SNMP traps и с чем их едят

Опубликовано 24.04.2012 автором asmi

В этой статье я хочу осветить вопрос работы с SNMP traps. В то время, когда я начинал разбираться с этой темой, при неплохом знании принципов работы протокола SNMP, я был удивительным образом перепутан результатами чтения документации по этому вопросу, всеми этими OID, которые нужно писать как параметры для команды snmptrap. Поиски внятной информации в интернет не дали результатов. Создавалось впечатление, что я один такой непонятливый среди тех, кто считает это элементарным.

Поиск

Свежие записи

- Mod_verto лёгкая Javascript сигнализация для freeswitch или как заменить тяжелый SIP на что то более простое
- Крошечный снипет, чтобы не показывать изображения, которые отсутствуют
- Основа настройки S-terra NME-RVPN часть 2
- Основа настройки S-terra NME-RVPN

В своей работе я использую пакет NetSNMP с открытым исходным кодом и поэтому, примеры относятся именно к этому пакету.

Я не буду останавливаться на описании дерева SNMP (считаю это действительно банальным), а начнем с пресловутого OID. OID это просто. OID, это Object ID, уникальный идентификатор объекта в дереве значений SNMP. OID может записываться в нескольких видах.

1. Числовой. Например, `1.3.6.1.4.1.2021`. Самый простой вид, по причине полного отсутствия каких либо трансформаций программным обеспечением. Именно в таком виде SNMP „думает“.
2. Текстовый, с использованием MIB файлов. Понятно, что дикие цифры OID запомнить категорически невозможно, поэтому с помощью MIB можно вводить текстовые мнемонические имена для более простого вспоминания и использования идентификаторов. Кроме того, текстовая запись позволяет избежать написания OID от корня дерева SNMP. Уникальные мнемонические имена уже содержат путь до необходимого узла дерева. Отсюда возможны записи вида enterprises.ucdavis.

Так как уникальность мнемонического имени в пределах всех возможных MIB никто не гарантирует и не всегда сразу вспомнишь к чему относится имя (например prTable), существует следующая, на мой взгляд самая приятная форма записи: `UCD-SNMP-MIB::prTable`. В таком формате, кроме самого имени указывается имя MIB (можно сказать, это пространство имен)

Единственным ограничением использования MIB является то, что при использовании программного обеспечения SNMP от разных производителей и различного железа необходимо единое „понимание“ этих волшебных

- MySQL, InnoDB, Foreign keys и большие буквы

Свежие комментарии

- asmi к записи Poor man kiosk или как настроить firefox для киоска
- Mihanbka к записи Poor man kiosk или как настроить firefox для киоска
- DaniCh к записи Poor man kiosk или как настроить firefox для киоска
- asmi к записи Основа настройки S-terra NME-RVPN часть 2
- asmi к записи SNMP traps и с чем их едят

Архивы

- Октябрь 2015
- Август 2015
- Март 2013
- Январь 2013
- Май 2012
- Апрель 2012
- Май 2010

Рубрики

- Cisco
- JS Frontend
- ЧАВО

Мета

- Войти
- RSS записей
- RSS комментариев
- WordPress.org

трансформаций. Поэтому на сайтах производителей управляемого оборудования есть файлы описания MIB для конкретных устройств.

С представлением OID разобрались. Теперь про трапы. Сразу оговорюсь, что все нижесказанное относится к версии 2c SNMP.

В протоколе SNMP есть трапы (traps) и есть информы (informs). Отличаются они тем, что трапы посылаются на станцию менеджера без гарантии их доставки, а информы гарантированно доставляются. Ну, или не доставляться, но тогда отправитель об этом узнает. Для простоты, я и то и другое называю трапом. Так нам будет прощеi.

Какая информация посылается менеджеру в трапе (информе)? А посылаются следующие данные:

- Uptime устройства в виде пары: стандартный OID uptime, значение
- OID, который говорит о том, что за событие приключилось. В MIB файлах есть специальные описания некоторого количества стандартных событий, вроде запуска или остановки агента. Также, мы можем определить собственные trap OIDs и даже описать их в собственном MIB файле.
- Любые пары OID и его значение (там еще есть и третье поле „тип значения“, но это нам побоку), которые могут дать дополнительную информацию. Например, когда свитч посылает трап „Падение линка“, дополнительное поле будет содержать информацию OID интерфейса, с которым приключилась неприятность — `IF-MIB:ifIndex.2`.

Теперь, наше повествование разделяется на направления. Первое направление: „Как нам послать SNMP trap ручками из командной строки„. Второе направлении „Как

нам заставить агента SNMP посылать трапы когда что то идет не по плану“.

Посылаем руками

Допустим, нам хочется послать сообщение менеджеру о том, что демон net-snmpd запустился и передать дополнительную информацию о размере swap файла и все это ручками ??? Ну пришла нам такая блажь. Допустим. Пишем:

```
snmpinform -c public -v 2c host.sample.com "" ucdavis.ucdTraps.ucdStart
memTotalSwap i 1024000
```

Что здесь что ?

- `""` — брать uptime по дефолту
- `ucdavis.ucdTraps.ucdStart` — послать этот трап
- `memTotalSwap i 1024000` — добавить в нагрузку OID memTotalSwap равное целому 1024000

И если на целевом хосте запущен демон `snmptrapd`, то в логе появится сообщение о приходе трапа.

Однако, иногда хочется странного, например послать трап о том, что инопланетяне тырят SCSI винчестеры прямо из корзинки или о том, что серверную заливает водой, о чем вас предупредительно оповестил самодельный датчик, которые весит на COM порту. Понятное дело, в стандартных MIB о таких странных ситуациях не упоминается. Поэтому, можно использовать произвольную последовательность цифирек для обозначения нового в мире события. Например:

```
snmptrap -c public -v 2c 127.0.0.1 "" 1.3.3.3.3.3.3.3 1.2.2.2.2.2.2 s "Aliens
opened the door"
```

И что характерно, это будет работать. В логе появится такая бредятинка:

```
Nov 22 14:08:24 snmptrapd[465]: localhost [127.0.0.1]: Trap , DISMAN-EVENT-
MIB::sysUpTimeInstance = Timeticks: (736247) 2:02:42.47, SNMPv2-MIB::snmpTrapOID.0
= OID: SNMPv2-SMI::org.3.3.3.3.3.3, iso.2.2.2.2.2.2 = STRING: " Aliens opened the
door"
```

В то же время, крутые администраторы должны все делать правильно, самодокументируемо, в соответствии с требованиями IETF и других компетентных организаций. Поэтому, будем описывать наши новые OID в своем собственном MIB файле.

```
ALLIENSATTACK-MIB DEFINITIONS ::= BEGIN
IMPORTS
    OBJECT-TYPE, NOTIFICATION-TYPE, MODULE-IDENTITY,
    Integer32, Opaque, enterprises, Counter32
        FROM SNMPv2-SMI

alliensattack MODULE-IDENTITY
    LAST-UPDATED "200209050000Z"
    ORGANIZATION "XCom"
    CONTACT-INFO
        "Planet Earth
        "
    DESCRIPTION
        "MIB for preventing aliens to stole our SCSI disks"
```

```
    ::= { enterprises 10050 }

attackTraps OBJECT IDENTIFIER ::= { alliensattack 1 }

attackStartTrap NOTIFICATION-TYPE
    STATUS      current
    DESCRIPTION
        "Notify about start of attack"
    ::= { attackTraps 1 }

attackStopTrap  NOTIFICATION-TYPE
    STATUS current
    DESCRIPTION
        "Notify about time to leave our hideouts"
    ::= { attackTraps 2 }

attackSource OBJECT IDENTIFIER ::= { aliensattack 2 }

attackFromDoors  OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Sent to manager when attack starts from doors"
   ::= { attackSource 1 }

attackFromWindows  OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "May be aliens are innocent? And SCSI drives disappeared in th
```

```
    ::= { attackSource 2 }

END
```

Здесь определяется enterptise с номером 10050 и ему даеся имя aliensattack. После этого вводятся две ветки: одна для трапов (начало и конец атаки), другая для OID дополнительной информации (атака через двери или это вовсе не инопланетяне).

Скажу честно, в формате MIB файла я понимаю мало. Поэтому внимательно изучив MIB файлы из стандартной поставки net-snmp, я сделал этот пример который работает, но кристальной ясности во всех его ключевых словах я не достиг.

Дальше! Укладываем этот файл с именем ALLIENSATTACK-MIB.txt в какой нибудь каталог поближе и запоминаем, что с этого момента все программы имеющие отношения к SNMP (snmpd, snmptrapd, snmpinform) мы запускаем с поддержкой этого MIB. Тоесть, к командной строке добавляем параметры

```
-M каталог/с_файлом -m +ALLIENSATTACK-MIB.
```

Знак „+“ перед именем MIB (не файла с MIB, а именно имя MIB), значит, что MIB будет добавлен к загружаемым по умолчанию.

Посылаем автоматически

Способ #2. Пишем скрипт, который проверяет системные параметры и посылает трап на станцию менеджера. Пишется на shell, perl или на любом известном языке.

Способ #1. Наиболее интересный. В 5-й версии net-snmp появиласть поддержка модуля DISMAN-EVENT-MIB котрый умеет отсылать сообщения при некоторых условиях. Но это уже другая история.

Запись опубликована в рубрике ЧАВО. Добавьте в закладки постоянную ссылку.

← Подключаемся к Avaya Xterm-om

Sendmail и разные хитрые фишки →

6 комментариев: *SNMP traps и с чем их едят*

Владимир *говорит:*

27.09.2012 в 17:43

Спасибо за годную статью.
Улыбнуло, и дало нужную информацию — о том, что OID трапа может быть произвольным.

Добавлю, что при выборе произвольного OID трапа главное сделать так, чтобы он не пересекся с каким-нибудь уже существующим стандартным OID. Иначе может получиться ситуация, когда ваш сервер trapd ждет сообщения о нападении инопланетян, а ему приходит обычное сообщение о поднятии линка.. Но их OID совпадают — и начинается паника.

Ответить

asmi *говорит:*

24.01.2013 в 15:56

Пожалуйста!
Паника при одинаковых OID точно будет!

Ответить

Сергей *говорит:*

28.03.2013 в 22:33

Приветствую. Немного не в тему .Пытаюсь настроить SNMP карту в UPSe IPPON , ищу информацию по настройке т.к. мануал с ней на аглицком,а машинный перевод не дает добрастся до сути. Вопрос такой как и куда загрузить MIB файлы с диска идущего с ней ,на карту если я правильно понимаю или они там находятся при поставке » железяки» с завода.

Ответить

asmi *говорит:*

29.03.2013 в 15:25

В случае с ucd-snmp файлы нужно класть туда, где хранятся другие файлы с MIB дистрибутива NET-SNMP. Давно я этим не занимался, там вроде еще есть файл index в котором перечисляются все MIB которые есть в директории.

Ответить

---

bush *говорит:*

20.09.2013 в 09:58

Здравствуйте!
Добавлю, что иногда получить нужные мибы от производителя непросто,

если невозможно..
В этом случае необходимо на оборудование «натравить» миб-браузер, используя GET BULK
Собственно смотрим до самого конца, что именно может ответить железяка по всем возможным параметрам.....
Например была необходимость получить серийные номера коммутаторов ZyXEL, нигде информации такое не было, и как раз с помощью булки серийники были обнаружены...
вот пример:

zyxel
3012F .1.3.6.1.4.1.890.1.5.8.11.1.10.0 0Z0720xxx5yy
3124-4F .1.3.6.1.4.1.890.1.5.8.26.1.10.0 S080Z4xxx77yy
3712F .1.3.6.1.4.1.890.1.5.8.48.1.10.0 S090Z1xxx43yy
Как видно из примера, даже у одного производителя видимо работает большая команда индусов, которая правит мибы как хочется..

Ответить

asmi *говорит:*

09.01.2014 в 10:24

Способ получения всех OID годный, однако, часто не понятно что под этими OID подразумевается. Остается только догадываться.

Ответить

Добавить комментарий

Ваш e-mail не будет опубликован. Обязательные поля помечены *

Имя *

E-mail *

Сайт

Комментарий

Можно использовать следующие HTML-теги и атрибуты: `<a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code> <del datetime=""> <em> <i> <q cite=""> <strike> <strong>`

Отправить комментарий