

Action.NET

SNMP Manager Communication Protocol

Versão1.0.1

Manual de Referência

00062.01
OUTUBRO, 2013

Action.NET – SNMP
Manager Communication Protocol

Versão 1.0.1

Manual de Referência

00062.01
OUTUBRO, 2013

Copyright2013®
Spin Engenharia de Automação Ltda
Todos os Direitos Reservados

Nenhuma parte deste documento pode ser reproduzida, copiada, fotocopiada, distribuída ou alterada sem a prévia e expressa autorização da Spin Engenharia de Automação Ltda.

NOTA

Notas gerais sobre o manual, texto centralizado com espaçamento de 60pt antes do texto, estilo:
NOTA – centralizado.

O alinhamento vertical dessa página é inferior, para tentar manter o texto sobre as alterações sempre alinhado com a margem inferior.

Sendo assim o tamanho dessas notas vai influenciar na posição do título “NOTA” lá no topo da página, quanto mais for escrito mais acima ele ficará.

Em virtude do contínuo desenvolvimento de seus produtos, a informação contida neste documento está sujeita a alterações e/ou modificações sem prévia notificação. A Spin não se considera responsável por erros de digitação ou interpretação das informações aqui contidas; e/ou por danos e prejuízos causados / gerados a terceiros. O conteúdo desta publicação poderá ser alterado a qualquer momento sem que exista a obrigação de notificar qualquer parte envolvida; isto não implicará, em nenhuma hipótese, em alterações, reclamações, ou extensão de garantia.

Nesta página estão exemplificados os estilos para serem utilizados com os ícones de atenção do texto. O estilo é **Atenção – ícones**.



Cuidado! Indica que o usuário deverá proceder exatamente como descrito neste manual, sob pena de danificar ou configurar errado o equipamento.



Dica. Indica informações úteis e rápidas para solução de pequenos problemas.



Perigo! Indica que o usuário deverá proceder exatamente como descrito neste manual, sob risco de choque ou descarga elétrica.

Sumário

1. INFORMAÇÃO GERAL.....	1
1.1 Sumário	1
1.2 Funcionamento Geral.....	1
2. CONFIGURAÇÃO DO CANAL.....	3
2.1 Opções de protocolo.....	3
2.2 Interfaces	3
3. CONFIGURAÇÃO DE “NODES”	4
3.1 Parâmetros	4
4. CONFIGURAÇÃO DE PONTOS	5
4.1 Tipos de pontos.....	5
4.2 Endereço do ponto.....	5
5. JANELA BROWSER	7
5.1 Geral	7
6. LISTA COM ALGUNS DOS OIDS MAIS RELEVANTES	9

1. INFORMAÇÃO GERAL

1.1 Sumário

Communication Driver Name: SNMP

Current Version: 1.0

Implementation DLL: T.ProtocolDriver. SNMP.dll, AVSNMPBrowser.dll e ANSNMP.dll.

Protocol: SNMP Manager communication protocol

Interface: TCP/IP

Description: O módulo de comunicação SNMP (*Simple Network Management Protocol*) Manager, ou Gerenciador, é composto pelas bibliotecas T.ProtocolDriver.SNMP.dll, AVSNMPBrowser.dll e ANSNMP.dll. O módulo implementa a funcionalidade SNMP gerenciador numa rede de agentes servidores SNMP. São suportadas as versões do protocolo SNMP até a SNMPv2c.

Protocol Options: Temporizadores para envio de mensagens de controle do protocolo.

Max number of nodes: User defined

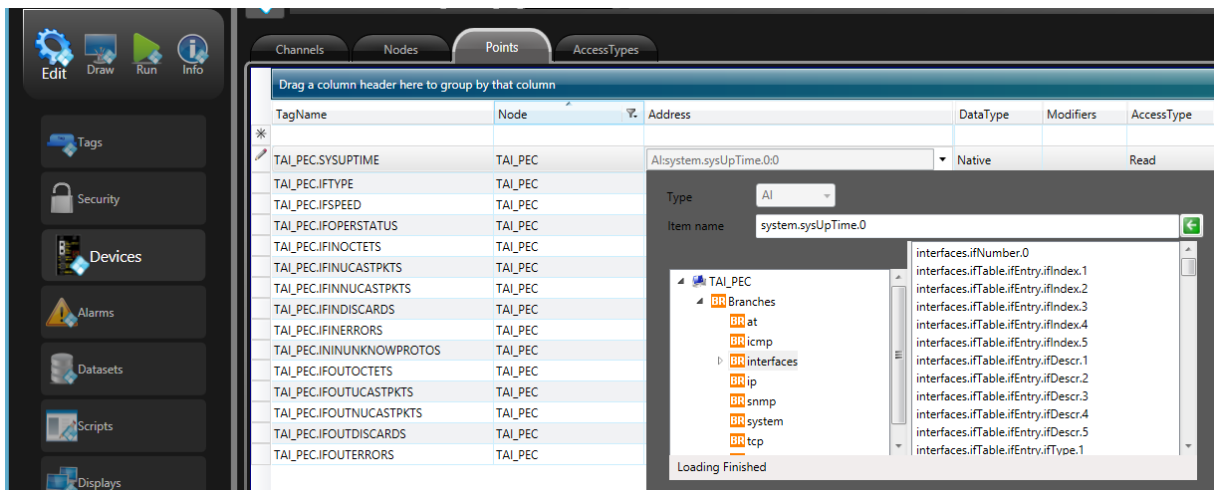
PC Hardware requirements: Standard PC Ethernet interface board;

PC Software requirements: Action.NET system.

1.2 Funcionamento Geral

No uso típico do SNMP um ou mais dispositivos chamados de gerenciadores, têm a tarefa de monitorar e/ou controlar um grupo de outros dispositivos chamados de agentes, os quais fazem o papel de servidores. Este gerenciamento dos agentes é feito pelos gerenciadores através de comandos e leituras de variáveis internas dos agentes, via UDP/IP. Estas variáveis, por sua vez, estão organizadas de maneira hierárquica como definido no padrão MIB (*Management Information Bases*). Alguns exemplos de dispositivos agentes são: roteadores, switches, computadores e impressoras.

Uma estrutura formatada como MIB pode ser entendida como uma árvore, onde cada nó/folha se refere a um OID (*object identifier*). Um OID, quando nó, pode ser entendido como uma estrutura complexa que contém debaixo de si vários nós e/ou folhas. Tem-se um exemplo mostrado na figura abaixo, onde um agente, nomeado TAI_PEC, tem o OID base (nó) **interfaces**, o qual tem diversos OID's folhas (isto é, sem filhos) debaixo de si.



The screenshot displays a software interface with a table of SNMP tags and a tree view of the MIB structure for TAI_PEC.

Table of Tags:

TagName	Node	Address	DataType	Modifiers	AccessType
TAI_PEC.SYSUPTIME	TAI_PEC	Alsystem.sysUpTime.0:0	Native		Read
TAI_PEC.IFTYPE	TAI_PEC				
TAI_PEC.IFSPPEED	TAI_PEC				
TAI_PEC.IFOPERSTATUS	TAI_PEC				
TAI_PEC.IFINOCTETS	TAI_PEC				
TAI_PEC.IFINUCASTPKTS	TAI_PEC				
TAI_PEC.IFINNUCASTPKTS	TAI_PEC				
TAI_PEC.IFINDISCARDS	TAI_PEC				
TAI_PEC.IFINERRORS	TAI_PEC				
TAI_PEC.ININUNKNOWPROTOS	TAI_PEC				
TAI_PEC.IFOUOCTETS	TAI_PEC				
TAI_PEC.IFOUUCASTPKTS	TAI_PEC				
TAI_PEC.IFOUTNUCASTPKTS	TAI_PEC				
TAI_PEC.IFOUTDISCARDS	TAI_PEC				
TAI_PEC.IFOUTERRORS	TAI_PEC				

MIB Tree View:

- TAI_PEC
 - Branches
 - at
 - icmp
 - interfaces
 - ip
 - snmp
 - system
 - tcp

The tree view shows a detailed structure of the MIB, including nodes like `interfaces.ifNumber.0`, `interfaces.ifTable.ifEntry.ifIndex.1`, `interfaces.ifTable.ifEntry.ifIndex.2`, `interfaces.ifTable.ifEntry.ifIndex.3`, `interfaces.ifTable.ifEntry.ifIndex.4`, `interfaces.ifTable.ifEntry.ifIndex.5`, `interfaces.ifTable.ifEntry.ifDescr.1`, `interfaces.ifTable.ifEntry.ifDescr.2`, `interfaces.ifTable.ifEntry.ifDescr.3`, `interfaces.ifTable.ifEntry.ifDescr.4`, `interfaces.ifTable.ifEntry.ifDescr.5`, and `interfaces.ifEntry.ifType.1`.

Figura 1 – Exemplo de árvore MIB usada pelo SNMP.

O módulo SNMP Gerenciador do Action.NET, trabalha basicamente com as seguintes funcionalidades previstas no protocolo:

- (1) Conexão em UDP na porta 161 com um ou mais agentes SNMP;
- (2) Navegação por uma rede de agentes SNMP apontando quais os agentes existentes e, para cada agente encontrado, mostrar sua estrutura interna de pontos;
- (3) Leitura por amostragem periódica individual dos objetos contidos na estrutura MIB de um agente SNMP;
- (4) Apontar erro de comunicação caso um agente alvo de amostragem periódica se mostre indisponível.

2. CONFIGURAÇÃO DO CANAL

2.1 Opções de protocolo

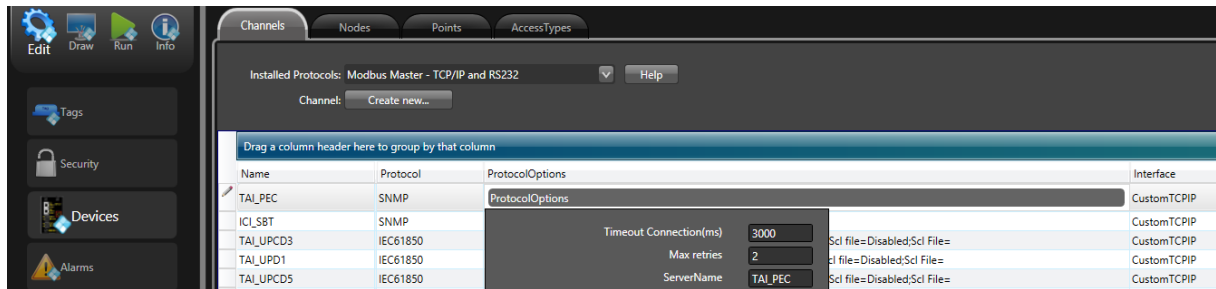


Figura 2 – Opções de protocolos

Timeout Connection(ms): Tempo em segundos máximo aceitável pela espera de uma solicitação ao agente servidor, após este tempo será disparado o *timeout*.

Max retries: Número máximo de tentativas de reenvio de uma solicitação ao agente servidor após o primeiro *timeout*.

ServerName- Nome “fantasia” do agente servidor, válido apenas internamente no Action.NET.

2.2 Interfaces

Interface:

- Custom TCP/IP: Esta opção deve ser escolhida para o protocolo em questão.

3. CONFIGURAÇÃO DE “NODES”

Cada nó (node) representa um agente servidor SNMP.

3.1 Parâmetros

Endereço IP – deve ser especificado o endereço IP do agente.

Port – Número da porta utilizada para conexão. A porta padrão para o protocolo SNMP é a porta 162.

Digital Sample Time – Período de tempo de amostragem em milissegundos de pontos configurados como entradas digitais na tabela Points do Action.NET. Se igual a zero nenhuma amostragem será feita.

Analog Sample Time – Período de tempo de amostragem em milissegundos de pontos configurados como entradas analógicas na tabela Points do Action.NET. Se igual a zero nenhuma amostragem será feita.

Backup Station – As mesmas configurações feitas para a estação principal podem ser feitas para um agente de backup (alternativa) se existir na instalação.

4. CONFIGURAÇÃO DE PONTOS

4.1 Tipos de pontos

Na tabela de pontos do Action.NET, podem ser usados os seguintes tipos de pontos:

Sigla	Tipo de Sinal	Tipo de Ponto	Descrição
AI	Analógico	Entrada	Entradas analógicas
BI	Digital	Entrada	Entradas digitais

Tabela 2 – Especificação dos tipos de pontos

4.2 Endereço do ponto

O endereçamento na tabela de pontos é feito usando os dois campos abaixo:

Node – Deve ser colocado o nome do Node referente ao agente servidor desejado.

Address – *String* completo de identificação de um OID “folha” do agente em questão. Para a obtenção dos nomes dos itens deve ser utilizada a Janela Browser SNMP, apresentada na próxima seção deste manual. Abaixo segue um exemplo de variáveis SNMP listadas na tabela Points do Action.NET:

TagName	Node	Address	DataType	Modifiers	AccessType	Scaling
TAL_PEC.SYSUPTIME	TAL_PEC	Alsystem.sysUpTime.0	Native		Read	None
TAL_PEC.IFATYPE	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifType.1	Native		Read	None
TAL_PEC.IFSPEED	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifSpeed.1	Native		Read	None
TAL_PEC.IFOPERSTATUS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifOperStatus.1	Native		Read	None
TAL_PEC.IFINOCTETS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifInOctets.1	Native		Read	None
TAL_PEC.IFINUCASTPKTS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifInUcastPkts.1	Native		Read	None
TAL_PEC.IFINNUCASTPKTS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifInNUcastPkts.1	Native		Read	None
TAL_PEC.IFINDISCARDS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifInDiscards.1	Native		Read	None
TAL_PEC.IFINERRORS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifInErrors.1	Native		Read	None
TAL_PEC.IFINUNKNOWNPROTOS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifInUnknownProtos.1	Native		Read	None
TAL_PEC.IFOUOCTETS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifOutOctets.1	Native		Read	None
TAL_PEC.IFOUUCASTPKTS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifOutUcastPkts.1	Native		Read	None
TAL_PEC.IFOUTNUCASTPKTS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifOutNUcastPkts.1	Native		Read	None
TAL_PEC.IFOUTDISCARDS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifOutDiscards.1	Native		Read	None
TAL_PEC.IFOUTERRORS	TAL_PEC	Alinterfaces.ifTable.ifEntry.ifOutErrors.1	Native		Read	None

Figura 3 – Exemplos de OIDs SNMP de um agente.

Para o SNMP a especificação SMI (Structure of Management information) define um número fixo de datatypes possíveis para representação dos atributos valores dos OIDs. A seguir, tem-se a lista dos datatypes (como definidos pelo SMI) e o tratamento interno realizado pelo Action.NET:

Integer, Integer32, Ulnteger, Counter32, Gauge32, TimeTicks - O módulo SNMP Gerenciador formatará a variável internamente em float de 32bits;

Object Identifier - O módulo SNMP Gerenciador formatará a variável internamente em um String representativo do nome do OID;

BitString, Opaque, NsapAddress - O módulo SNMP Gerenciador formatará a variável internamente em um String da representação hexadecimal dos dados;

IpAddress - O módulo SNMP Gerenciador formatará a variável internamente em um String sob a forma de 4 octetos separados por pontos;

Counter64- O módulo SNMP Gerenciador formatará a variável internamente em float de 64bits.

5. JANELA BROWSER

5.1 Geral

Para facilitar o cadastramento de OIDs existentes em um agente SNMP, está disponível a Janela Browser SNMP. Tal funcionalidade serve para realizar uma busca *online* na árvore do agente servidor, e então escolher um OID neste e associá-lo a um registro da tabela points Action.NET.

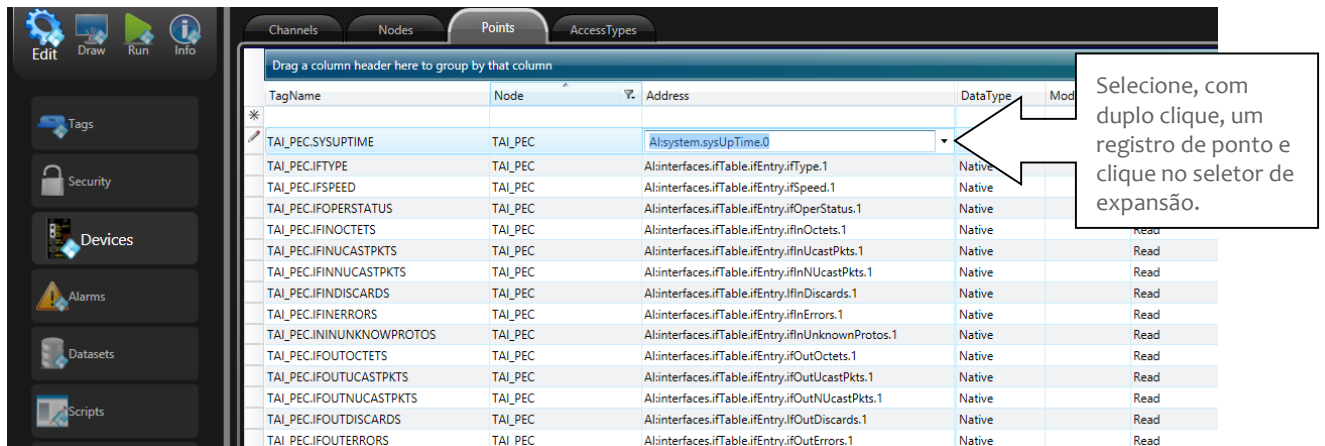


Figura 4 – Abertura da Janela Browser.

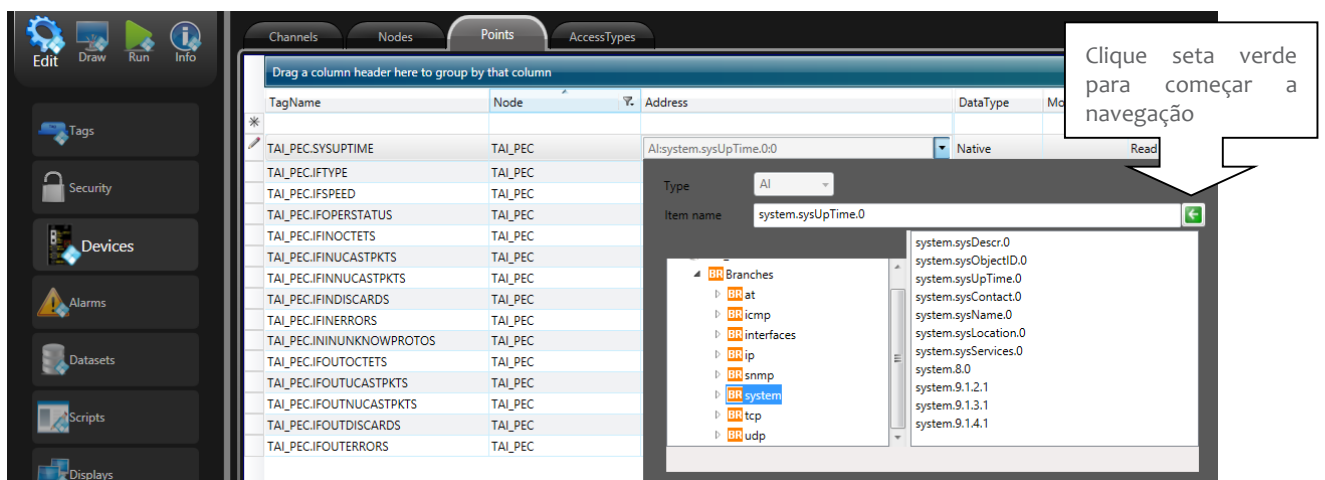


Figura 5 – Inicialização da navegação na Janela Browser.

A Janela é constituída de dois quadros apresentados lado a lado. No quadro da esquerda é apresentada uma lista hierárquica, em forma de uma árvore, com os dados da estrutura MIB do agente. Esta estrutura foi simplificada a três níveis de hierarquia, em ordem decrescente de nível hierárquico, tem-se: nome do agente servidor, OIDs base (chamados aqui ainda de nós ou de *branches*) e OIDs folhas (último nível hierárquico). Clicando-se, sobre o nome do servidor, ocorre a expansão para o próximo nível hierárquico, OIDs base, clicando-se nestes, chega-se às folhas. No quadro da direita serão apresentados os OIDs's folhas relativos ao OID *branch* clicado. Cada *branch* representa um grupo padronizado de OIDs como definido no padrão MIB-II (segunda versão do MIB). Exemplos de nomes de grupos padronizados pela norma: system (grupo de descrição básica), icmp (estatísticas apontadas pelo protocolo ICMP), tcp (estatísticas apontadas

pelo protocolo TCP), dentre outros. Se o agente estiver indisponível no momento, terá seu ícone marcado com um X na cor vermelha.

O quadro do lado direito serve para seleção dos OIDs folhas, ao ser selecionado este terá seu identificador transferido para o campo *Address* do registro em questão.

6. LISTA COM ALGUNS DOS OIDS MAIS RELEVANTES

Segue abaixo a lista de objetos relacionados a interfaces de rede. Para maiores detalhes, ver RFC 2233.

TIPO DE OBJETO: ifNumber

SINTAXE: Integer32

OID BASE: interfaces

DESCRIÇÃO:

O número de interfaces de rede (não importando seu estado atual) presentes no dispositivo.

TIPO DE OBJETO: ifIndex

SINTAXE: InterfaceIndex

OID BASE: interfaces

DESCRIÇÃO:

Um valor único, maior que zero, para cada interface. É recomendado que os valores comecem em 1 e sejam contínuos. O valor para cada interface deve se manter constante, exceto em caso de reinicialização do agente. Neste último caso o valor pode ser mudado.

TIPO DE OBJETO: ifDescr

SINTAXE: DisplayString (SIZE (0..255))

OID BASE: interfaces

DESCRIÇÃO:

Uma cadeia de caracteres contendo informação sobre a interface. Esta cadeia pode inclusive conter o nome do fabricante, nome do produto e o número de versão de hardware/software.

TIPO DE OBJETO: ifSpeed

SINTAXE: Gauge32

OID BASE: interfaces

DESCRIÇÃO:

Uma estimativa da largura de banda em bits por segundo.

TIPO DE OBJETO: ifPhysAddress

SINTAXE: Gauge32

OID BASE: interfaces

DESCRIÇÃO:

Endereço de físico da placa de rede.

TIPO DE OBJETO: ifAdminStatus

SINTAXE INTEIRO{

up(1), -- pronto para transmitir pacotes

down(2),

testing(3) -- em algum modo de teste

}

OID BASE: interfaces

DESCRIÇÃO:

O estado desejado pelo próprio agente para a interface. O estado atual é dado pelo tipo de objeto ifOperStatus, comentado no próximo item. O estado testing(3) indica que não pode haver

transmissão de pacotes no momento. Quando o agente inicia, todas as interfaces iniciam no estado down(2). Daí então, como de alguma ação explícita do agente ou de pré-configuração o status é alterado para up(1) ou testing(2).

TIPO DE OBJETO: ifOperStatus

SINTAXE INTEIRO {

up(1), -- pronto para transmitir pacotes

down(2),

testing(3), -- em algum modo de teste

unknown(4), -- impossível de determinar status

dormant(5),

notPresent(6), -- algum componente está faltando

lowerLayerDown(7) -- indisponível devido ao estado das interfaces das camadas inferiores

}

OID BASE: interfaces

DESCRIÇÃO:

O estado atual da interface. O estado testing(3) indica que não pode haver transmissão de pacotes no momento. Se o status de ifAdminStatus é down (2), então ifOperStatus deve ser down(2). Se ifAdminStatus mudou para up(1), então ifOperStatus deve trocar para up(1) se a interface está pronta para transmitir e receber tráfego de rede. Deve trocar para dormant(5) se a interface está esperando por ações externas. Deve travar (sem permitir trocas para o estado seguinte, up(1)) no estado down(2) se, e apenas se, existe falha que o impossibilite de seguir para o estado seguinte. O estado notPresent(6) geralmente indica falta/falha de componentes de hardware.

TIPO DE OBJETO: ifLastChange

SINTAXE: TimeTicks

OID BASE: interfaces

DESCRIÇÃO:

Retém o valor do objeto system.sysUpTime (contador de tempo deste que o agente foi ligado) quando a interface foi inicializada. Se o valor do estado foi atualizado anteriormente à última inicialização do agente, então o valor de ifLastChange deve ser zero.

TIPO DE OBJETO: ifInOctets

SINTAXE: Counter32

OID BASE: interfaces

DESCRIÇÃO:

O número total de octetos recebidos na interface, incluindo caracteres de framing.

TIPO DE OBJETO: ifInErrors

SINTAXE: Counter32

OID BASE: interfaces

DESCRIÇÃO:

Para interfaces orientadas a pacotes, indica o número de pacotes com algum tipo de erro que recebidos pela interface de rede. Estes mesmos não foram passados às camadas superiores. Para interfaces orientadas à caractere ou de comprimento fixo, indica o número de unidades de transmissão com algum tipo de erro que recebidos pela interface de rede. As quais também não foram passadas às camadas superiores.

TIPO DE OBJETO: ifInDiscards

SINTAXE: Counter32

OID BASE: interfaces

DESCRIÇÃO:

Indica o número de pacotes recebidos pela interface de rede que foram descartados mesmo sem conter erro algum. Estes mesmos não foram passados às camadas superiores. Uma possível razão poderia ser por falta de espaço em buffer de recepção.

TIPO DE OBJETO: ifOutOctets

SINTAXE: Counter32

OID BASE: interfaces

DESCRIÇÃO:

O número total de octetos transmitidos pela interface de rede, incluindo caracteres de framing.

TIPO DE OBJETO: ifOutDiscards

SINTAXE: Counter32

OID BASE: interfaces

DESCRIÇÃO:

Indica o número de pacotes a serem transmitidos pela interface de rede que foram descartados mesmo sem conter erro algum. Uma possível razão poderia ser por falta de espaço em buffer de envio.

TIPO DE OBJETO: ifOutErrors

SINTAXE: Counter32

OID BASE: interfaces

DESCRIÇÃO:

Para interfaces orientadas a pacotes, indica o número de pacotes com algum tipo de erro que, por isso tiveram sua transmissão cancelada. Para interfaces orientadas à caractere ou de comprimento fixo, indica o número de unidades de transmissão com algum tipo de erro que, por isso tiveram sua transmissão cancelada.

Segue abaixo a lista de objetos de informações básicas. Para maiores detalhes, ver RFC 3418.

TIPO DE OBJETO: sysUpTime

SINTAXE: TimeTicks

OID BASE: system

DESCRIÇÃO:

O tempo (em centésimos de segundo) desde que a funcionalidade de rede do host foi inicializado pela última vez.

TIPO DE OBJETO: sysName

SINTAXE: DisplayString (SIZE (0..255))

OID BASE: system

DESCRIÇÃO:

Nome administrativo do host.

Segue abaixo a lista de objetos de informações gerais do host. Para maiores detalhes, ver RFC 2790.

TIPO DE OBJETO: hrSystemUptime

SINTAXE:

OID BASE: host

DESCRIÇÃO:

O tempo desde o host (sobre o qual o agente é executado) foi inicializado pela última vez. Note que é diferente de sysUpTime, que é o tempo desde que a funcionalidade de rede do dispositivo foi inicializada pela última vez.

TIPO DE OBJETO: hrSystemDate

SINTAXE:

OID BASE: host

DESCRIÇÃO:

A noção do host de data e horário local.

TIPO DE OBJETO: hrSystemProcesses

SINTAXE: Gauge32

OID BASE: host

DESCRIÇÃO:

O número de processos de software atualmente sendo executados no host.

TIPO DE OBJETO: hrMemorySize

SINTAXE:

UNIDADE: KBytes

OID BASE: host

DESCRIÇÃO:

A quantidade de memória principal do host, tipicamente a RAM.

TIPO DE OBJETO: hrDeviceTypeSINTAXE:

OID BASE: host

DESCRIÇÃO:

Os diversos elementos(dispositivos) contidos no host são agrupados em 4 grupos:

- hrDeviceProcessor(3): indica que o dispositivo é um processador. Este elemento terá os objetos que o descrevem agrupados num objeto base hrProcessorTable.

- hrDeviceNetwork(4): indica que o dispositivo é um interface de rede. Este elemento terá os objetos que o descrevem agrupados num objeto base hrDeviceNetworkTable.

- hrDevicePrinter (5): indica que o dispositivo é um uma impressora. Este elemento terá os objetos que o descrevem agrupados num objeto base hrDevicePrinteTable.

- hrDeviceDiskStorage(6): indica que se trata de um elemento de armazenagem de dados. Este elemento terá os objetos que o descrevem agrupados num objeto base hrDeviceDiskStorageTable.

TIPO DE OBJETO: hrDeviceDescr

SINTAXE: DisplayString (SIZE (0..64))

OID BASE: host

DESCRIÇÃO:

Uma descrição textual do dispositivo, incluindo o fabricante, a versão e opcionalmente o número serial.

TIPO DE OBJETO: hrProcessorLoad

SINTAXE: Integer32 (0..100)

OID BASE: host

DESCRIÇÃO:

Porcentagem, sobre o último minuto, da carga de uso do processador em questão.

TIPO DE OBJETO: hrSWRunTable

SINTAXE: SEQUENCE OF

OID BASE: host
DESCRIÇÃO:
OID base que agrupa os OIDs relativos aos softwares sendo executados no host.

TIPO DE OBJETO: hrSWRunPerfCPU
SINTAXE: Integer32 (0..2147483647)
OID BASE: host
DESCRIÇÃO:
O número de centi-segundos consumidos dos recursos do host.

TIPO DE OBJETO: hrSWRunPerfMem
SINTAXE:
UNIDADES: KBytes
OID BASE: host
DESCRIÇÃO:
A quantidade total de memória consumida por este processo.

TIPO DE OBJETO: hrStorageIndex
SINTAXE: Integer32 (1..2147483647)
OID BASE: host
DESCRIÇÃO:
Um valor único para cada elemento de armazenamento contido no host.

TIPO DE OBJETO: hrStorageDescr
SINTAXE:
OID BASE: host
DESCRIÇÃO:
Uma descrição do tipo e instancia do elemento de armazenamento em questão

TIPO DE OBJETO: hrStorageAllocationUnits
SINTAXE: Integer32 (1..2147483647)
UNIDADE: Bytes
OID BASE: host
DESCRIÇÃO:
Unidade de alocação de memória, em bytes, usada como base para os objetos deste elemento de memória. Exemplo: os objetos hrStorageUsed e hrStorageSize (detalhados a seguir) de um objeto de memória, têm seus tamanhos dados na unidade hrStorageAllocationUnits do objeto de memória em questão.

TIPO DE OBJETO: hrStorageSize
SINTAXE: Integer32 (0..2147483647)
OID BASE: host
DESCRIÇÃO:
O tamanho total do elemento de memória na unidade definida no objeto hrStorageAllocationUnits do elemento. Exemplo: se a unidade citada acima for 1024 bytes e o valor do objeto hrStorageSize for 2, tem-se que o tamanho total do elemento de memória será 2048 bytes, ou 2KBytes.

TIPO DE OBJETO: hrStorageUsed

SINTAXE: Integer32 (0..2147483647)

OID BASE: host

DESCRIÇÃO:

O total de memória do elemento em questão usada no momento. Este total é medido na unidade definida no objeto hrStorageAllocationUnits do elemento.