

# 第 7 篇 园区网管理维护

---

第 28 章 园区网维护管理综述

第 29 章 SNMP 及日志管理

第 30 章 LLDP 技术

第 31 章 镜像技术

第 32 章 NTP

## 第28章 园区网管理维护综述

任何网络的安全可靠运行都由多方面决定，它以精心设计细心建设为基础，在完善周到的维护管理下健康运行。同样，在有了良好的设计和建设之后，必须采用科学有效的管理手段来维护才可以确保园区网的健康运行，提供不间断服务。但随着网络规模的增大，网络设备种类的繁杂给园区网管理带来了更大的挑战。如网络中的不同角色，路由器、交换机、服务器、应用终端等的管理，不同厂商的设备混合组网的管理都给网络管理带来了麻烦。

本章将重点根据实际需求提出网络管理面临的难题并为此找出对应的解决方案。

### 28.1 本章目标

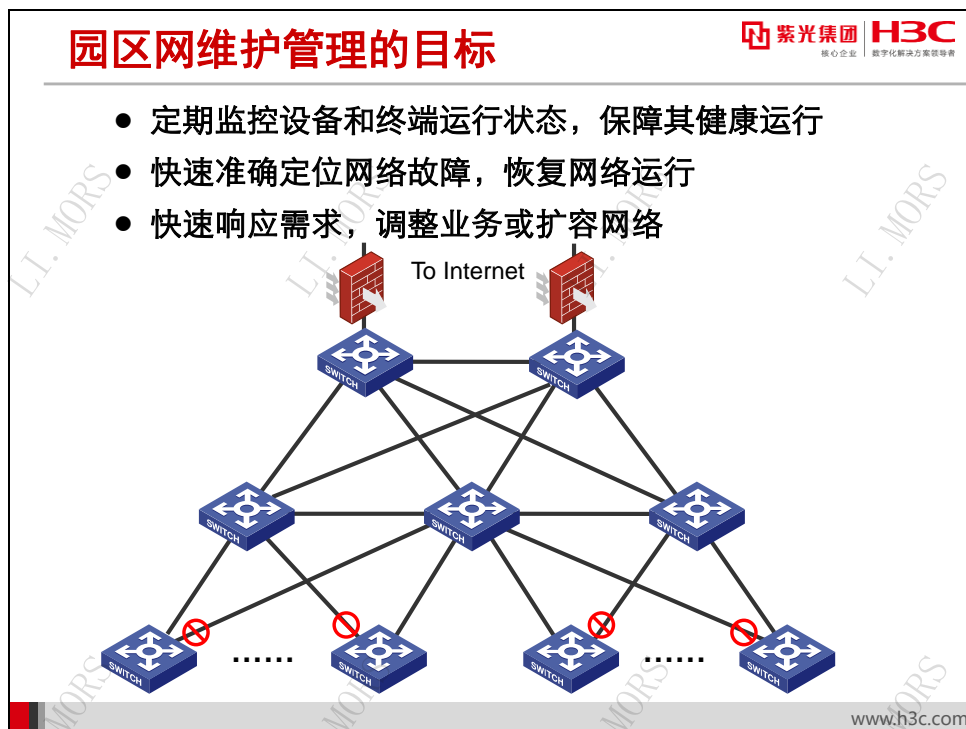
#### 课程目标

学习完本课程，您应该能够：

- 熟悉园区网存在的管理工作
- 了解园区网管理工作中的难题
- 了解园区网常见的管理手段和技术



## 28.2 园区网维护管理的目标及难题



当一个园区网建设完成并投入使用之后，网络管理员不得不面对的问题就是确保网络的健康运行。这也是网络维护管理的首要目标。为了达成此目标，必须采取相应的措施和手段实现网络设备的终端的实时监控或定期监控，使其工作状态了然于胸。但这并非网络维护管理工作的全部，任何网络设备和传输线路都不可能百分之百的不间断运行，当某个网络设备故障或者某个传输线路中断之后，网络管理员应当能够在最短时间内立即知晓故障点，故障原因并针对此故障采取有效措施快速恢复网络运行。

随着业务规模的扩大，新业务的集成应用，网络是否能够继续支撑当前业务的发展也是网络使用者和管理者关注的重要内容。因此掌握网络业务现状，快速响应新需求，调整业务部署和对网络进行升级扩容也是网络维护管理的重要目标之一。网络的维护管理工作做得是否到位，就取决于对上述重要目标的实现程度。

## 园区网维护管理的难题

紫光集团 H3C  
核心企业 数字化转型领导者

- 设备总类繁多，特性性能参差不齐
- 多厂商应用，实现机制千差万别
- 设备数量大，位置分散，管理任务繁重
- 网络安全无处不在，顾此失彼
- 网络瓶颈网络攻击不能及时发现
- 时间混乱，顺序无法准确反映

www.h3c.com

网络建设者为了平衡性能和成本各方面因素，选择的网络设备种类、设备厂商和型号越来越多样化，因此各网络设备的特性差异，性能差异也非常明显。使用同一套网络管理系统实现如此多样化设备的管理是网络管理必须解决的首要难题。要兼容不同厂商设备实现细节差异和私有协议的应用也是维护管理中的又一难题。

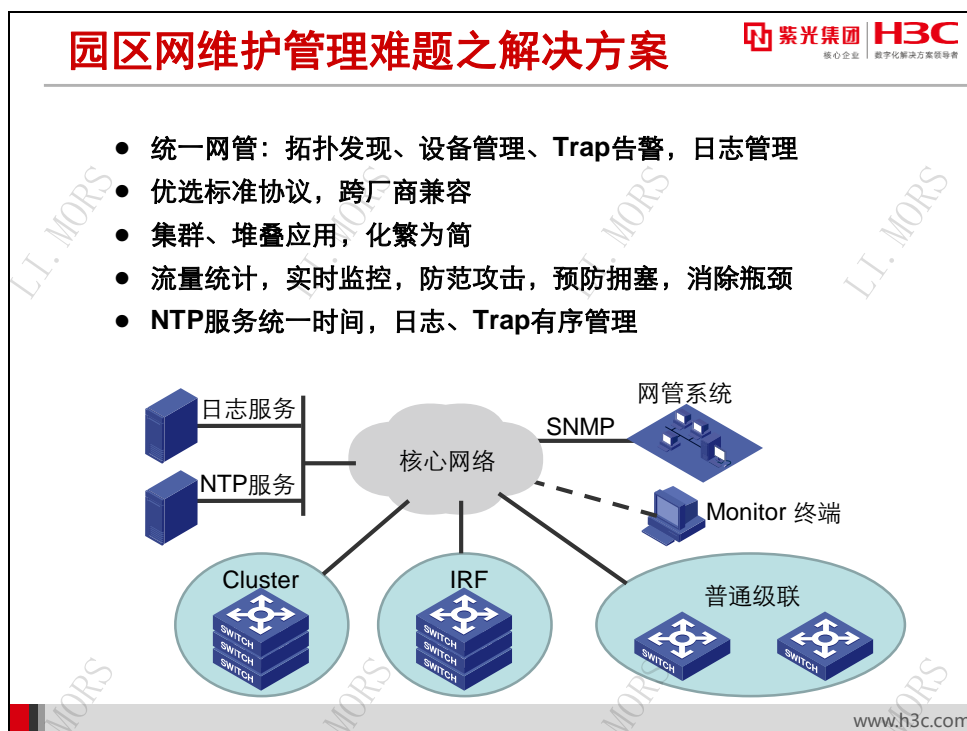
设备数量大，物理位置分散，管理任务繁重，要求网管系统性能优异，远程管理易于实施。化繁为简是大型园区网络管理新的课题。

不同的操作系统有不同的系统漏洞，从网络设备到网络终端到服务器以及传输线路处处都可能成为网络黑客的攻击切入点。网络建设和管理一旦考虑不周，很容易出现顾此失彼的现象。将网络置于危险境地。

网络设计的疏忽，新业务的开展都可能出现网络流量在某些物理链路或逻辑链路上形成拥塞。网络流量攻击更是网络拥塞的罪魁祸首。网络流量拥塞的及时发现并解决也是网络管理面临的难题之一。

网络设备记录日志和告警的时间混乱，导致日志告警产生的先后顺序无法准确判定给网络故障的定位和原因分析带来了困难，也许因为一条重要日志或告警信息的时间错误会将管理员的故障定位引入歧途，保证日志和告警信息的有序准确记录也是维护管理面临的难题之一。

## 28.3 网络维护管理的技术应用



针对网络维护管理工作面临的难题，网络管理员都分别找到了相应的应对策略和解决方案。

首先网络管理员依靠 **SNMP** 协议可以实现网络设备的统一管理。通过标准的 **MIB** 信息实现网络拓扑的发现和绘制，设备的基本信息查询和配置，设备 **Trap** 告警的统一管理。通过日志服务器的部署还可以实现网络设备重要日志的统一监管。

为了应对不同厂商的差异，应在网络规划和建设中选择业界标准协议，选择具有良好兼容性的网络设备。如选择通用的 **OSPF** 路由协议而非私有路由协议，标准的 **LLDP** (Link Layer Discovery Protocol) 协议而非私有的拓扑发现协议。

为了降低网管系统的管理任务，简化网络拓扑结构，选择恰当的网络拓扑是解决手段之一，同样部署堆叠和集群将多个物理设备联合成一个逻辑单元也能大幅降低网络管理单元数量，从而更进一步简化网络拓扑，让网络管理变得简单。

为了消除网络安全隐患，及时发现网络瓶颈和拥塞，流量统计、流量镜像、攻击防范等措施的部署是必要的。通过流量统计可以总结发现网络流量分布状况，预测网络流量的发展。通过各种流量镜像技术可以及时发现网络隐患，审计终端用户的网络行为。通过攻击防范部署可以有效的预防非法流量导致设备的宕机和网络瘫痪。

在网络中部署 **NTP** 服务器并运行 **NTP** 协议，可以保证所有网络设备具有统一的时钟参考，日志告警信息记录的时间准确无误，为网络故障定位提供准确可靠的依据。

## 28.4 本章总结

### 本章总结

- 明确维护管理工作的内容
- 熟悉维护管理工作面临的难题
- 解决维护管理难题的应对方案

www.h3c.com

## 28.5 习题和解答

### 28.5.1 习题

1. 园区网网络管理的主要目标是（ ）
  - A. 网络设备和终端的定期或实时监控，保障网络健康运行
  - B. 网络故障的快速定位和恢复
  - C. 快速响应业务需求，对网络进行正确的升级扩容
  - D. 部署最新的网络管理系统
2. 园区网网络管理维护面临的难题有（ ）
  - A. 兼容管理多厂商的网络设备
  - B. 及时发现网络安全隐患
  - C. 快速恢复网络故障
  - D. 准确监控和预测网络流量的变化
3. 园区网管理的措施中化繁为简的措施有（ ）
  - A. 统一网管
  - B. 集群部署
  - C. 堆叠部署
  - D. 选择标准协议

### 28.5.2 习题答案

1. ABC
2. ABCD
3. ABCD

## 第29章 SNMP 及日志管理

随着网络规模的不断扩大，网络拓扑环境 and 应用环境日趋复杂，而对网络进行精确管理的要求越来越高，并且自动化网络管理日益成为网络管理的一个趋势。

SNMP (Simple Network Management Protocol, 简单网络管理协议) 提供了一种从网络设备中收集网络管理信息的方法，也为设备向网络管理工作站报告问题和错误提供了一种方法。SNMP 可以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 已经成为 IP 领域网络管理的标准协议，并被广泛应用。本章主要介绍 SNMP 协议以及在设备上的配置。

### 29.1 本章目标

#### 课程目标

学习完本课程，您应该能够：

- 了解SNMP协议工作的C/S架构
- 熟悉SNMP的协议基础MIB的实现和分类
- 了解SNMP协议的历史以及历史版本的区别
- 掌握SNMP协议在园区网设备上的配置





## 29.2 SNMP的基本架构

### 29.2.1 网络管理关键功能

### 网络管理关键功能

紫光集团 H3C  
核心企业 数字化解决方案领导者

- ISO定义的网络管理关键功能：
  - 故障管理
  - 计费管理
  - 配置管理
  - 性能管理
  - 安全管理

www.h3c.com

ISO（International Organization for Standardization，国际化标准组织）定义的网络管理的关键功能有：

- **故障管理**：对网络中的问题或故障进行定位的过程。通过提供快速检查问题并启动恢复过程的工具，使网络的可靠性增强。
- **计费管理**：测量用户对网络的资源的使用情况，并据此建立度量标准，设定额度，确定费用以及给用户开具账单。
- **配置管理**：从网络获取数据，并使用这些数据对网络设备的配置进行管理的过程。目标是监视网络运行的环境和状态，改变和协调网络设备的配置，确保网络有效和可靠地运行。
- **性能管理**：保证网络保持在可通过和不拥塞的状态，为用户提供更好的服务。目标是通过监控网络的运行状态、调整网络参数来改善网络的性能，确保网络的安全运行。
- **安全管理**：通过控制信息的访问点保护网络中的敏感信息。

### 29.2.2 网络管理面临的挑战与 SNMP

## 网络管理面临的挑战与SNMP

紫光集团 H3C  
核心企业 数字化转型领导者

- **网络管理面临的挑战**
  - 网络规模越来越大
  - 网络设备越来越多样性
  - 自动化管理成为网络管理的基本需求
- **SNMP提供了一种对多供应商、可协同操作的网络管理工具，成为应用广泛的IP网络管理协议。**
  - SNMP协议实现简单，适合IP网络管理要求
  - SNMP能够花费最少的人力、设备、资金提供更智能的网络管理服务

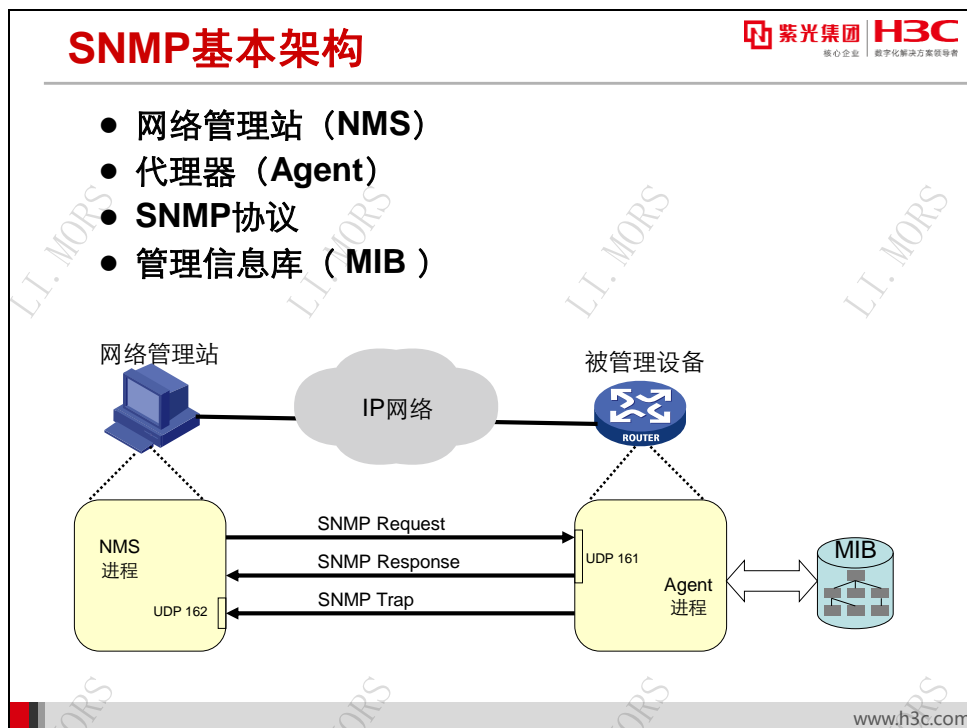
www.h3c.com

网络应用的发展对网络管理提出了更高的要求和挑战：

- 网络规模越来越大，网络设备之间的连接与业务越来越复杂，单纯靠人工管理这些设备和业务已经不可行。
- 在同一个网络中，设备的种类越来越多，必须要求使用一种标准的管理手段才能完成这些设备的统一管理。
- 网络管理要求实现不间断的管理，因此在无人值守的情况下网络管理必须提供一种自动化的工具来完成网络管理功能。

SNMP 作为网络管理协议，刚好解决了目前网络管理中遇到的几大挑战，并在花费最少的人力、设备、资金的前提下提供更智能的网络管理服务。

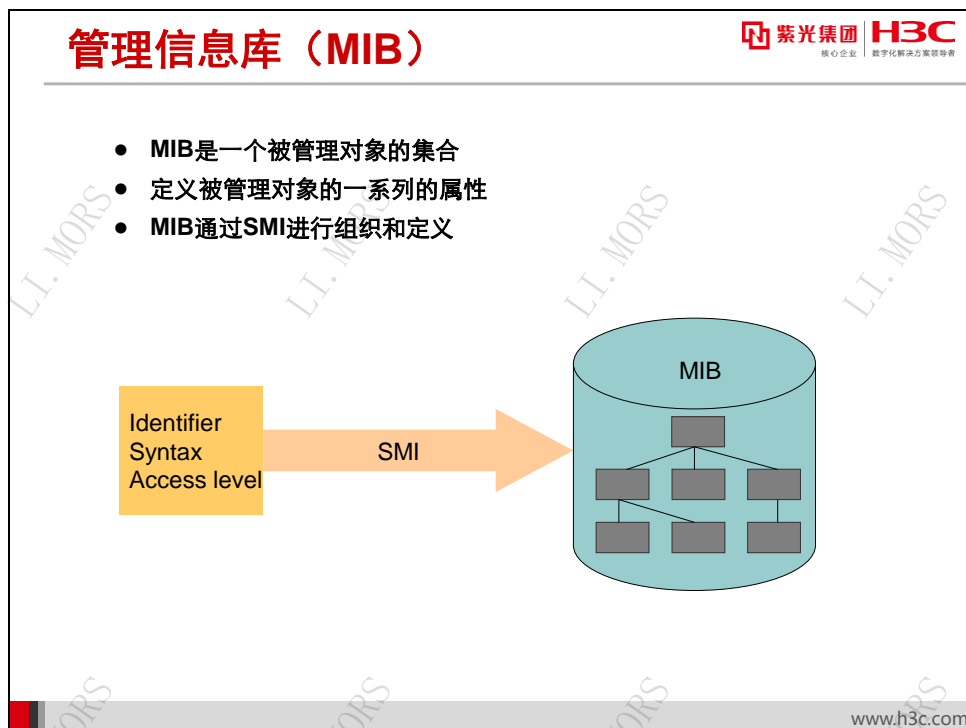
## 29.2.3 SNMP 基本架构



一个基于 SNMP 的网络管理模型，有四个重要的组成部分：

- **NMS (Network Management Station, 网络管理站)**: NMS 通常是一个独立的设备，上面运行着网络管理的应用程序。网络管理应用程序能够提供一个非常友好的人机交互界面，网络管理员能通过它来完成绝大多数的网络管理工作。NMS 通过 SNMP 协议从 SNMP Agent 获取管理信息，并且监听 UDP162 端口，接收 SNMP Agent 发送的 Trap/告警信息。同时 NMS 还努力地做到提供失效管理、安全管理、计费管理、配置管理和性能管理。
- **SNMP Agent (SNMP 代理)**: SNMP Agent 是驻留在被管理设备的一个软件模块，它主要负责如下管理任务：
  - ◆ 监听 UDP161 端口，接收和处理来自 NMS 的请求报文，并将处理结果返回给 NMS；
  - ◆ 在一些紧急情况下，SNMP Agent 还会主动发送 Trap 告警报文给 NMS。
- **SNMP (Simple Network Management Protocol, 简单网络管理协议)**: NMS 与被管理设备之间的交互遵循 SNMP 协议规定。
- **MIB (Management Information Base, 管理信息库)**: MIB 是存储在被管理设备中的管理信息数据库。

## 29.2.4 管理信息库



MIB（Management Information Base，管理信息库），是指被管理对象信息的集合，也就是所有代理进程包含的、并且能够被管理进程进行查询和设置的信息的集合。

SNMP Agent 通过 MIB 把被管对象按照一定的规则组织起来并以树状结构进行存储。NMS 通过 SNMP 协议向 Agent 发出查询，设置 MIB 等操作即可实现对被管理设备的管理操作。只有在被管理设备的 MIB 库中存在的对象才能被 SNMP 管理。

管理对象信息包括 OID（Object Identifier，对象标识符）、对象数据类型以及对象访问权限等属性信息，这些信息通过 SMI（Structure of Management Information，管理信息结构）规定的语法和组织形式组成一个树形的 MIB 数据库。

## MIB结构

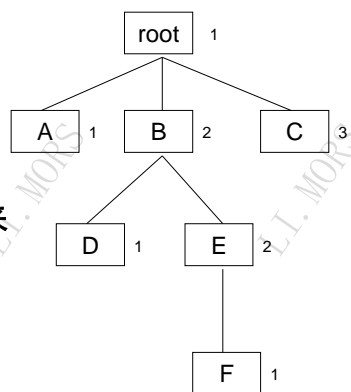
- **MIB是一种树形结构**

- 每个节点有一个名字和一个编号
- 名字不能重复
- 同一层节点的编号不能相同

- **节点可以通过名字或者OID来标识**

- 例如F节点可以表示为“1.2.2.1”

- **节点类型分为叶子节点和非叶子节点**



MIB 按照一种层次式属性结构排列。一个对象在树中的位置非常清楚地标识了如何访问该对象。在 MIB 树中，一个对象成为 MIB 树的一个节点，每个节点都有全局唯一的名字，并且具有同一个父节点的子节点的编号不能重复。

对一个节点的访问可以通过两种方式实现，一种是直接引用节点的名字，例如在图中节点 F 可以直接使用节点的名字“F”唯一标识该节点；另外一种方式是使用 OID（Object Identifier，对象标识符）来引用。

OID（Object Identifier，对象标识符）用于命名一个对象。它还标识了如何在 MIB 中访问该对象。OID 由 4 字节的整数序列组成，用来标识被管对象。MIB 树中的每个节点对应一个整数，从根节点到树中任意节点所经过的节点所对应的整数组成的有序整数序列，即为一个有效的 OID。例如 F 可以表示为“1.2.2.1”。显然可以通过提供 OID 来确定所要访问的 MIB 对象。

当对一个 MIB 节点进行访问的时候，所访问的是对象的一个特定实例，而不是一个对象类型。

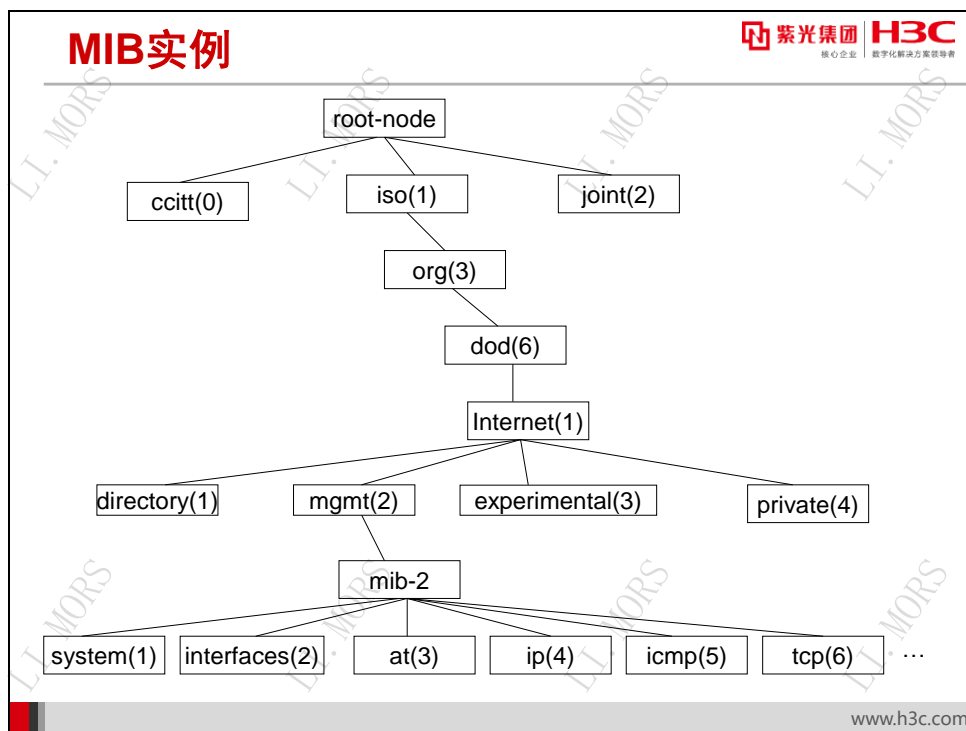
在 MIB 树中，节点根据其位置的不同，大致可以分为两类：

- **叶子节点**：不包含子节点的节点，称为叶子节点。在 MIB 树中，只有叶子节点是具有管理意义的节点。有的叶子节点不能被访问（not-accessible，不能被访问的叶子节点经常被用做表变量的索引项）；有的叶子节点只能被读出（read-only）；有的既允许读，又允许写（read-write）；有的不仅能读写，还能被创建（read-create，SNMPv2 概念；SNMPv1 也有创建功能，通过对具有 read-write 属性的表变量叶子节点，实行写操作来实现）。

- 非叶子节点：具有子节点的节点，称为非叶子节点。MIB 树中的非叶子节点表明了该节点的子孙节点的相关性，这些子孙节点一般具有大体相同的功能属性，通过访问这些节点，可以实现相对独立的功能。非叶子节点不能通过 SNMP 协议直接访问。

叶子节点又分为两类：

- 表型节点（TABULAR）：又称为列对象，这类节点在设备中代表一个抽象的对象，每个对象在设备中对应多个实例，NMS 获取的对象信息实际上是该对象的某一个实例信息的访问。例如在一个设备中存在多个接口，要想通过 MIB 获取一个接口的接口描述（ifDescr），必须指定要获取的接口的标识，接口标识实际使用的是接口索引（ifIndex），所以对于接口索引为 1 的接口，要获取该接口的描述信息，需要访问的实例为“ifDescr.1”。
- 标量节点（SCALAR）：标量节点在 MIB 中只有一个实例对象，对象和实例之间不存在模糊性。但是为了和表型节点的约定一致，并区别一个对象类型和一个对象实例，SNMP 规定一个标量对象的实例标识由它的对象标识符加上 0 组成。例如对于“sysObjectID”，一个设备只有一个值，因此“sysObjectID”为标量节点，该标量节点的实例标识为“sysObjectID.0”。



图中 MIB 树是一个标准的 MIB 树，MIB 树的根节点没有名字或编号，但是下面有三个子树：

- ccitt (0)：由 CCITT 管理
- iso (1)：由 ISO 管理
- joint-iso-ccitt：由 ISO 和 CCITT 共同管理

每一个子树下面根据需要又定义了不同的子树。

## 29.3 SNMP标准介绍

### 29.3.1 SNMP 版本

### SNMP版本



紫光集团 H3C  
核心企业 数字化解决方案领导者

- 常用的**SNMP**有三个版本
  - SNMPv1: RFC 1157定义
  - SNMPv2c: RFC 1901~RFC1908定义
  - SNMPv3: RFC3411~RFC3418定义
- 目前正式**SNMP**标准版本为**SNMPv3**

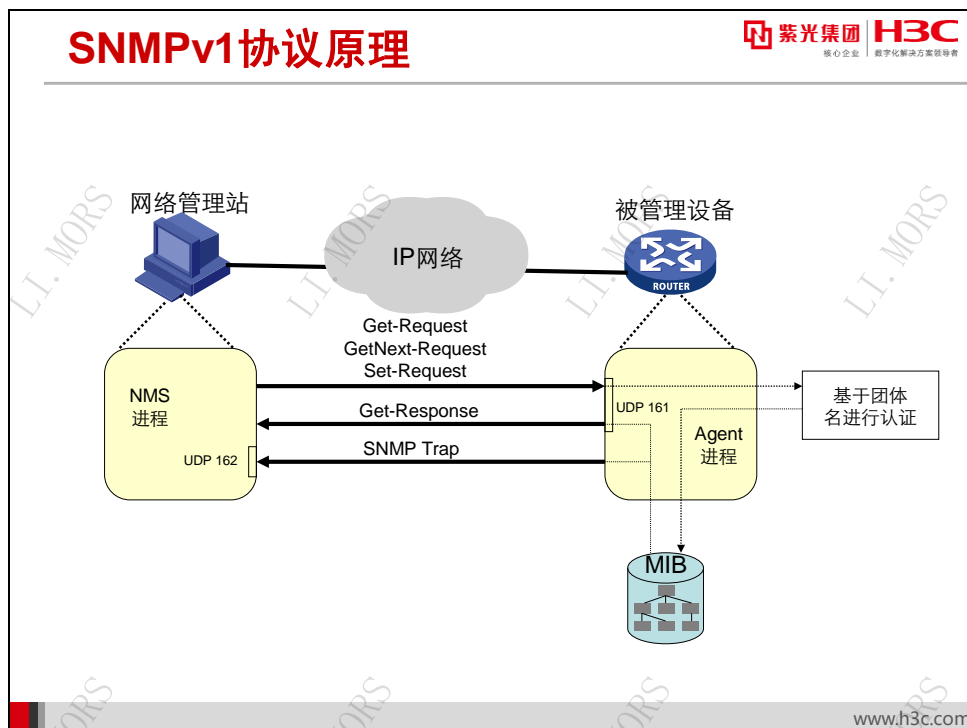
www.h3c.com

从 1988 年 SNMPv1 发布至今，SNMP 协议发展经历了多个版本的演变。目前常用的版本有：

- **SNMPv1**：该版本基于 SGMP（Simple Gateway Monitoring Protocol，简单网关监视协议）发展而来，于 1988 年发布，定义于 RFC1157。
- **SNMPv2c**：该版本使用了 SNMPv1 的消息封装以及团体名的概念，因此称为“基于团体的 SNMPv2”。该版本发布在 1996 年，在 RFC1901~1908 中定义。虽然该版本应用比较广泛，但是该版本一直没有成为一个标准协议，只能称为“事实上的标准”。
- **SNMPv3**：2002 年 SNMPv3 正式成为标准协议，以替代 SNMPv1 标准。SNMPv3 标准目前在 RFC3411~3418 中定义。



## 29.3.2 SNMPv1



SNMPv1 Agent 和 NMS 之间通过标准消息通信，使用 UDP 协议作为传输层协议，每一个消息都是一个单独的报文。UDP 使用无连接的服务，因此 SNMP 不需要依靠在 Agent 和 NMS 之间保持连接来传输信息。

SNMPv1 支持 5 种消息类型：Get-Request、GetNext-Request、Set-Request、Get-Response 和 Trap。NMS 使用 Get-Request 和 GetNext-Request 从拥有 SNMP Agent 的网络设备中检索信息，SNMP Agent 以 Get-Response 消息响应 Get-Request 和 GetNext-Request 消息。NMS 使用 Set-Request 实现设备中参数的远程配置，SNMP Agent 将配置结果以 Get-Response 消息反馈给 NMS。Trap 是 SNMP Agent 发送给 NMS 的非请求消息，这些消息通知 NMS 被管理设备发生了一个特定事件。例如可以通过 Trap 消息报告设备的某个接口从 linkup 状态变为 linkdown 状态。

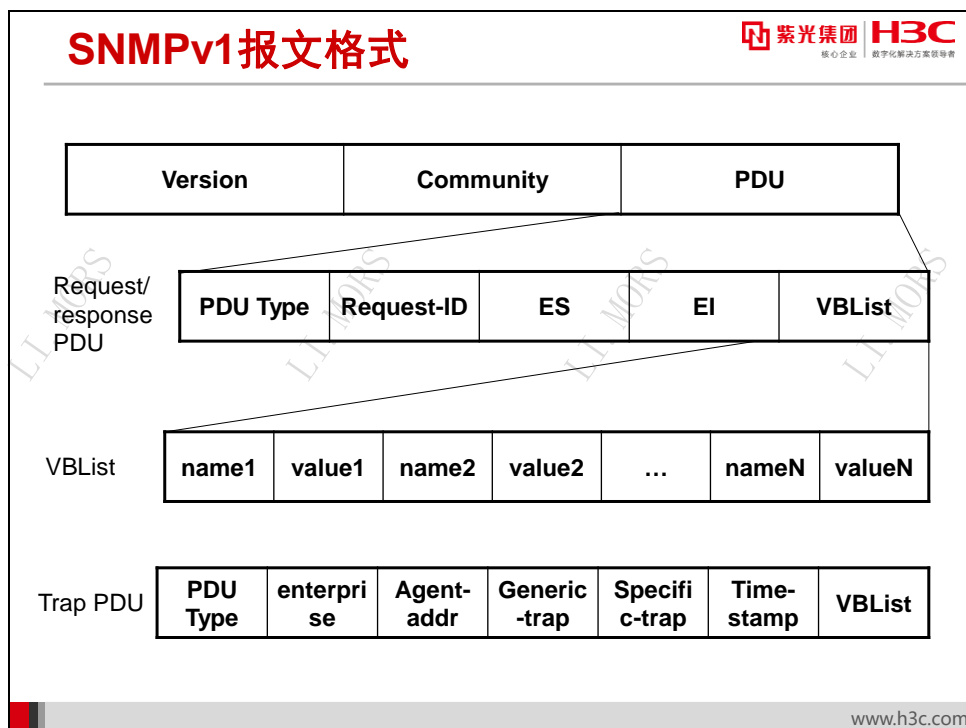
## SNMP团体



- **SNMPv1使用团体来进行安全机制管理**
- **团体是由Agent和若干个网络管理站应用程序组成**
- **每个团体通过团体名即一个字符串来区别**
- **团体名实际上是一个相关权限的密码**
  - 可以访问哪些节点
  - 访问的类型（读/设置）

www.h3c.com

SNMP Agent 和 NMS 之间是通过团体名(Community Name)来进行安全机制认证的，团体名以明文传输，因此 SNMPv1 的安全性较弱。团体是 SNMP Agent 和若干个经 SNMP Agent 授权的网络管理站应用程序组成的。每个团体都有自己的标识，即团体名，来区别于其他的团体。实质上，团体名充当了一个相关权限的密码。当 Agent 收到一个来自 NMS 的请求报文时，首先检查这个团体名是否存在，如果不存在，对此报文不再继续处理而直接丢弃；如果存在，则检查请求报文中所请求的节点是否是在该团体所被允许访问的对象集合中，并且该访问的行为是否被允许。



SNMPv1 报文由报文头加 PDU（Protocol Data Unit，协议数据单元）构成。报文头包含版本号（version）和团体名（community）两部分。

在 SNMPv1 中，Version 值为 1。团体名用作认证 SNMP 消息的口令。

在 SNMPv1 中，PDU 分为两类，一类是请求/响应 PDU，另一类是 Trap PDU。

SNMP PDU 由以下几个部分组成：

- **PDU Type:** 包括 GetRequest(0xA0)、GetNextRequest(0xA1)、GetResponse(0xA2)、SetRequest(0xA3) 和 Trap(0xA4) 共五种类型 PDU
- **Request-id:** 请求标识，用于在 NMS 和 SNMP Agent 之间对应请求和响应报文
- **ES (Error Status):** 用于指明报文出错的原因，在请求报文中该值一般为 0，只有在响应报文中该值才有意义。
- **EI (Error Index):** 用于指明变更绑定对中的第几个变量出现错误，在请求报文中该值一般为 0，只有在响应报文中该值才有意义。
- **VBLIST (Variable Binding List):** 是若干个由 OID 和该 OID 的实例值绑定对组成。

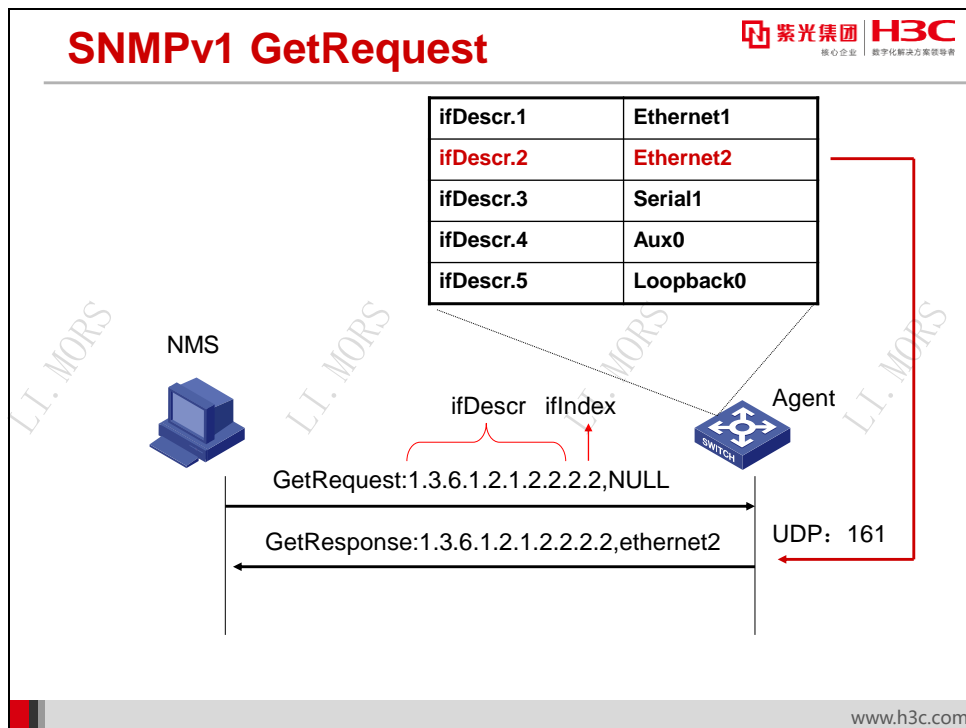
需要说明的是，在 SNMPv1 中，Trap PDU 格式比较特殊，与其它类型 PDU 不同，它除开拥有相同的 PDU type 域外，还包含如下域：

- **Enterprise:** 产生 Trap 的设备的标识，即设备的 sysObjectID。
- **Agent-addr:** 产生 Trap 的消息的设备 IP 地址。

- **Time-stamp:** 上次（重新）初始化网络实体和产生 Trap 之间的所持续的时间，取值为生成 Trap 时刻的 sysUpTime 的值。
- **generic-trap:** Trap 类型，具体取值请参考下表。
- **Specific-trap:** 当 generic-trap 取值为 6 时，表示该 Trap 被定义成企业特定的，并且使用 Specific-trap 号定义该 Trap。这样定义存在的一个问题是，作为一个通用网管，必须了解它所管理的设备 Specific-trap 号的含义，这样做对网管的适配性提出了很高的要求。这个问题在 SNMPv2 中进行了解决。

表29-1 generic-trap 标识的 Trap 类型

Trap 类型	名称	说明
0	coldStart	代理进程初始化，多数情况是设备重新启动
1	warmStart	代理进程初始化自己，但是管理对象没有更新
2	linkDown	接口从up状态变为down的状态
3	linkup	接口从down状态变为up的状态
4	authenticationFailure	一条SNMP消息收到，但是鉴别失败（例如团体名错误）
5	egpNeighborLoss	EGP邻居已经过渡到down的状态
6	enterpriseSpecific	企业自定义Trap，需要配合Specific-trap确定一个Trap。

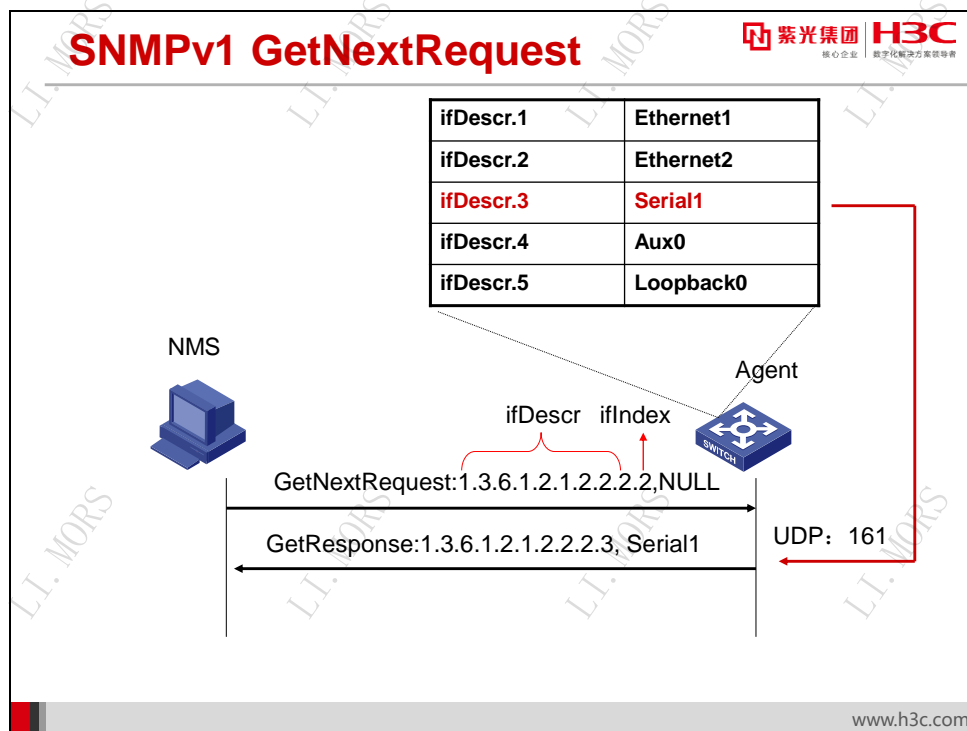


GetRequest 请求由 NMS 发送到 SNMP Agent，通过该请求 NMS 可以获取指定对象实例对应的值。SNMP Agent 通过 GetResponse 报文响应 GetRequest 请求。

在上图中，NMS 要获取第二个接口（ifIndex=2）对应的接口描述（ifDescr），通过一个 GetRequest 发送到 SNMP Agent，SNMP Agent 从本地 MIB 库中获取该对象的实例为“ethernet2”，并使用 GetResponse 返回给 NMS。

在 SNMPv1 中，GetRequest 请求是原子性的，即要么得到所有的值，要么一个也得不到。如果 SNMP Agent 能为请求中 VBLIST 列表中所有的变量提供值，那么在 GetResponse 中包括 VBLIST 域，并且对每个变量提供一个值。如果其中任意一个变量不能提供，那么将不返回任何值，可能发生的错误情况如下：

- 如果在 VBLIST 中指定的一个对象不存在，或者对象没有实例，那么需要返回一个 NoSuchName 错误状态，并在 GetResponse 报文的错误索引处填写出错变量在 VBLIST 中的位置索引。
- Agent 能够提供所有变量的值，但是最后得到的 GetResponse PDU 的大小超过了本地限制，这种情况下返回一个错误状态为 tooBig 报文，并且错误索引为 0。
- Agent 由于其他一些原因不能为 VBLIST 中的一个变量提供值，这种情况下返回一个 genErr 错误，错误索引为该变量在 VBLIST 中位置索引。

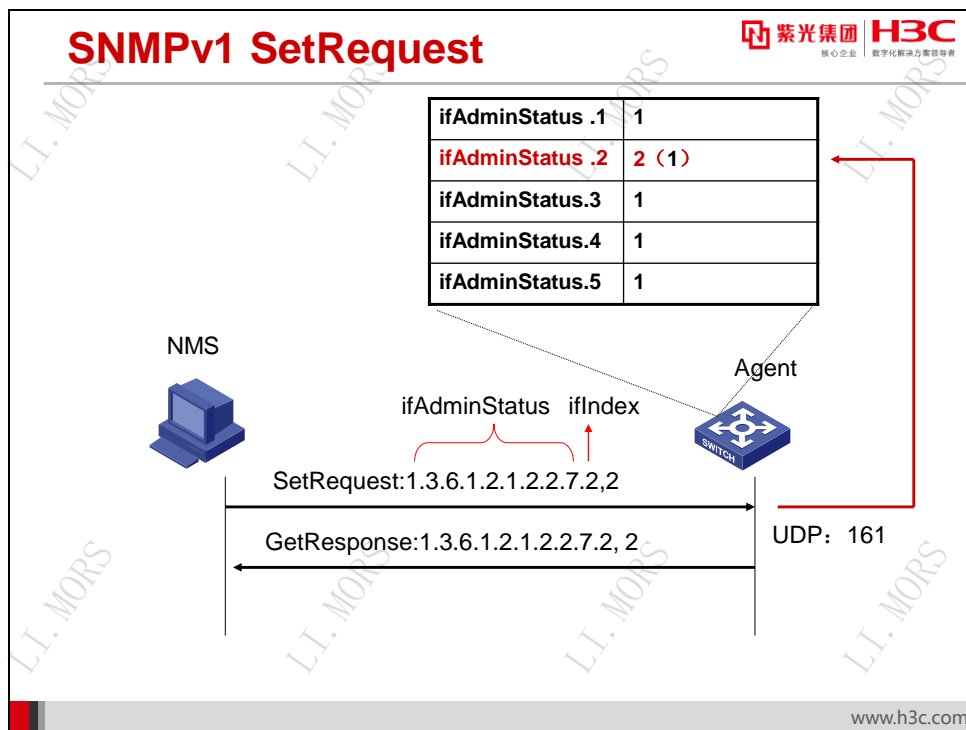


GetNextRequest 请求由 NMS 发送到 Agent，通过该请求 NMS 可以获取指定变量的字典序下一个实例对应的值。Agent 通过 GetResponse 报文响应 GetNextRequest 请求。

在上图中，NMS 要获取第二个接口（ifIndex=2）的下一个接口对应的接口描述（ifDescr），通过一个 GetNextRequest 发送到 Agent，Agent 进行本地数据库按照字典序查

找，找到下一个接口（ifIndex=3）对应的值为“ethernet3”，Agent 将该值使用 GetResponse 返回给 NMS。

在 SNMPv1 中，GetNextRequest 请求也是原子性的，要么所有的值都返回，要么一个也不返回。

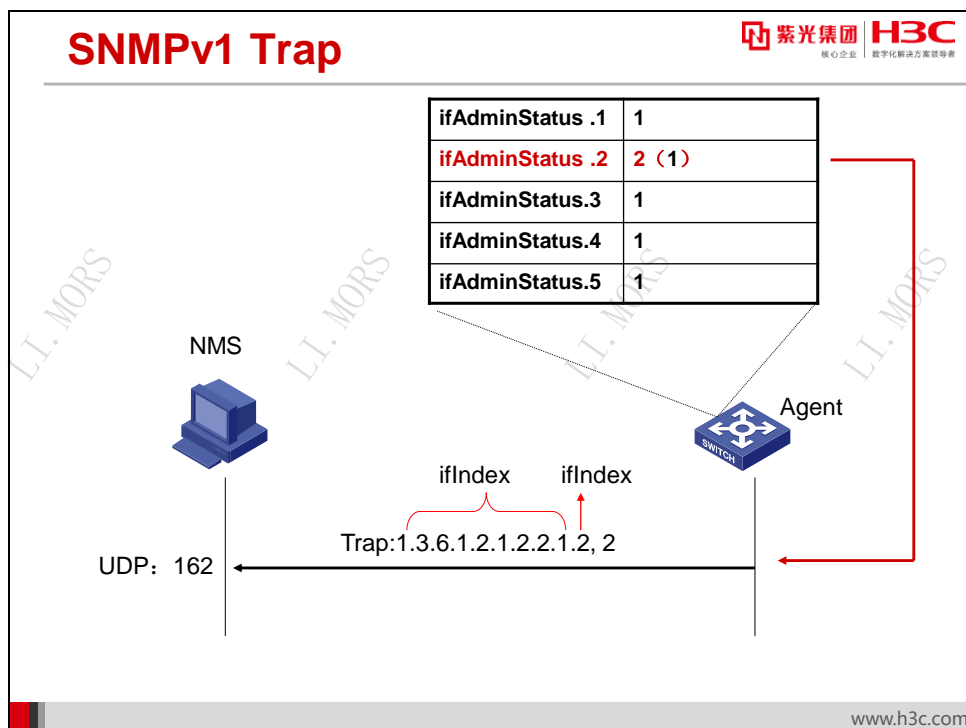


SetRequest 请求由 NMS 发送到 SNMP Agent，它用来配置一个对象的实例值。SNMP Agent 通过 GetResponse 报文响应 SetRequest 请求。

在图中，NMS 要将第二个接口（ifIndex=2）的管理状态（ifAdminStatus）设置为 2，通过一个 SetRequest 发送到 SNMP Agent，SNMP Agent 进行本地数据库查找并设置所要求的值，SNMP Agent 将 set 的操作结果使用 GetResponse 返回给 NMS。

在 SNMPv1 中，SetRequest 请求也是原子性的，要么所有的值都设置成功，要么一个值都不修改。

在进行 SetRequest 操作时，SNMP Agent 的错误处理与 GetRequest 类似。除此之外，当 set 的值与要求的值不一致的时候，返回一个“badValue”错误状态。



当 SNMP Agent 检测到特定事件发生的时候，将发送 Trap 消息到 NMS，NMS 实时监听 UDP 端口 162 来接收处理 Trap 消息。

如上图所示，接口 2（ifIndex=2）状态由 linkUp（1）变为 linkDown（2），此事件根据 SNMP 协议会触发一个 linkDown 的 Trap，该 Trap 封装成 SNMPv1 Trap 的格式被发送到 NMS。在该 Trap 的 PDU 的 VBLIST 中，包含了该接口的接口索引（ifIndex）。

由于 Trap 报文没有应答，因此它在网络中传输是不可靠的。

## SNMPv1的不足

- **SNMPv1的所有操作都是原子性的，效率低**
- **SNMPv1的错误状态有限**
- **不支持NMS到NMS之间的通信**
- **SNMPv1的Trap报文格式存在缺陷**
- **SNMPv1安全性较弱**

SNMPv1 的不足包括：

- SNMPv1 安全性差，只提供简单的身份验证和访问控制，容易被恶意攻击者破坏；
- SNMPv1 不支持 NMS 到 NMS 间的通信；
- SNMPv1 错误状态较少，导致 NMS 不能精确管理设备。
- SNMPv1 中所有操作是原子性的，只要有一个变量出错，就认为整个 PDU 处理失败，可能导致网管和代理数据不一致，同时也降低了处理效率。
- SNMPv1 agent 发送 Trap 后，不等 SNMP manager 应答，可靠性差。



## 29.3.3 SNMPv2c

## SNMPv2c vs SNMPv1



核心企业 | 数字化解决方案领导者

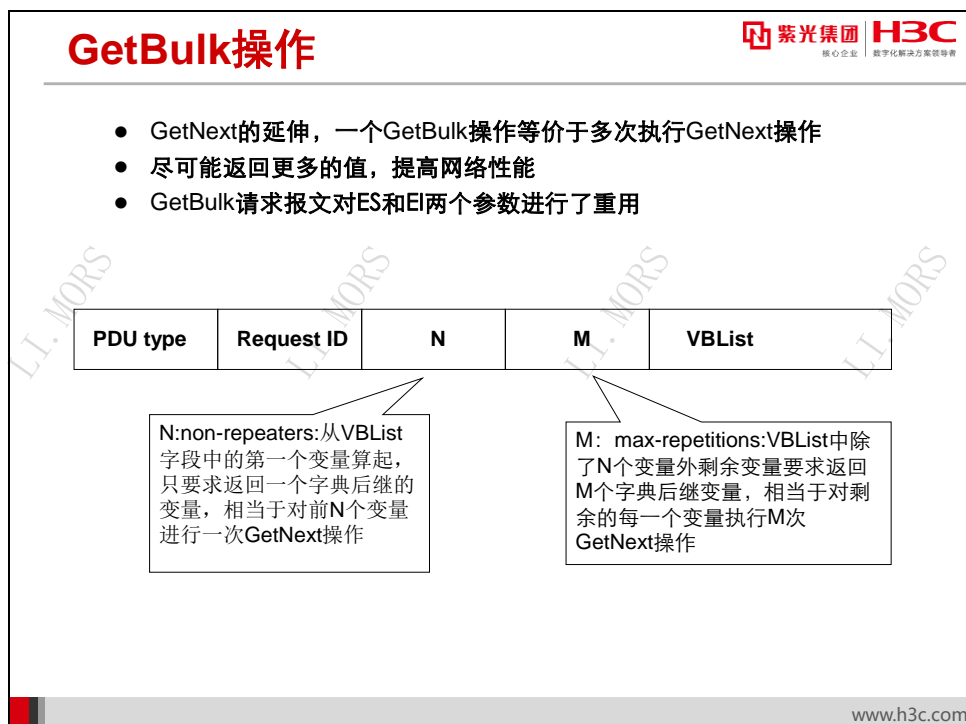
- **SNMPv2c也是基于团体名的安全机制**
- **SNMPv2c请求报文和响应报文格式与SNMPv1一致**
- **SNMPv2c相对SNMPv1改进**
  - 除了Set操作是原子性的，其他操作都是非原子性的；
  - 增加了GetBulk操作，操作效率更高；
  - 具有更丰富的错误状态和错误码；
  - 支持更丰富的数据类型
  - Trap报文格式与其他操作类型的报文格式进行了统一。

www.h3c.com

SNMPv2c 是在 SNMPv1 基础上发展起来的，消息格式与 SNMPv1 相同，Get、Get-next、Set 和 GetResponse PDU 都与 SNMPv1 相同。SNMPv2c 在安全性上没有提高，仍然采用了与 SNMPv1 相同的基于团体名的弱鉴别机制。

相对于 SNMPv1，SNMPv2c 的扩展有：

- 定义了 SNMPv2 Trap: SNMPv2 Trap PDU 格式与其它 PDU 格式（Get、GetNext、Set 等）相同，PDU 类型（PDU Type）值为 0xa7，其作用与 SNMPv1 Trap 的作用相同。在 SNMPv2 Trap 的 VBLIST 中，第一个变量提供了该 Trap 生成的时间（sysUptime），第二个变量是 snmpTrapOID.0，它表示了该 Trap 的类型，其它变量都是基于该 Trap 类型增加的。
- 定义了 GetBulk Request: GetBulk 操作基本思想是一次请求最大限度的获取更多信息。
- 定义了更丰富的错误状态和数据类型。
- 在 SNMPv2c 中，除了 Set 操作外，其它请求操作都是非原子性的。SNMPv2c 规定，即使 SNMP Agent 不能为所有变量提供取值，也返回一个 VBLIST 列表，如果发现与某个变量相关的异常情况，则将该变量名与一个异常指示组成一个变量绑定对放进 VBLIST 中，与其它正常的响应一起发送给 NMS。



GetBulkRequest 是 SNMPv2c 相对 SNMPv1 的主要增强之一。该操作的目的是使检索大批量管理信息所需的协议交互次数最小化。

GetBulkRequest 操作使用与 GetNextRequest 操作相同的选择原则，即所选择的总是按照字典序下一个对象实例，不同的是 GetBulkRequest 可以指定选择多个字典序后继。

GetBulkRequest 请求使用的 PDU 格式与其它 PDU 相同，只是启用了 ES（错误状态）和 EI（错误索引）两个新字段并命名为 N（non-repeaters）和 M（max-repetitions）。

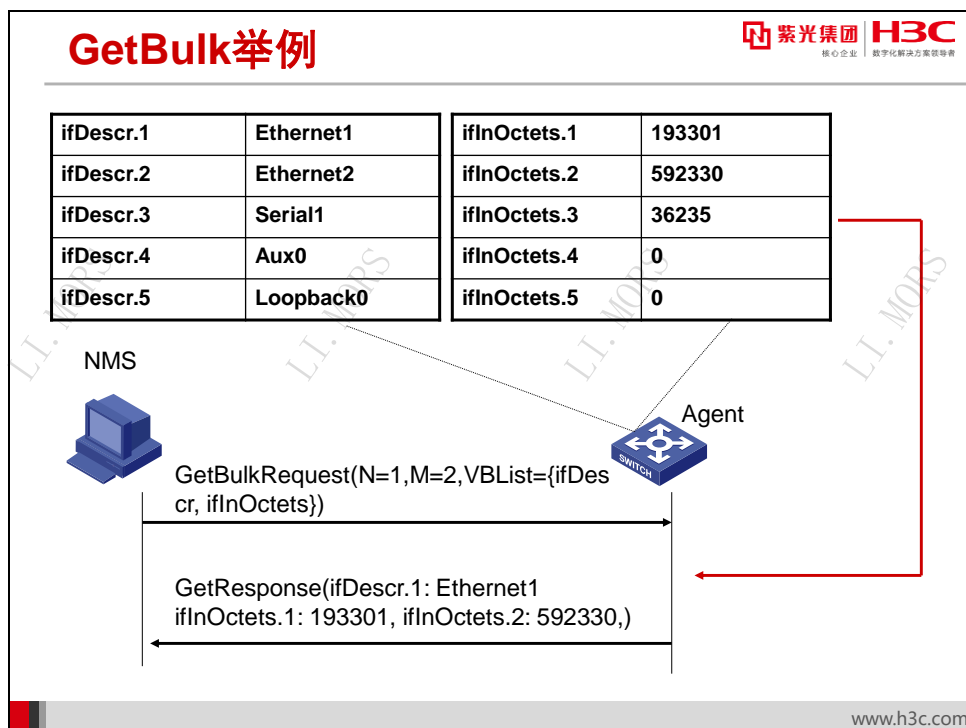
N 字段指定了在 VBLIST 中只返回单个字典序后继的变量的个数，M 指定了对 VBLIST 中其余变量将要返回字典序后继的数量。

为了解释该算法，作如下定义：

- $L = \text{GetBulkRequest PDU 中 VBLIST 中变量名的个数}$ ;
- $n = \text{从 VBLIST 中第一个变量开始，进行一次 GetNext 请求的变量个数}$ ;
- $r = \text{在开始的 N 个变量之后，请求多个字典序后继的变量个数}$ ;
- $m = \text{为最后 R 个变量中的每一个变量请求的字典序后继个数}$ 。

存在下列关系：

- $n = \text{MAX}[\text{MIN}(N, L), 0]$
- $m = \text{MAX}[M, 0]$
- $r = \text{MAX}[L - n, 0]$



如图示实例， $N=1$ ，表示对第一个变量进行一个 **GetNext** 操作， $M=2$ ，表示对剩余变量进行 2 次 **GetNext** 操作操作。

实际 **VBLIST** 中，第一个变量是 **ifDescr**，没有实例索引，因此根据 **GetNext** 原理，要取第一个接口的 **ifDescr**，也就是 **ifIndex=1** 对应的接口的接口描述。剩余的一个变量 **ifInOctets** 要进行两次 **GetNext**，获取的值分别是 **ifInOctets.1=193301** 和 **ifInOctets.2=592330**。

## SNMPv2c提供更丰富的错误码

紫光集团 H3C  
核心企业 数字化转型领导者

SNMPv1	SNMPv2c
badvalue	wrongValue
badvalue	wrongEncoding
badvalue	wrongType
badvalue	wrongLength
badvalue	inconsistentValue
noSuchName	noAccess
noSuchName	notWritable
noSuchName	noCreation
noSuchName	inconsistentName
genErr	resourceUnavailable
genErr	genErr
genErr	commitFailed
genErr	undoFailed

www.h3c.com

SNMPv2c 比 SNMPv1 中提供了更为细化的错误状态和错误码，如 SNMPv2c 中 wrongType, wrongEncoding, wrongLength, inconsistentValue 都对应于 SNMPv1 的 badValue。noAccess, notWritable, noCreation, inconsistentName 都对应于 SNMPv1 的 noSuchName。resourceUnavailable, GenErr, CommitFailed, undoFailed 都对应于 SNMPv1 的 genErr。

这种精细化的错误状态和错误码，能够更好的帮助网管理解代理的动作，从而为下一步的准确管理打下基础。例如在 SNMP 进行 set 操作的时候，set 操作失败的原因在 SNMPv1 中为 badvalue 的时候，对应的情况有很多种，为了修正这个错误，管理员需要从多个方面去检查 set 失败的原因，这样效率很低，SNMPv2c 中，对应 SNMPv1 的 badvalue 的情况进行了详细分类，这种分类可以让网管快速定位错误原因，从而提高了修正错误的效率。

## SNMPv1和SNMPv2c的安全性挑战

紫光集团 H3C  
核心企业 数字化转型领导者

- **SNMPv1和SNMPv2c安全性**

- 基于团体名(community name)的有限的安全机制
- 团体名是明文传输，入侵者很容易通过抓包工具来获取
- 报文不支持加密
- 限制了SNMP在非完全信任的网络中的使用

- **SNMPv3在继承了SNMPv2c的基础上，提供**

- 认证
- 加密
- 访问控制

www.h3c.com

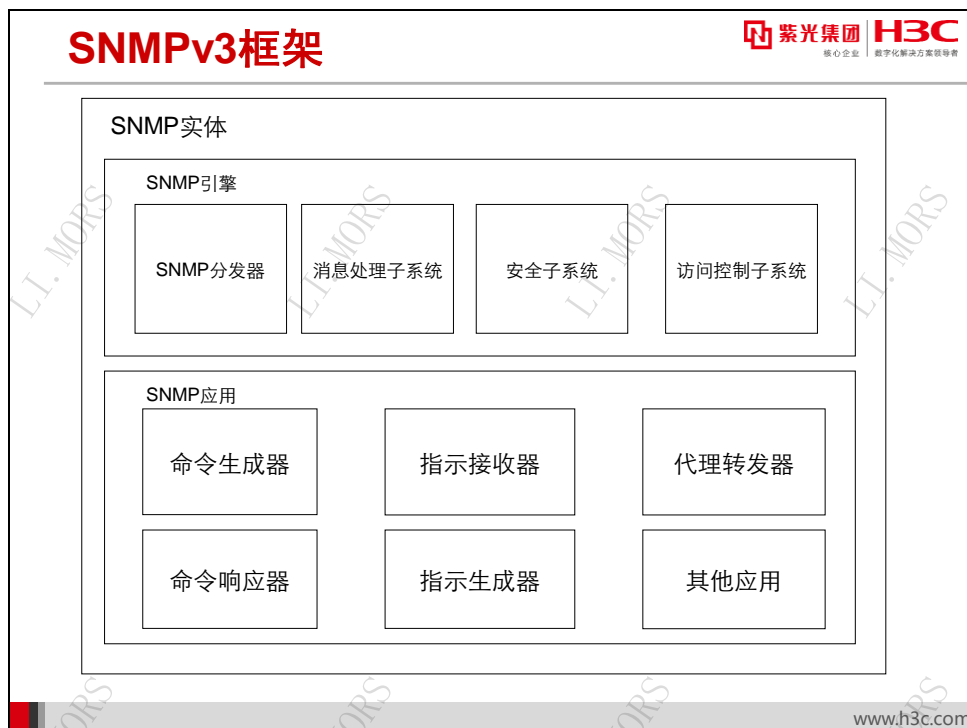
SNMPv1 和 SNMPv2c 在安全性上完全一样，都采用了基于团体名的鉴别机制，由于团体名在报文中采用明文方式传输，因此安全性非常有限。

另外 SNMPv1 和 SNMPv2c 都不支持报文的加密，因此也无法防止报文的恶意窃听。

因此 SNMPv1 和 SNMPv2c 作为一种简单的鉴别协议，可以用在完全信任的网络中，可以简化实现和提高管理效率。但是对于非信任网络，需要提供一种更安全可靠的管理协议，SNMPv3 就是在这种要求下产生的。

SNMPv3 在 SNMPv2c 基础上发展而来，协议操作上没有大的变化，在安全性方面有了实质的改进。它提供了基于指纹的认证机制、报文加密机制和基于 MIB 视图的访问控制机制。

## 29.3.4 SNMPv3

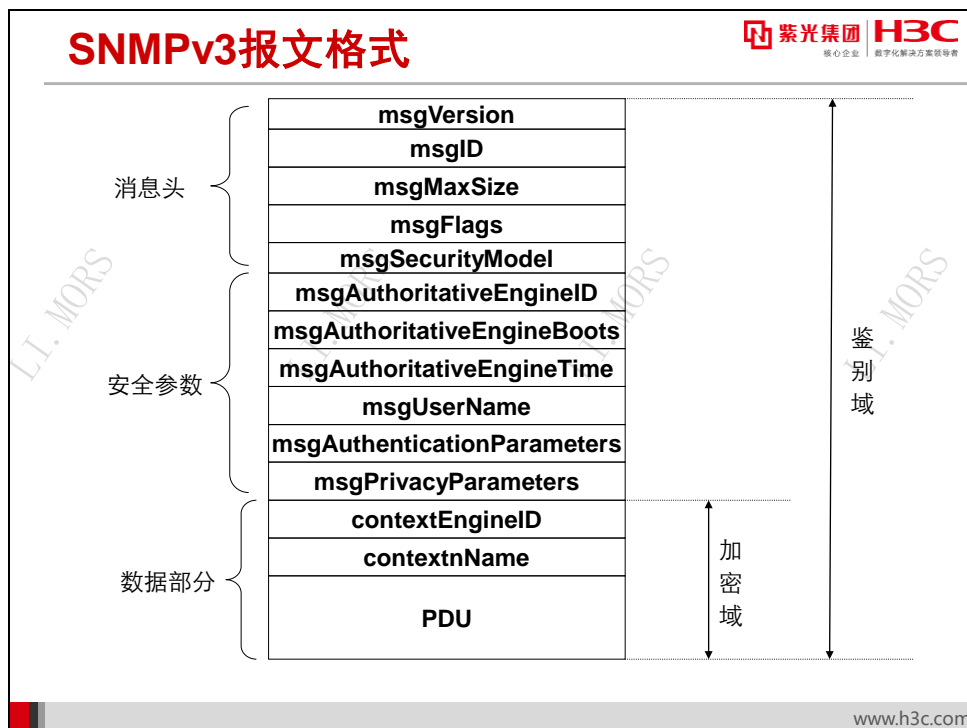


一个 SNMP 实体包括一个 SNMP 引擎和若干个 SNMP 应用。SNMP 引擎是 SNMP 实体中的核心部分，完成 SNMP 消息的收发、验证、提取 PDU、组装消息、与 SNMP 应用程序通信等功能。SNMP 应用处理 PDU，完成协议操作，存取 MIB。它包括命令生成器

（command Generator）、命令响应器（command Responder）、指示生成器（Notification Originator）、指示接收器（Notification Receiver）和代理转发器（Proxy Forwarder）等。拥有命令生成器或指示接收器的 SNMP 实体称为 SNMP manager，拥有命令响应器、指示生成器或代理转发器的 SNMP 实体称为 SNMP Agent。SNMP 实体也可以具有双重功能。

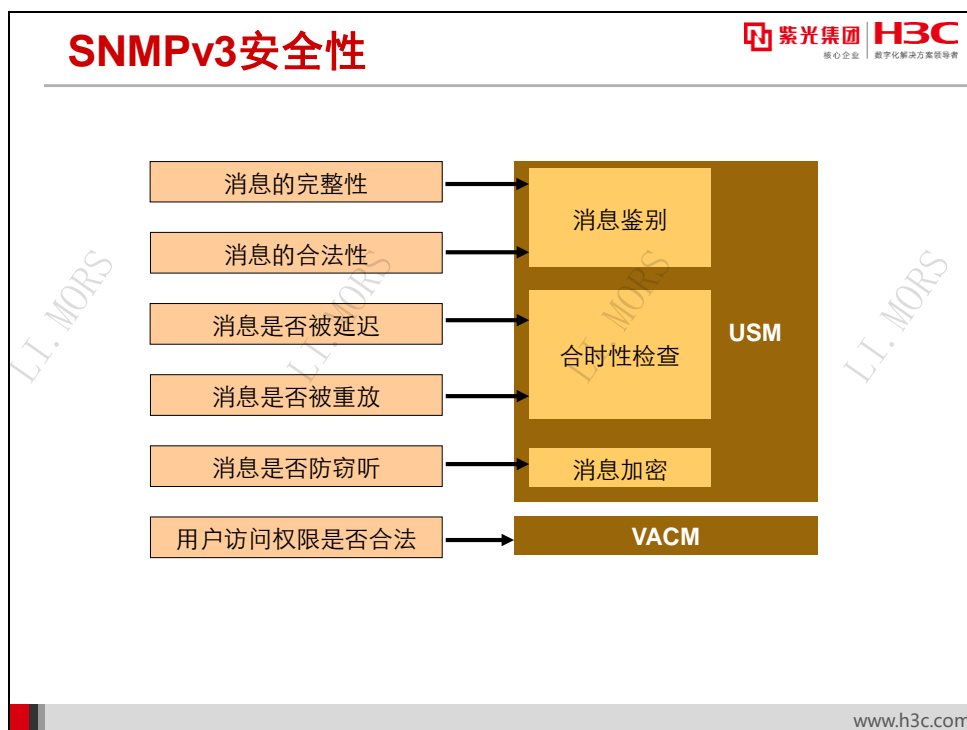
SNMP 引擎包括 SNMP 分发器（SNMP Dispatcher）、消息处理子系统（Message Processing Subsystem）、安全子系统（Security Subsystem）和访问控制子系统（Access Control Subsystem）等四个部分。子系统的一种具体描述称为模型。一个子系统中可以定义多个模型。在一个子系统的一种实现中，可以只实现一个模型，也可以同时实现多个模型。

SNMPv3 标准和 v1、v2 相比在很多方面有了提高，最主要的两点是在安全性和访问控制上作了详尽的规定。SNMPv3 定义了第三版消息处理模型（v3MP: v3 Message Processing），基于用户的安全模型（USM: User-based Security Model）和基于视图的访问控制（VACM: View-based Access Control Model）。



SNMPv3 虽然在 PDU 格式上与以往版本没有变化，但是为了适应安全性改进的需要，在报文头中增加了大量的安全参数。这些安全参数包括：msgAuthoritativeEngineID，msgAuthoritativeEngineBoots，msgAuthoritativeEngineTime，msgUsername，msgAuthenticationParameters，msgPrivacyParameters。

SNMPv3 提供了报文的鉴别和加密，鉴别针对整个报文，而加密仅针对数据部分，不包含安全参数和报文头。



SNMPv3 在安全性方面提供了两个安全模型，一个是 USM（User-based Security Model，基于用户的安全模型），另一个是 VACM（View-based Access Control Model，基于视图的访问控制模型）。

USM 模型包含消息鉴别、消息加密和合时性检查三个部分：

- **消息鉴别：**保证消息的完整性和合法性

当一个 SNMPv3 实体希望向一个命令实体 X 发送一个 SNMPv3 请求时，它将使用一个自己和 X 都知道的鉴别密钥来创建该消息的指纹，并将该指纹作为消息体的一部分发送给 X 实体。当 X 接收到这个消息后，使用相同的鉴别密钥计算出另一个指纹，如果该指纹和消息体中的指纹吻合，则消息鉴别通过。

SNMPv3 常用的鉴别协议有 HMAC-MD5-96 和 HMAC-SHA-96，前者基于 MD5 哈希算法，后者基于 SHA-1 哈希算法。

在计算指纹前，需要首先将 SNMPv3 报文头中的 msgAuthenticationParameters 字段设置为全 0，然后对整个消息使用相关协议计算消息认证指纹，最后将计算好的认证指纹填写到 msgAuthenticationParameters 字段中。

- **消息加密：**保证消息不被第三者窃听

与消息鉴别类似，消息加密也要双方共享一个密钥，基于该密钥进行消息加密和解密。

常用的加密协议有 DES、AES 等。一个 SNMPv3 消息中只有数据部分被加密，消息头和安全参数不被加密。

- **消息合时性检查：**保证消息及时到达，不被延迟与重放



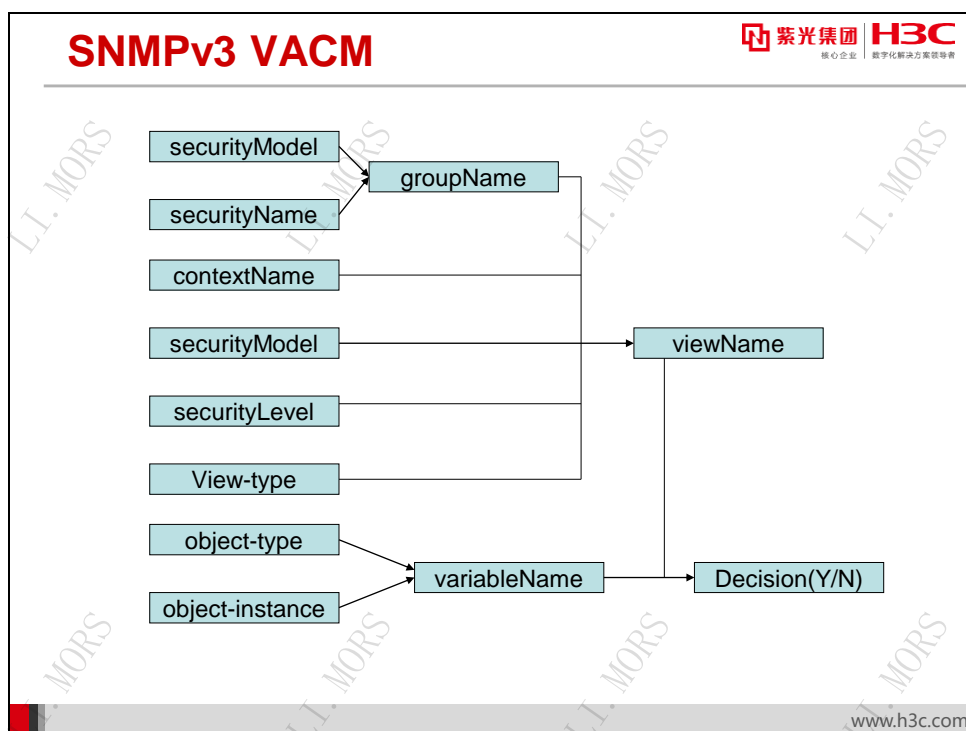
SNMPv3 的合时性检查是使用一种宽松的同步时钟的技术来保证的。

在 SNMPv3 中，进行通信的双方一个被认为是命令式的，另一个被认为是非命令式的，命令式实体维持一个“时钟”值用来保证同步，而获取并跟踪这个“时钟”值则是非命令实体的任务。一般来说，NMS 是非命令实体，而 Agent 是命令实体。需要同步的参数有两个，分别是系统启动次数和启动时间。

在 SNMPv3 报文头中由于需要携带启动次数和启动时间两个参数，因此通信的双方可以基于该参数进行合时性检查，任何一方接收到一个消息，如果判定消息在 150 秒的时间窗之外，就认为消息合时性检查失败。

合时性检查只有在使用消息鉴别的时候才会执行，因为如果不进行报文鉴别，任何恶意延迟或重放报文的攻击者都可以对报文的“时钟”信息进行修改，从而导致合时性检查和时钟同步毫无疑义。

VACM 模型可以对不同的网络管理者提供不同级别的访问权限，它基于访问用户名和 MIB 视图来实现。



VACM 用来确定一个 SNMP 协议操作是否能够访问一个 MIB 对象，它通过将用户与 MIB 视图关联起来的方法实现访问控制。

VACM 主要通过如下 MIB 完成访问控制：

- **vacmContextTable**: 定义了本地可用上下文。这个表只能读，不可通过 SNMP 配置。
- **vacmSecurityToGroupTable**: 将一个 securityModel 和 securityName 映射为一个 groupName。

- **vacmAccessTable:** 将一个 groupName、context 及安全信息映射为一个 MIB 视图。
- **vacmViewTreeFamilyTable:** 定义是否可以为一个给定的 MIB 视图访问一个对象标识符 (OID)。

对于一个具体的访问操作，采取以下步骤决定是否可以访问一个 PDU 的变量绑定中指定的 MIB 对象：

- 1) 使用 vacmSecurityToGroupTable 表将消息的 securityModel 和 securityName 映射为一个 groupName。
- 2) 使用 vacmAccessTable 将 groupName、context、securityModel 以及 securityLevel 映射到一个 MIB 视图。根据该消息的具体类型不同选取不同的 MIB 视图，例如该操作如果是 Set 操作，那么选择写视图，如果是 Get、GetNext 或者 GetBulk 操作，那么选择读视图，如果是通知消息，选择通知视图。
- 3) 根据 MIB 视图，利用 vacmViewTreeFamilyTable 表检查是否可以访问 PDU 变量绑定中的 MIB 对象。

### 29.3.5 SNMP v1/v2C/v3 对比

SNMP v1/v2c/v3对比				
	PDU支持情况	安全级别	认证	加密
SNMPv1	Get、GetNext、Set、Trap、GetResponse	noAuthNoPriv	community	NO
SNMPv2c	Get、GetNext、Set、Trap、Inform、GetResponse、GetBulk	noAuthNoPriv	community	NO
SNMPv3	Get、GetNext、Set、Trap、Inform、GetResponse、GetBulk	noAuthNoPriv AuthNoPriv AuthPriv	MD5 SHA	DES AES


在安全性上，SNMPv1 和 SNMPv2c 完全相同，SNMPv3 在安全性上有了重大改进。在实际使用中，虽然 SNMPv3 提供了高安全性，但是也要考虑其使用成本，因为消息鉴别和加密都要消耗设备的资源，对于被管理设备来说需要损耗一定性能。同时配置安全信息以及定期验证安全信息也需要一定的管理成本。

SNMPv2c 与 SNMPv3 在 PDU 上完全一致，而与 SNMPv1 相比，SNMPv1 不支持 Inform 消息，不支持 GetBulk 操作。同时 SNMPv1 的 Trap PDU 格式与 SNMPv2c 和 SNMPv3 有较大差别。

## 29.4 SNMP在交换机上的配置

### 29.4.1 SNMP 配置任务

SNMP配置任务



- 启动SNMP Agent服务
 

**[H3C]snmp-agent**
- 配置SNMP运行版本
 

**[H3C]snmp-agent sys-info { contact sys-contact / location sys-location / version {all | { v1 | v2c | v3 }\* } }**
- 创建MIB视图内容
 

**[H3C]snmp-agent mib-view { excluded | included } view-name oid-tree [ mask mask-value ]**

www.h3c.com

实现 SNMP 的配置，关键配置任务有 5 步。分别为启动 SNMP Agent 任务、打开 SNMP 协议开关、创建 MIB 视图、创建团体（SNMPv1&SNMPv2c）或创建组和创建用户（SNMPv3）：

- 启动 SNMP Agent 服务

**[H3C]snmp-agent**

snmp-agent 命令用来启动 SNMP Agent。缺省情况下，SNMP Agent 功能关闭。执行任何带 snmp-agent 关键字的配置命令都可以启动 SNMP Agent。

- 配置 SNMP 运行版本

**[H3C]snmp-agent sys-info version {all | v1| v2c |v3}**

缺省情况下开启 SNMPV3 版本。

- 配置 MIB 视图


**[H3C]snmp-agent mib-view { excluded | included } view-name oid-tree [ mask mask-value ]**

- 其中主要参数说明如下：

- ◆ **excluded:** 表示当前视图不包括该 MIB 子树的任何节点（即禁止访问 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（即允许访问 MIB 子树的所有节点）。
- ◆ **view-name:** 视图名，为 1~32 个字符的字符串。
- ◆ **oid-tree:** MIB 子树，用子树根节点的 OID（如 1.4.5.3.1）或名称（如“system”）表示。OID 是由一系列的整数组成，标明节点在 MIB 树中的位置，它能唯一标识一个 MIB 库中的对象。
- ◆ **mask mask-value:** 对象子树的掩码，十六进制数，长度为 1~32 中的偶数。

MIB 视图是 MIB 的子集，由视图名和 MIB 子树来唯一确定一个 MIB 视图。视图名相同但包含的子树不同，则认为是不同的视图。

## SNMP配置任务（续）



紫光集团 H3C  
核心企业 数字化解决方案领导者

- **创建SNMPv1 & SNMPv2c团体**

```
[H3C]snmp-agent community { read | write } [ simple | cipher ]
community-name [ mib-view view-name ] [ acl acl-number | acl
ipv6 ipv6-acl-number ] *
```
- **创建SNMPv3组**

```
[H3C]snmp-agent group v3 group-name [ authentication |
privacy ] [ read-view read-view ] [ write-view write-view ]
[ notify-view notify-view ] [ acl acl-number | acl ipv6 ipv6-acl-
number ] *
```
- **创建SNMPv3用户**

```
[H3C]snmp-agent usm-user v3 user-name group-name
[ remote { ip-address | ipv6 ipv6-address } [ vpn-instance vpn-
instance-name ] ] [ { cipher | simple } authentication-mode
{ md5 | sha } auth-password [ privacy-mode { aes128 | des56 }
priv-password ] ] [ acl acl-number | acl ipv6 ipv6-acl-number ] *
```

[www.h3c.com](http://www.h3c.com)

- 创建团体：SNMPv1 和 SNMPv2c 使用团体名进行访问鉴别，因此使用 SNMPv1 或 SNMPv2c 进行网络管理前，必须配置访问团体。下面命令用于团体的配置：

```
[H3C]snmp-agent community { read | write } { simple | cipher } community-name
[ mib-view view-name ] [ acl acl-number | acl ipv6 ipv6-acl-number ] *
```

其中主要参数说明如下：

- ◆ **read:** 表明对 MIB 对象进行只读的访问。NMS 使用该团体名访问 Agent 时只能执行读操作。

- ◆ **write**: 表明对 MIB 对象进行读写的访问。NMS 使用该团体名访问 Agent 时可以执行读、写操作。
- ◆ **simple**: 表示以明文方式配置团体名并以密文方式保存到配置文件中。
- ◆ **cipher**: 表示以密文方式配置团体名并以密文方式保存到配置文件中。
- ◆ **community-name**: 用来设置明文团体名或密文团体名, 区分大小写, 需要转义的字符请加 “\” 后输入。当以明文方式配置时, 团体名为 1~32 个字符的字符串; 当以密文方式配置时, 团体名为 33~73 个字符的字符串。
- ◆ **mib-view view-name**: 用来指定 NMS 可以访问的 MIB 对象的范围, **view-name** 表示 MIB 视图名, 为 1~32 个字符的字符串。不指定参数时, 默认的视图为 ViewDefault (启动 SNMP Agent 服务后系统创建的视图)。
- ◆ **acl acl-number**: 将团体名与基本 ACL 绑定, **acl-number** 表示访问列表号, 取值范围为 2000~2999。当未引用 ACL、或者引用的 ACL 不存在、或者引用的 ACL 为空时, 允许所有 NMS 访问设备; 当引用的 ACL 非空时, 则只有 ACL 中 permit 的 NMS 才能访问设备, 其它 NMS 不允许访问设备, 以免非法 NMS 访问设备。
- ◆ **acl ipv6 ipv6-acl-number**: 将团体名与基本 IPv6 ACL 绑定, **ipv6-acl-number** 表示访问列表号, 取值范围为 2000~2999。当未引用 IPv6 ACL、或者引用的 IPv6 ACL 不存在、或者引用的 IPv6 ACL 为空时, 允许所有 NMS 访问设备; 当引用的 IPv6 ACL 非空时, 则只有 IPv6 ACL 中 permit 的 NMS 才能访问设备, 其它 NMS 不允许访问设备, 以免非法 NMS 访问设备。

当使用 SNMPv3 进行管理时, 需要在 Agent 上配置 SNMPv3 相关参数, 这些参数中必须配置的有:

- 创建 SNMPv3 组: SNMP **snmp-agent group** 命令用来配置一个新的 SNMP 组, 并设置其访问权限, 属于该组的所有用户都具有该组的属性。执行下列命令可以创建 SNMP 组:

```
[H3C]snmp-agent group v3 group-name [ authentication | privacy ] [ read-view read-view ] [ write-view write-view ] [ notify-view notify-view ] [ acl acl-number | acl ipv6 ipv6-acl-number]
```

其中主要参数说明如下:

- ◆ **group-name**: SNMP 组名, 为 1~32 个字符的字符串, 区分大小写。
- ◆ **authentication**: 指明对报文进行认证但不加密。
- ◆ **privacy**: 指明对报文进行认证和加密。
- ◆ **read-view read-view**: 只读视图名, 为 1~32 个字符的字符串。缺省值为 ViewDefault。

- ◆ **write-view write-view:** 读写视图名，为 1~32 个字符的字符串。缺省情况下，未配置读写视图，即 NMS 不能对设备的所有 MIB 对象进行写操作。
- ◆ **notify-view notify-view:** 可以发 Trap 消息的视图名，为 1~32 个字符的字符串。缺省情况下，未配置 Trap 消息视图，即 Agent 不会向 NMS 发送 Trap 信息。
- 创建 SNMPv3 用户：SNMPv3 是基于用户的安全访问控制的，因此需要配置 SNMPv3 的用户信息。SNMPv3 用户属于一个 SNMPv3 组，该组的属性决定了该用户的一些安全和访问控制属性，与该用户是否配置安全参数无关。执行下列命令可以创建 SNMP 用户：

```
[H3C]snmp-agent usm-user v3 user-name group-name [ remote { ip-address | ipv6  
ipv6-address } [ vpn-instance vpn-instance-name ] ] [ { cipher | simple }  
authentication-mode { md5 | sha } auth-password [ privacy-mode { aes128 |  
des56 } priv-password ] ] [ acl acl-number | acl ipv6 ipv6-acl-number ] *
```

其中主要参数说明如下：

- ◆ **user-name:** 用户名，为 1~32 个字符的字符串，区分大小写。
- ◆ **group-name:** 该用户对应的组名，为 1~32 个字符的字符串，区分大小写。
- ◆ **remote { ip-address | ipv6 ipv6-address }:** 接收 Inform 信息的目的主机的 IP 地址或者 IPv6 地址，通常为 NMS 的 IP 地址或者 IPv6 地址。当设备需要向目的主机发送 SNMPv3 Inform 报文时，该参数必须配置。
- ◆ **vpn-instance vpn-instance-name:** 接收 Inform 报文的目的主机所属的 VPN。  
*vpn-instance-name* 表示 VPN 实例名称，为 1~31 个字符的字符串，区分大小写。如果未指定本参数，则表示目的主机位于公网中。
- ◆ **cipher:** 以密文方式设置认证密码和加密密码。当使用 16 进制字符作为密文密码时可以使用 `snmp-agent calculate-password` 命令来计算获得。
- ◆ **simple:** 以明文方式设置认证密码和加密密码。
- ◆ **authentication-mode:** 指明安全模式为需要认证。MD5 算法的计算速度比 SHA 算法快，而 SHA 算法的安全强度比 MD5 算法高。
- ◆ **auth-password:** 认证密码，区分大小写。采用明文设置认证密码时，认证密码的长度范围是 1~64 个字符。若采用密文设置认证密码时，如果选择 **md5** 参数，则 **auth-password** 为 32 个 16 进制字符构成的字符串；如果选择 **sha** 参数，则 **auth-password** 为 40 个 16 进制字符构成的字符串。
- ◆ **privacy-mode:** 指明安全模式为需要加密。加密算法的安全性由高到低依次是：AES、DES，安全性高的加密算法实现机制复杂，运算速度慢。对于普通的安全要求，DES 算法就可以满足需要。

- ◆ **priv-password:** 加密密码。若采用明文设置加密密码时，priv-password 表示明文密码，为 1~64 个字符的字符串。若采用密文设置加密密码时，priv-password 表示密文密码。如果选择认证模式为 md5 参数，则 priv-password 为 32 个 16 进制字符构成的字符串；如果选择认证模式为 sha 参数，则 priv-password 为 40 个 16 进制字符构成的字符串。

### 29.4.2 SNMP 配置示例

## SNMP配置示例



紫光集团 H3C  
核心企业 数字化转型方案领导者

- 要求网管使用SNMPv3管理设备，用户名为bob，该用户的访问要求认证加密，并且该用户对MIB-2中的非atTable内的节点有读或写权限

```
[H3C]snmp-agent mib-view included bobview mib-2
[H3C]snmp-agent mib-view excluded bobview atTable
[H3C]snmp-agent group v3 bobgroup privacy read-view bobview write-view bobview
[H3C]snmp-agent usm-user v3 bob bobgroup simple authentication-mode md5
bobauthkey privacy-mode des56 bobprivkey
```

www.h3c.com

在该配置实例中，首先配置了一个视图“bobview”，该视图包含了 MIB-2 中除 atTable 子树外的所有节点。

然后定义了一个 v3 组“bobgroup”，该组能够对 bobview 视图内的节点有读写权限。并且使用参数“privacy”指定了该组的安全级别为 AuthPriv。

最后定义一个 v3 用户“bob”，属于“bobgroup”组，并且配置该用户的鉴别密码和加密密码。

通过上述配置后，NMS 可以使用“bob”用户以鉴别加密的方式对设备“bobview”视图内的节点进行读写访问了。



## 29.5 本章总结

### 本章总结

- **SNMP**简单灵活，广泛应用于**IP**网络管理领域，成为**IP**网路管理的事实标准
- **SNMP**是网络管理的一系列标准，包括**SNMP**协议、**SMI**和**MIB**等
- **SNMP**目前存在三个版本，**SNMPv3**提供了较高安全性，是目前**SNMP**的现行标准。

www.h3c.com

## 29.6 习题和解答

### 29.6.1 习题

1. ISO 定义的网络管理的基本功能包括（ ）  
A. 故障管理    B. 配置管理    C. 计费管理    D. 性能管理
2. 以下关于 SNMP 描述中，正确的是（ ）  
A. SNMPv1 采用基于团体名的身份认证方式  
B. SNMPv2 增加了管理器之间的通信和数据块传送功能  
C. SNMPv2c 采用了加密和认证的安全机制  
D. SNMPv3 定义了安全机制和访问控制规则
3. 以下关于 MIB 描述中，正确的是（ ）  
A. MIB 是一种树形结构  
B. MIB 中每个节点都有全局唯一的一个名字，并且在同一个父节点下的子节点编号不能重复  
C. 在 MIB 中，只有叶子节点是可管理节点，叶子节点要么只读，要么可以读写。  
D. MIB 树中的标量节点在设备中唯一对应一个值，因此访问的时候不需要携带索引。
4. 下列哪些 PDU 类型不是 SNMPv1 的？（ ）  
A. Get-Request    B. GetNext-Request    C. GetBulk-Request    D. Inform
5. 关于 SNMPv3 消息鉴别，下列说法正确的是（ ）  
A. 消息鉴别只针对数据部分，其它部分不进行鉴别  
B. 消息鉴别是对整个消息体进行鉴别的  
C. 消息鉴别的作用是防止数据被窃听  
D. 消息鉴别的作用是防止数据被恶意篡改

### 29.6.2 习题答案

1. ABCD    2. ABD    3. AB    4. CD    5. BD

## 第30章 \*LLDP 技术

随着数据网络的发展，网络上的设备种类日益繁多。为了使不同厂商的设备能够在网络中相互发现并交互各自的系统及配置信息，需要有一个标准的信息交流平台。

在这种背景下，IEEE（Institute of Electrical and Electronics Engineer，电气与电子工程师协会）制定了 LLDP（Link Layer Discovery Protocol，链路层发现协议），提供了一种标准的链路层发现方式。设备可以将其主要能力、管理地址、设备标识、接口标识等信息组织成不同的 TLV（Type/Length/Value，类型/长度/值），并封装在 LLDPDU（Link Layer Discovery Protocol Data Unit，链路层发现协议数据单元）中发布给与自己直连的邻居，邻居收到这些信息后将其以标准 MIB（Management Information Base，管理信息库）的形式保存起来，以供网络管理系统查询及判断链路的通信状况。

### 30.1 本章目标

#### 课程目标


● 学习完本课程，您应该能够：

- 掌握LLDP基本工作原理
- 掌握LLDP常用TLV属性
- 了解LLDP-MED属性
- 掌握LLDP基本配置



## 30.2 LLDP简介

### LLDP简介



紫光集团 H3C  
核心企业 数字化解决方案领导者

- **LLDP所满足的需求**
  - LLDP是一个公共标准，使得不同厂商设备可以拓扑发现，获取对端的能力、配置等信息。
- **LLDP的基本功能**
  - 方便不同厂商以标准的方式发现网络拓扑信息。
  - 使网络管理系统有办法发现一些影响上层应用交互的配置不一致或错误。
  - 提供信息给网络管理系统，帮助定位不一致或错误问题。
  - 方便在VoIP环境里，以标准的方式部署和配置媒体终端设备，如IP电话。

www.h3c.com

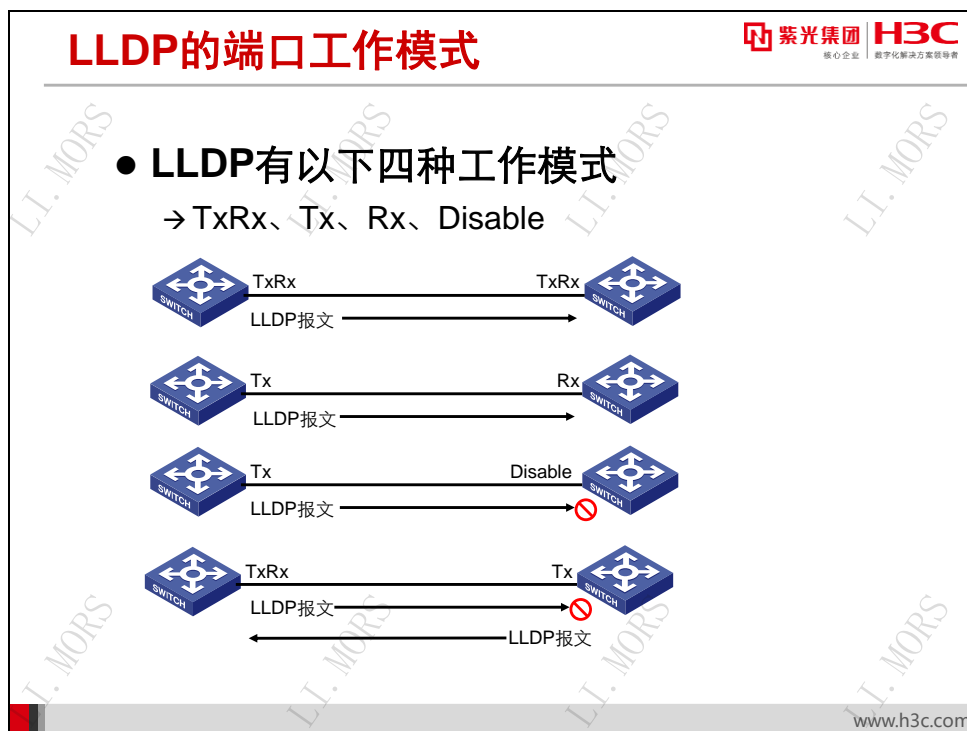
为了便于网络管理和维护，大多数设备厂商都制定了自己的链路层发现协议。但当多个厂商的设备联合组网时，这些私有的协议都将无能为力，因此制定一种能够在多个厂商设备间互操作的链路层发现协议变得迫在眉睫。LLDP 因此诞生并提供如下主要功能来解决网络维护管理的难题。

- 运行 LLDP 的网络设备以标准的方式发现并利用网络物理拓扑信息。例如，交换机 A 和交换机 B 连接，通过 LLDP 协议，在 A 设备上就可以知道对端的设备名称、MAC 地址、管理地址、端口名称、端口速率、是否支持聚合、是否支持 POE、是否具有二层网桥功能、是否具有三层路由功能等信息。
- LLDP MED (Media Endpoint Discovery, 媒体终端发现) 方案可以直接提供信息给网络管理系统，帮助定位不一致或错误问题。LLDP-MED TLV 为 VoIP (Voice over IP, 在 IP 上传送语音) 提供了许多高级的应用，包括基本配置、网络策略配置、地址信息以及目录管理等，满足了语音设备的不同生产厂商在成本有效、易部署、易管理等方面的要求，为语音设备的生产者、销售者以及使用者提供了便利。

LLDP 协议的设计目的是发布信息给远端设备，使其可以建立关于网路拓扑的管理信息库。LLDP 并不会修改对端设备的配置也不会对对端设备产生任何控制，也就是说，LLDP 为上层提供了链路发现的方法，但对发现的链路问题，并不提供解决方法。

## 30.3 LLDP基本工作原理

### 30.3.1 LLDP 的端口工作模式

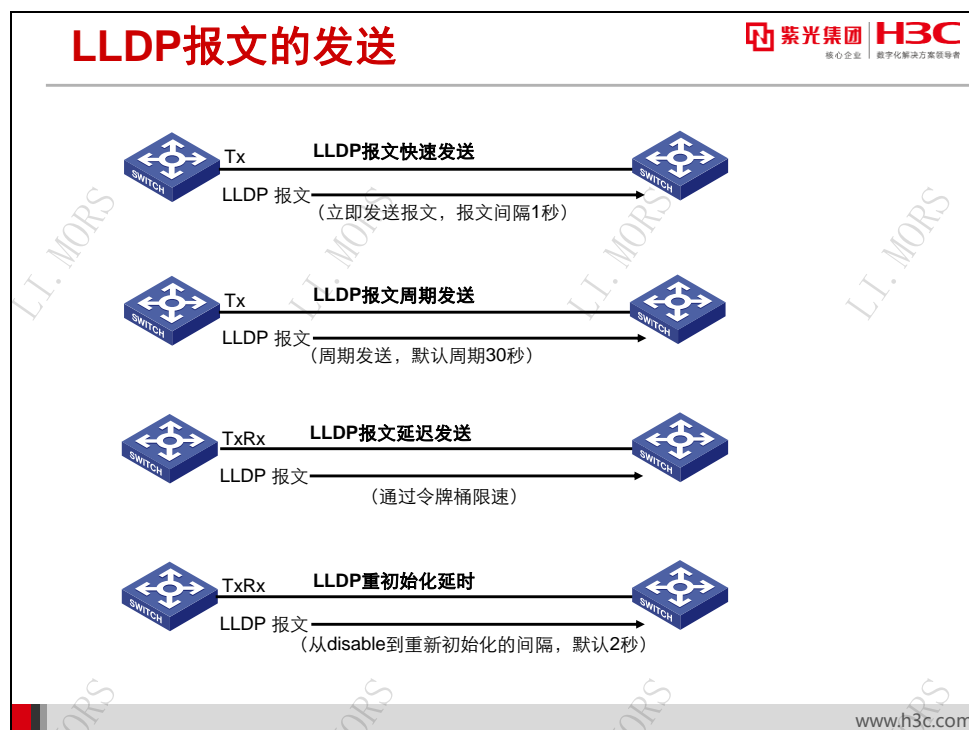


LLDP 有四种端口工作模式：

- **TxRx:** 端口既发送也接收 LLDP 报文；TxRx 模式是缺省模式，正常运行 LLDP 的设备之间都运行在 TxRx 模式。
- **Tx:** 端口只发送不接收 LLDP 报文；此模式下，发布自身信息，不保存邻居信息。
- **Rx:** 端口只接收不发送 LLDP 报文；此模式下，保存邻居信息，不发布自身信息。
- **Disable:** 端口既不发送也不接收 LLDP 报文；此模式下，不发布自身信息，也不保存邻居信息。

管理员可以为每个端口选择任何一种收发模式，以适应不同的需求。LLDP 邻居间的报文交互没有确认机制，发送方发送报文后并不需要等待对方应答。同理，接收方对接收到报文后也不用返回确认应答报文。

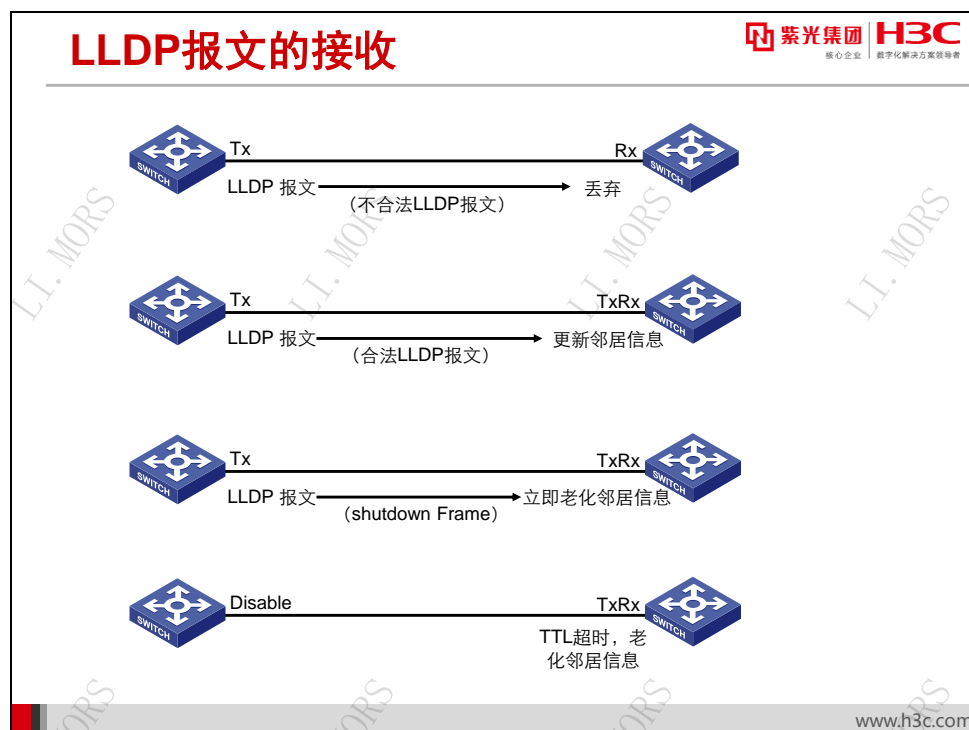
## 30.3.2 LLDP 报文的发送



LLDP 根据端口的工作状态决定报文的收发执行, 如果端口工作在 TxRx 或 Tx 模式时, 端口则按照如下规则进行 LLDP 报文的发送:

- **快速发送:** 为了支持 LLDP MED, LLDP 支持快速发送机制。在链路 UP、发送使能、发现新邻居等情况下, 端口将按照 1 秒钟的时间间隔发送一定数量的 LLDP 报文, 以保证尽快建立邻居关系。
- **周期发送:** 正常情况下, 设备将在端口上周期性的发送 LLDP 报文以维持邻居关系, 发送周期可配置, 默认为 30 秒。
- **延迟发送:** 当本地信息库里的信息变化时, 会触发 LLDP 报文发送, 这样邻居可以及时更新远端信息。但如果信息频繁变化, 会导致 LLDP 大量发送, 为避免这种情况, 使用令牌桶机制对 LLDP 报文发送作限速处理。目前默认限制发送报文速率的令牌桶大小为 5。
- **重初始化延时:** 为避免链路动荡时 LLDP 发送状态机的频繁初始化, 发送状态机变为非发送状态时, 需要等待一定的延时才执行重新初始化。此延时定时器默认为 2 秒。
- **发送 Shutdown 帧:** 当 LLDP 关闭或从发送模式切换为 Disable 或 Rx 模式时, 需要发送 Shutdown 帧。该报文只包含必要的必选 TLV。且其中的 TTL TLV 的 Value 字段值为 0。
- **发送统计:** 统计本端口发送的 LLDP 报文数量。

## 30.3.3 LLDP 报文的接收



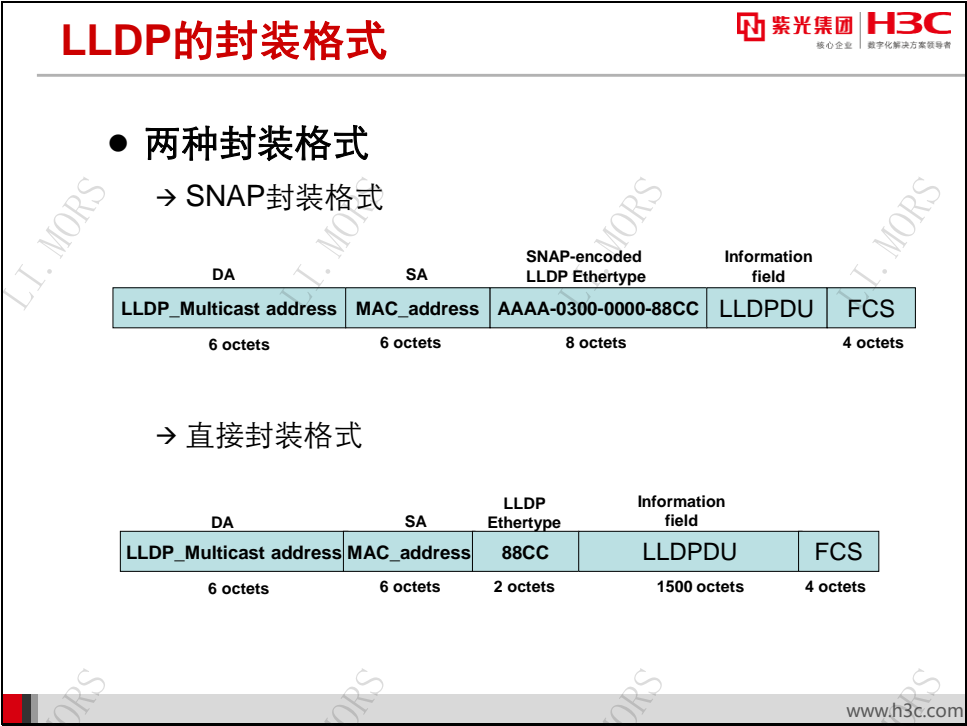
如果端口工作在 TxRx 或者 Rx 模式，端口则按照如下规则进行 LLDP 报文的接收：

- **合法性检查：**首先对 LLDP 报文格式、内容、TLV 的顺序、长度等信息进行合法性检查。如果合法性检查失败，则丢弃。如果合法性检查成功，则根据报文内容建立或更新邻居信息，如果报文的 TTL 值为 0，则立即老化该邻居信息。
- **多邻居处理：**一个端口可能收到多个邻居的信息，比如端口下挂一个 HUB。在这种情况下，防止资源被大量占用，需要限制允许接收的邻居数量，根据设备性能和实际需要可以灵活配置。
- **接收统计：**对接收的有效或无效 LLDP 报文进行统计。

#### 注意：

LLDP 报文只能在邻居设备间交互，不被邻居转发，不带 tag。如果设备不运行 LLDP 协议，则 LLDP 报文作为普通数据报文被转发。对于聚合端口，LLDP 可以在聚合组的任何一个子端口上运行，即 LLDP 运行在聚合特性之下。LLDP 协议只能运行在设备的二层以太网口、任何可以阻塞端口的特性不影响 LLDP 报文的收发。LLDP 报文每次发送，都必须提取所有允许发送的 TLV 封装到报文里，而不是增量式的发送。

30.3.4 LLDP 报文封装格式



LLDP 作为链路层协议，定义了两种协议报文封装类型：SNAP 和 Ethernet II。

- **SNAP 封装：**适用于 FDDI 和令牌环网，其中 LLC 字段为 AAAA03，SNAP 字段为 0x00000088CC。
- **直接封装：**适用于 802.3 标准的以太网，其 Type 字段为 0x88CC。

LLDP 采用保留的组播 MAC：01-80-c2-00-00-0e 为协议报文的目的 MAC，发送端口的端口 MAC 为协议报文的源 MAC。



## 30.3.5 LLDPDU 组成

## LLDPDU组成

**紫光集团**  
核心企业 | 数字化转型方案领导者

●**LLDPDU由下面的4个必选TLV和若干可选TLV组成，必选TLV的封装顺序不能改变**

- Chassis ID TLV
- Port ID TLV
- Time to Live TLV
- 可选 TLV
- End Of LLDPDU TLV

Chassis ID TLV	Port ID TLV	Time to Live TLV	Optional TLV	...	Optional TLV	End Of LLDPDU TLV
M	M	M				M

M - mandatory TLV – 必选TLV

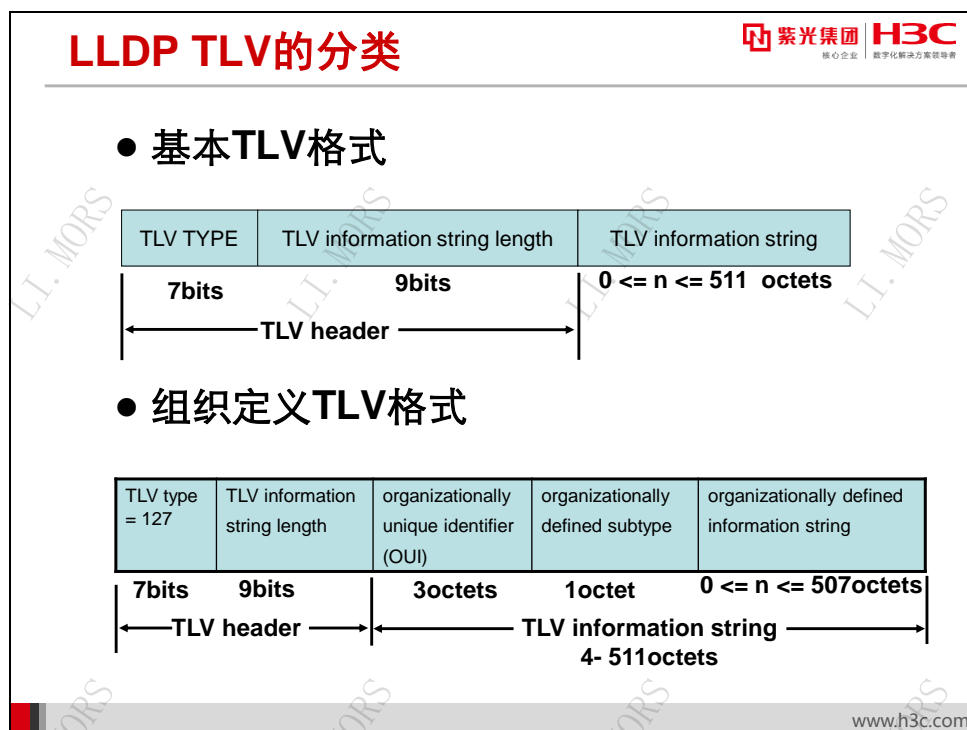
www.h3c.com

LLDPDU 依次包含下列 TLV：

- Chassis ID TLV
- Port ID TLV
- Time to Live TLV
- 可选 TLV
- End of LLDPDU TLV

Chassis ID TLV、Port ID TLV、Time To Live TLV、End of LLDP TLV 是必选 TLV，也就是说任何一个 LLDPDU 必须包含这 4 个必选 TLV。必选 TLV 在 LLDPDU 中的先后顺序也不能改变且不允许重复，否则，报文在合法性检查中将被视为非法而丢弃。End of LLDP TLV 视为 LLDPDU 的结束标记，该 TLV 以后的报文内容都不会被处理。任何一个 LLDPDU 可以包含若干个可选 TLV 或者不包含可选 TLV。

## 30.3.6 LLDP TLV 分类

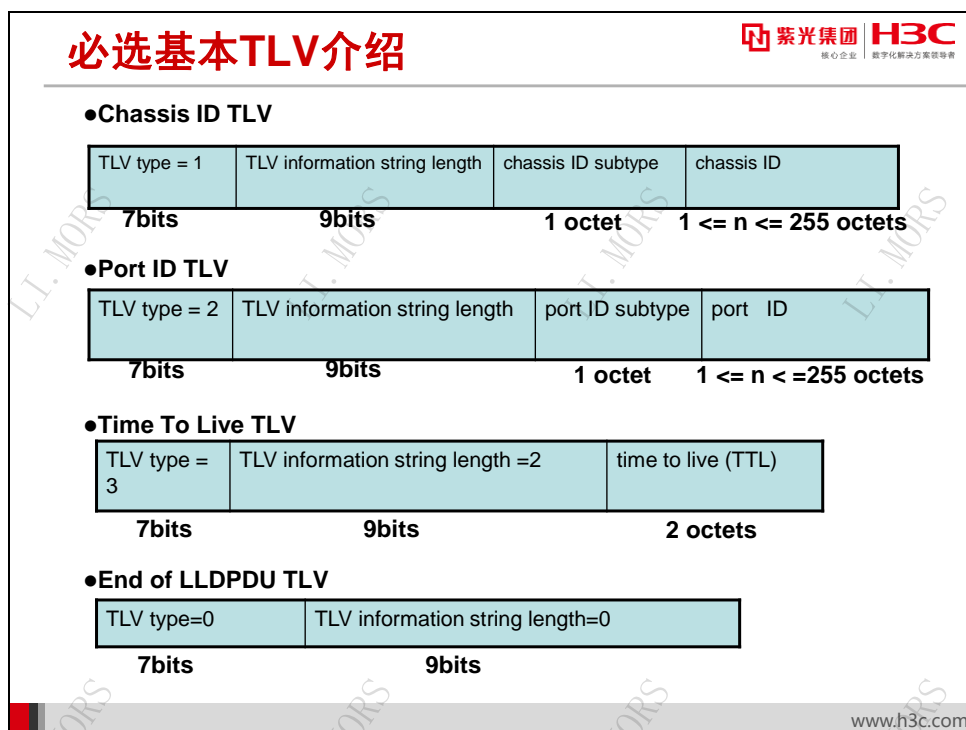


LLDP TLV 分为两类，基本 TLV 和组织定义 TLV。基本 TLV 描述基本的网络管理信息，是 LLDP 提供的基本功能。组织定义 TLV 是标准化组织（比如 802.1 和 IEEE 802.3）或其它机构，为特定媒介或协议定义的扩展 TLV。

基本 TLV 类型的取值范围是 0-8，包含 4 个必选 TLV 和 5 个可选 TLV。必选 TLV 包括 End of LLDPDU，Chassis ID TLV，Port ID TLV，Time To Live TLV；可选 TLV 包括 Port Description TLV，System Name TLV，System Description TLV，System Capabilities TLV，Management Address。

组织定义 TLV，类型值固定为 127。每个组织都由 OUI（Organizationally Unique Identifier，组织唯一标识）标识。组织定义 TLV 的子类型值指示该 TLV 的信息类型。H3C 目前实现的组织定义 TLV 有 802.1 组织 TLV、802.3 组织 TLV、MED TLV。

## 30.3.7 必选基本 TLV 介绍



LLDP 的必选基本 TLV 包括：

- Chassis ID TLV:** 必选的基本 TLV。TLV 类型为 1，信息字符串长度在 2 到 256 之间，信息域的第一个字节表示 Chassis ID 的子类型，往后是 Chassis ID 描述。Chassis ID TLV 是 LLDPDU 中的第一个 TLV。按照不同的分类原则，可以分成多个子类型，H3C 使用桥 MAC 描述 Chassis ID，对应的子类型值为 4。LLDP 邻居关系维持过程中，发送的 LLDPDU 中的 Chassis ID TLV 应该保持不变。
- Port ID TLV:** 必选的基本 TLV。TLV 类型为 2，信息字符串长度在 2 到 256 之间，信息域的第一个字节表示 Port ID 的子类型，往后是 Port ID 描述。Port ID TLV 是 LLDPDU 中的第二个 TLV。按照不同的分类原则，可以分成多个子类型，如果 LLDPDU 中携带有 LLDP-MED TLV，其内容为端口的 MAC 地址，对应的子类型值为 3；否则，其内容为端口的名称，对应的子类型值为 5。LLDP 邻居关系维持过程中，端口发送的 LLDPDU 中的 Port ID TLV 应该保持不变。
- Time To Live TLV:** 必选的基本 TLV。TLV 类型为 3，信息字符串长度为 2，信息域是 0-65535 之间一个整数。Time To Live TLV 是 LLDPDU 中的第三个 TLV。TTL 表示接收信息的设备保存此邻居信息的时间，即信息的有效时间，到时后信息将被老化。TLV 为 0（比如 shutdown 帧），表示立即老化当前邻居信息。一般的，TTL 取值为 LLDP 报文基本发送周期的倍数，默认是 4 倍。比如 LLDP 报文基本发送周期为 30 秒，那么，对应的 LLDP 报文中的 TTL 值为 120 秒。当然这个倍数（乘数）的值也是可配置的。

- **End of LLDPDU TLV:** 是必选基本 TLV。TLV 类型为 0，TLV 信息字符串长度也为 0，没有信息域，由全 0 的两个字节组成。该 TLV 是 LLDPDU 的最后一个 TLV，标志 LLDPDU 的结束。一些 802 的 MAC 系统要求帧的数据长度必须满足一个最小值，如果用户数据小于该长度，报文在封装过程中会在数据的后面增加填充数据，以满足最小长度的要求，End of LLDPDU TLV 可以防止填充数据被识别成 LLDP 报文内容。

### 30.3.8 可选基本 TLV 介绍

# 可选基本TLV介绍

紫光集团 H3C  
核心企业 数字化解决方案领导者

## ●Port Description TLV

TLV type = 4 7bits	TLV information string length 9bits	port description 0 ≤ n ≤ 255 octets
-----------------------	--	--

## ●System Name TLV

TLV type = 5 7bits	TLV information string length 9bits	system name 0 ≤ n ≤ 255 octets
-----------------------	--	-----------------------------------

## ●System Description TLV

TLV type = 6 7bits	TLV information string length 9bits	system description 0 ≤ n ≤ 255 octets
-----------------------	--	--

## ●System Capabilities TLV

TLV type = 7 7bits	TLV information string length 9bits	system capabilities 2 octets	enabled capabilities 2 octets
-----------------------	--	---------------------------------	----------------------------------

## ●Management Address TLV

TLV type = 8 7bits	TLV information string length 9bits	Management address string length 1 octet	Management address subtype 1 octet	Management address 1-31 octets	Interface numbering subtype 1 octet	Interface number 4 octet	OID string length 1 octet	Object identifier 0-128 octets
-----------------------	--	---	---------------------------------------	-----------------------------------	--	-----------------------------	------------------------------	-----------------------------------

www.h3c.com

LLDP 的可选基本 TLV 包括：

- **Port Description TLV:** TLV 类型为 4。该 TLV 允许网管通告 IEEE 802 局域网站的端口描述信息。在 LLDPDU 中，端口描述 TLV 不能多于一个。
- **System Name TLV:** TLV 类型为 5。该 TLV 允许网管通告系统名。在 LLDPDU 中，系统名 TLV 不能多于一个。
- **System Description TLV:** TLV 类型为 6。该 TLV 允许网管通告系统描述。在 LLDPDU 中，系统描述 TLV 不能多于一个。
- **System Capabilities TLV:** TLV 类型为 7。该 TLV 通告系统的主要能力，以及系统的主要能力是否开启。在 LLDPDU 中，系统描述 TLV 不能多于一个。系统的主要能力用一个 2 字节的 bit 影射表表示。比如影射表 0 序的第 bit 位 1 集线器功能，bit 为 2 表示网桥功能，bit 为 3 表示无线 AP 功能，bit 为 4 表示路由器功能等等。影射表中相应的 bit 位的值为 1，表示开启该功能。

- **Management Address TLV:** TLV 类型为 8。该 TLV 表示和本地 LLDP 代理（运行 LLDP 的协议实体）相关的管理地址，网络管理者可以通过该地址对设备进行访问和控制。该 TLV 还提供了 LLDP 运行端口的端口号和管理地址相关的 OID（对象标识，Object identifier）。OID，是用来标识与管理地址相关的硬件或协议实体的类型，一个 LLDPDU 可以携带多个互不重复的管理地址 TLV。

### 30.3.9 组织定义 TLV 介绍

## IEEE 802.1组织定义TLV介绍

核心企业 数字化解决方案领导者

●IEEE802.1组织定义格式

TLV type = 127	TLV information string length	00-80-C2 ( 802.1 OUI )	802.1 subtype	802.1 defined information string
-------------------	----------------------------------	---------------------------	---------------	-------------------------------------

●IEEE802.1组织定义TLV类型

IEEE 802.1 subtype	TLV name
0	Reserved
1	Port VLAN ID
2	Port And Protocol VLAN ID
3	VLAN Name
4	Protocol Identity
...	...

[www.h3c.com](http://www.h3c.com)

IEEE 802.1 组织定义 TLV 描述了虚拟局域网的各种相关属性，OUI 为 0080C2，目前包括 Port VLAN ID TLV、Port And Protocol VLAN ID TLV、VLAN Name TLV、Protocol Identity TLV 等。

- **Port VLAN ID TLV:** 子类型为 1，用于通告端口的 PVID（Port VLAN Identifier，端口 VLAN 标识），在 LLDPDU 中最多包含一个 Port VLAN ID TLV。
- **Port And Protocol VLAN ID TLV:** 子类型为 2，用于通告端口是否支持协议 VLAN，是否启动协议 VLAN，协议 VLAN ID，LLDPDU 可以包含多个不同的 Port And Protocol VLAN ID TLV。如果 LLDP 报文中包含不支持协议 VLAN 但开启协议 VLAN 的矛盾信息，则该报文将作为错误报文丢弃。
- **VLAN Name TLV:** 子类型为 3，用于通告站点所配置的任何 VLAN 的名称。一个 LLDPDU 可以携带多个互不重复的 VLAN Name TLV。
- **Protocol Identity TLV:** 子类型为 4，用于通告端口支持的协议类型，该 TLV 的协议标识字段包含协议二层地址头后面的 N 个字节，这些信息应该能使报文的接受者识别

出该协议的类型和版本号。目前 H3C 公司的设备的可以接受并识别协议标识 TLV，但不主动发送该 TLV。

## IEEE 802.3组织定义TLV介绍

**紫光集团** **H3C**  
核心企业 | 数字化解决方案领导者

### ●IEEE802.3组织定义TLV格式

TLV type = 127	TLV information string length	00-12-0F ( 802.3 OUI )	802.3 subtype	802.3 defined information string
-------------------	----------------------------------	---------------------------	---------------	-------------------------------------


### ●IEEE803.3组织定义TLV类型

IEEE 802.3 subtype	TLV name
0	Reserved
1	MAC/PHY Configuration/Status
2	Power Via MDI
4	Maximum Frame Size
...	...

[www.h3c.com](http://www.h3c.com)

IEEE 802.3 组织定义 TLV 描述了 802.3 局域网端口相关的各种属性，OUI 为 00120F 目前包括 MAC/PHY Configuration/Status TLV、Power Via MDI TLV、Maximum Frame Size TLV 等。

- **MAC/PHY Configuration/Status TLV:** 子类型为 1，该 TLV 发布端口支持的速率和双工，是否支持自动协商，当前自动协商状态（是否使能），当前的速率和双工状态。在一个 LLDPDU 中至多包含一个。
- **Power Via MDI TLV:** 子类型为 2，供电能力 TLV。在一个 LLDPDU 中至多包含一个该 TLV。
- **Maximum Frame Size TLV:** 子类型为 4，表示端口的 MAC 和物理层所支持的最大帧长度。


  
 紫光集团 H3C
   
核心企业 数字化转型领导者

## LLDP-MED TLV介绍

### ●LLDP-MED TLV格式与类型

TLV type = 127	TLV information string length	00-12-BB (TIA OUI)	LLDP-MED subtype	LLDP-MED defined information string
-------------------	----------------------------------	-----------------------	---------------------	--

MED TLV Subtype	TLV Name
1	LLDP-MED Capabilities
2	Network Policy
3	Location Identification
4	Extended Power-via-MDI
5	Hardware Revision
6	Firmware Revision
7	Software Revision
8	Serial Number
9	Manufacturer Name
10	Model Name
11	Asset ID
12-255	Reserved for future standardization

[www.h3c.com](http://www.h3c.com)

媒体终端发现协议，在 LLDP 协议的基础上，增加了一些 TLV。LLDP MED 区分网络链接设备（如交换机）和媒体终端设备（如 IP 电话），媒体终端设备又可以分成三类，分别是 MED 一般终端、MED 媒体终端和 MED 通讯设备终端。每种设备发布和接收 TLV 的能力不同，作用也不同。

为了让媒体终端设备能够快速响应，LLDP-MED 使用快速开始机制，即网络连接设备连接的媒体终端设备上线时（链路 UP），需要启动快速发送机制，立即发送一定数量（数目可自行配置）带有 LLDP-MED TLV 的 LLDP 报文，而不是等待发送周期到时再发送报文。

为了节省 LLDPDU 的空间，只有当设备发现连接的是具有 LLDP-MED 能力的媒体终端设备时（根据接收到的报文中是否携带 LLDP MED Capability TLV 以及该 TLV 中的 MED 设备类型字段来判断），才开始发送带有 LLDP-MED TLV 的报文。

LLDP-MED TLV 使用 TIA（Telecommunications Industry Association，电信工业协会）的 OUI：0012BB。目前使用的 TLV 子类型有 11 个，取值 1-11，12-255 预留。

主要的 MED TLV 有：

- **LLDP-MED Capabilities TLV**：网络设备用该 TLV 来标识其所支持的各种 LLDP-MED TLV，TLV 子类型为 1。所有的终端设备应该同时支持 LLDP-MED Capabilities TLV 的发送和接收，而 LLDP-MED 网络连接设备可以有选择地发送该 TLV，但应当能接收该 TLV。在 LLDPDU 中，该 TLV 最多只能有一个。
- **Network Policy TLV**：网络连接设备和终端设备表征端口的 VLAN 类型、VLAN ID 以及二三层与具体的应用类型相关的优先级，TLV 子类型为 2。对于该 TLV，网络连接

设备必须发送，终端设备只需接收。在 LLDPDU 中，可以发布多个不同的 Network Policy TLV。

- **Location Identification TLV:** 网络连接设备使用该 TLV 发布合适的位置标识信息，供终端在基于位置的应用中使用。TLV 子类型为 3，一个 LLDPDU 最多只能包含一个该 TLV。该 TLV，网络连接设备必须发送，终端设备只需接收。
- **Extended Power-via-MDI TLV:** 该 TLV 允许发布设备能源相关的信息（依照 IEEE 802.3af）。TLV 子类型为 4，一个 LLDPDU 最多只能包含一个该 TLV，网络连接设备和终端设备必须都要支持发送和接收。



## 30.4 LLDP基本配置

### 30.4.1 使能 LLDP 功能

### 使能LLDP功能

● 全局启动LLDP

[H3C]lldp global enable

● 端口下启动LLDP

[H3C-GigabitEthernet1/0/1]lldp enable

● 端口下配置LLDP的工作模式

[H3C-GigabitEthernet1/0/1]lldp admin-status { disable | rx | tx | txrx }

紫光集团 H3C

核心企业 数字化解决方案领导者

www.h3c.com

全局使能 LLDP 功能后，设备才会把 LLDP 报文作为协议报文处理，否则设备将 LLDP 报文当着业务报文转发。全局使能 LLDP 功能的命令是：

```
[H3C]lldp global enable
```

缺省情况下，全局 LLDP 使能之后，端口上的 LLDP 也处于使能状态，如果需要在特定端口上关闭和重新使能可以使用端口视图下的命令进行单独控制。

端口视图下使能 LLDP 的命令是：

```
[H3C-GigabitEthernet1/0/1]lldp enable
```

开启 LLDP 功能后，需要配置合适的 LLDP 端口工作模式，支持四种工作模式：


```
[H3C-GigabitEthernet1/0/1]lldp admin-status { disable | rx | tx | txrx }
```

其中参数说明如下：

- **disable**：端口工作在 disable 模式，此模式下端口即不接受也不发送 LLDP 报文。
- **rx**：端口工作在 rx 模式，此模式下端口不发送 LLDP 报文但接收 LLDP 报文。
- **tx**：端口工作在 tx 模式，此模式下端口至发送 LLDP 报文但不接收 LLDP 报文。
- **txrx**：端口工作在 txrx 模式，此模式下端口既发送也接收 LLDP 报文。

## 30.4.2 配置 LLDP 全局参数

## 配置LLDP全局参数

 紫光集团 **H3C**  
核心企业 数字化解决方案领导者

- 配置快速发送报文数
 

[H3C]lldp fast-count *count*
- 配置TTL乘数
 

[H3C]lldp hold-multiplier *value*
- 配置LLDP定时器
 

[H3C]lldp timer notification-interval *interval*

[H3C]lldp timer reinit-delay *delay*

[H3C]lldp timer tx-interval *interval*

www.h3c.com

系统视图下配置快速发送报文数，快速发送报文个数缺省值 4，取值范围 1-8：

```
[H3C]lldp fast-count count
```

系统视图下配置 TTL 乘数，TTL 乘数缺省值 4，取值范围 2-10：

```
[H3C]lldp hold-multiplier value
```

系统视图下配置 LLDP trap 定时器，缺省值 30 秒，取值范围 5-3600：

```
[H3C]lldp timer notification-interval interval
```

系统视图下配置 LLDP 重初始化延时计时器，缺省值 2 秒，取值范围 1-10：

```
[H3C]lldp timer reinit-delay delay
```

系统视图下配置 LLDP 报文发送周期，缺省值 30 秒，取值分为 5-32768：

```
[H3C]lldp timer tx-interval interval
```

## 30.4.3 配置端口 LLDP 运行参数

## 配置端口LLDP运行参数

紫光集团 H3C  
核心企业 数字化转型方案领导者

- 配置封装类型

```
[H3C-GigabitEthernet1/0/1]lldp encapsulation snap
```
- 配置发送TLV类型

```
[H3C-GigabitEthernet1/0/1]lldp tlv-enable { basic-tlv { all | port-description | system-capability | system-description | system-name | management-address-tlv [ ip-address ] } | dot1-tlv { all | port-vlan-id | link-aggregation | dcbx | protocol-vlan-id [ vlan-id ] | vlan-name [ vlan-id ] | management-vid [ mvlan-id ] } | dot3-tlv { all | mac-physic | max-frame-size | power } | med-tlv { all | capability | inventory | network-policy | power-over-ethernet | location-id { civic-address device-type country-code { ca-type ca-value }&<1-10> | elin-address tel-number } } }
```
- 启动LLDP trap

```
[H3C-GigabitEthernet1/0/1]lldp notification remote-change enable
```

www.h3c.com

端口视图下配置 LLDP 报文的封装格式为 SNAP 封装，缺省情况下，LLDP 报文的封装格式为 Ethernet II 格式：

```
[H3C-GigabitEthernet 1/0/1]lldp encapsulation snap
```

端口视图下配置允许发送的 TLV 类型：

```
[H3C-GigabitEthernet1/0/1]lldp tlv-enable { basic-tlv { all | port-description | system-capability | system-description | system-name | management-address-tlv [ ip-address ] } | dot1-tlv { all | port-vlan-id | link-aggregation | dcbx | protocol-vlan-id [ vlan-id ] | vlan-name [ vlan-id ] | management-vid [ mvlan-id ] } | dot3-tlv { all | mac-physic | max-frame-size | power } | med-tlv { all | capability | inventory | network-policy | power-over-ethernet | location-id { civic-address device-type country-code { ca-type ca-value }&<1-10> | elin-address tel-number } } }
```

主要参数含义如下：

- basic-tlv: 基本可选 TLV。
- dot1-tlv: IEEE802.1 组织定义 TLV。
- dot3-tlv: IEEE802.3 组织定义 TLV。
- med-tlv: LLDP-MED TLV。如果禁止发布 802.3 的组织定义的 MAC/PHY Configuration/Status TLV，则 LLDP-MED TLV 将不会被发布，不论其是否被允许发

布；如果禁止发布 LLDP-MED Capabilities TLV，则其它 LLDP-MED TLV 将不会被发布，不论其是否被允许发布。

端口视图下启动 LLDP trap 功能：

[H3C-GigabitEthernet1/0/1]lldp notification remote-change enable

### 30.4.4 LLDP 的显示与调试

LLDP的显示与调试	
操作	命令
查看LLDP本地信息	<code>display lldp local-information [ global   interface interface-type interface-number ]</code>
查看LLDP远端信息	<code>display lldp neighbor-information [ [ interface interface-type interface-number ] [ verbose ] ] list [ system-name system-name ]</code>
显示LLDP统计信息	<code>display lldp statistics [ global   interface interface-type interface-number ]</code>
显示LLDP状态	<code>display lldp status [ interface interface-type interface-number ]</code>
显示TLV配置信息	<code>display lldp tlv-config [ interface interface-type interface-number ]</code>
LLDP的调试命令	<code>debugging lldp { all   error   event   fsm [ interface interface-type interface-number ]   packet [ receive   transmit ] [ interface interface-type interface-number ] [ verbose ] }</code>

配置完 LLDP 后，可以通过 `display lldp local-information` 命令察看 LLDP 本地信息库信息。

```
[H3C]display lldp local-information
Global LLDP local-information:
Chassis ID       : 70ba-ef6a-76f9
System name      : H3C
System description : H3C Comware Platform Software, Software Version
7.1.045,
                  Release 2311P04
                  H3C S5820V2-54QS-GE
                  Copyright (c) 2004-2014 Hangzhou H3C Tech. Co., Ltd. All
                  rights reserved.
System capabilities supported : Bridge, Router, Customer Bridge, Service
Bridge
System capabilities enabled   : Bridge, Router, Customer Bridge

MED information:
Device class                : Connectivity device
MED inventory information of master board:
HardwareRev                  : Ver.A
FirmwareRev                  : 132
SoftwareRev                  : 7.1.045 Release 2311P04
SerialNum                    : 210235A0XAH148000037
```

Manufacturer name : H3C  
 Model name : H3C S5820V2-54QS-GE  
 Asset tracking identifier : Unknown

LLDP local-information of port 1[GigabitEthernet1/0/1]:

Port ID type : Interface name  
 Port ID : GigabitEthernet1/0/1  
 Port description : GigabitEthernet1/0/1 Interface  
 LLDP agent nearest-bridge management address:  
 Management address type : IPv4  
 Management address : 1.1.1.1  
 Management address interface type : IfIndex  
 Management address interface ID : 971  
 Management address OID : 0  
 LLDP agent nearest-nontpmr management address:  
 Management address type : IPv4  
 Management address : 1.1.1.1  
 Management address interface type : IfIndex  
 Management address interface ID : 971  
 Management address OID : 0  
 LLDP agent nearest-customer management address:  
 Management address type : IPv4  
 Management address : 1.1.1.1  
 Management address interface type : IfIndex  
 Management address interface ID : 971  
 Management address OID : 0

DCBX Control info:

Oper version : Standard

DCBX ETS configuration info:

CBS : False

Max TCs : 8

CoS	Local Priority	Percentage	TSA
0	2	6	ETS
1	0	2	ETS
2	1	4	ETS
3	3	8	ETS
4	4	9	ETS
5	5	17	ETS
6	6	25	ETS
7	7	29	ETS

DCBX ETS recommendation info:

CoS	Local Priority	Percentage	TSA
0	2	6	ETS
1	0	2	ETS
2	1	4	ETS
3	3	8	ETS
4	4	9	ETS
5	5	17	ETS
6	6	25	ETS
7	7	29	ETS

DCBX PFC info:

P0-0	P1-0	P2-0	P3-0	P4-0	P5-0	P6-0	P7-0
------	------	------	------	------	------	------	------

Number of traffic classes supported: 8

Value of MBC: 0

Port VLAN ID(PVID) : 1

Port and protocol VLAN ID(PPVID) : 0

Port and protocol VLAN supported : No

Port and protocol VLAN enabled : No

VLAN name of VLAN 1 : VLAN 0001

Management VLAN ID : 0

Link aggregation supported : Yes

Link aggregation enabled : No

Aggregation port ID : 0

Auto-negotiation supported : Yes

Auto-negotiation enabled : Yes

```

OperMau          : Speed(1000)/Duplex(Full)
Power port class  : PSE
PSE power supported : No
PSE power enabled  : No
PSE pairs control ability : No
Power pairs       : Signal
Port power classification : Class 0
Maximum frame size : 10000

```

通过以上信息我们可以知道，LLDP 本地的全局和端口信息，这些信息包括全局的信息，全局的 MED 信息，端口下的端口信息、VLAN 信息、802.3 局域网端口信息、LLDP-MED 信息。

可以通过 **display lldp neighbor-information verbose** 命令察看 LLDP 邻居信息库的全部信息：

```

<H3C>display lldp neighbor-information verbose
LLDP neighbor-information of port 1[GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
  LLDP neighbor index : 1
  Update time         : 0 days, 11 hours, 59 minutes, 16 seconds
  Chassis type        : MAC address
  Chassis ID          : 0023-8928-74ae
  Port ID type        : Interface name
  Port ID             : GigabitEthernet1/0/1
  Time to live        : 120
  Port description    : GigabitEthernet1/0/1 Interface
  System name         : 5800
  System description  : H3C Comware Platform Software, Software Version 5.20,
  Rel ease 1808P27
  H3C S5800-60C-PWR
  Copyright (c) 2004-2014 Hangzhou H3C Tech. Co., Ltd. All
  rights reserved.
  System capabilities supported : Bridge, Router
  System capabilities enabled  : Bridge, Router
  Port VLAN ID(PVID) : 1
  Port and protocol VLAN ID(PPVID) : 0
  Port and protocol VLAN supported : Yes
  Port and protocol VLAN enabled  : No
  VLAN name of VLAN 1 : 1
  Link aggregation supported : Yes
  Link aggregation enabled  : No
  Aggregation port ID      : 0
  Auto-negotiation supported : Yes
  Auto-negotiation enabled  : Yes
  OperMau                  : Speed(1000)/Duplex(Full)
  Power port class         : PSE
  PSE power supported      : Yes
  PSE power enabled       : No
  PSE pairs control ability : No
  Power pairs             : Signal
  Port power classification : Class 0
  Maximum frame size      : 10000

```

可以通过 **display lldp neighbor-information** 参数显示邻居的简单信息：

```

<H3C>display lldp neighbor-information
LLDP neighbor-information of port 1[GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
  LLDP neighbor index : 1
  ChassisID/subtype   : 0023-8928-74ae/MAC address
  PortID/subtype      : GigabitEthernet1/0/1/Interface name
  Capabilities        : Bridge, Router

```

可以按列表显示邻居信息：

```
<H3C>display lldp neighbor-information list
Chassis ID : * -- -- Nearest nontpmr bridge neighbor
            # -- -- Nearest customer bridge neighbor
            Default -- -- Nearest bridge neighbor
System Name      Local Interface Chassis ID      Port ID
5800             GE1/0/1             0023-8928-74ae GigabitEthernet1/0/1
```

可以通过 **display lldp statistics** 察看 LLDP 的各种统计信息：

```
<H3C>display lldp statistics
LLDP statistics global information:
LLDP neighbor information last change time:0 days, 11 hours, 59 minutes, 16
seconds
The number of LLDP neighbor information inserted : 2
The number of LLDP neighbor information deleted : 1
The number of LLDP neighbor information dropped : 0
The number of LLDP neighbor information aged out : 0

LLDP statistics information of port 1 [GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
The number of LLDP frames transmitted : 101
The number of LLDP frames received : 64
The number of LLDP frames discarded : 0
The number of LLDP error frames : 0
The number of LLDP TLVs discarded : 0
The number of LLDP TLVs unrecognized : 0
The number of LLDP neighbor information aged out : 0
The number of CDP frames transmitted : 0
The number of CDP frames received : 0
The number of CDP frames discarded : 0
The number of CDP error frames : 0

LLDP agent nearest-nontpmr:
The number of LLDP frames transmitted : 0
The number of LLDP frames received : 0
The number of LLDP frames discarded : 0
The number of LLDP error frames : 0
The number of LLDP TLVs discarded : 0
The number of LLDP TLVs unrecognized : 0
The number of LLDP neighbor information aged out : 0
The number of CDP frames transmitted : 0
The number of CDP frames received : 0
The number of CDP frames discarded : 0
The number of CDP error frames : 0

LLDP agent nearest-customer:
The number of LLDP frames transmitted : 0
The number of LLDP frames received : 0
The number of LLDP frames discarded : 0
The number of LLDP error frames : 0
The number of LLDP TLVs discarded : 0
The number of LLDP TLVs unrecognized : 0
The number of LLDP neighbor information aged out : 0
The number of CDP frames transmitted : 0
The number of CDP frames received : 0
The number of CDP frames discarded : 0
The number of CDP error frames : 0
```

通过 **display lldp status** 命令察看 LLDP 模块状态，主要包括 LLDP 全局和端口下的启动情况，当前系统的邻居的总数，邻居信息库最近一次的变化时间，LLDP 全局配置参数，端口配置情况，端口邻居统计等：

```

<H3C>display lldp status
Global status of LLDP: Enable
Bridge mode of LLDP: customer-bridge
The current number of LLDP neighbors: 1
The current number of CDP neighbors: 0
LLDP neighbor information last changed time: 0 days, 11 hours, 59 minutes, 16
seconds
Transmit interval           : 30s
Fast transmit interval      : 1s
Transmit max credit         : 5
Hold multiplier             : 4
Reinit delay                : 2s
Trap interval               : 30s
Fast start times            : 4

LLDP status information of port 1 [GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
Port status of LLDP        : Enable
Admin status               : TX_RX
Trap flag                  : No
MED trap flag              : No
Polling interval           : 0s
Number of LLDP neighbors   : 1
Number of MED neighbors    : 0
Number of CDP neighbors    : 0
Number of sent optional TLV : 10
Number of received unknown TLV : 0

LLDP agent nearest-nontpmr:
Port status of LLDP        : Enable
Admin status               : Disable
Trap flag                  : No
MED trap flag              : No
Polling interval           : 0s
Number of LLDP neighbors   : 0
Number of MED neighbors    : 0
Number of CDP neighbors    : 0
Number of sent optional TLV : 0
Number of received unknown TLV : 0

LLDP agent nearest-customer:
Port status of LLDP        : Enable
Admin status               : Disable
Trap flag                  : No
MED trap flag              : No
Polling interval           : 0s
Number of LLDP neighbors   : 0
Number of MED neighbors    : 0
Number of CDP neighbors    : 0
Number of sent optional TLV : 0
Number of received unknown TLV : 0

```

通过 **display lldp tlv-config** 命令显示各个端口下的 TLV 发送配置:

```

<H3C>display lldp tlv-config
LLDP tlv-config of port 1[GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
NAME                               STATUS   DEFAULT
Basic optional TLV:
Port Description TLV               YES     YES
System Name TLV                   YES     YES
System Description TLV             YES     YES
System Capabilities TLV           YES     YES
Management Address TLV            YES     YES
IEEE 802.1 extend TLV:

```



Port VLAN ID TLV	YES	YES
Port And Protocol VLAN ID TLV	NO	NO
VLAN Name TLV	NO	NO
DCBX TLV	NO	NO
EVB TLV	NO	NO
Link Aggregation TLV	YES	YES
Management VID TLV	NO	NO
IEEE 802.3 extend TLV:		
MAC-Physic TLV	YES	YES
Power via MDI TLV	YES	YES
Maximum Frame Size TLV	YES	YES
LLDP-MED extend TLV:		
Capabilities TLV	YES	YES
Network Policy TLV	YES	YES
Location Identification TLV	NO	NO
Extended Power via MDI TLV	YES	YES
Inventory TLV	YES	YES

## LLDP agent nearest-nontpmr:

NAME	STATUS	DEFAULT
------	--------	---------

## Basic optional TLV:

Port Description TLV	NO	NO
System Name TLV	NO	NO
System Description TLV	NO	NO
System Capabilities TLV	NO	NO
Management Address TLV	NO	NO

## IEEE 802.1 extend TLV:

Port VLAN ID TLV	NO	NO
Port And Protocol VLAN ID TLV	NO	NO
VLAN Name TLV	NO	NO
DCBX TLV	NO	NO
EVB TLV	YES	YES
Link Aggregation TLV	NO	NO
Management VID TLV	NO	NO

## IEEE 802.3 extend TLV:

MAC-Physic TLV	NO	NO
Power via MDI TLV	NO	NO
Maximum Frame Size TLV	NO	NO

## LLDP-MED extend TLV:

Capabilities TLV	NO	NO
Network Policy TLV	NO	NO
Location Identification TLV	NO	NO
Extended Power via MDI TLV	NO	NO
Inventory TLV	NO	NO

## LLDP agent nearest-customer:

NAME	STATUS	DEFAULT
------	--------	---------

## Basic optional TLV:

Port Description TLV	YES	YES
System Name TLV	YES	YES
System Description TLV	YES	YES
System Capabilities TLV	YES	YES
Management Address TLV	YES	YES

## IEEE 802.1 extend TLV:

Port VLAN ID TLV	YES	YES
Port And Protocol VLAN ID TLV	NO	NO
VLAN Name TLV	NO	NO
DCBX TLV	NO	NO
EVB TLV	NO	NO
Link Aggregation TLV	YES	YES
Management VID TLV	NO	NO

## IEEE 802.3 extend TLV:

MAC-Physic TLV	NO	NO
Power via MDI TLV	NO	NO
Maximum Frame Size TLV	NO	NO

可以通过 **debug** 调试命令，观察 LLDP 协议实体运行过程，包括协议状态机的变化、收信息、协议运行过程中的相关事件、各种错误等：

- 691 -

\*Jan 1 12:41:29:616 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=0.

\*Jan 1 12:41:30:609 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=1.

\*Jan 1 12:41:30:613 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=55.

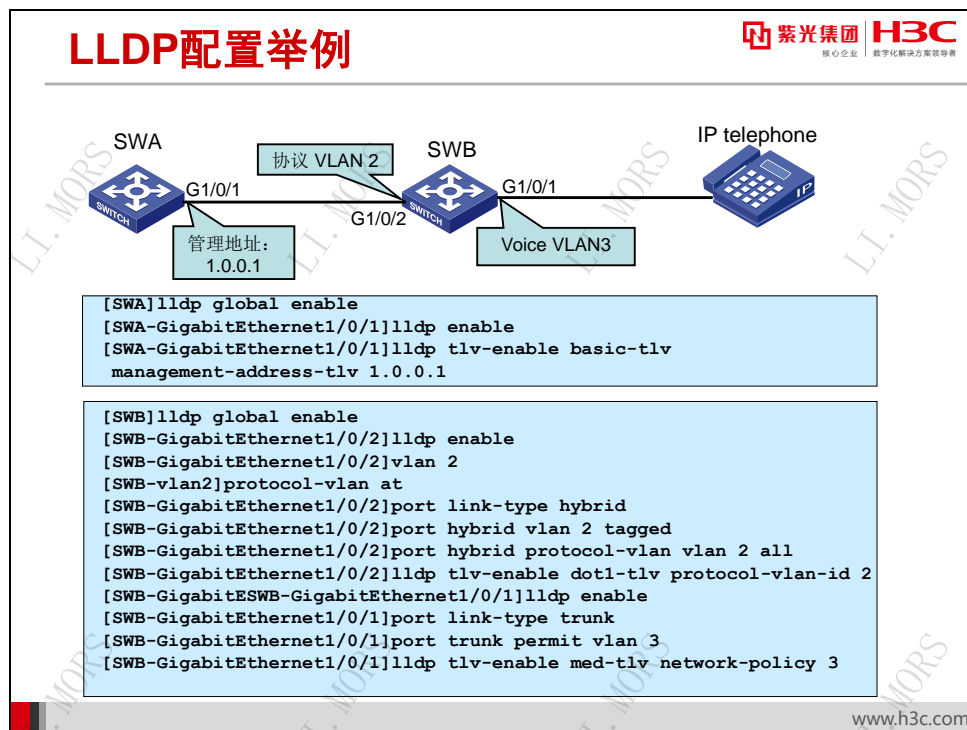
\*Jan 1 12:41:30:617 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=54.

\*Jan 1 12:41:30:620 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=53.

\*Jan 1 12:41:30:624 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=52.

\*Jan 1 12:41:30:629 2011 H3C LLDP/7/Event: Thread proc: type=1, subtype=0, curque=51.

### 30.4.5 LLDP 配置示例



如上图所示，网络设备默认发送各种基本和组织定义 TLV。其中 SWA 设备发布的管理地址为 1.0.0.1，应该做如下配置：

```

[SWA]lldp global enable
[SWA-GigabitEthernet1/0/1]lldp enable
[SWA-GigabitEthernet1/0/1]lldp tlv-enable basic-tlv management-address-tlv 1.0.0.1
  
```

SWB 设备和 SWA 设备的连接端口属于配置协议 VLAN 2，并且通过 802.1 组织定义 TLV 发布该信息，应该做如下配置：

```
[SWB]lldp global enable
[SWB-GigabitEthernet1/0/2]lldp enable
[SWB-GigabitEthernet1/0/2]vlan 2
[SWB-vlan2]protocol-vlan at
[SWB-GigabitEthernet1/0/2]port link-type hybrid
[SWB-GigabitEthernet1/0/2]port hybrid vlan 2 tagged
[SWB-GigabitEthernet1/0/2]port hybrid protocol-vlan vlan 2 all
[SWB-GigabitEthernet1/0/2]lldp tlv-enable dot1-tlv protocol-vlan-id 2
```

SWB 设备的 G1/0/1 端口连接 IP 电话，交换机通过 LLDP 将 Voice vlan 信息通报给话机。

```
[SWB-GigabitEthernet1/0/1]lldp enable
[SWB-GigabitEthernet1/0/1]port link-type trunk
[SWB-GigabitEthernet1/0/1]port trunk permit vlan 3
[SWB-GigabitEthernet1/0/1]lldp tlv-enable med-tlv network-policy 3
```

## 30.5 本章总结

### 本章总结

- LLDP为不同厂商设备组网提供统一的拓扑发现和交互系统配置信息提供统一平台
- LLDP提供两种封装格式，对应不同的网络类型
- LLDP PDU由四种必选TLV和若干可选TLV组成
- LLDP TLV包括基本TLV、组织定义TLV、LLDP-MED扩展TLV
- LLDP工作模式包括TXRX、TX、RX、disable
- LLDP支持快速开始发送和基本周期发送

www.h3c.com

## 30.6 习题和解答

### 30.6.1 习题

1. LLDP 报文中必须包含下面哪些 TLV? ( )  
A. Chassis ID TLV                      B. Port ID TLV  
C. Time to Live TLV                    D. System Name TLV
2. 下面哪些 TLV 在 LLDPBPDU 中可以存在多个? ( )  
A. Time to Live TLV                    B. End Of LLDPDU TLV  
C. Port And Protocol VLAN ID          D. Network Policy TLV
3. LLDP 的端口工作模式有哪些? ( )  
A. TxRx    B. Tx    C. Rx    D. default
4. 如果 A 设备的远端信息库里有 B 设备信息, 则 B 设备的远端信息库里也会有 A 设备的信息。( )  
T. 正确            F. 错误
5. STP 的阻塞端口也能接收和发送 LLDP 报文 ( )  
T. 正确            F. 错误
6. 链路聚合的逻辑口可以支持 LLDP 协议 ( )  
T. 正确            F. 错误

### 30.6.2 习题答案

1. ABC
2. CD
3. ABC
4. F
5. T
6. F

## 第31章 镜像技术

在日常网络维护过程中，掌握网络中转发传输的是什么流量，流量从哪个网络节点出发到哪个网络节点终止，目前网络带宽占用实际比例是多少，是一个网络管理员最为关心的事情，也是必须关心的事情。镜像技术就是掌握上述信息的一种重要辅助手段。

另外在 IT 技术日益发达和广泛应用的今天，对重要数据网络和公共信息网络进行安全审查也显得尤为重要。尽管这些任务大多都交给防火墙、IPS、IDS 等安全设备，但将流量送给某些安全设备也需要采用镜像技术来完成报文的复制，使得在监控信息的同时不影响业务的正常开展。

### 31.1 本章目标

#### 课程目标

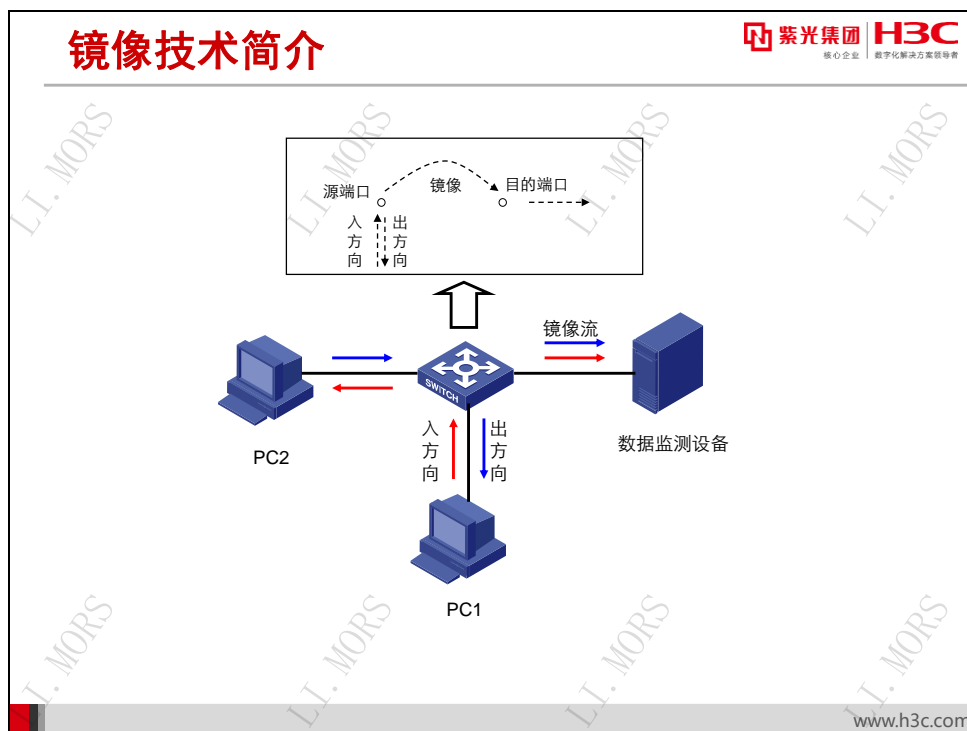
● 学习完本课程，您应该能够：

- 了解目前的几种镜像方法
- 掌握端口镜像的操作
- 掌握远程镜像的原理和操作
- 掌握流镜像的操作



## 31.2 镜像技术概述和原理

### 31.2.1 镜像技术简介

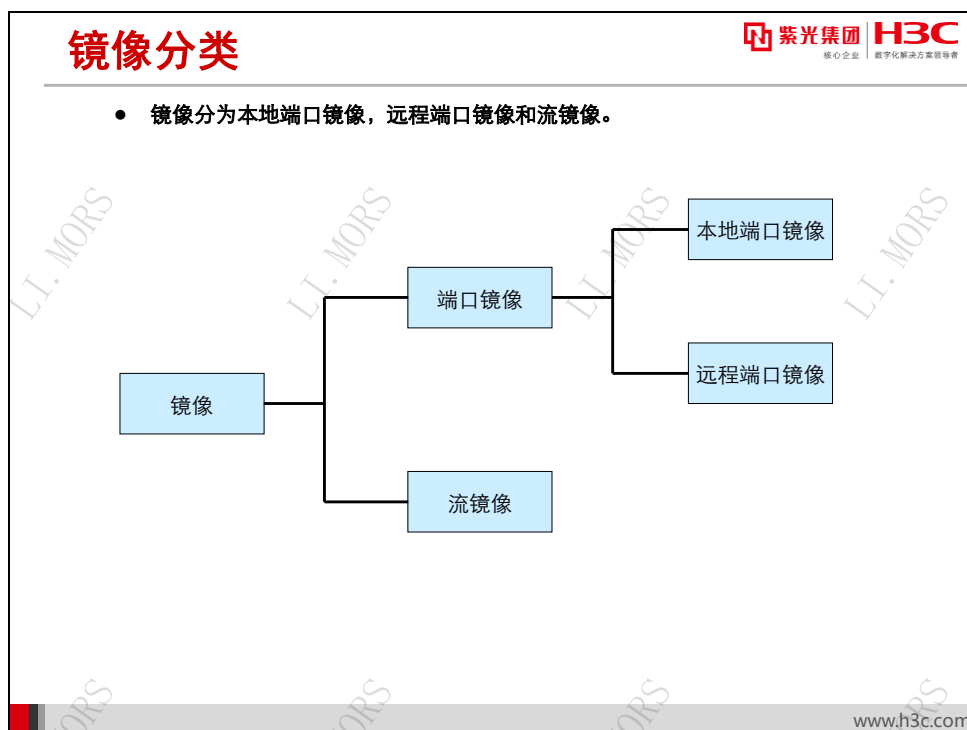


镜像就是将指定端口的报文或者符合指定规则的报文复制到目的端口。用户可利用镜像技术，进行网络监管和故障排除。其中镜像技术中应用最为广泛，实现最为简单的当属端口镜像，为了便于认识镜像技术，首先介绍端口镜像中涉及的几个基本概念：

- **源端口：**源端口是被监控的端口，用户可以对通过该端口的报文进行监控和分析。
- **目的端口：**目的端口也可称为监控端口，该端口将接收到的报文转发到数据监测设备，以便对报文进行监控和分析。
- **镜像的方向：**端口镜像的方向分为入方向、出方向和双向。
  - ◆ 入方向：仅对源端口接收的报文进行镜像。
  - ◆ 出方向：仅对源端口发送的报文进行镜像。
  - ◆ 双向：对源端口接收和发送的报文都进行镜像。



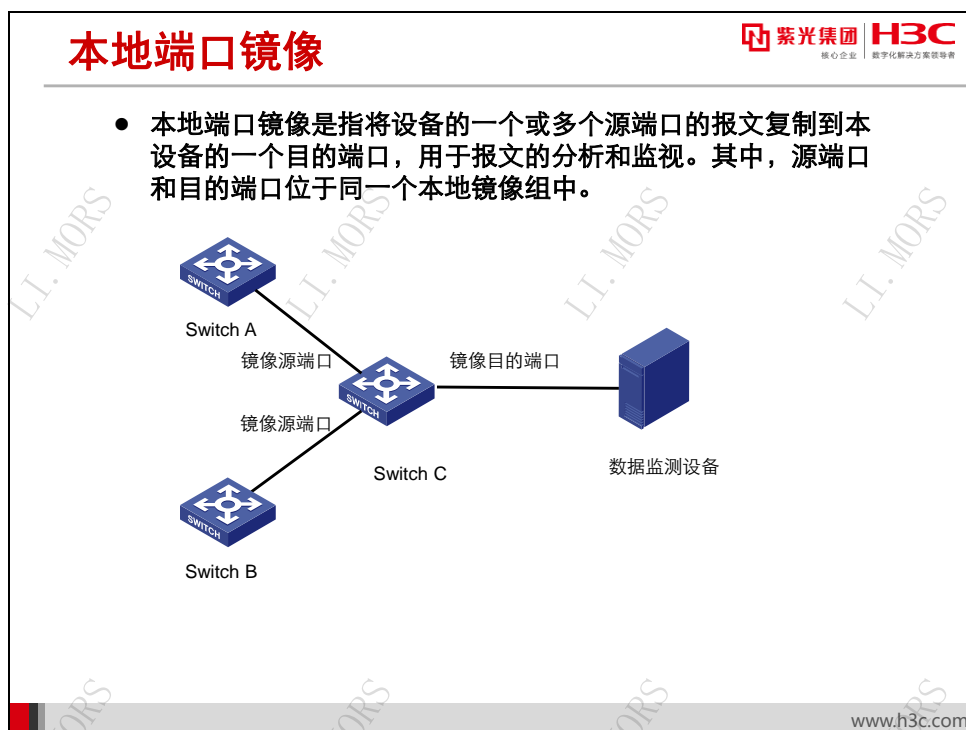
## 31.2.2 镜像分类



镜像技术包括如下三种：

- **本地端口镜像：**是指将设备的一个或多个端口（源端口）的报文复制到本设备的一个监视端口（目的端口），用于报文的监视和分析。其中，源端口和目的端口必须在同一台设备上。
- **远程端口镜像：**是指将设备的一个或多个端口的报文复制并通过中间网络设备转发到指定目的交换机上的目的端口。它突破了源端口和目的端口必须在同一台设备上的限制，使源端口和目的端口间可以跨越多个网络设备。
- **流镜像：**是指通过 ACL 等规则将具有某特征的数据流复制到目的端口。它通过 QoS 策略实现，即使用流分类技术来定义需要被镜像的报文的匹配条件，再通过配置流行为将符合条件的报文镜像至指定的方向。因此流镜像可以避免过多的冗余流量被复制到目的端口。

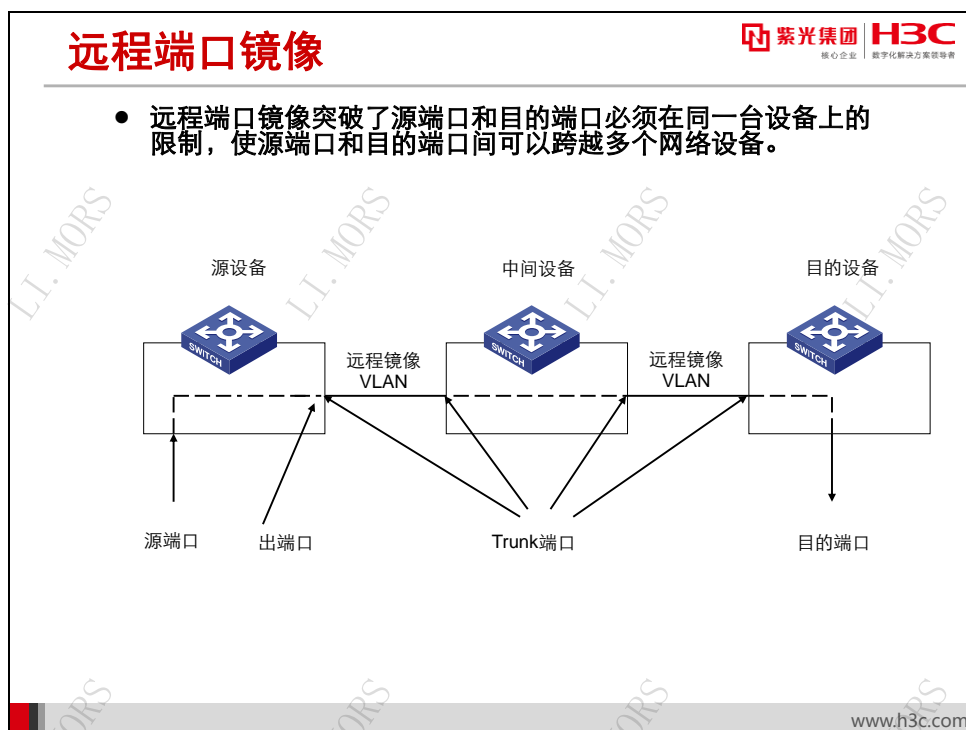
## 31.2.3 本地端口镜像



本地端口镜像是最为简单的镜像技术，H3C 网络设备通过本地镜像组的方式来实现。同一个镜像组可以包含一个或多个镜像源端口，但只能有一个镜像目的端口，且可以根据实际需要指定源端口上被镜像报文的方向。当完成端口镜像的源端口和目的端口配置之后，交换机按照正常的转发规则转发所有报文外，还在所有镜像源端口根据实际配置的镜像方向将报文复制到镜像目的端口。

由于源端口可以是多个，因此在做本地端口镜像时需要注意实际被镜像流量不要超过目的端口的物理带宽，否则容易导致镜像报文因拥塞而丢弃。同时由于镜像目的端口将发送大量的被镜像报文，因此此端口不宜用作正常业务转发。

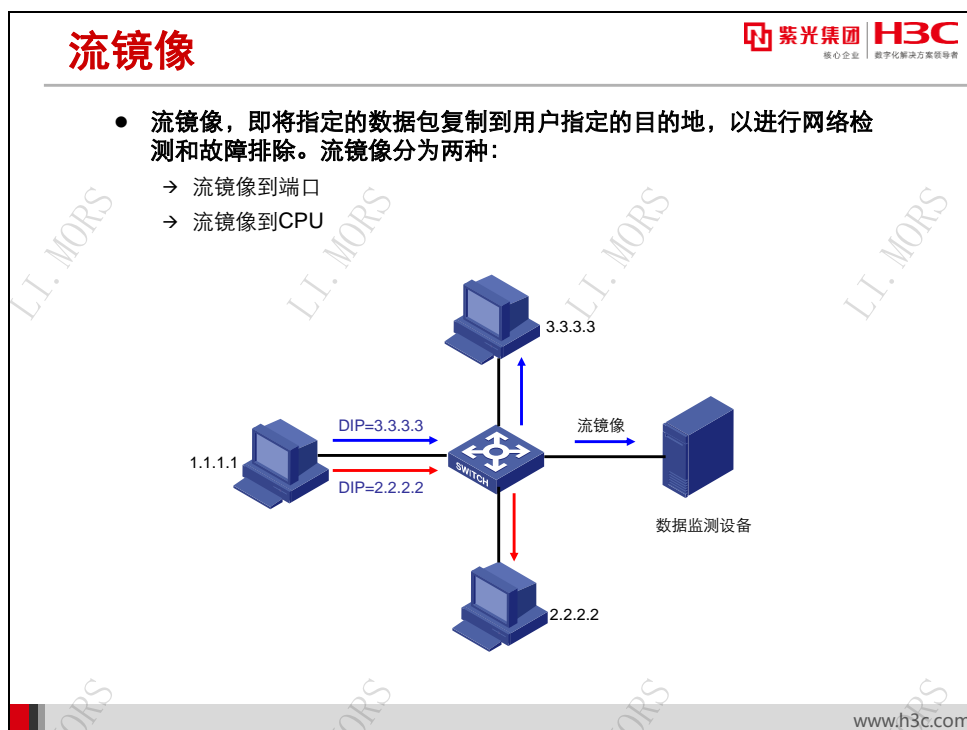
## 31.2.4 远程端口镜像



远程端口镜像通过远程源镜像组和远程目的镜像组互相配合的方式实现。在整个远程端口镜像过程中，源设备负责将源端口的报文在特定 VLAN 内通过出端口转发出去。网络中间设备在指定的 VLAN 内将报文转发到目的设备。目的设备从指定 VLAN 内接收镜像报文并转发给目的端口。如图所示，完成远程镜像至少包含源设备和目的设备，大多数情况下还存在中间设备：

- **源设备：**源端口所在的设备，用户需要在源设备上创建远程源镜像组。本设备负责将源端口的报文复制一份，在远程镜像 VLAN 中通过出端口将报文转发出去，传输给中间设备或目的设备。
- **中间设备：**网络中处于源设备和目的设备之间的设备。中间设备负责将镜像报文传输给下一个中间设备或目的设备。如果源设备与目的设备直接相连，则不存在中间设备。用户需要确保远程镜像 VLAN 内源设备到目的设备的二层互通性。且在实现端口的双向镜像时需要保证中间设备可以禁止该 VLAN 内 MAC 地址的学习，从而可以正常的将镜像报文从一个端口转发到另一个端口。
- **目的设备：**远程镜像目的端口所在的设备，用户需要在目的设备上创建远程目的镜像组。目的设备收到报文后，比较报文的 VLAN ID 和远程目的镜像组的远程镜像 VLAN 是否相同，如果相同，则将该报文通过镜像目的端口转发给监控设备。

## 31.2.5 流镜像



流镜像，即将指定的数据包复制到用户指定的目的地，以进行网络检测和故障排除。

流镜像根据实际镜像的目标不同可以分为两种：

- **流镜像到端口：**是把通过配置了流镜像的端口的符合要求的数据包复制一份，然后发送到目的端口。
- **流镜像到 CPU：**是把通过配置了流镜像的端口的符合要求的数据包复制一份，然后发送到 CPU 以供分析诊断。在特定的条件下可以通过将流量镜像到 CPU，并通过设备的 debugging 命令查看收到报文的具体内容而不需要额外的监控终端。

## 31.3 配置端口镜像

### 31.3.1 端口镜像配置命令

## 配置端口镜像

**紫光集团 H3C**  
核心企业 | 数字化解决方案领导者

- 创建一个本地镜像组
 

**[H3C]mirroring-group group-id local**
- 配置源端口
 

**[H3C]mirroring-group group-id mirroring-port interface-list**  
**{ both | inbound | outbound }**
- 配置目的端口
 

**[H3C]mirroring-group group-id monitor-port interface-type**  
**interface-number**

www.h3c.com

端口镜像的配置可分为三个步骤：

**第1步：**在系统视图下创建本地镜像组。**mirroring-group local** 命令用来创建一个本地镜像组。

**[H3C]mirroring-group group-id local**

**group-id：**表示端口镜像组的组号。不同的产品取值范围不同，具体取值范围参考相关产品配套手册。

**第2步：**在系统视图或者端口视图下配置源端口。**mirroring-group mirroring-port** 命令用来为镜像组配置源端口。

**[H3C]mirroring-group group-id mirroring-port interface-list { both | inbound | outbound }**

**[H3C-GigabitEthernet1/0/1]mirroring-group group-id mirroring-port { both | inbound | outbound }**

其中主要参数说明如下：

- **group-id：**端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。

- **interface-list**: 端口列表，表示一个或多个源端口。表示方式为 **interface-list = { interface-type interface-number [ to interface-type interface-number ] }&<1-8>**。其中，**interface-type interface-number** 为端口类型和端口编号。**&<1-8>** 表示前面的参数最多可以输入 8 次。当使用 **to** 参数配置端口范围时，起始端口和终止端口必须是相同设备上相同类型的端口，且终止端口的端口编号必须大于等于起始端口的端口编号。
- **both**: 表示对端口接收和发送的报文都进行镜像。
- **inbound**: 表示仅对端口接收的报文进行镜像。
- **outbound**: 表示仅对端口发送的报文进行镜像。

**第3步:** 可在系统视图或者端口视图下配置目的端口。**mirroring-group monitor-port** 命令用来为镜像组配置目的端口。

```
[H3C]mirroring-group group-id monitor-port interface-type interface-number
```

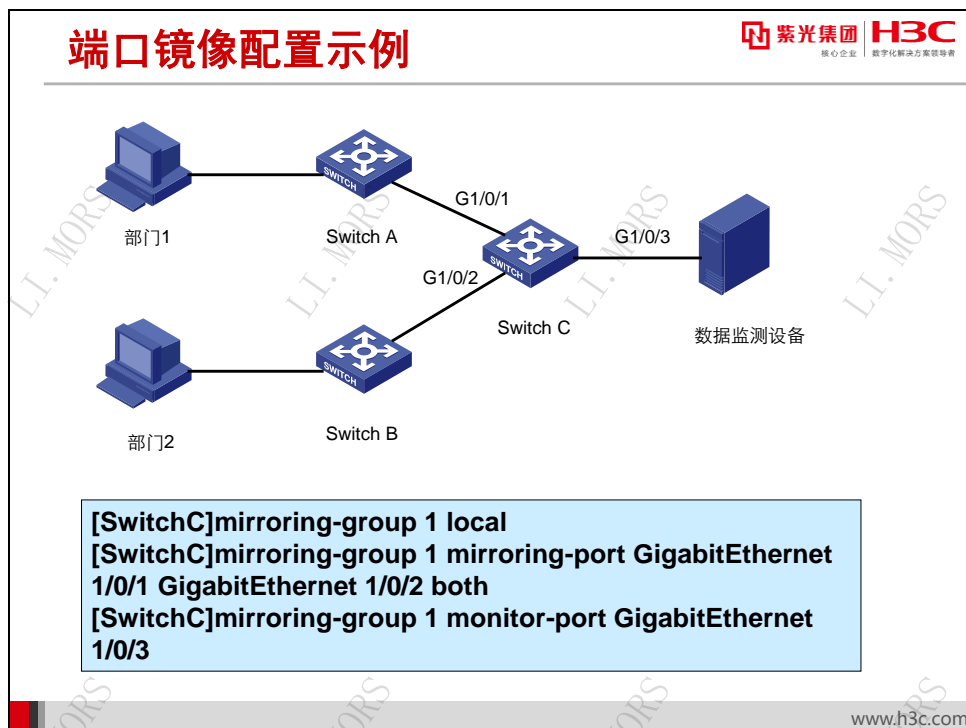
```
[H3C-GigabitEthernet1/0/1]mirroring-group group-id monitor-port
```

其中主要参数说明如下：

- **group-id**: 端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。
- **interface-type interface-number**: 目的端口。其中，**interface-type** 为端口类型，**interface-number** 为端口号。

同一镜像组只能配置一个镜像目的端口，要想修改镜像组的目的端口必须先删除原有目的端口。

## 31.3.2 端口镜像配置示例



在如图所示的组网中，在交换机 **SwitchC** 上通过本地端口镜像将端口 **GigabitEthernet1/0/1** 和 **GigabitEthernet1/0/2** 两个端口的双向流量镜像到目的端口 **GigabitEthernet1/0/3**，已实现对两个部门之间流量的监控。其具体配置步骤和配置命令如下：

#### #创建本地镜像组 1

```
[SwitchC]mirroring-group 1 local
```

#配置镜像源，对端口 **GigabitEthernet1/0/1** 和端口 **GigabitEthernet1/0/2** 的入/出数据流进行镜像

```
[SwitchC]mirroring-group 1 mirroring-port GigabitEthernet 1/0/1 GigabitEthernet
1/0/2 both
```

#### #配置 **GigabitEthernet1/0/3** 为镜像目的端口，即监控端口

```
[SwitchC]mirroring-group 1 monitor-port GigabitEthernet 1/0/3
```

## 31.4 配置远程镜像

### 31.4.1 远程镜像配置任务

### 配置远程镜像

紫光集团 H3C  
核心企业 数字化解决方案领导者

- 远程端口镜像通过远程源镜像组和远程目的镜像组互相配合的方式实现。
  - 在源设备上配置远程源镜像组
    - 源端口
    - 出端口
    - 远程镜像VLAN
  - 配置中间设备
    - 配置远程镜像VLAN，禁止MAC地址学习
    - 配置TRUNK端口
  - 在目的设备上配置远程目的镜像组
    - 远程镜像VLAN
    - 目的端口

www.h3c.com

远程端口镜像需要分别在源设备，目的设备和中间设备上配置，具体配置任务包括：

- **在源设备上配置远程源镜像组：**首先需要在系统视图下创建远程源镜像组，然后为镜像组配置源端口、出端口和 **Proble VLAN**。
- **配置中间设备：**中间设备只需要在指定 VLAN 内禁止 MAC 地址学习并将互联端口配置为 Trunk 端口且允许指定 VLAN 通过。
- **在目的设备上配置目的镜像组：**首先需要在系统视图下创建远程目的镜像组，然后为镜像组配置目的端口和 **Proble VLAN**。



## 31.4.2 配置源设备

## 配置源设备

**紫光集团** **H3C**  
核心企业 | 数字化转型方案领导者

- 配置远程源镜像组
  - 创建远程源镜像组
 

**[H3C]mirroring-group group-id remote-source**
  - 配置源端口
 

**[H3C]mirroring-group group-id mirroring-port interface-list { both | inbound | outbound }**
  - 配置出端口
 

**[H3C]mirroring-group group-id monitor-egress interface-type interface-number**
  - 配置远程镜像VLAN
 

**[H3C]mirroring-group group-id remote-probe vlan vlan-id**

www.h3c.com

远程镜像的源设备配置分为四个步骤：

**第1步：**在系统视图下创建远程源镜像组。mirroring-group remote-source 命令用来创建一个远程源镜像组。

**[H3C]mirroring-group group-id remote-source**

其中参数 *group-id* 为端口镜像组的组号，不同的产品取值范围不同，具体取值范围参考相关产品配套手册。

**第2步：**在系统视图或者端口视图下配置源端口。mirroring-group mirroring-port 命令用来为镜像组配置源端口。

**[H3C]mirroring-group group-id mirroring-port interface-list { both | inbound | outbound }**

**[H3C-GigabitEthernet1/0/17]mirroring-group group-id mirroring-port { both | inbound | outbound }**

其中参数说明如下：

- *group-id*: 端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。
- *interface-list*: 端口列表，表示多个端口。表示方式为 *interface-list* = { **interface-type interface-number [ to interface-type interface-number ]** }&<1-8>。其中，**interface-type** 为端口类型，*interface-number* 为端口号。&<1-8>表示前面的参数最多可以输入

8 次。当使用 **to** 参数配置端口范围时，起始端口和终止端口必须是相同设备上相同类型的端口，且终止端口的端口编号必须大于等于起始端口的端口编号。

- **both**: 表示对端口接收和发送的报文都进行镜像。
- **inbound**: 表示仅对端口接收的报文进行镜像。
- **outbound**: 表示仅对端口发送的报文进行镜像。

**第3步:** 可在系统视图或者端口视图下配置出端口。**mirroring-group monitor-egress** 命令用来为镜像组配置出端口。

```
[H3C]mirroring-group group-id monitor-egress interface-type interface-number
```

```
[H3C-GigabitEthernet1/0/18]mirroring-group group-id monitor-egress
```

其中主要参数说明如下:

- **group-id**: 端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。
- **interface-type interface-number**: 表示出端口。其中，**interface-type interface-number** 为端口类型和端口编号。

**第4步:** 在系统视图下创建远程镜像 VLAN。**mirroring-group remote-probe vlan** 命令用来配置远程镜像 VLAN。


```
[H3C]mirroring-group group-id remote-probe vlan vlan-id
```

其中主要参数说明如下:

- **group-id**: 端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。
- **vlan-id**: 远程镜像 VLAN ID，该 VLAN 必须已经存在并且为静态 VLAN。被配置成远程镜像 VLAN 后，该 VLAN 不能直接删除，必须先删除远程镜像 VLAN 的配置才能够删除这个 VLAN。

## 31.4.3 配置中间设备

## 配置中间设备

紫光集团 H3C  
核心企业 数字化转型方案领导者

- 配置中间设备
  - 创建远程镜像VLAN，禁止MAC地址学习功能（假设为VLAN2）

```
[H3C]vlan 2
[H3C-vlan2]undo mac-address mac-learning enable
```
  - 配置TRUNK端口（GigabitEth1/0/1、GigabitEth1/0/2为例）

```
[H3C]interface GigabitEthernet 1/0/1
[H3C-GigabitEthernet1/0/1] port link-type trunk
[H3C-GigabitEthernet1/0/1] port trunk permit vlan 2
[H3C-GigabitEthernet1/0/1] quit
[H3C] interface GigabitEthernet 1/0/2
[H3C-GigabitEthernet1/0/2] port link-type trunk
[H3C-GigabitEthernet1/0/2] port trunk permit vlan 2
```

www.h3c.com

远程镜像的中间设备配置可分为两个步骤：

**第1步：**在 VLAN 视图下配置远程镜像 VLAN 属性。**remote-probe vlan enable** 命令用来将当前 VLAN 配置为远程镜像 VLAN。此步骤根据设备支持情况可选。

```
[H3C-vlan2]undo mac-address mac-learning enable
```



**第2步：**将与源设备/目的设备相连的端口设置为 TRUNK 属性，并允许远程镜像 VLAN 通过。

```
[H3C-GigabitEthernet1/0/1]port link-type { trunk | Hybrid | access }
```

```
[H3C-GigabitEthernet1/0/1]port trunk permit vlan vlan-id
```

## 31.4.4 配置目的设备

## 配置目的设备

 紫光集团   
核心企业 | 数字化解决方案领导者

- 配置远程目的镜像组
  - 创建远程目的镜像组

[H3C]mirroring-group group-id remote-destination
  - 配置远程镜像VLAN

[H3C]mirroring-group group-id remote-probe vlan vlan-id
  - 配置目的端口

[H3C]mirroring-group group-id monitor-port interface-type interface-number
  - 将目的端口加入远程镜像VLAN（access 端口为例）

[H3C-GigabitEthernet1/0/21]port access vlan vlan-id

www.h3c.com

远程镜像的目的设备配置可分为四个步骤：

**第1步：**在系统视图下创建远程目的镜像组。**mirroring-group remote-destination** 命令用来创建一个远程目的镜像组。

```
[H3C]mirroring-group group-id remote-destination
```

其中参数 *group-id* 为端口镜像组的组号。

**第2步：**在系统视图下创建远程镜像 VLAN。**mirroring-group remote-probe vlan** 命令用来为远程源镜像组或者远程目的镜像组配置远程镜像 VLAN。

```
[H3C]mirroring-group group-id remote-probe vlan vlan-id
```

其中主要参数说明如下：

- *group-id*：端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。
- *vlan-id*：远程镜像 VLAN ID，该 VLAN 必须已经存在并且为静态 VLAN。被配置成远程镜像 VLAN 后，该 VLAN 不能直接删除，必须先删除远程镜像 VLAN 的配置才能够删除这个 VLAN。

**第3步：**在系统视图或者端口视图下配置目的端口。**mirroring-group monitor-port** 命令用来为镜像组配置目的端口。

```
[H3C]mirroring-group group-id monitor-port interface-type interface-number
```

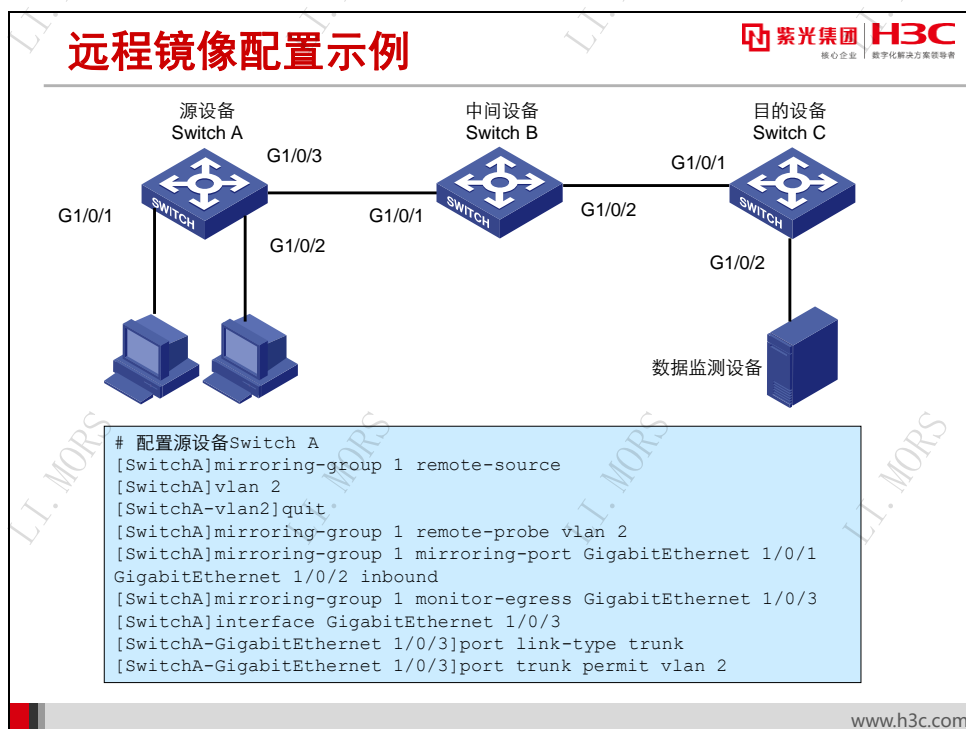
其中主要参数说明如下：

- **group-id**: 端口镜像组的组号，在配置此命令前必须保证此镜像组已经创建。
- **interface-type interface-number**: 目的端口，其中，**interface-type** 为端口类型，**interface-number** 为端口号。

**第4步：**将目的端口加入远程镜像 VLAN 中。

[H3C-GigabitEthernet1/0/21]port access vlan *vlan-id*

### 31.4.5 远程镜像配置示例



- 配置源设备 Switch A

#### #创建远程源镜像组 1

```
[SwitchA]mirroring-group 1 remote-source
```

#### #创建静态 VLAN2，配置 VLAN2 为远程镜像 VLAN

```
[SwitchA]vlan 2
[SwitchA]mirroring-group 1 remote-probe vlan 2
```

#### #对端口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 的入方向数据流量进行镜像

```
[SwitchA]mirroring-group 1 mirroring-port GigabitEthernet 1/0/1
GigabitEthernet 1/0/2 inbound
```

#### #端口 G1/0/3 为出端口

```
[SwitchA]mirroring-group 1 monitor-egress Ethernet 1/0/3
```

#### #设置端口 GigabitEthernet1/0/3 的 TRUNK 属性，并允许远程镜像 VLAN 通过

```
[SwitchA]interface Ethernet 1/0/3
[SwitchA-GigabitEthernet1/0/3]port link-type trunk
```

```
[SwitchA-GigabitEthernet1/0/3]port trunk permit vlan 2
```

## 远程镜像配置示例（续）



```
# 配置中间设备Switch B(S5820V2交换机为例)
[SwitchB]vlan 2
[SwitchB-vlan2]undo mac-address mac-learning enable
[SwitchB-vlan2]quit
[SwitchB]interface GigabitEthernet 1/0/1
[SwitchB-GigabitEthernet1/0/1]port link-type trunk
[SwitchB-GigabitEthernet1/0/1]port trunk permit vlan 2
[SwitchB-GigabitEthernet1/0/1]quit
[SwitchB]interface GigabitEthernet 1/0/2
[SwitchB-GigabitEthernet1/0/2]port link-type trunk
[SwitchB-GigabitEthernet1/0/2]port trunk permit vlan 2
# 配置目的设备Switch C
[SwitchC]interface GigabitEthernet 1/0/1
[SwitchC-GigabitEthernet 1/0/1] port link-type trunk
[SwitchC-GigabitEthernet 1/0/1] port trunk permit vlan 2
[SwitchC-GigabitEthernet 1/0/1] quit
[SwitchC]mirroring-group 1 remote-destination
[SwitchC]vlan 2
[SwitchC-vlan2]quit
[SwitchC]mirroring-group 1 remote-probe vlan 2
[SwitchC]mirroring-group 1 monitor-port GigabitEthernet 1/0/2
[SwitchC]interface GigabitEthernet 1/0/2
[SwitchC-GigabitEthernet 1/0/2]port access vlan 2
```

www.h3c.com

### ● 配置中间设备 Switch B

#配置端口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 的 TRUNK 属性，并允许远程镜像 VLAN 通过。在远程镜像 vlan 中禁止 mac 地址学习。

```
[SwitchB]vlan 2
[SwitchB-vlan2]undo mac-address mac-learning enable

[SwitchB]interface GigabitEthernet 1/0/1
[SwitchB-GigabitEthernet1/0/1]port link-type trunk
[SwitchB-GigabitEthernet1/0/1]port trunk permit vlan 2
[SwitchB] interface GigabitEthernet 1/0/2
[SwitchB-GigabitEthernet1/0/2]port link-type trunk
[SwitchB-GigabitEthernet1/0/2]port trunk permit vlan 2
```

### ● 配置目的设备 Switch C

#配置端口 GigabitEthernet1/0/1 的 TRUNK 属性，并允许远程镜像 VLAN 通过

```
[SwitchC]interface GigabitEthernet 1/0/1
[SwitchC-GigabitEthernet1/0/1]port link-type trunk
[SwitchC-GigabitEthernet1/0/1]port trunk permit vlan 2
```

#创建远程目的镜像组 1

```
[SwitchC]mirroring-group 1 remote-destination
```

#创建静态 VLAN2，配置 VLAN2 为远程镜像 VLAN

```
[SwitchC]vlan 2
[SwitchC]mirroring-group 1 remote-probe vlan 2
```

#配置镜像目的端口

```
[SwitchC]mirroring-group 1 monitor-port GigabitEthernet 1/0/2
```

### #配置镜像目的端口允许远程镜像 vlan 通过

```
[SwitchC]interface GigabitEthernet 1/0/2  
[SwitchC-GigabitEthernet 1/0/2]port access vlan 2
```

## 31.5 配置流镜像

### 31.5.1 流镜像配置命令

## 流镜像配置命令

**紫光集团 H3C**  
核心企业 | 数字化解决方案领导者

- 配置ACL，并指定ACL序号
 

**[H3C]acl basic *acl-number***
- 定义规则（基本ACL为例）
 

**[H3C-acl-ipv4-basic-2000]rule [ *rule-id* ] { **deny | permit** } [ **counting | fragment | logging | source { *sour-addr sour-wildcard | any* } | time-range *time-name*** ]**
- 配置流分类规则
 

**[H3C]traffic classifier *classifier-name* [ **operator { and | or }** ]**  
**[H3C-classifier-1]if-match *match-criteria***
- 进入流行为模式，配置流镜像目的接口
 

**[H3C] traffic behavior *behavior-name***  
**[H3C-behavior-1]mirror-to interface *interface-type interface-number***
- 配置QoS策略
 

**[H3C]qos policy *policy-name***  
**[H3C-qospolicy-policy-name]classifier *classifier-name* behavior *behavior-name***
- 将QoS策略应用到端口上
 

**[H3C-GigabitEthernet1/0/1]qos apply policy *policy-name* inbound**

www.h3c.com

流镜像的配置可分为 6 个步骤，其具体配置参数含义详细参考 QoS Policy 的配置：

**第1步：**在系统视图下创建 ACL 并进入相应 ACL 视图。具体命令如下：

```
[H3C]acl basic acl-number
```

**第2步：**在 ACL 视图下定义规则（以基本 ACL 2000 为例）。

```
[H3C-acl-ipv4-basic-2000] rule [ rule-id ] { deny | permit } [ counting | fragment | logging | source { sour-addr sour-wildcard | any } | time-range time-range-name ]
```

**第3步：**在系统视图下创建流，并在流视图下定义匹配报文的规则。具体命令如下：

```
[H3C]traffic classifier classifier-name [ operator { and | or } ]  
[H3C-classifier-1]if-match match-criteria
```

**第4步：**在系统视图下创建流行为，并在流行为视图下配置流镜像目的端口。

```
[H3C]traffic behavior behavior-name  
[H3C-behavior-1]mirror-to interface interface-type interface-number
```

**第5步：**在系统视图下创建策略，并在策略中为类指定流行为，使类与流行为关联。

```
[H3C]qos policy policy-name
```

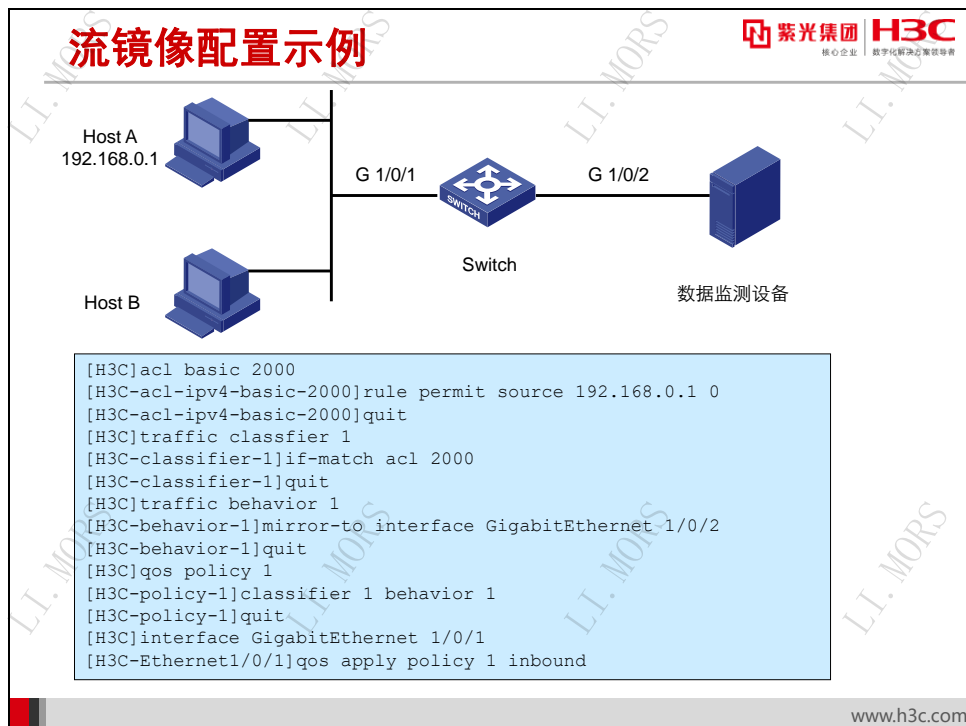


[H3C-qospolicy-policy-name]**classifier** *tcl-name* **behavior** *behavior-name*

**第6步：**在端口视图下，把 QoS 策略应用到端口上。

[H3C-GigabitEthernet1/0/1]**qos apply policy** *policy-name* **inbound**

### 31.5.2 流镜像配置示例



# 配置基本 ACL 2000，匹配源 IP 地址为 192.168.0.1 的报文。

```

[H3C]acl basic 2000
[H3C-acl-ipv4-basic-2000]rule permit source 192.168.0.1 0
[H3C-acl-ipv4-basic-2000]quit
  
```

# 配置流分类规则，使用基本 ACL 2000 进行流分类。

```

[H3C]traffic classifier 1
[H3C-classifier-1]if-match acl 2000
  
```

# 配置流行为，定义流镜像到 GigabitEthernet 1/0/2 的动作。

```

[H3C]traffic behavior 1
[H3C-behavior-1]mirror-to interface GigabitEthernet 1/0/2
  
```

# 配置 QoS 策略 1，为流分类 1 指定流行为 1。

```

[H3C]qos policy 1
[Switch-policy-1]classifier 1 behavior 1
  
```

# 将 QoS 策略应用到端口 GigabitEthernet 1/0/1 上。

```

[H3C-GigabitEthernet1/0/1]qos apply policy 1 inbound
  
```

## 31.6 镜像显示及注意事项

### 镜像显示及注意事项



紫光集团 H3C  
核心企业 数字化解决方案领导者

- 镜像显示
 

```
[sysname] display mirroring-group { groupid | all | local |
remote-destination | remote-source }
```
- 镜像配置注意事项
  - 一个镜像组可以配置多个源端口，但只能配置一个目的端口；
  - 远程镜像VLAN必须为已经创建的静态VLAN；
  - 远程镜像VLAN不要做其它用途，仅用于远程镜像；
  - 在对源端口双向的数据流进行镜像情况下，远程镜像VLAN最好关闭MAC地址学习

www.h3c.com

镜像配置完成之后，为了方面检查配置的完整性，设备提供相关命令用于检查配置镜像组的配置。**display mirroring-group** 命令用来显示端口镜像组的信息。可以指定镜像组显示，也可以分类别显示或全部显示。如果显示所有镜像组则按照镜像组号的大小顺序进行。该命令可在任意视图下执行。

**[H3C]display mirroring-group { groupid | all | local | remote-destination | remote-source }**

其中主要参数说明如下：

- **group-id**: 显示指定镜像组的组号。
- **all**: 显示所有镜像组。
- **local**: 显示本地镜像组。
- **remote-destination**: 远程目的镜像组。
- **remote-source**: 远程源镜像组。

由于同一设备支持多种镜像技术和多个镜像组，在配置镜像是我们需要注意如下注意事项：

- 一个镜像组可以配置多个源端口，但只能配置一个目的端口；
- 远程镜像 VLAN 必须为已经创建的静态 VLAN；
- 远程镜像 VLAN 不要做其它用途，仅用于远程镜像；

- 在对源端口双向的数据流进行镜像情况下,远程镜像 VLAN 最好关闭 MAC 地址学习;

## 31.7 本章总结

### 本章总结

- 镜像可分为端口镜像、远程端口镜像和流镜像
- 端口镜像通过镜像组方式实现，流镜像通过QoS来实现
- 镜像VLAN 和镜像目的端口不要做其它用途
- 镜像目的端口不要开启STP相关协议

www.h3c.com

## 31.8 习题和解答

### 31.8.1 习题

1. 镜像源端口最多可以有\_\_\_\_\_个。  
A 1  
B 2  
C 4  
D 没有限制
2. 聚合端口也可以作为镜像源端口。  
T. 正确      F. 错误
3. 端口镜像的方向有\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。
4. 远程镜像中，Probe VLAN 内应该禁止 MAC 地址学习。  
T. 正确      F. 错误

### 31.8.2 习题答案

1. D
2. T
3. inbound, outbound, both
4. T

## 第32章 NTP

网络设备记录的日志信息是网络状态监控和网络故障定位的重要依据，保证其具备严格的先后顺序是非常必要的。频繁配置设备时间耗会费管理员的大量精力，且不论配置得如何频繁如何细心，都无法确保各设备的时钟分秒不差。

要满足这种网络设备保持时间一致性的需求，需要利用 NTP 协议在各设备之间自动同步时间，使所有设备的时间都达到一致。NTP 在其他诸多方面也都有着广泛应用。

### 32.1 本章目标

#### 课程目标

学习完本课程，您应该能够：

- 了解NTP的基本功能和工作原理
- 熟悉NTP的各种工作模式
- 熟练掌握NTP的配置方法



## 32.2 NTP简介

### 32.2.1 NTP 的作用

### NTP简介

紫光集团 H3C  
核心企业 数字化解决方案领导者

- **NTP用于对网络上的设备进行时钟同步**
- **典型应用**
  - 日志信息、调试信息记录
  - 计费系统
  - 多个系统协同处理
  - 备份服务器和客户端之间进行增量备份
- **经历了五个版本**
  - V0 (RFC958) 、 V1 (RFC1059)
  - V2 (RFC1119) 、 V3 (RFC1305)
  - V4(RFC 5905)

www.h3c.com

对于网络中的各网络设备来说，如果依靠管理员手工输入命令来修改系统时钟，不但工作量巨大，而且也不能保证时钟的精确性。相反，通过 **NTP** 自动同步，可以很快将网络设备的时钟同步，同时也能保证很高的精度。

**NTP**（**Network Time Protocol**，网络时间协议）由 **RFC 1305** 规定，用来在分布式时间服务器和客户端之间进行时间同步的协议。**NTP** 基于 **UDP** 进行传输，使用的 **UDP** 端口号为 123。

使用 **NTP** 的目的是对网络内所有具有时钟的设备进行时钟同步，使网络内所有设备的时钟保持一致，从而使设备能够提供基于统一时间的多种应用。

运行 **NTP** 的本地系统，既可以接受来自其他时钟源的同步，又可以作为时钟源同步其它的时钟，并且可以和其它设备互相同步。

**NTP** 主要应用于需要网络中所有设备时钟保持一致的场合，比如：

- 在网络管理中，对于从不同设备采集来的日志信息、调试信息进行分析的时候，需要以时间作为参照依据。
- 计费系统要求所有设备的时钟保持一致。
- 定时重启网络中的所有设备，要求所有设备的时钟保持一致。

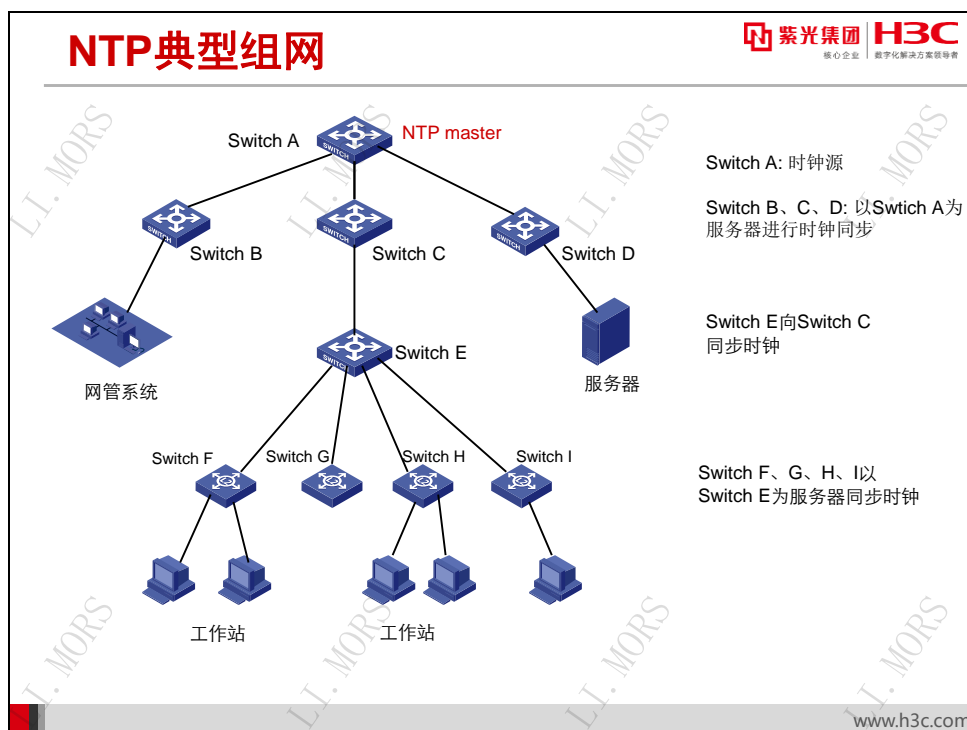
- 多个系统协同处理同一个比较复杂的事件时，为保证正确的执行顺序，多个系统必须参考同一时钟。
- 在备份服务器和客户端之间进行增量备份时，要求备份服务器和所有客户端之间的时钟同步。

NTP 协议最早由美国 Delaware 大学的 Mills 教授设计提出，到目前为止经历了五个版本：v0（RFC958）、v1（RFC1059）、v2（RFC1119）、v3（RFC1305）、v4（RFC5905）。

NTP v3 相对前 3 个版本，并没有对协议做重大改进，也没有撤消以前的版本及已经商用的部署，而是在继承原有结构的情况下，提出了在高速的 Gigabit 网络中如何部署更合理、更稳定、更精确的商业模型。比如完善了校验字段，降低了丢包、重传对同步的影响，修改本地时间算法来保证稳定与精确等等。

最新的 NTP 版本是 NTPv4，它继承自 RFC 1305 所描述的 NTP v3。[网络时间同步技术](#)将向更高精度、更强的兼容性和多平台的适应性方向发展。

### 32.2.2 NTP 基本架构



在实际网络中 NTP 采用 Client/Server 结构运行，但服务器和客户端的概念是相对的，提供时间标准的设备称为时间服务器，接收时间服务的设备称为客户端。作为客户端的 NTP 设备同时还可以作为其它 NTP 设备的服务器。因此相互进行时钟同步的各网络设备最终组成树状网络结构。从时钟服务器的根到各分支逐级进行时钟同步。

NTP 采用分层（Stratum）的方法来定义时钟的准确性。NTP 设备的时钟层数越大，说明时钟精度越低。实际网络中，通常将从权威时钟（如原子时钟）获得时钟同步的 NTP 服务器

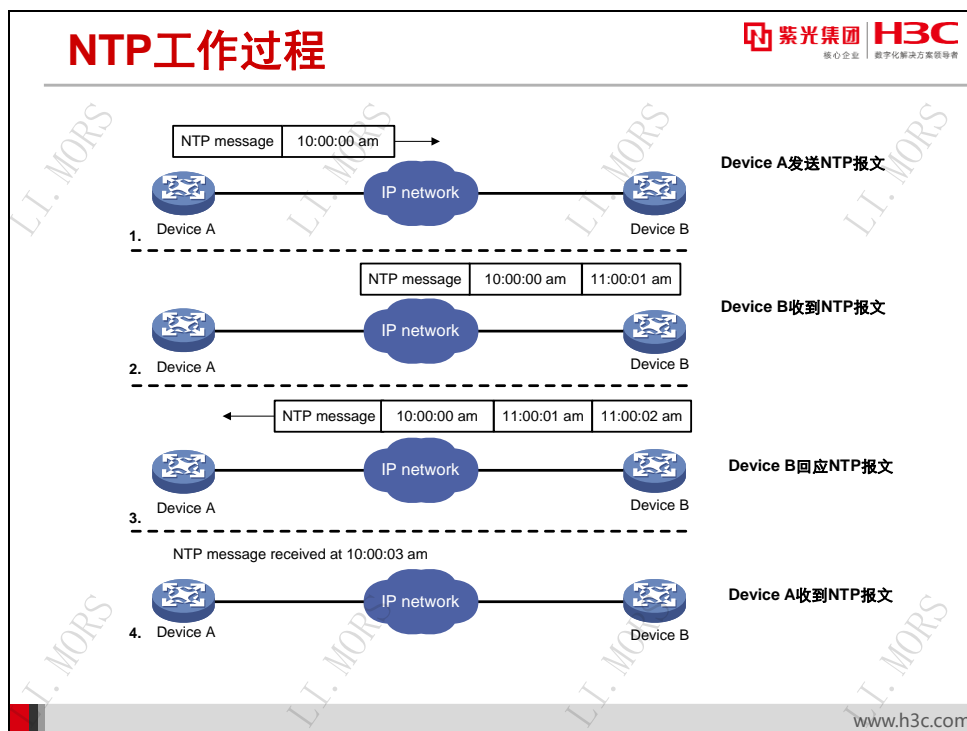


的层数设置为 1，并将其作为主参考时钟源，用于同步网络中其它设备的时钟。网络中的设备与主参考时钟源的 NTP 距离，即 NTP 同步链上 NTP 服务器的数目，决定了设备上时钟的层数。

配置本地时钟作为参考时钟或者已经与上一级时钟源同步后，本地设备也可以作为时钟源同步网络中的其他设备。

## 32.3 NTP 原理

### 32.3.1 NTP 工作过程



NTP 的基本工作原理如图所示。Device A 和 Device B 通过网络相连，它们都有自己独立的系统时钟，需要通过 NTP 实现各自系统时钟的自动同步。为了便于理解，作如下假设：

- 在 Device A 和 Device B 的系统时钟同步之前，Device A 的时钟设定为 10:00:00am，Device B 的时钟设定为 11:00:00am。
- Device B 作为 NTP 时间服务器，即 Device A 将使自己的时钟与 Device B 的时钟同步。
- NTP 报文在 Device A 和 Device B 之间单向传输所需要的时间为 1 秒。

在上述假设情况下，系统时钟同步的一个完整工作过程描述如下：

**第1步：**Device A 首先发送一个 NTP 报文给 Device B，该报文包含它离开 Device A 时的时间戳，该时间戳为 10:00:00am (T1)。

**第2步：**当此 NTP 报文到达 Device B 时，Device B 加上自己的时间戳，该时间戳为 11:00:01am (T2)。

**第3步：**Device B 正确处理此报文并返回响应报文，此 NTP 响应报文离开 Device B 时，Device B 再加上自己的时间戳，该时间戳为 11:00:02am (T3)。

**第4步：**当 Device A 接收到该响应报文时，Device A 的本地时间为 10:00:03am (T4)。

**第5步：**Device A 根据上述 4 个时间计算出如下两个重要参数并据此设定自己的时钟从而达到时间同步。

NTP 报文的往返时延  $\text{Delay} = (T4 - T1) - (T3 - T2) = 2$  秒。

Device A 相对 Device B 的时间差  $\text{offset} = ((T2 - T1) + (T3 - T4)) / 2 = 1$  小时。

以上内容只是对 NTP 工作原理的一个粗略描述，实际计算过程相对要复杂得多，具体计算过程以及算法请参阅 RFC 1305。

### 32.3.2 NTP 报文结构



NTP 有两种不同类型的报文，一种是时钟同步报文，另一种是控制报文。控制报文仅用于需要网络管理的场合，它对于时钟同步功能来说并不是必需的，因此本章不做详细介绍。


NTP 时钟同步报文基于 UDP 传送，其格式如图所示。

- **LI (Leap Indicator):** 长度为 2 比特，值为 11 时表示告警状态，时钟未被同步。为其它值时 NTP 本身不做处理。
- **VN (Version Number):** 长度为 3 比特，表示 NTP 的版本号，目前的最新版本为 3。
- **Mode:** 长度为 3 比特，表示 NTP 的工作模式。其中 0 未定义，1 表示主动对等体模式，2 表示被动对等体模式，3 表示客户端模式，4 表示服务器模式，5 表示广播模式或组播模式，6 表示此报文为 NTP 控制报文，7 预留给内部使用。

- **Stratum:** 长度为 8 比特，表示系统时钟的层数，取值范围为 1~16，它定义了时钟的准确度。层数为 1 的时钟准确度最高，准确度从 1 到 16 依次递减，层数为 16 的时钟处于未同步状态，不能作为参考时钟。
- **Poll:** 轮询时间，即两个连续 NTP 报文之间的时间间隔。
- **Precision:** 系统时钟的精度。
- **Root Delay:** 本地到主参考时钟源的往返时间。
- **Root Dispersion:** 系统时钟相对于主参考时钟的最大误差。
- **Reference Identifier:** 参考时钟源的标识。
- **Reference Timestamp:** 系统时钟最后一次被设定或更新的时间。
- **Originate Timestamp:** NTP 请求报文离开发送端时发送端的本地时间。
- **Receive Timestamp:** NTP 请求报文到达接收端时接收端的本地时间。
- **Transmit Timestamp:** 应答报文离开应答者时应答者的本地时间。
- **Authenticator:** 验证信息。

### 32.3.3 NTP 工作模式

NTP工作模式


  
紫光集团 H3C  
核心企业 数字化转型领导者

- 设备可以采用多种NTP工作模式进行时间同步
  - 客户端/服务器模式
  - 对等体模式
  - 广播模式
  - 组播模式
- 选择适合的模式
  - 服务器和对等体模式：需指定地址，设备从指定的服务器或对等体获得时钟同步，增加了时钟的可靠性
  - 广播和组播模式：在不能确定服务器或对等体IP地址、网络中需要同步的设备很多等情况下

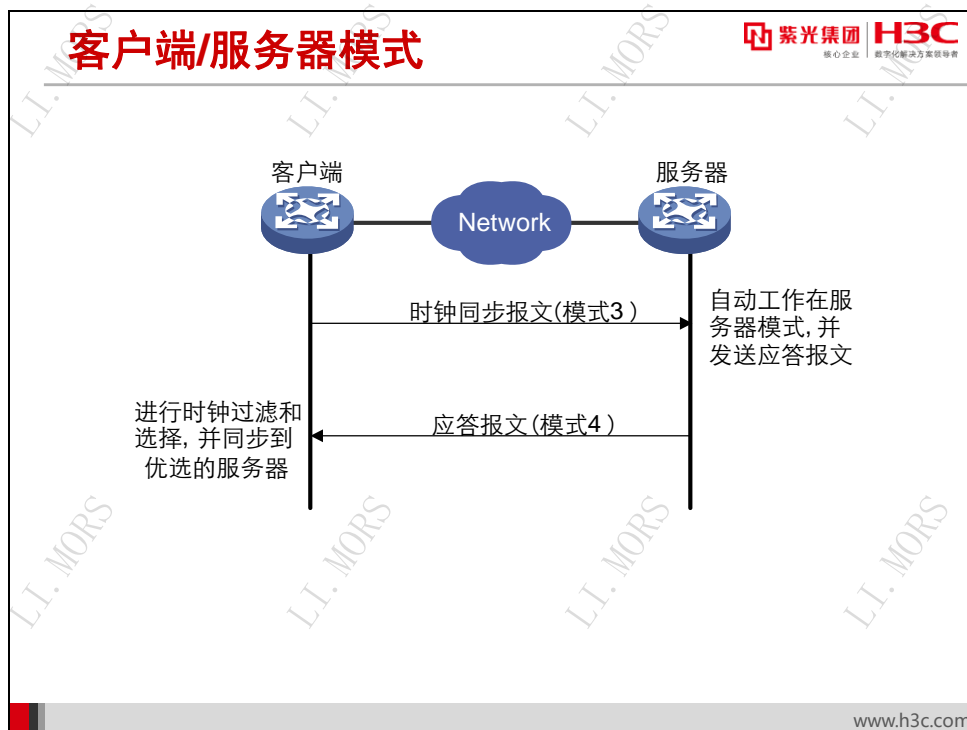
[www.h3c.com](http://www.h3c.com)

设备可以采用多种 NTP 工作模式进行时间同步：

- 客户端/服务器模式
- 对等体模式

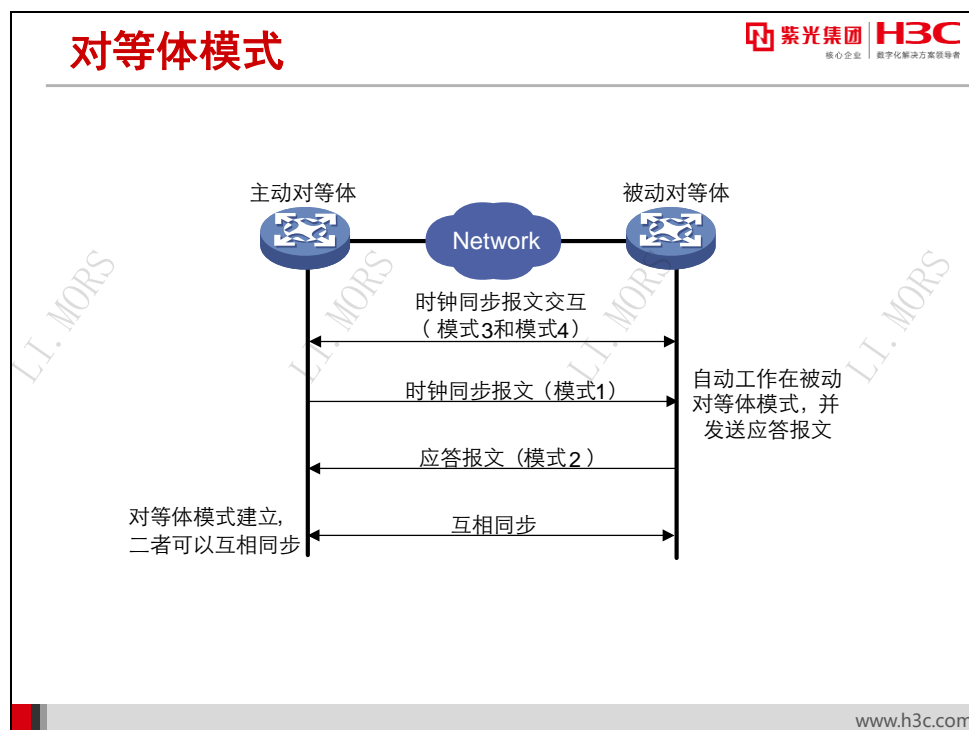
- 广播模式
- 组播模式

用户可以根据需要选择合适的工作模式。在不能确定服务器或对等体 IP 地址、网络中需要同步的设备很多等情况下，可以通过广播或组播模式实现时钟同步；服务器和对等体模式中，设备从指定的服务器或对等体获得时钟同步，增加了时钟的可靠性。

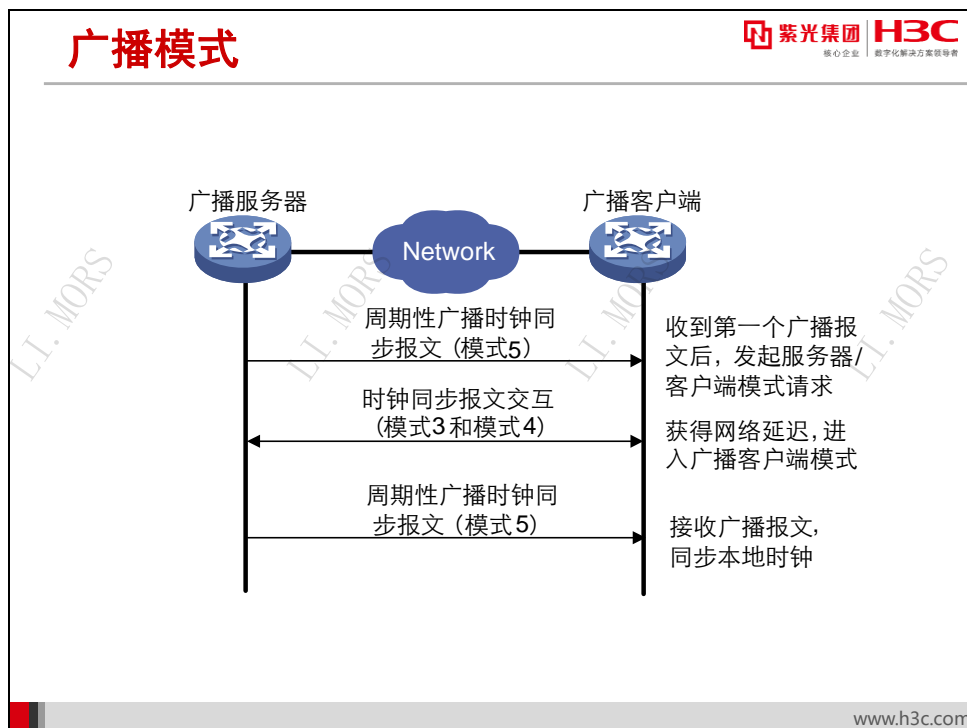


在客户端/服务器模式中，客户端向服务器发送时钟同步报文，报文中的 **Mode** 字段设置为 3（客户模式）。服务器端收到报文后会自动工作在服务器模式，并发送应答报文，报文中的 **Mode** 字段设置为 4（服务器模式）。客户端收到应答报文后，进行时钟过滤和选择，并同步到优选的服务器。

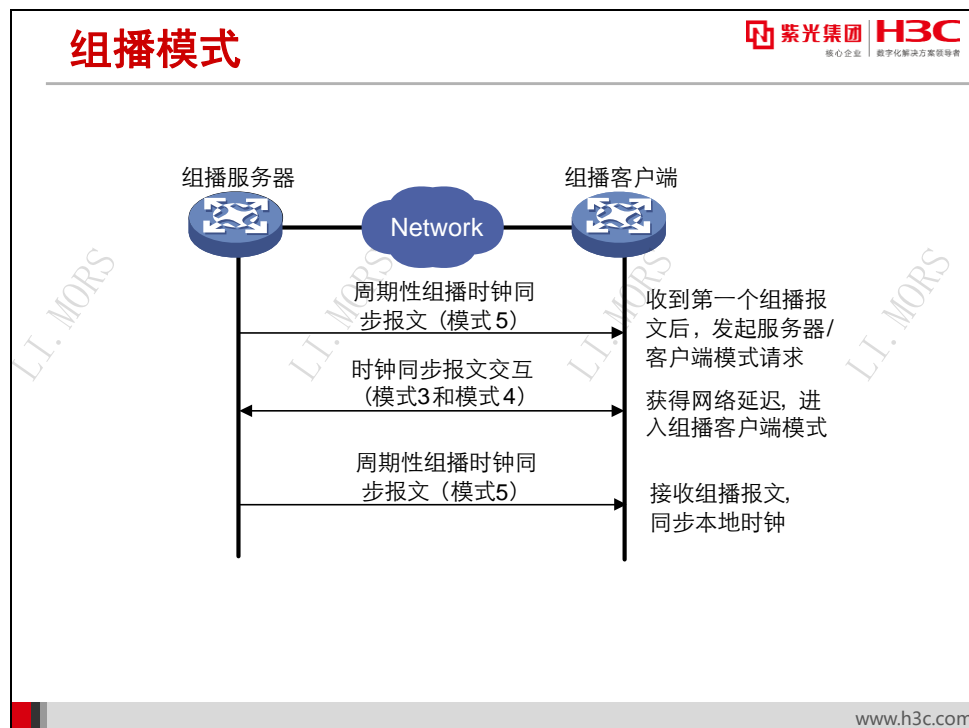
在该模式下，客户端能同步到服务器，而服务器无法同步到客户端。



在对等体模式中，主动对等体和被动对等体之间首先交互 **Mode** 字段为 3（客户端模式）和 4（服务器模式）的 NTP 报文。之后，主动对等体向被动对等体发送时钟同步报文，报文中的 **Mode** 字段设置为 1（主动对等体），被动对等体收到报文后自动工作在被动对等体模式，并发送应答报文，报文中的 **Mode** 字段设置为 2（被动对等体）。经过报文的交互，对等体模式建立起来。主动对等体和被动对等体可以互相同步。如果双方的时钟都已经同步，则以层数小的时钟为准。



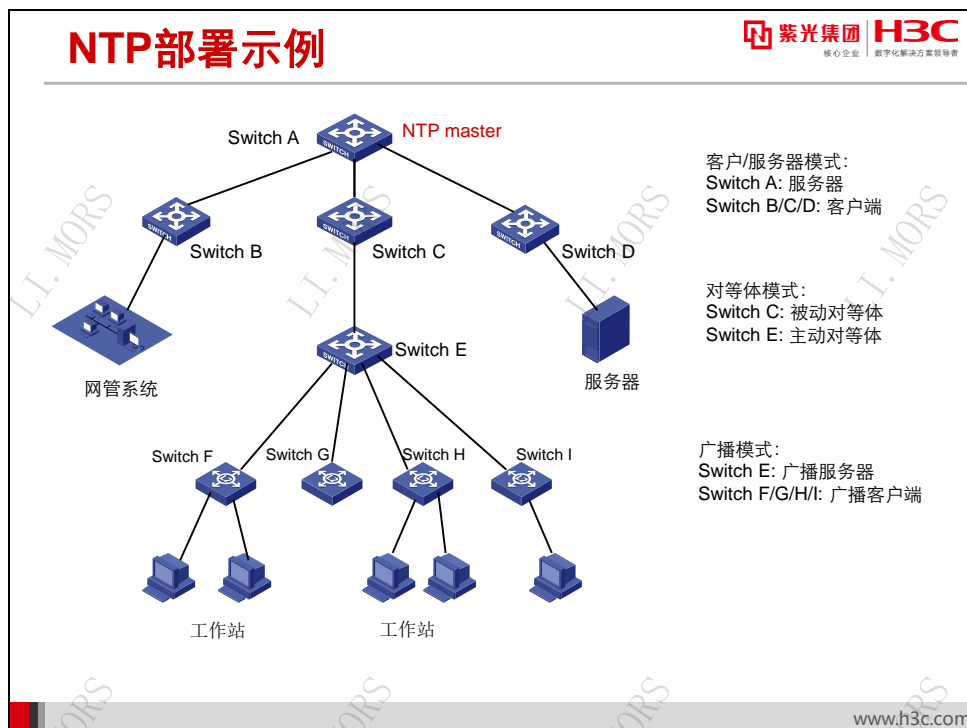
在广播模式中，服务器端周期性地向广播地址 **255.255.255.255** 发送时钟同步报文，报文中的 **Mode** 字段设置为 **5**（广播模式）。客户端侦听来自服务器的广播报文。当客户端接收到第一个广播报文后，客户端与服务器交互 **Mode** 字段为 **3**（客户端模式）和 **4**（服务器模式）的 NTP 报文，以获得客户端与服务器间的网络延迟。之后，客户端就进入广播客户端模式，继续侦听广播报文的到来，根据到来的广播报文对系统时钟进行同步。



在组播模式中，服务器端周期性地向用户配置的组播地址（若用户没有配置组播地址，则使用默认的 NTP 组播地址 224.0.1.1）发送时钟同步报文，报文中的 **Mode** 字段设置为 5（组播模式）。客户端侦听来自服务器的组播报文。当客户端接收到第一个组播报文后，客户端与服务器交互 **Mode** 字段为 3（客户模式）和 4（服务器模式）的 NTP 报文，以获得客户端与服务器间的网络延迟。之后，客户端就进入组播客户端模式，继续侦听组播报文的到来，根据到来的组播报文对系统时钟进行同步。



## 32.3.4 NTP 部署示例



在实际 NTP 部署中，可以根据网络拓扑选择恰当的模式来配置 NTP。通常可以选择核心设备作为最高级的 NTP 时钟源，部分单点连接的网络设备之间采用对等体模式，局部星形拓扑连接的设备采用广播模式或客户端/服务器模式。


例如，在图示网络拓扑中推荐采用如下方案：

- Switch A 为 NTP master 即时钟源，Switch B、Switch C、Switch D 作为客户端，指向 Switch A 的地址与其同步。它们工作在客户端/服务器模式。
- Switch C 与 Switch E 工作在对等体模式，可互为备份。
- Switch E 是 Switch F、Switch G、Switch H、Switch I 的连接中心，选择广播或组播模式则相对简单。以 Switch E 作为广播服务器为其它设备提供时钟同步源，其它设备作为客户端同步 Switch E 的时钟。

## 32.3.5 NTP 验证

## NTP验证

- 针对安全性要求较高的网络
- 客户端和服务端配置密码验证
- 客户端只与通过验证的设备进行同步
- 服务器端的配置与客户端的配置应保持一致



核心企业 | 数字化解决方案领导者

www.h3c.com

在一些对安全性要求较高的网络中，运行 NTP 协议时需要启用验证功能。通过客户端和服务端端的密码验证，可以保证客户端只与通过验证的设备进行同步，提高网络安全性。

NTP 验证功能可以分为客户端的 NTP 验证和服务端端的 NTP 验证两个部分。在应用 NTP 验证功能时，应注意以下原则：

- 对于所有同步模式，如果使能了 NTP 验证功能，应同时配置验证密钥并将密钥设为可信密钥。否则，无法正常启用 NTP 验证功能。
- 对于客户端/服务器模式和对等体模式，还应在客户端（对等体模式中的主动对等体）将指定密钥与对应的 NTP 服务器（对等体模式的被动对等体）关联；对于广播服务器模式和组播服务器模式，应在广播服务器或组播服务器上将指定密钥与对应的 NTP 服务器关联。否则，无法正常启用 NTP 验证功能。
- 对于客户端/服务器同步模式，如果客户端没有成功启用 NTP 验证功能，不论服务器端是否使能 NTP 验证，客户端均可以与服务器端同步；如果客户端上成功启用了 NTP 验证功能，则客户端只会同步到提供可信密钥的服务器，如果服务器提供的密钥不是可信的密钥，那么客户端不会与其同步。
- 对于所有同步模式，服务器端的配置与客户端的配置应保持一致。

## 32.4 NTP基本配置

### 32.4.1 NTP 配置命令

配置NTP客户端/服务器模式

**紫光集团**  
核心企业 数字化解决方案领导者

- 配置NTP服务器参考本地时钟

```
[H3C]ntp-service enable
[H3C]ntp-service refclock-master [ ip-address ]
[ stratum ]
```

- 配置NTP客户端

```
[H3C]ntp-service enable
[H3C]ntp-service unicast-server { server-name |
ip-address } [ vpn-instance vpn-instance-name ]
[ authentication-keyid keyid | priority | source
interface-type interface-number | version number ]
*
```

www.h3c.com

实际网络中，通常将从权威时钟（如原子时钟）获得时钟同步的 NTP 服务器的层数设置为 1，并将其作为主参考时钟源同步网络中其他设备的时钟。网络中的设备与主参考时钟源的 NTP 距离，即 NTP 同步链上 NTP 服务器的数目，决定了设备上时钟的层数。在 NTP 的网络中必须存在至少一个 NTP 服务器或者说 NTP 时钟源。

当网络没有标准的时钟源作为 NTP 服务器时，可以选择具有本地实时时钟的网络设备作为时钟服务器或时钟源。在被选定的网络设备上配置本地时钟作为参考源后，即可作为 NTP 服务器为 NTP 客户端提供时钟同步。但在大型网络中须谨慎配置设备参考本地时钟，以免导致网络中设备的时钟错误。其配置设备参考本地时钟的命令为：

```
[H3C]ntp-service refclock-master [ ip-address ] [ stratum ]
```

其中参数 *ip-address* 只能配置为 127.127.1.u，u 的取值范围为 0~3，表示 NTP 的进程 ID。参数 *stratum* 为本地时钟所处的层数，取值范围为 1~15，缺省值为 8。

网络中有了可以同步的时钟源之后，即可在 NTP 客户端上配置如下命令进行时钟同步：

```
[H3C]ntp-service unicast-server { server-name | ip-address } [ authentication-keyid
keyid | priority | source interface-type interface-number | version number ] *
```


其中主要参数说明如下：

- *server-name*: NTP 服务器的主机名。

- **ip-address**: NTP 服务器的 IP 地址。
- **vpn-instance vpn-instance-name**: 指定 NTP 服务器所属的 VPN，为 1~31 个字符的字符串，区分大小写。如果未指定本参数，则表示 NTP 服务器位于公网中。
- **authentication-keyid**: 配置 NTP 验证中向对端发送报文时使用的 key ID。
- **priority**: 指定此参数可以在同等条件下，优先同步此服务器的时钟。
- **source interface-type interface-number**: 指定发送 NTP 报文的源接口。
- **version**: 指定 NTP 运行的版本号，缺省为 4。

同一个 NTP 客户端可以重复配置此命令指定多个 NTP 服务器，客户端依据时钟优选来选择最优的时钟源。

## 配置 NTP 对等体模式



紫光集团 H3C  
核心企业 | 数字化解决方案领导者

- 配置主动对等体

```
[H3C]ntp-service enable
[H3C]ntp-service unicast-peer { peer-name | ip-address } [ vpn-instance vpn-instance-name ]
[ authentication-keyid keyid | priority | source interface-type interface-number | version number ] *
```

- 配置被动对等体

```
[H3C]ntp-service enable
```

www.h3c.com

当设备采用对等体模式进行 NTP 时钟同步时，需要在主动对等体上指定被动对等体。被动对等体上需要执行 `ntp-service enable` 命令来开启 NTP 服务，否则被动对等体不会处理来自主动对等体的 NTP 报文。同时保证主动对等体和被动对等体的时钟至少要有有一个处于同步状态，否则它们的时间都将无法同步。

配置主动对等体的 Peer 的具体命令为：


```
[H3C]ntp-service unicast-peer { peer-name | ip-address } [ vpn-instance vpn-instance-name ] [ authentication-keyid keyid | priority | source interface-type interface-number | version number ] *
```

其中参数 *ip-address* 为被动对等体的 IP 地址。

对于对等体模式中的被动对等体没有固定的配置，要么选择配置参考本地时钟，要么选择 NTP 的任何一种工作模式并正确同步即可。

多次执行 **ntp-service unicast-peer** 命令配置多个被动对等体可以提供时钟同步的备份。

## 配置NTP广播模式

 紫光集团 **H3C**  
核心企业 数字化解决方案领导者

- **配置NTP广播模式**
  - 配置NTP广播服务器

```
[H3C]ntp-service enable
[H3C-Vlan-interface]ntp-service broadcast-server
[ authentication-keyid keyid | version number ]
```

  - 配置NTP广播客户端

```
[H3C]ntp-service enable
H3C-Vlan-interface]ntp-service broadcast-client
```

www.h3c.com

在 NTP 广播模式下，需要配置一个 NTP 广播服务器。NTP 广播服务器周期性地向外发送本地广播的 NTP 报文，工作在 NTP 广播客户端模式的设备将回应这个报文，从而开始时钟同步。

NTP 广播服务器的使能和指定 NTP 广播报文的发送接口通过如下配置命令完成，并且只有当服务器自己的时钟已经同步后，才能为广播客户端提供时钟同步服务。具体配置命令为：

```
[H3C]ntp-service enable
```

```
[H3C-Vlan-interface]ntp-service broadcast-server [ authentication-keyid keyid |
version number ]
```

NTP 广播客户端的使能和指定广播报文的接收接口通过执行如下命令即可：

```
[H3C]ntp-service enable
```


```
[H3C-Vlan-interface]ntp-service broadcast-client
```

## 配置NTP组播模式

- 配置NTP组播模式
  - 配置NTP组播服务器
 

```
[H3C]ntp-service enable
[H3C-Vlan-interface]ntp-service multicast-server
[ ip-address ] [ authentication-keyid keyid | ttl ttl-
number | version number ]
```
  - 配置NTP组播客户端
 

```
[H3C]ntp-service enable
[H3C-Vlan-interface]ntp-service multicast-client
[ ip-address ]
```



紫光集团 H3C  
核心企业 | 数字化解决方案领导者

www.h3c.com

在 NTP 组播模式下，NTP 组播服务器以组播形式周期性地发送时钟同步报文，工作在 NTP 组播客户端模式的设备收到 NTP 组播报文后响应此报文，从而开始时钟同步。

NTP 组播服务器只有当其时钟同步后，才能去同步组播客户端。配置组播服务器的具体命令为：

**[H3C]ntp-service enable**

**[H3C-Vlan-interface]ntp-service multicast-server [ ip-address ] [ authentication-keyid keyid | ttl ttl-number | version number ]**

其中参数 *ip-address* 为组播 IP 地址，取值范围为 224.0.1.0~224.0.1.255，缺省值为 224.0.1.1。参数 *ttl-number* 为组播报文的生存周期，缺省参数为 16，可取值范围为 1-255。

NTP 组播客户端的使能和指定组播报文的接收接口通过执行如下命令即可：

**[H3C]ntp-service enable**

**[H3C-Vlan-interface] ntp-service multicast-client [ ip-address ]**

其中参数 *ip-address* 为组播 IP 地址，取值范围为 224.0.1.0~224.0.1.255，缺省值为 224.0.1.1。为了正确的接收 NTP 组播服务器的报文，客户端和服务器必须配置相同的组播 IP。

## NTP的常用维护命令

紫光集团 H3C  
核心企业 数字化转型领导者

- 显示NTP会话及详细信息

```
[H3C]display ntp-service sessions [ verbose ]
```

- 显示NTP服务的状态信息

```
[H3C]display ntp-service status
```

www.h3c.com

在完成 NTP 各种工作模式的配置之后，可以采用设备提供的维护命令检查 NTP 服务的工作状态。常用的维护命令有显示 NTP 会话和 NTP 服务的状态。如执行 **display ntp-service session** 命令显示的 NTP 会话信息如下：

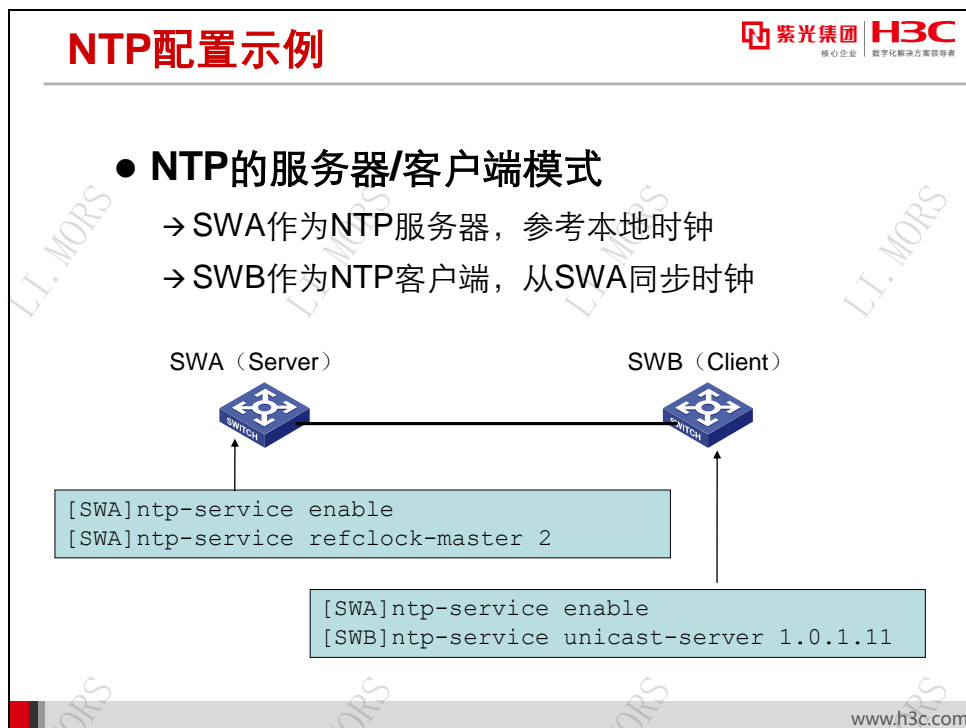
```
[H3C]display ntp-service sessions
      source          reference      stra reach poll now offset delay disper
*****
[12345]1.1.1.1        127.127.1.0      2      1 64 7 -9.600 1.8920 0.0610
Notes: 1 source(master), 2 source(peer), 3 selected, 4 candidate, 5
configured.
```

Total sessions: 1

要先检查设备是否已经处于同步状态可以使用 **display ntp-service status** 命令。

```
[H3C]display ntp-service status
Clock status: synchronized
Clock stratum: 3
System peer: 1.1.1.1
Local mode: client
Reference clock ID: 1.1.1.1
Leap indicator: 00
Clock jitter: 0.004456 s
Stability: 0.000 pps
Clock precision: 2^-15
Root delay: 1.89209 ms
Root dispersion: 24.84131 ms
Reference time: d82ec5c3.2d451c6c Sun, Dec 7 2014 12:25:39.176
```

## 32.4.2 NTP 配置示例



NTP 的配置相对简单，在基本连通性配置完成的情况下，只需要少量命令即可让 NTP 正常工作起来，达到网络设备的时间全网同步。此处以 NTP 的服务器/客户端模式为例，介绍两个网络设备进行时钟同步的配置。

首先需要在网络中选择其中一台设备作为时钟服务器。在如图组网中，选择 SWA 作为时钟服务器。注意由于作为时钟服务器的网络设备必须自己的时钟已经同步或者存在本地时钟，因此 SWA 必须是具有本地时钟的网络设备。

在 SWA 上使能 NTP 服务且配置参考本地时钟，并制定时钟层数为 2。

```
[SWA]ntp-service enable
[SWA]ntp-service refclock-master 2
```

然后在 SWB 上使能 NTP 服务且配置 NTP 服务器，并制定服务器地址为 SWA 的接口 IP。

```
[SWA]ntp-service enable
[SWB]ntp-service unicast-server 1.0.1.11
```

完成上述配置之后，即可使用 NTP 维护显示命令显示 SWB 的时钟同步状态。

```
[SWB]display ntp-service status
Clock status: synchronized
Clock stratum: 3
System peer: 1.0.1.11
Local mode: client
Reference clock ID: 1.0.1.11
Leap indicator: 00
Clock jitter: 0.004456 s
```



```
Stability: 0.000 pps
Clock precision: 2^-15
Root delay: 1.89209 ms
Root dispersion: 24.84131 ms
Reference time: d82ec5c3.2d451c6c Sun, Dec 7 2014 12:25:39.176
```

从中可以确认 **SWB** 已经同步，当前时钟层数为 **3**，在 **SWA** 的时钟层数上增加了 **1**。还有更详细的时钟参考源，时钟精度等参数。

## 32.5 本章总结

### 本章总结

- NTP的网络应用和发展
- NTP时钟同步工作原理
- NTP的工作模式及其正确选择
- NTP的配置和维护

www.h3c.com

## 32.6 习题和解答

### 32.6.1 习题

1. NTP 的当前常用版本是 ( )
  - A. v0
  - B. v1
  - C. v2
  - D. v3
2. NTP 的工作模式有哪些? ( )
  - A. 客户端/服务器模式
  - B. 广播模式
  - C. 组播模式
  - D. 对等体模式
3. NTP 的时钟层数为\_\_\_\_\_表示时钟未同步。
4. 在对等体模式下, 主动对等体可以同步被动对等体, 被动对等体也可以同步主动对等体。
  - T. 正确
  - F. 错误
5. 请简述 NTP 客户端/服务器模式下 NTP 时钟同步的过程。

### 32.6.2 习题答案

1. D
2. ABCD
3. 16
4. T
5. 略