

- GetRequest、GetNextRequest 和 GetBulkRequest PDU 都是管理实体向代理发送的，以请求位于该代理所在的被管设备中的一个或多个 MIB 对象值。其值被请求的 MIB 对象的对象标识符定义在该 PDU 的变量绑定部分。GetRequest、GetNextRequest 和 GetBulkRequest 的差异在于它们的数据请求粒度。GetRequest 能够请求 MIB 值的任意集合；多个 GetNextRequest 能用于顺序地读取 MIB 对象的序列或表；GetBulkRequest 允许读取大块数据，能够避免因发送多个 GetRequest 或 GetNextRequest 报文可能导致的额外开销。在所有这三种情况下，代理用包括该对象标识符和它们相关值的 Response PDU 进行响应。
- SetRequest PDU 是管理实体用来设置在被管设备中的一个或多个 MIB 对象值的。代理用带有“无错 (noError)”差错状态的 Response PDU 来回答，证实该值的确已经被设置。
- InformRequest PDU 是一个管理实体用来通知另一个管理实体的，其中 MIB 信息是接收实体不能访问的。该接收实体用带有“无错 (noError)”差错状态的 Response PDU 来确认收到了 InformRequest PDU。
- SNMPv2 PDU 的最后一种类型是陷阱报文。陷阱报文是异步产生的，即它们不是为了响应接收到的请求而产生的，而是为了响应管理实体所需要的通知事件而产生的。RFC 3418 定义了周知的陷阱类型，其中包括设备进行的冷启动或热启动，链路的从故障状态变为工作状态或相反，找不到相邻设备，或鉴别失效事件。接收到的陷阱请求不要求从管理实体得到响应。

给出了 SNMPv2 请求响应性质后，注意到尽管 SNMP PDU 能够通过许多不同的运输协议传输，但 SNMP PDU 通常还是在 UDP 数据报的负载中传输的。事实上，RFC 3417 认为 UDP 是“首选的运输映射”。由于 UDP 是一种不可靠的运输协议，因而不能确保一个请求或它的响应将能够被它的目的地接收到。管理实体用该 PDU 的“请求 ID”字段为它向代理发送的请求编号；代理的响应从接收到的请求中获取该“请求 ID”。因此，该“请求 ID”字段能被管理实体用来监测丢失的请求或回答。如果在一定量时间后还没有收到对应的响应，由管理实体来决定是否重传一个请求。特别是，SNMP 标准没有强制任何特殊的重传过程，哪怕首先进行了重传。它只是要求管理实体“需要根据重传的频率和周期可靠地动作”。这当然让人们想知道一个“可靠的”协议行为应当如何做！

9.3.4 安全性和管理

SNMPv3 的设计者们说过“SNMPv3 能被看成是具有附加的安全性和管理能力的 SNMPv2” [RFC 3410]。SNMPv3 比起 SNMPv2 来无疑有一些变化，但是其他地方的变化都没有管理和安全性领域的变化那么明显。在 SNMPv3 中安全性的重要性尤为突出，因为缺乏安全性导致 SNMP 只能用于监视而不能用于控制（例如，在 SNMPv1 中很少使用 SetRequest）。

SNMP 经历了 3 个版本后已经成熟，它的功能有了增加，但不幸的是，与 SNMP 相关的标准文档的数量增加了。这能够由下述事实来印证：现在甚至有了“描述用于描述 SNMP 管理框架的体系结构的 RFC” [RFC 3411]！虽然描述一个框架的体系结构这个概念也许使人有些不太好理解，但 RFC 3411 的目的是值得肯定的，它引入了一种共同语言，用于描述功能和由 SNMPv3 代理或管理实体所采取的动作。SNMPv3 实体的体系结构是简单明了的，浏览该体系结构将有助于我们进一步理解 SNMP。