**15 Лекция. Служебные протоколы. Протокол SNMP.**

**Служба SNMP**

Создана для стандартизации управления оборудованием разных производителей.

Например, в сети установлены разные маршрутизаторы (cisco, motorola, linux и т.д.), все настраиваются поразному. Нужно изменить настройки на всех маршрутизаторах. Обычным методом придется каждый настраивать индивидуально (интерфейсы у всех разные). С помощью SNMP можно настроить используя один интерфейс.

SNMP служит примером системы управления, в которой для достижения нужного результата не выдается команда, а осуществляется обмен информацией, решение принимается "на месте" в соответствии с полученными данными.

Объекты сети:

* Сервера управления - порт по умолчанию 162.
* Клиенты - сетевая станция, на который находится агент (программный модуль), позволяющий серверу управлять и наблюдать за ней. Порт по умолчанию 161.

Компоненты SNMP:

* **Протокол SNMP**
* **SMI - (Structure of Management Information)** - структура информации управления.
* **MIB (Management Information Base)** - информационная база управления.

**Протокол SNMP**

**SNMP (Simple Network Management Protocol)** - простой протокол управления сетью.

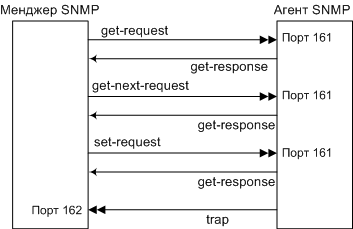
Первый стандарт SNMP определен в RFC1067 (Simple Network Management Protocol J.D. Case, M. Fedor, M.L. Schoffstall, J. Davin Aug-01-1988)

Последняя версия [RFC1157](http://ipm.kstu.ru/internet/doc/rfc/rfc1157.txt) (STD0015 Simple Network Management Protocol (SNMP) J.D. Case, M. Fedor, M.L. Schoffstall, J. Davin May-01-1990)

RFC1592 (Simple Network Management Protocol Distributed Protocol Interface Version 2.0 B. Wijnen, G. Carpenter, K. Curran, A. Sehgal, G. Waters March 1994)

Протокол прикладного уровня работает по умолчанию поверх **UDP,** но может работать по TCP.

Клиент и сервер обмениваются сообщениями.

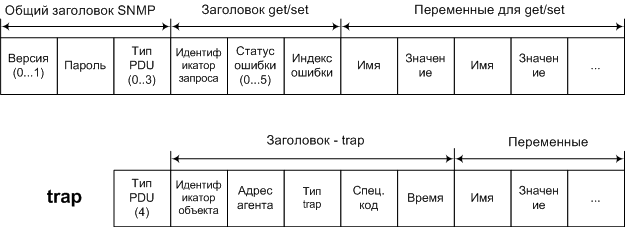


Взаимодействие клиент-сервера SNMP. Приведены все пять типов сообщений.

**PDU (Protocol Data Unit)** - блок (пакет) данных протокола.

Типы PDU сообщений SNMP

|  |  |  |
| --- | --- | --- |
| **Тип PDU** | **Имя** | **Значение** |
| 0 | get-request | Получить значение переменных |
| 1 | get-next-request | Получить следующие переменные после этой |
| 2 | set-request | Установить значение переменных |
| 3 | get-response | Выдать значение переменных (Посылает агент в ответ на get-request, get-next-request, set-request) |
| 4 | trap | Уведомить менеджера, когда что-либо произошло с агентом |



Формат SNMP-сообщений. Trap приведен отдельно.

Версия - содержит версию SNMP минус 1, т.е. для SNMPv1 версия=0

Статус ошибки - это целое число, которое возвращается агентам

Индекс ошибки - это целое смещение, указывающее на то, в какой переменной произошла ошибка.

Значения статуса ошибки SNMP.

|  |  |  |
| --- | --- | --- |
| **Статус ошибки** | **Имя** | **Значение** |
| 0 | noError | Все в порядке |
| 1 | noError | Клиент не может поместить отклик в одно SNMP сообщение |
| 2 | noSuchName | Оператор указывает на несуществующую переменную |
| 3 | badValue | В команде set использовано недопустимое значение или неправильный синтаксис |
| 4 | readOnly | Менеджер попытался изменить переменную, которая помечена как "только для чтения" |
| 5 | genErr | неопознанная ошибка |

Типы trap.

|  |  |  |
| --- | --- | --- |
| **Тип trap** | **Имя trap** | **Значение** |
| 0 | coldStart | Установление начального состояния объекта |
| 1 | wannStart | Восстановление начального состояния объекта |
| 2 | linkDown | Интерфейс выключился. Первая переменная в сообщении идентифицирует интерфейс |
| 3 | linkUp | Интерфейс включился. Первая переменная в сообщении идентифицирует интерфейс |
| 4 | authenticationFailure | От менеджера получено SNMP-сообщение с неверным паролем |
| 5 | egpNeighborLoss | EGP-партнер отключился. Первая переменная в сообщении определяет IP-адрес партнера |
| 6 | entrpriseSpeclfic | Информация о trap содержится в поле "Специальный код" |

Поле "Специальный код" для типов trap 0…4 поле должно быть равно нулю.

Время - содержит число сотых долей секунды (число тиков) с момента инициации объекта управления.

**Структура информации управления (SMI)**

**SMI - (Structure of Management Information)** - структура информации управления

Первый стандарт SMI определен в RFC1155 (Structure and identification of management information for TCP/IP-based internets M.T. Rose, K. McCloghrie May-01-1990)

Последний стандарт для версии SMIv1 RFC1155 (Structure and identification of management information for TCP/IP-based internets M.T. Rose, K. McCloghrie May-01-1990)

Последний стандарт для версии SMIv2 [RFC2578](http://ipm.kstu.ru/internet/doc/rfc/rfc2578.txt) (Structure of Management Information Version 2 (SMIv2) K. McCloghrie, D. Perkins, J. Schoenwaelder April 1999)

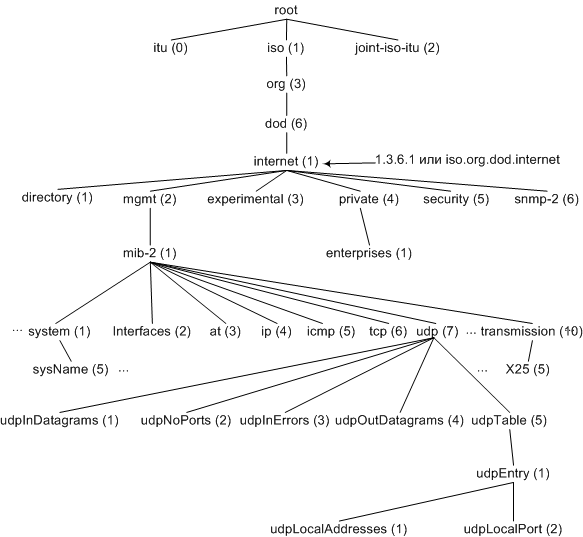
Некоторые типы данных:

* INTEGER - целое число
* OCTET STRING - восьмеричная строка, каждый байт имеет значение от 0 до 255.
* DisplayString - строка, каждый байт должен быть символом из набора ASCII NVT.
* OBJECT IDENTIFIER - идентификатор объекта - OID.
* NULL - ноль, означает, что у соответствующей переменной нет значения.
* IpAddress - IP адрес. восьмеричная строка длиной 4 байта.
* PhysAddress - физический адрес, восьмеричная строка, содержит физический адрес (например, 6-байтный Ethernet адрес).
* Counter - счетчик.
* Gauge - критерий.
* TimeTicks (тики времени). Счетчик, который считает время в сотых долях секунды с какой-либо исходной точки.
* SEQUENCE - последовательность.
* SEQUENCE OF - чего последовательность.

11.2.2.1 Дерево идентификаторов объектов

Все управляемые объекты глобальной сети расположены в дереве.

Идентификатор объекта (OID) это последовательность целых десятичных чисел, разделенных точками (1.4.2.1.6). Эти целые числа представляют собой древовидную структуру, напоминающую DNS



Дерево идентификаторов объектов в информационной базе управления

Все переменные в MIB начинаются с идентификатора объекта - **1.3.6.1.2.1 (iso.org.dod.internet.mgmt.mib-2)**

root - корень не имеет идентификатора

iso - администрирует дерево

org - организационный узел

dod - министерство обороны США

directory - справочник

mgmt - управление

enterprise - предприятия, например:

348 (1.3.6.1.2.1.348) - Procter&Gamble

743 (1.3.6.1.2.1.743) - ЦРУ

Например, ветвь 1.3.6.1.2.1.4 (iso.org.dod.internet.mgmt.mib-2.ip) дает информацию необходимую для управления компьютерами и маршрутизаторами.

**Информационная база управления (MIB)**

Первый стандарт MIB определен в RFC1066 (Management Information Base for network management of TCP/IP-based internets K. McCloghrie, M.T. Rose Aug-01-1988 )

Последний стандарт для версии MIB-I RFC1156 (Management Information Base for network management of TCP/IP-based internets K. McCloghrie, M.T. Rose May-01-1990)

Последний стандарт для версии MIB-II [RFC1213](http://ipm.kstu.ru/internet/doc/rfc/rfc1213.txt) (STD0017 Management Information Base for Network Management of TCP/IP-based internets:MIB-II K. McCloghrie, M.T. Rose Mar-01-1991)

Ветвь 1.3.6.1.2.1.4 (iso.org.dod.internet.mgmt.mib-2).

Расмотрим подробнее ветвь UDP (рис. выше).

Группа UDP содержит четыре переменные, и одну таблицу (udpTable) из двух переменных.

Переменные группы udp

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Тип данных** | **Чтение/Запись** | **Описание** |
| udpInDatagrams | Counter | Только чтение | Количество UDP датаграмм, доставленных пользовательским процессам. |
| udpNoPorts | Counter | Только чтение | Количество доставленных UDP датаграмм, для которых не оказалось порта назначения. |
| udpInErrors | Counter | Только чтение | Количество недоставленных UDP датаграмм по другим причине (например, ошибка контрольной суммы UDP). |
| udpOutDatagrams | Counter | Только чтение | Количество отправленных UDP датаграмм. |

Как видно из таблицы эти переменные обеспечивают полный сбор статистики для UDP-протокола.

Переменные в udpTable.

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Тип данных** | **Чтение/Запись** | **Описание** |
| udpLocalAddress | IpAddress | Только чтение | Локальный IP адрес слушающего процесса. |
| udpLocalPort | INTEGER [0..65535] | Только чтение | Локальный номер порта слушающего процесса. |

index = <udpLocalAddress>.<udpLocalPort>

Примеры идентификации

Каждая переменная в MIB должна быть идентифицирована.

Обращение осуществляется только к тупиковым (в дереве) узлам.

Простые переменные

На то, что эта переменная простая, указывает ".0", добавленный к идентификатору объекта переменной. Например, к счетчику udpInErrors , c идентификатором объекта 1.3.6.1.2.1.7.3, можно обратиться как 1.3.6.1.2.1.7.3.0. Текстовое имя при подобном обращении будет iso.org.dod.internet.mgmt.mib.udp.udpInErrors.0.

Обращения к этой переменной чаше делаются в сокращенном виде, udpInErrors.0, т.к. реально идет обращение к идентификатору объекта 1.3.6.1.2.1.7.1.0.

Таблицы

Рассмотрим идентификацию пунктов таблицы udpTable более подробно.

Для каждой таблицы в MIB указан один или несколько индексов. Для таблицы слушающего процесса UDP, MIB определяет индекс как комбинацию двух переменных udpLocalAddress (локальный IP адрес) и udpLocalPort (локальный UDP порт), индекс в данном случае - целое число.

Пример таблицы udpTable.

|  |  |
| --- | --- |
| udpLocalAddress | udpLocalPort |
| 0.0.0.0 | 53 |
| 0.0.0.0 | 67 |
| 0.0.0.0 | 161 |

Из таблицы видно, что система готова принимать UDP датаграммы с любого интерфейса (0.0.0.0) для портов 53 (DNS), 67 (BOOTP) и 161 (SNMP).

11.2.3.1.3 Абстрактная форма записи

**ASN.1 (Abstract Syntax Notation 1) -**абстрактная форма записии.

Все поля в MIB и SNMP сообщениях описываются с использованием ASN.1.

Например, ASN.1 определение переменной udpNoPorts выглядит так:

udpNoPorts OBJECT-TYPE  
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
"The total number of received UDP datagrams for which there  
was no application at the destination port."  
::= { udp 2 }

**SNMP версии 2**

Новое в этой версии:

* get-bulk-request - позволяет менеджеру эффективно обрабатывать большие блоки данных.
* inform-request - позволяет одному менеджеру посылать информацию другому менеджеру.
* Определены два новых MIB: MIB SNMPv2 и MIB SNMPv2-M2M (менеджер-менеджер).
* Улучшенна секретность.

**Программы для работы с SNMP**

**Whatsup** - многофункциональная утилита (официальный сервер - <http://www.ipswitch.com/> ).

**SolarWinds** - многофункциональная утилита (официальный сервер - <http://www.solarwinds.net/> ).

**MRTG** - сбор статистики по SNMP, предоставляет данные в WEB-интерфейсе [http://people.ee.ethz.ch/~oetiker/webtools/mrtg/](http://people.ee.ethz.ch/%7Eoetiker/webtools/mrtg/)