LI, Jianxin; LEON, Benjamim J. *A Formal Approach to Model SNMP Network Management Systens*. IEEE: 1995

(LIANG, 2004) – Liang, T.-J.; Shyu, J.-L. *Improved DSP-controlled online UPS system with high real output power.* IEE Proc.-Electr. Power Appl., Vol. 151, No. 1, January 2004.

(LINZ, 2001) - Linz, Stephan; Heggelbacher, Oliver; Wolf, Matthias. *A System-Design for UPS-Equipment for Long-Term Backup Times*. Masterguard KG, Germany: 2001.

(LIU, 2003) – Liu, Shufen; Han, Lu; ZhangLiu, Xinjia; Nie, Kai. *Study of Network Performance Measurement Based on SNMP*. IEEE: 2003.

(LOU, 1999) - LOU, Rui – An; THAM, Chen – Khong. *Real-Time Data Transfer Using a Real-Time SNMP MIB*. IEEE: 1999.

(MANUAL DO RHINO, 2003) - Manual de UPS Microsol. *UPS Rhino*, Microsol Tecnologia LTDA: Março de 2003.

(MANUAL DO SOLIS, 2005) - Manual de UPS Microsol, UPS Solis. *UPS Solis*, Microsol Tecnologia LTDA: 12 de setembro de 2005.

(MOLINA, 1999) - MOLINA, Jean-Marc; GRAFF, Claude. *UPS European Guide*. The CEMEP UPS Group: 1999.

(MURRAY, 1998) - MURRAY, James D. *Windows NT SNMP*. USA: O'Reilly & Associates, Inc, 1998.

(MYERSON, 2002) – Myerson, Judith M. *Identifying enterprise network vulnerabilities*. John Wiley & Sons Ltd. INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT: 7 de fevereiro de 2002.

(NARA, 1996) - NARA, Koichi; HASEGAWA, Jun. *Future Flexible Power Delivery System and Its Intelligent Functions*. IEEE: 1996.

(PARTRIDGE, 1987) - PARTRIDGE, C; TREWITT, G. *The high-level entity management system (hems),* RFC 1021, October 1987.
< http://www.ietf.org/rfc/rfc1021.txt> Último acesso em 25/04/2006.

_____. *Hems variable definitions,* RFC 1024, October 1987.
< http://www.ietf.org/rfc/rfc1024.txt> Último acesso em 25/04/2006.

_____. *The high-level entity management protocol (hemp),* RFC 1022,October 1987.
< http://www.ietf.org/rfc/rfc1022.txt> Último acesso em 25/04/2006.

(PARTRIDGE, 1988) - PARTRIDGE, C; TREWITT, G. *HEMS Monitoring and Control Language*, RFC 1076, November 1988.
< http://www.ietf.org/rfc/rfc1076.txt> Último acesso em 25/04/2006.

(ROSE, 1990) - ROSE, M; MCCLOGHRIE K. *Structure and Identification of Management Information for TCP/IP-based Internets*, RFC 1155: USA, 1990.
< http://www.ietf.org/rfc/rfc1155.txt> Último acesso em 25/04/2006.

(ROSE, 1991) - ROSE, M; MCCLOGHRIE K. *Concise MIB Definitions,* RFC 1215, March 1991.
< http://www.ietf.org/rfc/rfc1215.txt> Último acesso em 25/04/2006.

_____. *Management Information Base for Network Management of TCP/IP-based internets: MIB-II.* RFC 1213, March 1991.
< http://www.ietf.org/rfc/rfc1213.txt> Último acesso em 25/04/2006.

(SANTIAGO, 2003) - SANTIAGO, Reuber S. *Desenvolvimento de uma UPS de 10KVA dupla conversão on-line*, *Dissertação de mestrado*, Universidade Federal do Ceará, Programa de Pós-graduação em Engenharia Elétrica, Março, 2003.

(SARKINEN, 1998) - SARKINEN, Johan; LUNDIN, Ola. *Integrate Internet Solutions Into Your Energy Management Network*. IEEE: 1998.

(SCHONWALDER, 2005) – Schonwalder, J. *Characterization of SNMP MIB Modules*. IEEE: 2005.

(SEC UPS, 2002) - SEC UPS Protocol. <http://us1.networkupstools.org/protocols/sec.html> Último acesso em 25/04/2006.

(SEUNG-HYUN, 2003) - Seung-hyun, Park; Myong-soon, Park. *An Efficient Transmission for Large MIB Tables in Polling-based SNMP*. IEEE: 2003.

(SILVEIRA, 2004) – Silveira, Jarbas; Castro, Helano. *SNMP Management system with remote authentication: UPS Applications*. VI Induscon: 2004.

(SKOK, 2004) - Skok, Srdjan, Ph.D.; Skok, Minea, M Se.; Vrkic, Niksa, B.Sc. *Electrical Performance Test Procedure For Uninterruptible Power Supplies.* IEEE: 2004.

(SMITH, 1995) - SMITH, Marvin W; McNally, John. *The UPS as a System Element*. ISBN: 1995.

(SOLTER, 2002) – Solter, Wilhelm. *A New International UPS Classification*. IEEE: 2002.

(STALLINGS, 1999) - STALLINGS, W. *SNMP, SNMPV2, SNMPV3 and RMON 1 and 2*. Third Edition. USA: Addison-Wesley, Inc., 1999.

(TANENBAUM, 2000) - TANENBAUM, Andrew S; Woodhull Albert S. *Sistemas Operacionais, Projeto e Implementação*. Segunda Edição. Porto Alegre: Bookman, 2000.

(TANENBAUM, 2001) - TANENBAUM, Andrew S. *Modern Operating Systems*. Second Edition. New Jersey: Prentice Hall, 2001.

(WAN, 2000) - WAN, Tat C; GOH, Alwyn; NG, Chin K; POH, Geong S. *Integrating Public Key Cryptography into the Simple Network Management Protocol (SNMP) Framework*. IEEE: 2000.

(WARRIER, 1987) - WARRIER U; BESAW, L. *A Simple Gateway Monitoring Protocol,* RFC 1028, November 1987.
< http://www.ietf.org/rfc/rfc1028.txt> Último acesso em 25/04/2006.

(WARRIER, 1989) - WARRIER U; BESAW, L. *The Common Management Information Services and Protocol over TCP/IP(CMOT)*, RFC 1095, April 1989
< http://www.ietf.org/rfc/rfc1095.txt> Último acesso em 25/04/2006.

(WARRIER, 1990) - WARRIER U; BESAW, L; LABARRE, L; HANDSPICKER, L B. *The Common Management Information Services and Protocols for the Internet (CMOT and CMIP),* RFC 1189, October 1990.
< http://www.ietf.org/rfc/rfc1189.txt> Último acesso em 25/04/2006.

(ZAPF, 1999) - ZAPF M., HERRMANN, K; Geihs K. *Decentralized SNMP Management with Mobile Agents*. Goethe-Universitat, Dept. of Computer Science: 1999.

# Apêndice

## Apêndice A – Mensagens SNMP

### A.1 – Estrutura geral de uma mensagem SNMP

Toda mensagem SNMP é codificada utilizando a codificação BER. Como mostrado na Figura A.11, esta mensagem sempre consiste de um preâmbulo, sendo seguido de uma unidade de dados chamada APDU (Application Protocol Data Unit). A função do preambulo é de identificação e validação da mensagem SNMP. O cabeçalho da PDU contém informações específicas da mensagem SNMP, como o tipo de operação a ser realizada. Os dados da PDU contêm a mensagem SNMP, sendo os dados que tornarão possível a operação SNMP proposta.

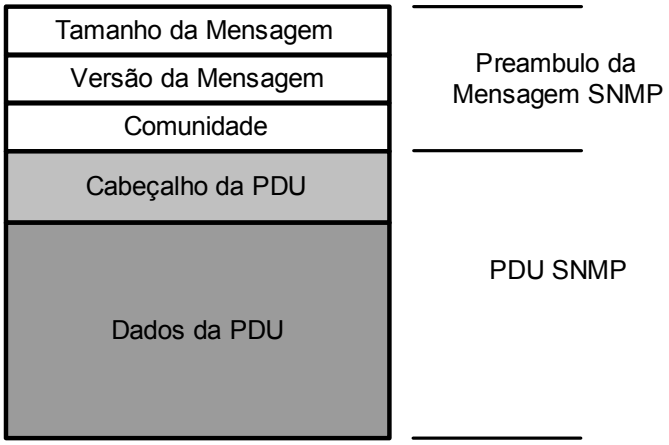


Figura A.1: estrutura geral de uma mensagem SNMPv1.

O preâmbulo da mensagem SNMP possui três campos, os quais estão presentes em todas as mensagens SNMPv1. Estes campos são:

- Tamanho da Mensagem: este campo indica o tamanho da mensagem, excluindo o próprio tamanho do campo, que são dois octetos;

- Versão da Mensagem: usado para indicar qual a versão do protocolo SNMP. Os valores são 0 para SNMPv1 e 1 para SNMPv2;

- Comunidade: este campo tem tamanho variável e especifica qual a comunidade de gerenciamento que deve aceitar esta mensagem SNMP. Tem tamanho máximo de 128 octetos.

O campo comunidade é a relação entre o gerente SNMP e o conjunto de agentes gerenciados, ou seja, o agente deve estar habilitado a receber e processar mensagens da comunidade a qual o mesmo está inserido. A comunidade estabelece uma relação de senha de acesso entre o gerente e o agente, mas esta função não engloba o conceito de segurança, sendo somente um elemento de delimitação entre um ambiente de gerenciamento SNMP. Como descrito na RFC 1157, o SNMP prevê uma esquema trivial de autenticação, sendo necessário envolver processos de encriptação e decripitação para aumentar o nivel de segurança no ambiente da rede. Por este motivo, alguns administradores de rede se limitam a usar somente as funções de busca, evitando usar funções de armazenamento de valores da MIB (STALLINGS, 1999).

O cabeçalho da PDU é semelhante para os grupos de PDU que são do tipo Request ou Response, dentre eles: Get-Request, Get-NextRequest, Set-Request e Get-Response.

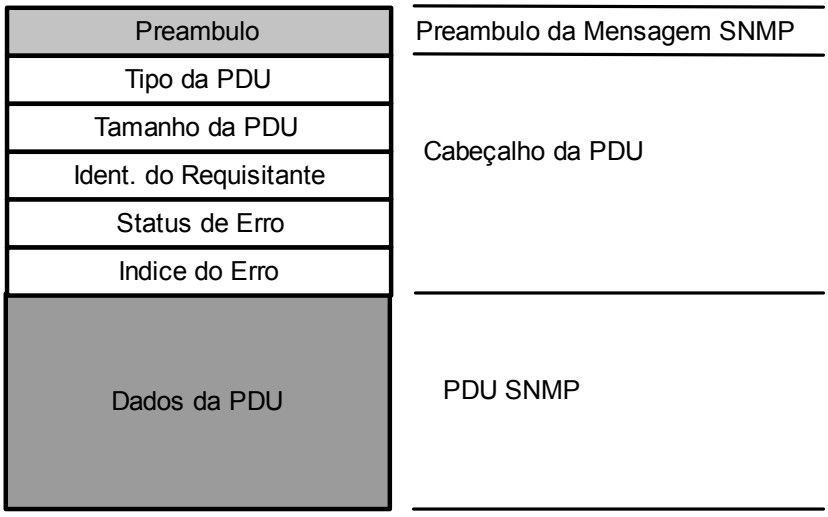


Figura A.2: cabeçalho de PDU Get ou Set.

Descreveremos a seguir, conforme indicado na Figura A.12, cada campo do cabeçalho de PDU do tipo Get-Request, Get-NextRequest, Get-Response ou Set-Request:

- Tipo da PDU: Indica o tipo de operação a que se propõe a PDU. Pode ter os valores: 0 (Get-Request), 1 (Get-NextRequest), 2 (Get-Response), 3 (Set-Request) e 4 (Trap), como descrito na RFC 1157;
- Tamanho da PDU: Indica o número de octetos que deverão existir após o seu próprio campo;
- Identificação do Requisitante: Este campo é usado para relacionar as mensagens de Get–Response ao Get-Request correspondente. Este campo é bastante útil quando um gerente faz vários pedidos simultâneos a um mesmo agente SNMP;
- Status de Erro: Este campo indica se o resultado da operação foi alcançado. Um valor diferente de zero indica que a operação não obteve sucesso. Os possíveis códigos para erros estão descritos na Tabela A.1.

- Índice do Erro: Indica qual a posição da seqüência de variável foi a causadora do erro. Se o Status de Erro for diferente de zero e o Índice de erro for zero, não foi possível localizar a posição do erro.

Tabela A.1: valores de erros em mensagens SNMP.

| *Valor* | *Mensagem* | *Descrição* |
|---|---|---|
| 0 | Nenhum Erro | Operação com sucesso |
| 1 | Muito Grande | A resposta de mensagem é muito grande para ser transmitida. |
| 2 | Nenhum Nome | O agente não conhece o OID solicitado |
| 3 | Valor Errado | O valor ou tipo a ser escrito na variável não é válido |
| 4 | Somente Leitura | A variável a qual foi solicitada um Set-Request é do tipo somente leitura |
| 5 | Erro Genérico | Um erro desconhecido do protocolo ocorreu |

A PDU Trap tem um cabeçalho diferente das outras mensagens SNMP. Este fato se deve a esta PDU ser um evento assíncrono, o que acarreta maior necessidade de informações para que o gerente possa tratar esta mensagem SNMP. Como o cabeçalho da mensagem SNMP Trap tem muitas informações relacionadas à própria PDU em questão, ele será esclarecido quando da descrição da PDU.

## A.2 – A Mensagem SNMP Get-Request

A Mensagem SNMP Get-Request tem a função de solicitar ao agente SNMP a leitura de uma variável da MIB. Esta operação é executada pelo gerente, que dentro da mensagem pode solicitar o valor de uma ou mais variáveis. Estas variáveis podem estar presentes em diferentes MIBs, desde que as mesmas estejam implementadas no agente. A PDU Get-Request é composta por uma seqüência de identificadores de variáveis, que são os OID (Object Identifiers), representando a identificação de cada variável da MIB. A cada variável, dentro do corpo da mensagem Get-Request, são associados dados de identificação dessa variável, como tamanho, tipo e valor, como mostrado na Figura A.3.

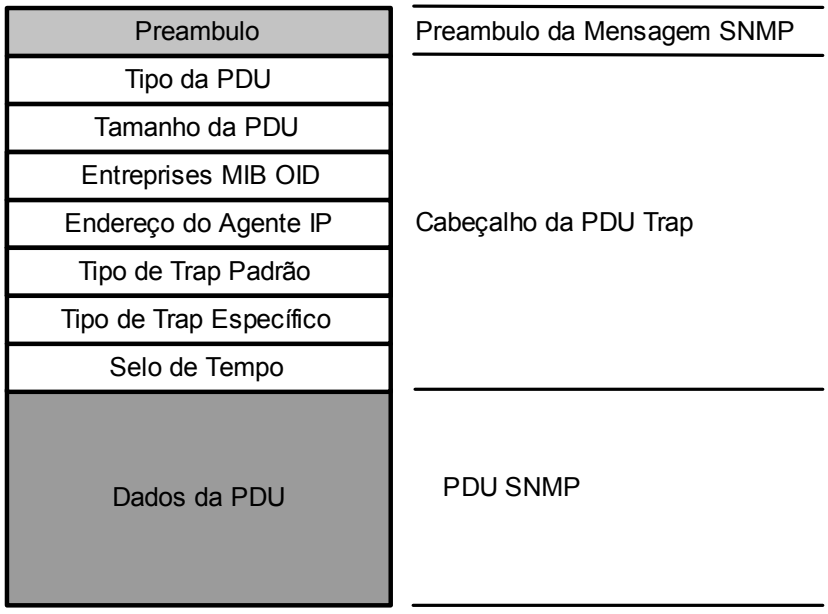


Figura A.3: a PDU Get-Request.

O Campo de dados da PDU Get-Request possui os seguintes campos, os quais se repetem para cada variável que é solicitada dentro da mensagem, excetuando-se o tamanho da lista de variáveis:

- Tamanho da Variável: Indica o tamanho da seqüência que segue, incluindo o próprio campo. Este tamanho compreende também o OID, o tipo e valor, que no pedido do Get-Request, sempre é do tipo NULL;
- OID da Variável: Indica o endereço da variável da MIB, representado numa notação numérica em que ramo da árvore se encontra. Em sua indicação textual é separado por pontos;
- Tipo da variável: Indica qual o tipo da variável que está sendo solicitada;
- Valor da Variável: Este campo somente é enviado para seguir uma padrão do ASN.1, pois sempre é do tipo NULL no pedido de Get-Request e somente será útil na PDU Get-Response.

Exemplificando uma PDU genérica em formato codificado, na Tabela A.2 é mostrado como a mensagem seria encapsulada em um datagrama UDP, além de visualizarmos a codificação BER em SNMP, como comentamos anteriormente. Foi anexado também um campo de descrição em ASN.1, que indica a forma de cada campo. Os valores numéricos encontram-se em formato hexadecimal assim como prevê a codificação BER.

Tabela A.2: PDU Get-Request codificada.

| **MENSAGEM SNMP Get-Request** | | |
|---|---|---|
| **VALOR CODIFICADO** | **CAMPO** | **DESCRIÇÃO (ASN.1)** |
| 30 35 | Tamanho da Mensagem | Seqüência (30h) seguida do comprimento total da mensagem |
| 02 01 00 | Versão do Protocolo | Tipo, Tamanho e Valor da Versão do Protocolo |
| 04 06 70 75 62 6C 69 63 | Comunidade | Tipo (String de Octetos), tamanho e dado |
| A0 | Tipo da Mensagem | Valor correspondente ao Get-Request |
| 1B | Tamanho da PDU | Comprimento do restante da PDU |
| 02 01 00 | Ident. Do Requisitante | Tipo, Tamanho e Valor |
| 02 01 00 | Status do Erro | Tipo, Tamanho e Valor |
| 02 01 00 | Índice do Erro | Tipo, Tamanho e Valor |
| 30 1B | Tamanho da Lista | Seqüência e Tamanho |
| 30 0C 06 08 | Tamanho / Tipo -1.a VAR | Seqüência, Tamanho, Tipo e Tamanho |
| 2B 06 01 02 01 04 03 00 | ID da 1.a Variável | ID da Variável 1.3.6.1.2.1.4.3. O valor 1.3 está compactado em 2B como já descrito |
| 05 00 | Valor da 1.a Variável | Tipo (Null) e Valor |
| 30 0D 06 09 | Tamanho / Tipo -2.a VAR | Seqüência, Tamanho, Tipo e Tamanho |
| 2B 06 01 02 01 04 04 0B 00 | ID da 2.a Variável | ID da Variável 1.3.6.1.2.1.4.4.11 |
| 05 00 | Valor da 2.a Variável | Tipo (Null) e Valor |

## A.3 – A Mensagem SNMP Get-NextRequest

A mensagem SNMP Get-NextRequest tem também a função de efetuar uma leitura de dados na MIB do agente, sendo indicada para permitir a leitura de valores de tabelas de tamanho desconhecido. Ela é muito útil para ler, por exemplo, a tabela ARP de um roteador. Esta tabela armazena o endereço MAC e o respectivo IP ao qual está associado nesta rede. Isto varia dinamicamente com as condições da rede e, portanto, esta tabela pode variar de tamanho. Esta mensagem apresenta um baixo desempenho para transmissão de grande quantidade de dados, sendo apresentada na versão 2 do protocolo SNMP uma nova PDU, a *Get-Bulk*, para melhorar este desempenho (SEUNG-HYUN, 2003).

O Get-NextRequest tem formato idêntico ao Get-Request. Ao fim de cada seqüência em uma PDU do tipo Get-NextRequest deve ser acrescido o valor 1, para indicar que o próximo valor, na mesma seqüência, deverá ser enviado. Por exemplo, caso desejemos ler a seqüência de variáveis da OID 1.3.6.1.2.1.1.2, sendo esta a entrada da tabela, devemos enviar uma mensagem Get-NextRequest na OID:

*1.3.6.1.2.1.1.2.1.0,*

após o que receberemos um Get-Response com o valor do OID:

*1.3.6.1.2.1.1.2.2.0,*

e assim por diante até recebermos um valor fora do índice procurado:

*1.3.6.1.2.1.1.2.0,*

indicando o fim da tabela. Na Tabela A.3 é mostrada uma PDU do tipo Get-NextRequest codificada em linguagem BER e sua construção em ASN.1.

Tabela A.3: PDU Get-NextRequest codificada.

| **MENSAGEM SNMP Get-NextRequest** | | |
|---|---|---|
| **VALOR CODIFICADO** | **CAMPO** | **DESCRIÇÃO (ASN.1)** |
| 30 35 | Tamanho da Mensagem | Seqüência (30h) seguida do comprimento total da mensagem |
| 02 01 00 | Versão do Protocolo | Tipo, Tamanho e Valor da Versão do Protocolo |
| 04 06 70 75 62 6C 69 63 | Comunidade | Tipo (String de Octetos), tamanho e dado |
| A1 | Tipo da Mensagem | Valor correspondente ao Get-NextRequest |
| 1B | Tamanho da PDU | Comprimento do restante da PDU |
| 02 01 00 | Ident. Do Requisitante | Tipo, Tamanho e Valor |
| 02 01 00 | Status do Erro | Tipo, Tamanho e Valor |
| 02 01 00 | Índice do Erro | Tipo, Tamanho e Valor |
| 30 1B | Tamanho da Lista | Seqüência e Tamanho |
| 30 0C 06 08 | Tamanho / Tipo -1.a VAR | Seqüência, Tamanho, Tipo e Tamanho |
| 2B 06 01 02 01 04 03 00 | ID da 1.a Variável | ID da Variável 1.3.6.1.2.1.4.3. O valor 1.3 está compactado em 2B como já descrito |
| 05 00 | Valor da 1.a Variável | Tipo (Null) e Valor |
| 30 0D 06 09 | Tamanho / Tipo -2.a VAR | Seqüência, Tamanho, Tipo e Tamanho |
| 2B 06 01 02 01 04 04 0B 00 | ID da 2.a Variável | ID da Variável 1.3.6.1.2.1.4.4.11 |
| 05 00 | Valor da 2.a Variável | Tipo (Null) e Valor |

## A.4 – A Mensagem SNMP Get-Response

A mensagem SNMP Get-Response é usada para que o agente SNMP responda ao gerente as requisições de leituras das variáveis. Após receber uma mensagem Get-Request ou Get-NextRequest, o agente SNMP deve:

- Analisar o OID recebido na mensagem;

- Executar uma operação de leitura na MIB;

- Encapsular o OID e seu respectivo valor em uma mensagem Get-Response;

- Responder ao gerente SNMP que solicitou o pedido.

O formato da PDU Get-Response é semelhante aos outros formatos das mensagens Get-Request e Get-NextRequest. Na lista de variáveis, é adicionado, após o campo “Tipo”, o valor da variável que foi lido na MIB. Na Tabela A.4 temos uma mensagem Get-Response acompanhada de sua respectiva Get-Request.

Tabela A.4: mensagem Get-Request com respectiva Get-Response.

| | | |
|---|---|---|
| 52 | Tamanho da Mensagem | 65 |
| 0 | Versão do Protocolo | 0 |
| Public | Comunidade | Public |
| 0 | Tipo da PDU | 2 |
| 39 | Tamanho da PDU | |
| 131 | Ident. do Requisitante | 131 |
| 0 | Status de Erro | 0 |
| 0 | Índice do Erro | 0 |
| 13 | Tamanho da Lista | |
| 11 | Tamanho da 1.a VAR | 11 |
| 1.3.6.1.2.1.4.3 | OID da 1.a Variável | 1.3.6.1.2.1.4.3 |
| INTEGER | Tipo da 1.a Variável | INTEGER |
| NULL | Valor da 1.a Variável | 32 |
| Get Request | | Get Response |

A mensagem Get-Response é enviada ao gerente SNMP, caso não ocorra nenhum erro na leitura da variável da MIB. Os erros mais comuns de acontecer são: Nenhum Nome, Muito Grande e Erro Genérico, como já descritos anteriormente.

Na Tabela A.5 é mostrada uma PDU do tipo Get-Response codificada em linguagem BER e ao lado a respectiva construção em ASN.1.

Tabela A.5: PDU Get-Response codificada.

| **MENSAGEM SNMP Get-Response** | | |
|---|---|---|
| VALOR CODIFICADO | CAMPO | DESCRIÇÃO (ASN.1) |
| 30 27 | Tamanho da Mensagem | Seqüência (30h) seguida do comprimento total da mensagem |
| 02 01 00 | Versão do Protocolo | Tipo, Tamanho e Valor da Versão do Protocolo |
| 04 06 70 75 62 6C 69 63 | Comunidade | Tipo (String de Octetos), tamanho e dado |
| A2 | Tipo da Mensagem | Valor correspondente ao Get-Response |
| 1B | Tamanho da PDU | Comprimento do restante da PDU |
| 02 01 00 | Ident. Do Requisitante | Tipo, Tamanho e Valor |
| 02 01 00 | Status do Erro | Tipo, Tamanho e Valor |
| 02 01 00 | Índice do Erro | Tipo, Tamanho e Valor |
| 30 1B | Tamanho da Lista | Seqüência e Tamanho |
| 30 0C 06 08 | Tamanho / Tipo -1.a VAR | Seqüência, Tamanho, Tipo e Tamanho |
| 2B 06 01 02 01 04 03 00 | ID da 1.a Variável | ID da Variável 1.3.6.1.2.1.4.3. O valor 1.3 está compactado em 2B como já descrito |
| 02 83 | Valor da 1.a Variável | Tipo (Null) e Valor |

## A.5 – A Mensagem SNMP Set-Request

A Mensagem SNMP Set-Request é usada para efetuar modificações no valor das variáveis da MIB. É através desta PDU que o gerente consegue realizar a operação de armazenamento em SNMP. A mensagem Set-Request tem estrutura semelhante às PDUs do tipo Get. Dentro da PDU, uma lista de variáveis é adicionada a cada seqüência de variáveis. Após o OID da variável, segue o valor que se deseja armazenar.

Para cada Set-Request enviado por um gerente SNMP, deve ser recebido um Get-Response, confirmando a escrita nas respectivas variáveis da MIB do agente. O agente, ao receber a mensagem Set-Request, analisa a seqüência de variáveis e executa as operações de armazenamento e, caso não haja nenhum erro, monta uma mensagem de Get-Response e a envia ao gerente. Esta mensagem enviada ao gerente deve ter o mesmo valor no campo "Identificação do Requisitante", presente no cabeçalho das mensagens Get e Set. Isto serve para que o gerente possa identificar qual o Set-Request que está associado àquele Get-Response.

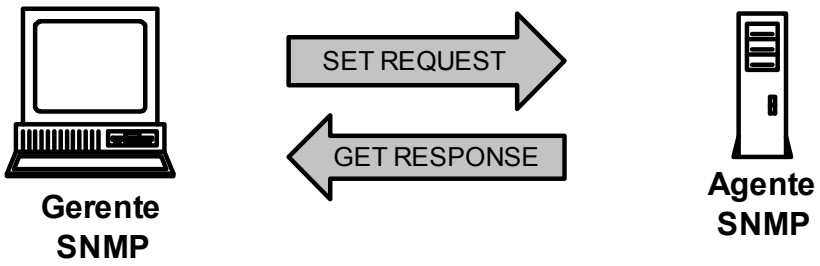


Figura A.4: operação de Set-Request simples.

Algumas aplicações sugerem que se utilize uma redundância na verificação do sucesso da operação do Set-Request. Como descrito em (MURRAY, 1998), quando o agente recebe a mensagem Set-Request e analisa a lista de variáveis, o mesmo somente verifica a existência e a permissão do respectivo OID, e efetua a operação de escrita, realizando assim uma operação

de Set-Request simples, como mostrada na Figura A.14. Muitas vezes este valor está associado a um evento particular do sistema que impede que a operação de escrita seja efetuada. A mensagem de confirmação do Set-Request, o Get-Response, pode não ter conhecimento desta falha, acarretando uma falha na informação recebida pelo gerente, podendo ser assim entendida como uma operação de Set-Request simples. Para tornarmos esta operação robusta, como mostrado na Figura A.5, devemos primeiro efetuar uma operação de Get-Request, para nos certificarmos do valor da variável, para depois efetuarmos o Set-Request. Após recebido o Get-Response de confirmação, alguns segundos são esperados, podendo este tempo variar de acordo com o tipo da aplicação, para depois ser enviado outro Get-Request para a verificação do valor da variável. A esta operação chamamos de operação de Set-Request robusta.

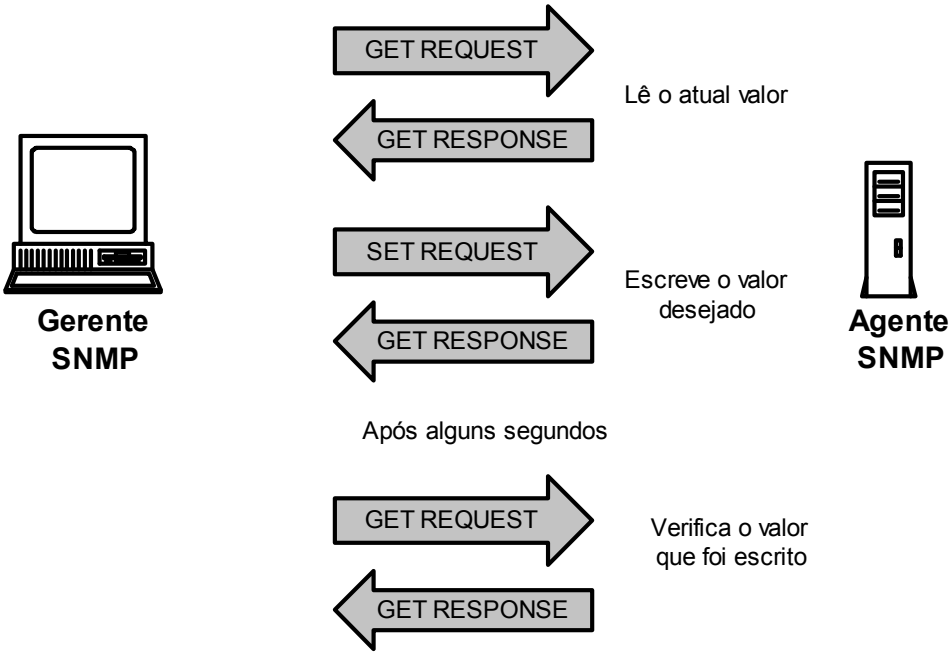


Figura A.5: operação de Set-Request Robusta.

Quando algum erro ocorre na execução de uma operação de escrita, uma mensagem Get-Response é enviada ao gerente, notificando a ocorrência de um erro. Um erro muito comum em sistemas embarcados é o erro “Muito Grande”; isto indica que o agente não pode enviar uma mensagem Get-Response com o tamanho da variável escrita na MIB. Como descrito em

(MURRAY, 1998), agentes não são “obrigados” a aceitar mensagens SNMP maiores que 484 bytes, embora elas possam ser implementadas, caso sejam suportadas pelo sistema de gerenciamento. Na Tabela A.6 temos uma PDU do tipo Set-Request codificada em BER e sua respectiva construção em ASN.1.

Tabela A.6: PDU Set-Response codificada.

| **MENSAGEM SNMP Set-Request** | | |
|---|---|---|
| VALOR CODIFICADO | CAMPO | DESCRIÇÃO (ASN.1) |
| 30 27 | Tamanho da Mensagem | Seqüência (30h) seguida do comprimento total da mensagem |
| 02 01 00 | Versão do Protocolo | Tipo, Tamanho e Valor da Versão do Protocolo |
| 04 06 70 75 62 6C 69 63 | Comunidade | Tipo (String de Octetos), tamanho e dado |
| A3 | Tipo da Mensagem | Valor correspondente ao Get-Response |
| 1B | Tamanho da PDU | Comprimento do restante da PDU |
| 02 01 00 | Ident. Do Requisitante | Tipo, Tamanho e Valor |
| 02 01 00 | Status do Erro | Tipo, Tamanho e Valor |
| 02 01 00 | Índice do Erro | Tipo, Tamanho e Valor |
| 30 1B | Tamanho da Lista | Seqüência e Tamanho |
| 30 0C 06 08 | Tamanho / Tipo -1.a VAR | Seqüência, Tamanho, Tipo e Tamanho |
| 2B 06 01 02 01 04 03 00 | ID da 1.a Variável | ID da Variável 1.3.6.1.2.1.4.3. O valor 1.3 está compactado em 2B como já descrito |
| 02 83 | Valor da 1.a Variável | Tipo (Null) e Valor |

## A.6 – A Mensagem SNMP Trap

A Mensagem SNMP Trap tem a função de enviar alertas ao gerente sobre alterações previstas nos dados da MIB. Estas mensagens são do tipo não solicitadas e têm estímulo próprio do agente, tendo portanto comportamento assíncrono. A ocorrência de uma Trap deve indicar a ocorrência de alguma alteração de condição que pode interessar ao gerente.

O formato da mensagem Trap difere do formato do restante das mensagens SNMP. A PDU Trap tem um preambulo que é semelhante ao das outras mensagens, mas difere em seu cabeçalho, trazendo informações relativas ao corpo da mensagem. A seguir alguns dos campos do cabeçalho são descritos:

- Enterprises MIB OID: A OID da classe de gerenciamento principal da aplicação;
- Endereço IP do Agente: O endereço IP do agente que enviou a mensagem Trap;
- Tipo de Trap Padrão: Identifica o tipo de Trap padrão, caso não seja um trap específico esse valor é 6;
- Tipo de Trap Específico: Identifica qual o tipo de Trap específico. Só é válido quando o tipo de Trap padrão é 6;
- Selo de Tempo: indica o número de centisegundos desde que o agente foi iniciado até o momento em que a mensagem foi enviada. Tem tamanho de 32 bits.

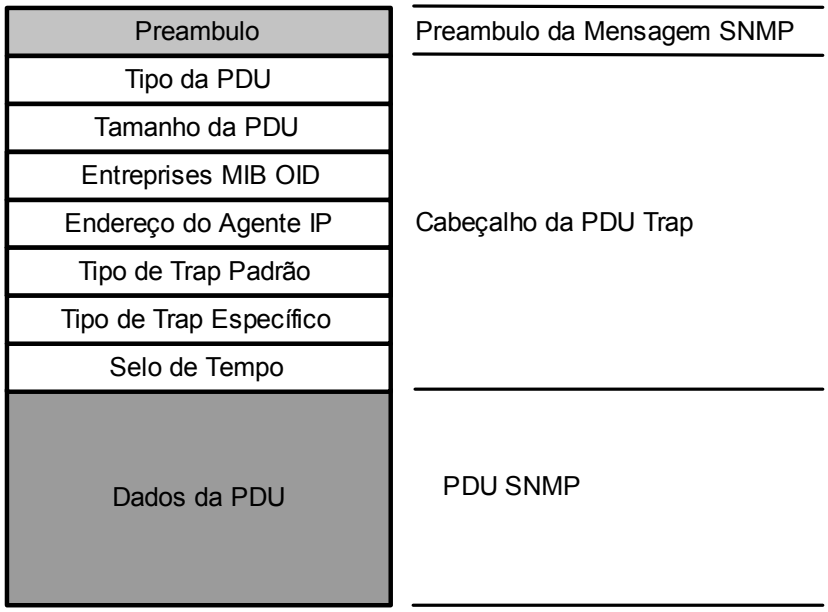


Figura A.6: cabeçalho de PDU Trap.

A mensagem SNMP Trap tem uma particularidade especial que é definida em seu cabeçalho, através dos campos Trap Padrão e Trap Específico. Caso o Trap Padrão tenha um valor de 0 a 5, identificando que o mesmo pertence à classe de Traps genéricos definidos na MIB-II, o campo Entreprises deverá ter o valor:

*1 . 3 . 6 . 1 . 2 . 1 . 11 ,*

ou descrevendo textualmente

*iso.org.dod.internet.mib.snmp ,*

e o campo Trap Específico terá valor 0. Esta mensagem não conterá lista de variáveis pois toda a sua informação já está contida no campo, e ela é dita como uma Mensagem Trap Padrão. Caso o valor do campo Trap Padrão seja 6, o campo Enterprise conterá a identificação da OID da aplicação e o campo

Trap Específico terá o valor definido em sua MIB que indica qual o alarme que está sendo sinalizado. O corpo da Mensagem Trap conterá o OID do alarme seguido dos seus campos descritores. Na Tabela A.7 está contida a descrição dos tipos padrões de Traps.

Tabela A.7: tipos Padrões de Mensagens Trap.

| Valor | Tipo | Descrição |
|---|---|---|
| 0 | ColdStart | Indica que um agente foi inicializado e suas configurações anteriores são de origem. Os dados armazenados em memória volátil provavelmente foram perdidos. |
| 1 | WarmStart | Indica que o dispositivo foi inicializado mas suas configurações anteriores não foram mudadas. |
| 2 | LinkDown | Indica que a comunicação com a rede deste dispositivo falhou. |
| 3 | LinkUp | Indica que a comunicação com a rede foi estabelecida novamente. |
| 4 | AuthenticationFailure | Indica que um agente recebeu uma mensagem SNMP que não foi autenticada, sendo transmitida para uma comunidade errada ou não autorizada. |
| 5 | EgpNeighborloss | Indica que um agente falhou na comunicação com um dispositivo EGP (Exterior Gate Protocol). Este Trap é obrigatório para roteadores que implementam o protocolo EGP. (RFC 904) |
| 6 | EnterpriseSpecific | Indica que a mensagem é uma Trap Especifica |

Na Tabela A.8 é mostrada uma PDU do tipo Trap codificada em linguagem BER e ao lado a respectiva construção em ASN.1.

Tabela A.8: PDU Trap Cold Start.

| **MENSAGEM SNMP Trap** | | |
|---|---|---|
| VALOR CODIFICADO | CAMPO | DESCRIÇÃO (ASN.1) |
| 30 2C | Tamanho da Mensagem | Seqüência (30h) seguida do comprimento total da mensagem |
| 02 01 00 | Versão do Protocolo | Tipo, Tamanho e Valor da Versão do Protocolo |
| 04 06 70 75 62 6C 69 63 | Comunidade | Tipo (String de Octetos), tamanho e dado |
| A3 | Tipo da Mensagem | Valor correspondente ao Trap |
| 1F | Tamanho da PDU | Comprimento do restante da PDU |
| 06 0C 2B 06 01 04 01 82 37 01 01 03 01 01 | Enterprises MIB OID | Cadeia de Strings (06H) de tamanho 0C, com OID 1.3.6.1.4.1.311.1.1.3.1.1 |
| 40 04 80 80 80 03 | Endereço do Agente IP | Tipo IP (40H): 128.128.128.3 |
| 02 01 00 | Tipo de Trap Padrão | Trap Padrão (00h): Cold Start |
| 02 01 00 | Tipo de Trap Específico | Trap Específico (00H) |
| 43 01 00 00 00 | Selo de Tempo | Selo de tempo de 1 centisegundo (0h: 00m: 00s |

# Anexos

## Artigo Publicado