SNMP 技术白皮书

Copyright © 2019紫光恒越技术有限公司 版权所有，保留一切权利。
非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。
除紫光恒越技术有限公司的商标外，本手册中出现的其它公司的商标、产品标识及商品名称，由各自权利人拥有。本文中的内容为通用性技术信息，某些信息可能不适用于您所购买的产品。

目 录

1 概述 ····· 1
1.1 产生背景 ····· 1
1.2 技术优点 ····· 1
2 SNMP 技术实现 ····· 1
2.1 SNMP 网络架构 ····· 1
1.2.1 NMS 简介 ····· 1
1.2.2 Agent 简介 ····· 2
1.2.3 MIB 简介 ····· 2
2.2 SNMP 版本 ····· 3
1.2.4 SNMPv1 ····· 4
1.2.5 SNMPv2c ····· 4
1.2.6 SNMPv3 ····· 4
2.3 SNMP 操作 ····· 4
2.4 SNMP 报文 ····· 4
1.2.7 SNMPv1 报文 ····· 5
1.2.8 SNMPv2c 报文 ····· 6
1.2.9 SNMPv3 报文 ····· 6
2.5 SNMP 协议原理 ····· 7
1.2.10 SNMPv1 和 SNMPv2c 实现机制 ····· 7
1.2.11 SNMPv3 实现机制 ····· 8
2.6 SNMP 静默功能 ····· 9
2 实现的技术特色 ····· 9 3 典型组网应用 ····· 10 3.1 组网图 ····· 10
4 参考文献 ····· 10

1 概述

1.1 产生背景

如今，网络与人们的生活和工作联系越来越紧密，但随着网络的普及引发了两大问题：

- 网络规模逐渐增大，网络设备的数量成级数增加，网络管理员很难及时监控所有设备的状态、发现并修复故障。
- 网络设备可能来自不同的厂商，如果每个厂商都提供一套独立的管理接口（比如命令行），将使网络管理变得越来越复杂。

为解决以上两大问题，一套覆盖服务、协议和管理信息库的标准（SNMP）应运而生。

1.2 技术优点

SNMP 是管理进程（NMS）和代理进程（Agent）之间的通信协议。它规定了在网络环境中对设备进行监视和管理的标准化管理框架、通信的公共语言、相应的安全和访问控制机制。网络管理员使用 SNMP 功能可以查询设备信息、修改设备的参数值、监控设备状态、自动发现网络故障、生成报告等。

SNMP 具有以下技术优点：

- 基于 TCP/IP 互联网的标准协议，传输层协议一般采用 UDP。
- 自动化网络管理。网络管理员可以利用 SNMP 平台在网络上的节点检索信息、修改信息、发现故障、完成故障诊断、进行容量规划和生成报告。
- 屏蔽不同设备的物理差异，实现对不同厂商产品的自动化管理。SNMP 只提供最基本的功能集，使得管理任务与被管设备的物理特性和实际网络类型相对独立，从而实现对不同厂商设备的管理。
- 简单的请求—应答方式和主动通告方式相结合，并有超时和重传机制。
- 报文种类少，报文格式简单，方便解析，易于实现。
- SNMPv3 版本提供了认证和加密安全机制，以及基于用户和视图的访问控制功能，增强了安全性。

2 SNMP 技术实现

2.1 SNMP网络架构

SNMP 网络架构由三部分组成：NMS、Agent 和 MIB。

1.2.1 NMS简介

NMS 是网络中的管理者，是一个利用 SNMP 协议对网络设备进行管理和监视的系统。NMS 既可以指一台专门用来进行网络管理的服务器，也可以指某个设备中执行管理功能的一个应用程序。

NMS 可以向 Agent 发出请求，查询或修改一个或多个具体的参数值。同时，NMS 可以接收 Agent 主动发送的 Trap 信息，以获知被管理设备当前的状态。

1.2.2 Agent简介

Agent 是网络设备中的一个应用模块，用于维护被管理设备的信息数据并响应 NMS 的请求，把管理数据汇报给发送请求的 NMS。

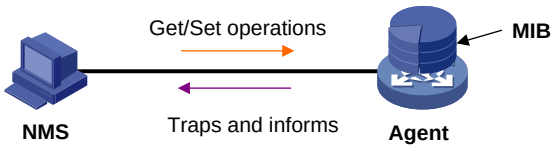
Agent 接收到 NMS 的请求信息后，完成查询或修改操作，并把操作结果发送给 NMS，完成响应。同时，当设备发生故障或者其他事件的时候，Agent 会主动发送 Trap 信息给 NMS，通知设备当前的状态变化。

1.2.3 MIB简介

1. MIB

任何一个被管理的资源都表示成一个对象，称为被管理的对象。MIB 是被管理对象的集合。它定义了被管理对象的一系列属性：对象的名称、对象的访问权限和对象的数据类型等。每个 Agent 都有自己的 MIB。MIB 也可以看作是 NMS 和 Agent 之间的一个接口，通过这个接口，NMS 可以对 Agent 中的每一个被管理对象进行读/写操作，从而达到管理和监控设备的目的。NMS、Agent 和 MIB 之间的关系如图 1 所示。

图1 NMS、Agent 和 MIB 关系图



2. MIB 视图

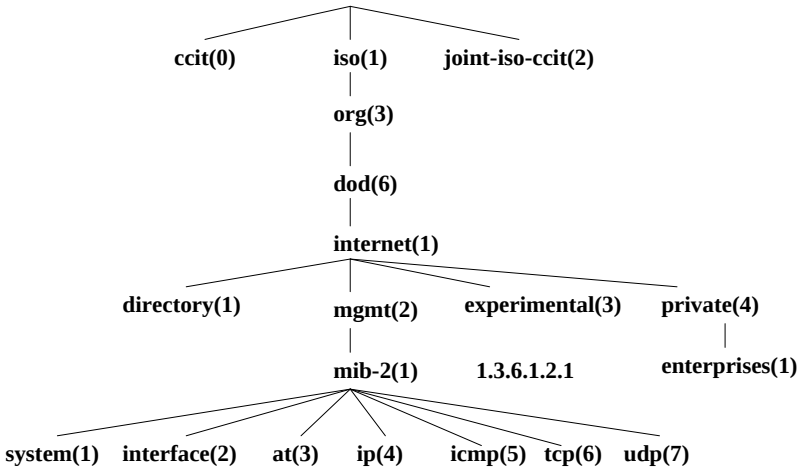
MIB 视图是 MIB 的子集合，配置 Agent 时用户可以将团体名/用户名与 MIB 视图绑定，从而限制 NMS 能够访问的 MIB 对象。用户可以配置 MIB 视图内的对象为 excluded 或 included。excluded 表示当前视图不包括该 MIB 子树的所有节点；included 表示当前视图包括该 MIB 子树的所有节点。

3. OID 和子树

MIB 是以树状结构进行存储的。树的节点表示被管理对象，它可以用从根开始的一条路径唯一地识别，这条路径就称为 OID）。如图 2 所示。管理对象 system 可以用一串数字{1.3.6.1.2.1.1}唯一标识，这串数字就是 system 的 OID。

子树可以用该子树根节点的 OID 来标识。如以 private 为根节点的子树的 OID 为 private 的 OID——{1.3.6.1.4}。

图2 MIB 树结构示意图



4. 子树掩码

子树掩码可以和子树 OID 共同来确定一个视图的范围。子树掩码用十六进制格式表示，转化成二进制后，每个比特位对应 OID 中的一个小节，其中，

- 1 表示精确匹配，即要访问的节点 OID 与 MIB 对象子树 OID 对应小节的值必须相等；
- 0 表示通配，即要访问的节点 OID 与 MIB 对象子树 OID 对应小节的值可以不相等。

例如：子树掩码为 0xDB（二进制格式为 11011011），子树 OID 为 1.3.6.1.6.1.2.1，则对应关系如图 3 所示，所确定的视图就包括子树 OID 为 1.3.*.1.6.*.2.1（*表示可为任意数字）的子树下的所有节点。

图3 子树 OID 与子树掩码对应关系图

子树OID	1	3	6	1	6	1	2	1
子树掩码	1	1	0	1	1	0	1	1

说明

- 若子树掩码的 bit 数目大于子树 OID 的小节数，则匹配时，子树掩码的第一位与子树 OID 的第一小节对齐，第二位与第二小节对齐，以此类推，子树掩码中多出的 bit 位将被忽略
- 若子树掩码的 bit 数目小于子树 OID 的小节数，则匹配时，子树掩码的第一位与子树 OID 的第一小节对齐，第二位与第二小节对齐，以此类推，子树掩码中不足的 bit 位将自动设置为 1;
- 如果没有指定子树掩码，则使用缺省子树掩码（全 1）。

2.2 SNMP版本

SNMP 主要有 SNMPv1、SNMPv2c、SNMPv3 几种最常用的版本。

1.2.4 SNMPv1

SNMPv1 是 SNMP 协议的最初版本，提供最小限度的网络管理功能。SNMPv1 的 SMI 和 MIB 都比较简单，且存在较多安全缺陷。

SNMPv1 采用团体名认证。团体名的作用类似于密码，用来限制 NMS 对 Agent 的访问。如果 SNMP 报文携带的团体名没有得到 NMS/Agent 的认可，该报文将被丢弃。

1.2.5 SNMPv2c

SNMPv2c 也采用团体名认证。在兼容 SNMPv1 的同时又扩充了 SNMPv1 的功能：它提供了更多的操作类型（GetBulk 操作等）；支持更多的数据类型（Counter32 等）；提供了更丰富的错误代码，能够更细致地区分错误。

1.2.6 SNMPv3

SNMPv3 主要在安全性方面进行了增强，它采用了 USM 和 VACM 技术。USM 提供了认证和加密功能，VACM 确定用户是否允许访问特定的 MIB 对象以及访问方式。

1. USM（基于用户的安全模型）

USM 引入了用户名和组的概念，可以设置认证和加密功能。认证用于验证报文发送方的合法性，避免非法用户的访问；加密则是对 NMS 和 Agent 之间传输的报文进行加密，以免被窃听。通过有无认证和有无加密等功能组合，可以为 NMS 和 Agent 之间的通信提供更高的安全性。

2. VACM（基于视图的访问控制模型）

VACM 技术定义了组、安全等级、上下文、MIB 视图、访问策略五个元素，这些元素同时决定用户是否具有访问的权限，只有具有了访问权限的用户才能管理操作对象。在同一个 SNMP 实体上可以定义不同的组，组与 MIB 视图绑定，组内又可以定义多个用户。当使用某个用户名进行访问的时候，只能访问对应的 MIB 视图定义的对象。

2.3 SNMP操作

SNMP 支持多种操作，主要为以下几种基本操作：

- Get 操作：NMS 使用该操作从 Agent 获取一个或多个参数值。
- GetNext 操作：NMS 使用该操作从 Agent 获取一个或多个参数的下一个参数值。
- Set 操作：NMS 使用该操作设置 Agent 一个或多个参数值。
- Response 操作：Agent 返回一个或多个参数值。该操作是前面三种操作的响应。
- Trap 操作：Agent 主动发出的操作，通知 NMS 有某些事情发生。
- Inform 操作：Agent 主动发出的操作，通知 NMS 有某些事情发生。NMS 收到 Inform 报文后，需要给 Agent 发送确认报文。

缺省情况下，执行前四种操作时设备使用 UDP 协议采用 161 端口发送报文，执行 Trap 和 Inform 操作时设备使用 UDP 协议向 NMS 的 162 端口发报文。由于收发采用了不同的端口号，所以一台设备可以同时作为 Agent 和 NMS。

2.4 SNMP报文

根据 SNMP 的不同版本和不同操作，定义了以下报文格式：

1.2.7 SNMPv1报文

图4 SNMPv1 报文格式

SNMP message

Version	Community	SNMP PDU

Get/GetNext/Set PDU

PDU type	Request ID	0	0	Variable bindings

Response PDU

PDU type	Request ID	Error status	Error index	Variable bindings

Trap PDU

PDU type	enterprise	Agent addr	Generic trap	Specific trap	Time stamp	Variable bindings

Variable bindings

Name1	Value1	Name2	Value2	…	Namen	Valuen

从图 4 可以看出，SNMP 消息主要由 Version、Community、SNMP PDU 几部分构成。其中，报文中的主要字段定义如下：

- Version：SNMP 版本。
- Community：团体名，用于 Agent 与 NMS 之间的认证。团体名有可读和可写两种，如果是执行 Get、GetNext 操作，则采用可读团体名进行认证；如果是执行 Set 操作，则采用可写团体名进行认证。
- Request ID：用于匹配请求和响应，SNMP 给每个请求分配全局唯一的 ID。
- Error status：用于表示在处理请求时出现的状况，包括 noError、tooBig、noSuchName、badValue、readOnly、genErr。
- Error index：差错索引。当出现异常情况时，提供变量绑定列表（Variable bindings）中导致异常的变量的信息。
- Variable bindings：变量绑定列表，由变量名和变量值对组成。
- enterprise：Trap 源（生成 Trap 信息的设备）的类型。
- Agent addr：Trap 源的地址。
- Generic trap：通用 Trap 类型，包括 coldStart、warmStart、linkDown、linkUp、authenticationFailure、egpNeighborLoss、enterpriseSpecific。
- Specific trap：企业私有 Trap 信息。
- Time stamp：上次重新初始化网络实体和产生 Trap 之间所持续的时间，即 sysUpTime 对象的取值。

1.2.8 SNMPv2c报文

图5 SNMPv2c 报文格式

GetBulk PDU

PDU type	Request ID	Non repeaters	Max repetitions	Variable bindings

Trap PDU (SNMPv2c)

PDU type	Request ID	0	0	Variable bindings				
				sysUp Time.0	Value1	snmpTrap OID.0	Value2	……

比较 SNMPv1 而言，SNMPv2c 新增了 GetBulk 操作报文。GetBulk 操作所对应的基本操作类型是 GetNext 操作，通过对 Non repeaters 和 Max repetitions 参数的设定，高效率地从 Agent 获取大量管理对象数据。

SNMPv2c 修改了 Trap 报文格式。SNMPv2c Trap PUD 采用 SNMPv1 Get/GetNext/Set PDU 的格式，并将 sysUpTime 和 snmpTrapOID 作为 Variable bindings 中的变量来构造报文。

1.2.9 SNMPv3报文

SNMPv3 修改了消息的格式，但是 PDU 部分的格式同 SNMPv2c 是保持一致的。图 6 只列出消息格式。

图6 SNMPv3 消息格式

SNMPv3 message

Version	RequestID	MaxSize	Flags	Security Model	Security Parameters	Context EngineID	Context Name	PDU

其中，整个 SNMPv3 消息可以使用认证机制，并对 EngineID、ContextName、PDU 消息体部分进行加密。RequestID、MaxSize、Flags、SecurityModel、SecurityParameters 构成 SNMPv3 消息头。

报文中的主要字段定义如下：

- RequestID：请求报文的序列号。
- MaxSize：消息发送者所能够容纳的消息最大字节，同时也表明了发送者能够接收到的最大字节数。
- Flags：消息标识位，占一个字节，只有最低的三个比特位有效，比如 0x0 表示不认证不加密，0x1 表示认证不加密，0x3 表示认证加密，0x4 表示发送 report PDU 标志等。
- SecurityModel：消息的安全模型值，取值为 0～3。0 表示任何模型，1 表示采用 SNMPv1 安全模型，2 表示采用 SNMPv2c 安全模型，3 表示采用 SNMPv3 安全模型。
- ContextEngineID：唯一识别一个 SNMP 实体。对于接收消息，该字段确定消息该如何处理；对于发送消息，该字段在发送一个消息请求时由应用提供。
- ContextName：唯一识别在相关联的上下文引擎范围内部特定的上下文。

SecurityParameters 又包括以下主要字段：

- AuthoritativeEngineID：消息交换中权威 SNMP 的 snmpEngineID，用于 SNMP 实体的识别、认证和加密。该取值在 Trap、Response、Report 中是源端的 snmpEngineID，对 Get、GetNext、GetBulk、Set 中是目的端的 snmpEngineID。

- AuthoritativeEngineBoots：消息交换中权威 SNMP 的 snmpEngineBoots。表示从初次配置时开始，SNMP 引擎已经初始化或重新初始化的次数。
- AuthoritativeEngineTime：消息交换中权威 SNMP 的 snmpEngineTime，用于时间窗判断。
- UserName：用户名，消息代表其正在交换。NMS 和 Agent 配置的用户名必须保持一致。
- AuthenticationParameters：认证参数，认证运算时所需的密钥。如果没有使用认证则为空。
- PrivacyParameters：加密参数，加密运算时所用到的参数，比如 DES CBC 算法中形成初值 IV 所用到的取值。如果没有使用加密则为空。

2.5 SNMP协议原理

1.2.10 SNMPv1和SNMPv2c实现机制

SNMPv1/SNMPv2c 实现机制基本一致，SNMPv2c 丰富了错误码，新增了 GetBulk 操作。下面以在 SNMPv1 版本环境执行 Get、GetNext 和 Set 操作为例来描述 SNMPv1/SNMPv2c 的实现机制。

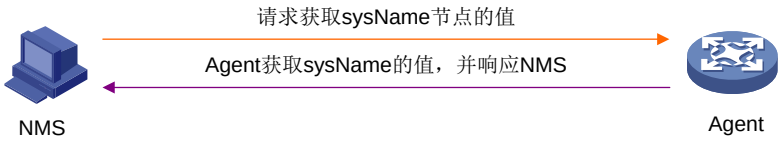
1. Get 操作

NMS 想要获取被管理设备 MIB 节点 sysName 的值（sysName 对象在允许访问视图内），使用 public 为可读团体名，过程如下：

(1) NMS 给 Agent 发送 Get 请求，请求报文主要字段将被设置为：Version 字段的值为 1，Community 字段的值为 public，PDU 里 Variable bindings 中 Name1 字段的值为 sysName.0。

(2) Agent 给 NMS 发送 Get 响应，说明是否获取成功。如果成功，则 Response PDU 里 Variable bindings 中 Value1 字段的值为设备的名字（比如 Agent010）；如果获取失败，则在 Error status 字段填上出错的原因，在 Error index 填上出错的位置信息。

图7 Get 操作示意图



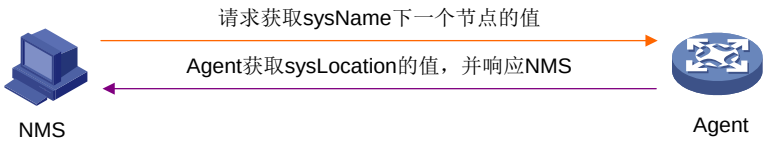
2. GetNext 操作

NMS 想要获取被管理设备 MIB 节点 sysName 的下一个节点 sysLocation 的值（sysName 和 sysLocation 对象都在允许访问视图内），使用 public 为可读团体名，过程如下：

(1) NMS 给 Agent 发送 GetNext 请求，请求报文主要字段将被设置为：Version 字段的值为 1，Community 字段的值为 public，PDU 里 Variable bindings 中 Name1 字段的值为 sysName.0。

(2) Agent 给 NMS 发送 GetNext 响应。如果成功，则 Response PDU 里 Variable bindings 中 Name1 字段值为 sysName.0 的下一个节点 sysLocation.0， Value1 字段的值为（比如 Beijing China）；如果获取失败，则在 Error status 字段填上出错的原因，在 Error index 填上出错的位置信息。

图8 GetNext 操作示意图



3. Set 操作

NMS 想要设置被管理设备 MIB 节点 sysName 的值为 Device01，使用 private 为可写团体名，过程如下：

(1) NMS 给 Agent 发送 Set 请求，请求报文主要字段将被设置为：Version 字段的值为 1，Community 字段的值为 private，PDU 里 Variable bindings 中 Name1 字段的值为 sysName.0，Value1 字段的值填为 Device01。

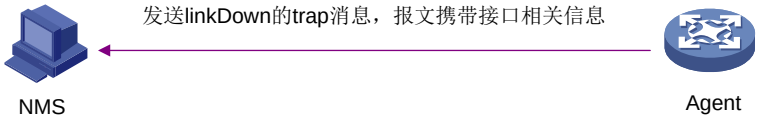
(2) Agent 给 NMS 发送 Set 响应，说明是否设置成功。如果成功，则 Response PDU 里 Variable bindings 中 Value1 字段的值填为设备的新名字（比如 Device01）；如果设置失败，则在 Error status 字段填上出错的原因，在 Error index 填上出错的位置信息。

图9 Set 操作示意图

4. Trap 操作

当设备发生某些异常需要通知 NMS 时，Agent 会主动发出 Trap 报文。例如：设备某端口网线被拔出，Agent 发送 linkDown 的 Trap 消息给 NMS。Version 字段的值为 1，Community 字段的值为 public，PDU 中 enterprise 字段为 sysObjectID.0 的取值（比如为 enterprises.25506），Generic trap 字段值为 linkDown，Variable bindings 字段携带接口相关信息。

图10 Trap 操作示意图



1.2.11 SNMPv3实现机制

SNMPv3 各操作的实现机制同 SNMPv1 和 SNMPv2c 基本一样，其主要区别在于 SNMPv3 新增加了认证和加密、解密的处理。下面以 SNMPv3 使用认证和加密方式执行 get 操作为例来描述其实现机制，过程如下：

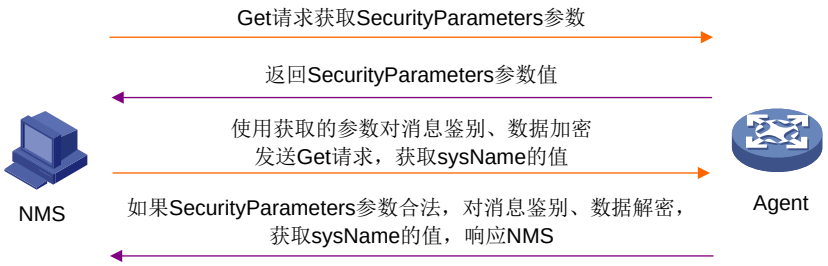
(1) NMS 首先发送不带任何认证和加密参数的 Get 请求，Flags 字段设置为 0x4，以获取 contextEngineID、contextName、AuthoritativeEngineID、AuthoritativeEngineBoots、AuthoritativeEngineTime 等相关参数的值。

(2) Agent 解析消息，发送 report 报文，并携带上述相关参数的值。

(3) NMS 再次给 Agent 发送 Get 请求，请求报文主要字段将被设置为：Version 字段的值为 3，将（2）获取到的参数值填入相应字段，PDU 里 Variable bindings 中 Name1 字段的值为 sysName.0，并且根据配置的认证算法计算出 AuthenticationParameters，使用配置的加密算法计算出 PrivacyParameters，并使用配置的加密算法对 PDU 数据进行加密。

(4) Agent 首先对消息进行认证，认证通过后对 PDU 报文进行解密。解密成功后，则获取 sysName.0 对象的值，并将 Response PDU 里 Variable bindings 中 Value1 字段的值填为设备的名字（比如 Agent010）。如果认证、解密失败或者获取参数值失败，则在 Error status 字段填上出错的原因，在 Error index 填上出错的位置信息。最后对 PDU 进行加密，设置 contextEngineID、contextName、AuthoritativeEngineID、AuthoritativeEngineBoots、AuthoritativeEngineTime、AuthenticationParameters、PrivacyParameters 等参数，发送响应报文。

图11 SNMPv3 Get 操作示意图



2.6 SNMP静默功能

设备使用 SNMP 静默功能可以自动检测并防御 SNMP 攻击。

其原理为：用户开启 SNMP 功能后，设备会自动创建 SNMP 静默定时器，并统计 1 分钟内收到的认证失败的 SNMP 报文的个数：

- 如果个数小于 100，则自动重新开始计数。
- 如果个数大于等于 100，则认为设备受到了 SNMP 攻击，SNMP 模块会进入静默状态，设备将不再响应收到的任何 SNMP 报文。静默时间为 5 分钟，5 分钟后，自动重新开始计数。

2 实现的技术特色

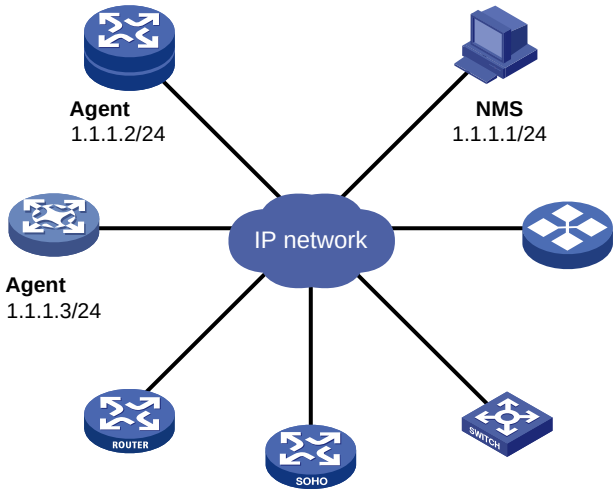
设备支持SNMPv1、SNMPv2c、SNMPv3三个版本，为了兼容SNMPv3，SNMPv1 和SNMPv2c版本也可以配置组、用户和视图，这时，只需要将 NMS 侧的团体名参数设置为设备上配置的用户名即可。用户可以同时使能多个版本，但需要和 NMS 侧的版本一致。

3 典型组网应用

3.1 组网图

如图 12 所示，网络中存在不同厂商的设备，以及同一厂商的不同型号的设备。NMS 通过 SNMP 协议对 Agent 进行监控管理，Agent 只接受 IP 地址为 1.1.1.1 的 NMS 的管理，Agent 发生故障时能主动向 NMS 发送告警信息。

图12 SNMP 典型组网图



4 参考文献

与 SNMP 相关的协议规范有：

- RFC 1155：Structure and Identification of Management Information for TCP/IP-based Internets
- RFC 2578：Structure of Management Information Version 2 (SMIv2)
- RFC 2579：Textual Conventions for SMIv2
- RFC 3411：An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks
- RFC 3412：Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)
- RFC 3414：User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
- RFC 3415：View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)