

Конфигурация сигнала тревоги RMON и настройки событий с помощью команд SNMP

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Общие сведения](#)

[Пошаговая процедура](#)

[Создание события](#)

[Создание сигнала](#)

[Пример](#)

[Проверка](#)

[Устранение неполадок](#)

[Дополнительные сведения](#)

Введение

Этот документ предоставляет пример конфигурации для сигнала тревоги удаленного мониторинга (RMON) и параметров настройки события с помощью команд SNMP.

Предварительные условия

Требования

Для этого документа отсутствуют особые требования.

Используемые компоненты

Чтобы выполнить процедуры данного документа, устройство должно поддерживать RMON-MIB. Можно проверить это в [Программных средствах MIB Cisco IOS \(только зарегистрированные клиенты\)](#).

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

Условные обозначения

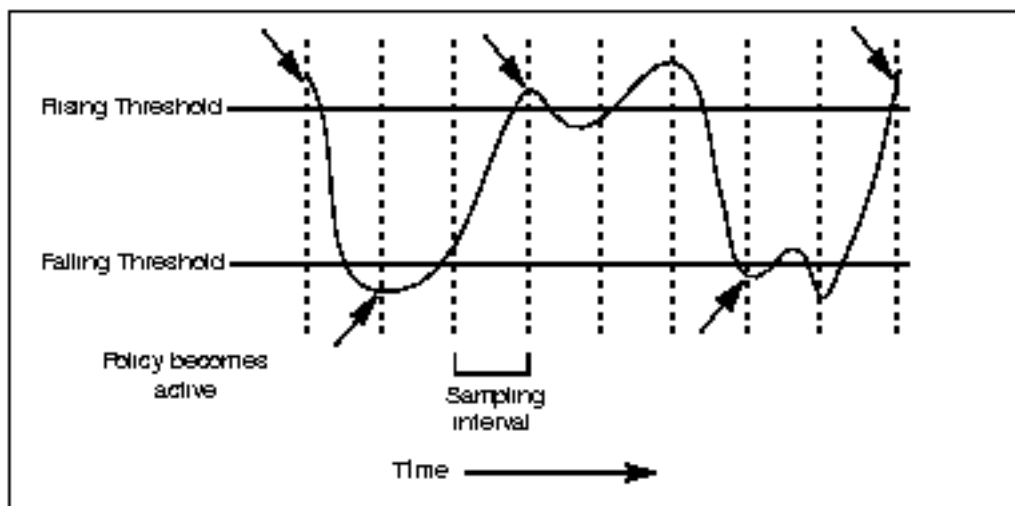
Общие сведения

Назначение сигнала и события удаленного мониторинга (RMON) заключается в том, чтобы контролировать определенный объект MIB на устройстве и предостерегать системного администратора, если одно из этих значений выходит из установленного диапазона.

Когда условие (падение или верхний порог) достигнуто, сигнал тревоги контролирует конкретный объект в MIB и инициирует событие.

Событие является trap-сообщением или журналом, генерируемым, когда сигнал тревоги инициирует его. Пример порога подъема и падения:

n =value monitored by the alarm. The falling threshold is "5" and the rising threshold is "15"
Сигналы тревоги запускают событие при достижении одного из двух значений.



Значение	Трап-сообщение	Комментарий
n1=16	повышение	достигнутый возрастающее значение: 15
n2=4	падение	достигнутый падающее значение: 5
n3=6	нет	между 5 и 15
n4=6	повышение	достигнутый возрастающее значение: 15
n5=13	нет	хорошее состояние
n6=20	нет	выше 15, но не пошел под 5 начиная с последнего события
n7=4	падение	достигнутый падающее значение: 5

n8=20	нет	под 5, но не выходил за предел 15 начиная с последнего события
n9=16	повышение	достигнутый возрастающее значение: 15

Можно настроить Сигнальное оповещение "rmon", и событие на маршрутизаторах с помощью интерфейса командной строки (CLI) (обратитесь к [Сигнальному оповещению "rmon" Настройки и Параметрам настройки события от Интерфейса командной строки](#)), и на маршрутизаторах и коммутаторах с помощью команд Протокола SNMP. Параметры для изменения являются частью [RMON-MIB](#).

Пошаговая процедура

Создание события

Используйте следующую команду для формирования события:

```
# snmpset -c <read_write_community> <device_name> .1.3.6.1.2.1.16.9.1.1.x.y <variable type>
<value>
```

Сначала выберите идентификатор события (переменная y).

Выполните эту процедуру для создания события. Для каждого шага существует описание шага, название объекта MIB для изменения, идентификатор объекта (OID), <тип переменной> и <value> от групповой команды.

1. Стереть завершенное устаревшее событие, которое имело идентификатор "y" (убедитесь, что в нем больше нет необходимости). В противном случае используйте другой ID).* eventStatus / .1.3.6.1.2.1.16.9.1.1.7.y
* variable type=integer
* value=4

Примечание: Используйте ту же команду для стирания события при необходимости.

2. Войдите в режим создания событий:* eventStatus / .1.3.6.1.2.1.16.9.1.1.7.y
* variable type=integer
* value v=2
3. Задайте описание события:* eventDescription / .1.3.6.1.2.1.16.9.1.1.2.y
* variable type=string (for Net-snmp) or octetsting (for Openview)
* value = a description of the event
4. Задайте требуемый тип события:* eventType / .1.3.6.1.2.1.16.9.1.1.3.y
* variable type=integer
* value =
"1" => none
"2" => log
"3" => snmp-trap
"4" => log-and-trap
5. Определите строку имени и пароля сообщества для ловушки: * eventCommunity / .1.3.6.1.2.1.16.9.1.1.4.y
* variable type=string (for Net-snmp) or octetsting (for Openview)
* value="<trap_community_string>"
6. Укажите владельца события:* eventOwner / .1.3.6.1.2.1.16.9.1.1.6.y
* variable type=string (for Net-snmp) or octetsting (for Openview)

```
* value=<event_owner>"
```

7. Активизируйте событие: * eventStatus / .1.3.6.1.2.1.16.9.1.1.7.y

```
* variable type=integer
```

```
* value=1
```

Создание сигнала

Используйте данную команду, чтобы создать сигнал:

```
# snmpset -c .1.3.6.1.2.1.16.3.1.1.x.y <read_write_community> <device_name> <variable type> <value>
```

1. Сотрите возможный старый сигнал тревоги, который использовал бы ID=y (первая проверка, если вам больше не нужен он. В противном случае используйте другой ID):*

```
alarmStatus / .1.3.6.1.2.1.16.3.1.1.12.y
```

```
* variable type=integer
```

```
* value=4
```

2. Введите сигнальный режим создания: * alarmStatus / .1.3.6.1.2.1.16.3.1.1.12.y

```
* variable type=integer
```

```
* value=2
```

3. Установите интервал (в секундах), по истечении которого данные измеряются и сравниваются с пороговыми величинами возрастания и падения: * alarmInterval / .1.3.6.1.2.1.16.3.1.1.2.y

```
* variable type=integer
```

```
* value=<n_seconds>
```

4. Укажите OID, который требуется контролировать: * alarmVariable /

```
.1.3.6.1.2.1.16.3.1.1.3.y
```

```
* variable type=objid (for Net-snmp) or objectidentifier (for Openview)
```

```
* value=<oid_to_check>
```

5. Определите тип выборки, которую вы хотите: * alarmSampleType /

```
.1.3.6.1.2.1.16.3.1.1.4.y
```

```
* variable type=integer
```

```
* value=<rising_threshold> "1" => absoluteValue "2" => deltaValue
```

6. Определите условие формирования сигнала тревоги: * alarmStartupAlarm /

```
.1.3.6.1.2.1.16.3.1.1.6.y
```

```
* variable type=integer
```

```
* value=
```

```
"1" => risingAlarm
```

```
"2" => fallingAlarm
```

```
"3" => risingOrFallingAlarm
```

7. Определите верхний порог: * alarmRisingThreshold / .1.3.6.1.2.1.16.3.1.1.7.y

```
* variable type=integer
```

```
* value=<rising_threshold>
```

8. Определите порог снижения: * alarmFallingThreshold / .1.3.6.1.2.1.16.3.1.1.8.y

```
* variable type=integer
```

```
* value=<falling_threshold>
```

9. Определите идентификатор события, которое инициируется при превышении верхнего порога: * alarmRisingEventIndex / .1.3.6.1.2.1.16.3.1.1.9.y

```
* variable type=integer
```

```
* value=<event_ID>
```

10. Укажите ID события, когда был пересечен нижний порог: * alarmFallingEventIndex /

```
.1.3.6.1.2.1.16.3.1.1.9.y
```

```
* variable type=integer
```

```
* value=<event_ID>
```

11. Определите владельца сигнала тревоги: * alarmOwner / .1.3.6.1.2.1.16.3.1.1.11.y

```
* variable type=string (for Net-snmp) or octetsting (for Openview)
```

```
* value=<owner>
```

12. Активируйте сигнал тревоги: * alarmStatus / .1.3.6.1.2.1.16.3.1.1.12.y

- * variable type=integer
- * value=1

Пример

Когда количество байтов, входящих в интерфейс 12 в течение прошлых двух минут, выше 14000000 или ниже 10, в данном примере **сафари** используется для передачи trap-сообщения.

Safari является программным обеспечением Cisco IOS 2500 (C2500-JS-L), Версия 12.1 (9), РЕЛИЗ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ (fc1).

Данный пример также попробовали успешно на программном обеспечении WS-C6506, Версия NmpSW: 6.1 (1b).

Примечание: В коммутаторах Catalyst не предусмотрена команда CLI для проверки конфигурации. Для проверки конфигурации выполните команду snmpwalk на сервере.

На маршрутизаторе и коммутаторе данная конфигурация сохраняет работоспособность после перезагрузки.

```
safari# show rmon events Event table is empty # snmpset -c private safari
.1.3.6.1.2.1.16.9.1.1.7.123 integer 4 16.9.1.1.7.123 = 4 # snmpset -c private safari
.1.3.6.1.2.1.16.9.1.1.7.123 integer 2 16.9.1.1.7.123 = 2 safari#show rmon events Event 123 is
under creation, owned by Description is Event firing causes nothing, last fired 00:00:00 #
snmpset -c private safari .1.3.6.1.2.1.16.9.1.1.2.123 string "test_event" 16.9.1.1.2.123 =
"test_event" # snmpset -c private safari .1.3.6.1.2.1.16.9.1.1.3.123 integer 4 16.9.1.1.3.123 =
4 # snmpset -c private safari .1.3.6.1.2.1.16.9.1.1.4.123 string "public" 16.9.1.1.4.123 =
"public" # snmpset -c private safari .1.3.6.1.2.1.16.9.1.1.6.123 string "event_owner"
16.9.1.1.6.123 = "event_owner" # snmpset -c private safari .1.3.6.1.2.1.16.9.1.1.7.123 integer 1
16.9.1.1.7.123 = 1 safari# show rmon events Event 123 is active, owned by event_owner
Description is test_event Event firing causes log and trap to community public, last fired
00:00:00 safari# show rmon alarm Alarm table is empty # snmpset -c private safari
.1.3.6.1.2.1.16.3.1.1.12.321 integer 2 16.3.1.1.12.321 = 2 safari# show rmon alarm Alarm 321 is
under creation, owned by Monitors ccitt.0 every 10 second(s) Taking absolute samples, last value
was 0 Rising threshold is 0, assigned to event 0 Falling threshold is 0, assigned to event 0 On
startup enable rising or falling alarm # snmpset -c private safari .1.3.6.1.2.1.16.3.1.1.2.321
integer 120 16.3.1.1.2.321 = 120 # snmpset -c private safari .1.3.6.1.2.1.16.3.1.1.3.321 objid
.1.3.6.1.2.1.2.2.1.10.12 16.3.1.1.3.321 = OID: interfaces.ifTable.ifEntry.ifInOctets.12 #
snmpset -c private safari .1.3.6.1.2.1.16.3.1.1.4.321 integer 2 16.3.1.1.4.321 = 2 # snmpset -c
private safari .1.3.6.1.2.1.16.3.1.1.6.321 integer 3 16.3.1.1.6.321 = 3 # snmpset -c private
safari .1.3.6.1.2.1.16.3.1.1.7.321 integer 14000000 16.3.1.1.7.321 = 14000000 # snmpset -c
private safari .1.3.6.1.2.1.16.3.1.1.8.321 integer 10 16.3.1.1.8.321 = 10 # snmpset -c private
safari .1.3.6.1.2.1.16.3.1.1.9.321 integer 123 16.3.1.1.9.321 = 123 # snmpset -c private safari
.1.3.6.1.2.1.16.3.1.1.10.321 integer 123 16.3.1.1.10.321 = 123 # snmpset -c private safari
.1.3.6.1.2.1.16.3.1.1.11.321 string "alarm_owner" 16.3.1.1.11.321 = "alarm_owner" # snmpset -c
private safari .1.3.6.1.2.1.16.3.1.1.12.321 integer 1 16.3.1.1.12.321 = 1 safari# show rmon
alarm Alarm 321 is active, owned by alarm_owner Monitors ifEntry.10.1 every 120 second(s) Taking
delta samples, last value was 130244 Rising threshold is 14000000, assigned to event 123
Falling threshold is 10, assigned to event 123 On startup enable rising or falling alarm
```

Проверка

В настоящее время для этой конфигурации нет процедуры проверки.

Устранение неполадок

Для этой конфигурации в настоящее время нет сведений об устранении проблем.

Дополнительные сведения

- [Сигнальное оповещение "rmon" Настройки и параметры настройки события от интерфейса командной строки](#)
- [Поддержка события MIB](#)
- [RFC 1757](#)
- [Техническая поддержка - Cisco Systems](#)