tr-069-1-6-1.pdf

1、

2、

3、程序和要求

本节连同本节中引用的附录定义了CPE广域网管理协议的规范性要求。
本节还引用了构成CPE广域网管理协议一部分的多个标准和其他规范。除非另有规定，CPE和ACS必须遵守这些被引用规范的要求。

3.1 ACS发现

CPE广域网管理协议定义了CPE可以用来发现其关联ACS地址的以下机制：

1. CPE可以针对每个CWMP端点本地配置ACS的URL。例如，这可以通过局域网侧的CPE自动配置协议来完成。如果需要，CPE将使用DNS从URL中的主机名部分解析ACS的IP地址。
2. 作为IP层自动配置的一部分，接入网络上的DHCP服务器可以配置为包含ACS URL作为DHCP选项[17] / [25] / [38]。如果需要，CPE将使用DNS从URL中的主机名部分解析ACS的IP地址。在这种情况下，还可以使用额外的DHCP选项来设置：
    - 配置代码（ProvisioningCode），该代码可用于向ACS指示主要服务提供商及其他配置信息。
    - CWMP重试最小等待间隔（CWMPRetryMinimumWaitInterval），用于设置CWMP会话重试最小等待间隔的初始值，如第3.2.1.1节所述。
    - CWMP重试间隔乘数（CWMPRetryIntervalMultiplier），用于设置CWMP会话重试间隔乘数的初始值，如第3.2.1.1节所述。

CPE通过在DHCPv4供应商类标识符（选项60）、DHCPv4 V-I供应商类选项（选项124）或DHCPv6供应商类（选项16）的供应商类别数据项中任意位置包含字符串“dslforum.org”（全部小写）来向DHCP服务器表明支持此方法。

CPE可以在DHCPv4参数请求列表（选项55）中包含DHCPv4选项43或DHCPv4选项125（但不能同时包含两者），以表示对选项43或选项125的支持并请求它们。如果CPE不以这种方式使用选项55，则服务器可以假设它支持并请求选项43（而不是选项125）。同样地，CPE可以在DHCPv6选项请求选项（选项6）中包含DHCPv6选项17。

CPE可以使用从DHCP服务器接收到的供应商特定信息（DHCPv4选项43 / DHCPv4选项125 / DHCPv6选项17）中的值来设置表2中列出的相应参数。如果同时接收到了两个DHCPv4选项，CPE必须使用它在DHCPv4参数请求列表（选项55）中包含的那个DHCP选项；除非它没有使用选项55，在这种情况下CPE必须使用DHCPv4选项43的值。如果同时收到了DHCPv4和DHCPv6选项，CPE必须优先使用DHCPv6选项而不是DHCPv4选项。此DHCP选项按照[17] / [25] / [38]定义的格式编码为一个或多个封装的供应商特定选项的列表。除了这里列出的之外，该列表还可以包括其他供应商特定选项。

包含IANA企业编号的DHCP消息，即DHCPv4选项124/125和DHCPv6选项16/17，必须使用宽带论坛的IANA企业编号，该编号是十进制的3561（“ADSL Forum”条目在IANA私有企业编号注册表[21]中）。

如果CPE通过DHCP获得了ACS URL但无法到达ACS，CPE必须使用DHCP重新发现ACS URL。如果CPE不能在每个由ACS URL解析出的IP地址上建立TCP连接达300秒，则CPE必须认为ACS不可达。如果CPE没有收到DHCP响应，它必须根据[23] / [38]尝试重试。

当CPE需要联系ACS时，在以下情况下它必须使用DHCP发现机制：

- 如果CPE的ManagementServer.URL参数为空值，或者
- 如果CPE无法联系到ACS，并且CPE最初（最近一次出厂重置后的第一次成功时）是通过DHCP获得其ACS URL的。
  这种行为使CPE能够在未预先配置ACS URL的情况下回退到使用DHCP来查找ACS。例如，这可以处理在CPE上设置了错误的ACS URL的情况。这种行为并不意在作为ACS故障转移机制。

CPE必须记住每次出厂重置后用于定位ACS的机制。如果CPE没有使用DHCP来发现ACS URL，那么它不应该回退到使用DHCP进行ACS发现。如果CPE最初使用DHCP进行ACS发现，那么当它无法联系ACS时，它必须通过DHCP执行重新发现。即使ACS URL随后通过非DHCP机制设置，这一要求仍然适用。

Table 2 – Encapsulated Vendor Specific Options【见 原文档】

所有封装的选项值必须表示为字符串，并且必须是它们对应参数的有效值。指定的URL必须是一个绝对URL。封装的选项值不得以空字符结尾。如果CPE接收到一个以空字符结尾的封装选项值，CPE必须接受提供的值，并且不得将空字符解释为值的一部分。

3. CPE可以有一个默认的ACS URL，在没有提供其他URL时使用。

ACS URL必须是有效的HTTP或HTTPS URL [6]的形式。使用HTTPS URL表示CPE必须与ACS建立SSL或TLS连接。

一旦CPE通过CWMP端点建立了与ACS的连接，ACS可以在任何时候修改存储在CPE内的ACS URL参数（ManagementServer.URL，如[27]、[34]和[35]中定义）。一旦修改，CPE必须对所有后续与ACS的连接使用修改后的URL。

ACS URL中的"主机"部分用于当使用基于证书的身份验证时CPE验证来自ACS的证书。由于这依赖于ACS URL的准确性，所以该协议的整体安全性取决于ACS URL的安全性。

CPE应当限制本地配置ACS URL的能力，仅限于需要严格安全性的机制。CPE还可以进一步限制本地设置ACS URL的能力，只允许在初始设置时进行，一旦与ACS成功建立初始连接后，就防止进一步的本地配置，这样只有现有的ACS才能随后更改此URL。CPE应防止尝试通过跨站脚本[54]和跨站请求伪造[55]攻击来更改ACS URL。

使用DHCP来配置ACS URL应该仅限于服务提供商能够确保DHCP服务器与CPE之间链接安全的情况下。由于DHCP本身不包含安全机制，因此应提供其他方法来确保这种安全性。

ACS URL可以包含DNS主机名或IP地址。解析ACS主机名时，DNS服务器可能返回多个IP地址。在这种情况下，CPE应从列表中随机选择一个IP地址。当CPE无法到达ACS时，它应该从列表中随机选择另一个IP地址，并尝试在新的IP地址上联系ACS。这种行为确保了如果多个IP地址代表不同的ACS，CPE将在这些ACS之间均衡它们的请求。

除非CPE无法联系DNS服务器以获取更新，否则CPE不应缓存超过DNS服务器返回的时间生存期（TTL）的DNS服务器响应。这种行为符合DNS RFC 1034 [5]的要求，并提供了DNS服务器更新过时数据的机会。

进一步建议CPE实现对特定ACS IP地址的亲和性。对给定IP地址的亲和性意味着CPE将尽可能长时间地尝试使用相同的IP地址，只要它能够通过该地址联系到ACS。这创建了一个更稳定的系统，并且由于更好的缓存机制，可以使ACS表现得更好。为了实现这种亲和性，CPE应该将最后成功使用的IP地址以及从中选择的IP地址列表存储在持久存储中。CPE应继续像往常一样执行DNS查询，但只要它能够联系到ACS并且DNS返回的IP地址列表没有变化，就应继续使用相同的IP地址。每当IP地址列表发生变化或CPE无法联系到ACS时，CPE应该选择一个新的IP地址。这为服务提供商提供了重新配置其网络的机会。

IANA已为CPE广域网管理协议分配了端口7547（参见[20]），ACS可以在其URL中使用此端口。

3.2 连接建立

3.2.1 CPE连接发起

CPE可以在任何时候通过CWMP端点使用预设的ACS地址（见第3.1节）发起与ACS的连接。在以下情况下，CPE必须建立与ACS的连接并发出Inform RPC方法（遵循第3.7.1.1节中描述的程序）：

- 在初始安装时CPE首次建立与接入网络的连接。
- 在上电或复位时。
- 每隔一个ManagementServer.PeriodicInformInterval（例如，每24小时一次）。
- 当被可选的ScheduleInform方法指示时。
- 无论何时CPE从ACS收到有效的连接请求（见第3.2.2节）。
- 无论何时ACS的URL发生变化。
- 无论何时修改了需要在变更时触发Inform的方法的参数。
- 无论何时被ACS通过SetParameterAttributes方法标记为"主动通知"的参数值因外部原因（非ACS自身的原因）而被修改。通过SetParameterValues由ACS自身所做的参数更改不应导致新的会话被启动。如果在CPE能够发起会话进行通知之前某个参数被多次修改，CPE只需执行一次通知。

如果在会话进行过程中，由于外部原因修改了一个参数，那么这种变更将在当前会话终止后导致一个新的会话被建立（它不应影响当前会话）。

为了避免向ACS发送过多的流量，CPE可以对参数变更通知的频率设定本地限制。这个限制应该被定义得足够宽松，仅在异常情况下才会被超过。如果超过了这个限制，CPE可以延迟由本地指定的时间量来发起一个会话以通知ACS。延迟之后，CPE必须发起一个与ACS的会话，并指出自上次此类通知以来发生的所有相关参数变化（那些被标记为需要通知的参数）。

- 每当下载或上传完成（无论成功与否），只要CPE策略指示需要通知ACS关于下载或上传完成的情况。
    - ACS始终需要被告知由ACS特别请求的下载或上传的完成情况。
    - CPE策略必须决定是否通知ACS未被特别请求的下载或上传的完成情况。
        - 注意：此CPE策略可以通过ManagementServer.AutonomousTransferCompletePolicy对象中定义的参数进行远程配置。例如，CPE可能被配置为只有在非ACS请求的下载或上传未能成功完成时才通知ACS。
- 每当根据第3.2.1.1节中规定的会话重试策略重新尝试未成功终止的会话时。

当CPE或ACS上不再有未处理的消息时，CPE不应保持与ACS的连接打开状态。有关CPE会话终止标准的详细信息，请参阅第3.7.1.4节。

3.2.1.1 会话重试策略

CPE必须重试失败的会话，以尝试重新传递之前未能成功传递的事件，并允许ACS及时发出额外请求。第3.7.1.5节详细说明了成功传递事件、重试事件传递以及在未能传递事件后丢弃事件的规则。CPE必须跟踪其尝试重试失败会话的次数。

如果CPE无法建立会话，可能是因为CPE支持CPE广域网管理协议v1.1（或更高版本），而ACS仅支持v1.0。如果怀疑这种情况（见第3.7.2.1节），CPE在重试失败会话时必须回退到v1.0。

CPE必须在等待表3中指定的时间间隔后或者当新的事件发生时重试失败的会话，以先发生的为准。CPE必须通过从重启后的会话重试计数给定范围内随机选择一个秒数来确定等待间隔。在经历了一次重启之后重试失败的会话时，CPE必须重置它所选择的等待间隔，就如同它是第一次进行会话重试一样。换句话说，如果会话因BOOT以外的新事件而被重试，这不会重置等待间隔，尽管新事件的持续出现可能会导致会话发起频率高于表格所示。无论前一次会话失败的原因是什么，或是触发会话重试的情况如何，CPE都必须向ACS传达会话重试计数。

等待间隔的范围由两个参数控制：最小等待间隔和间隔乘数，每个参数都对应于数据模型中的一个参数，具体描述如下表所示。

Descriptive Name	Symbol	Default	Data Model Parameter Name
Minimum wait interval	m	5 seconds	ManagementServer.CWMPRetryMinimumWaitInterval
Interval multiplier	k	2000	ManagementServer.CWMPRetryIntervalMultiplier

这些参数的出厂默认值必须是先前版本的CPE广域网管理协议中硬编码的值，即"默认"列中的值。如第3.1节所述，这些值可以通过DHCP获得的值来覆盖。ACS也可以随时更改这些值。

从第十次重启后的会话重试尝试开始，CPE必须从表3中显示的固定最大范围内选择等待间隔。CPE必须继续重试失败的会话，直到会话成功终止，或者直到遵循表8中"重试/丢弃策略"列定义的规则为止。一旦会话成功终止，CPE必须将会话重试计数重置为零，并且不再应用会话重试策略来决定何时发起下一个会话。

Table 3 – Session Retry Wait Intervals

以下是您提供的表格内容的文本形式：

Post Session Count	Reboot Retry	Default Wait Interval Range (min-max seconds)	Actual Wait Interval Range (min-max seconds)
#1		5-10	m – m.(k/1000)
#2		10-20	m.(k/1000) – m.(k/1000)^2
#3		20-40	m.(k/1000)^2 – m.(k/1000)^3
#4		40-80	m.(k/1000)^3 – m.(k/1000)^4
#5		80-160	m.(k/1000)^4 – m.(k/1000)^5
#6		160-320	m.(k/1000)^5 – m.(k/1000)^6
#7		320-640	m.(k/1000)^6 – m.(k/1000)^7
#8		640-1280	m.(k/1000)^7 – m.(k/1000)^8
#9		1280-2560	m.(k/1000)^8 – m.(k/1000)^9
#10 and subsequent		2560-5120	m.(k/1000)^9 – m.(k/1000)^10

3.2.1.2 随机源端口的使用

每次CPE在重启后首次连接到ACS时，都应使用不同的临时TCP源端口，以避免重复使用上次使用的相同端口。如果自上次连接以来的时间少于ACS配置的TCP TIME_WAIT值，重用相同的端口可能导致ACS拒绝该连接。

为了尽量减少连续情况下使用相同临时端口号的概率，端口的选择应使用强随机化机制。

3.2.2 ACS连接发起

ACS可以在任何时候请求CWMP端点使用连接请求机制发起与ACS的连接。CPE必须支持此机制，而ACS则建议支持该机制。

该机制依赖于ACS能够访问到CPE。如果CPE位于防火墙之后，或者在ACS和CPE之间存在NAT设备，则ACS可能根本无法访问CPE。附录K定义了一种机制，允许ACS联系那些不能直接被ACS访问到的CPE。

该机制依赖于ACS至少有一次与CPE发起的交互中的CWMP端点进行过通信。在这次交互中，如果ACS希望允许将来的由ACS发起的事务，它会使用ManagementServer.ConnectionRequestURL参数的值（参见[27]、[34]和[35]）。如果用于管理访问的URL发生变化，CPE必须通过Inform消息通知ACS。

IANA已为CPE广域网管理协议分配了端口7547（参见[20]），CPE可以在连接请求URL中使用此端口。

3.2.2.1 通用连接请求要求

连接请求机制有以下通用要求（意味着这些要求适用于任何独立于所使用的传输协议的连接请求通信）：

- CPE必须接受来自任何具有目标CPE正确认证参数来源的连接请求。
- 为了进一步减少拒绝服务攻击的可能性，CPE应该限制在特定时间段内对某个特定CWMP端点接受的连接请求数量。如果CPE因此选择拒绝一个连接请求，则CPE必须用特定于传输的错误响应该连接请求。
- 如果CPE成功地验证并如上所述响应了一个针对特定CWMP端点的连接请求，并且它尚未为请求的CWMP端点建立会话，则它必须在发送响应后的30秒内尝试与预定的ACS地址（参见第3.1节）建立会话，在此过程中Inform中包含“6 CONNECTION REQUEST”事件代码。
- 注意 - 实际操作中可能存在一些特殊情况，导致CPE偶尔无法满足这一要求。
- 如果ACS收到一个成功的连接请求响应，但在至少30秒后CPE仍未成功建立包括“6 CONNECTION REQUEST”事件代码在内的会话，ACS可以对该CPE重试连接请求。
- 如果CPE成功验证并响应了一个连接请求，但在与ACS建立会话之前，收到了针对同一个CWMP端点的一个或多个成功验证的连接请求，那么CPE必须对每一个这样的连接请求返回一个成功的响应，但不得由于这些额外的连接请求而为同一个CWMP端点发起任何额外的会话，无论在这段时间内它接收到了多少个这样的请求。
- 如果CPE在已经与至少一个CWMP端点同ACS处于会话状态时收到一个或多个连接请求，它不得因此提前终止任何CWMP端点的会话。相反，CPE必须采取特定于传输的替代措施。
  这个要求适用于CPE认为自己与至少一个CWMP端点处于会话期间任何时候收到的连接请求，包括CPE正在建立会话的过程中。
- 除了上述描述的原因外，CPE不得以任何其他理由拒绝正确验证的连接请求。如果CPE因为上述原因之一拒绝了连接请求，则不得因此连接请求与ACS发起会话。

3.2.2.2 HTTP特定的连接请求要求

当通过HTTP传输时，连接请求机制还有以下要求：

- 连接请求端口必须仅用于TR-069连接请求。它不得与其他任何协议共享。
- 连接请求必须使用HTTP 1.1 GET方法访问由CPE指定的特定URL。该URL值作为只读参数在CPE上可用。此URL路径应由CPE随机生成，以确保每个CPE都是唯一的。
- 连接请求必须使用HTTP，而不是HTTPS。相关的URL必须是HTTP URL。
- 在连接请求的HTTP GET中不携带任何数据。CPE应当忽略可能包含的任何数据。
- CPE在继续之前必须使用[8]定义的HTTP认证以及[9]中定义的HTTP摘要认证方案来验证ACS的身份——如果认证未成功，CPE不得发起与ACS的连接。
- 对于成功验证的连接请求，CPE的响应必须使用“200 (OK)”或“204 (No Content)”HTTP状态码。CPE必须在成功验证后立即发送此响应，在其启动由此产生的会话之前。HTTP响应中的消息体长度（参见第3.3节/RFC 7230 [6]）必须为零。

- 为了进一步减少拒绝服务攻击的可能性，CPE应该限制在特定时间段内对某个特定CWMP端点接受的连接请求数量。如果CPE因此选择拒绝一个连接请求，则CPE必须用HTTP 503状态码（服务不可用）响应该连接请求。此外，CPE不应在响应中包括HTTP Retry-After头部。
- 如果CPE在已经与至少一个CWMP端点同ACS处于会话状态时收到一个或多个连接请求，它不得因此提前终止任何CWMP端点的会话。相反，CPE必须采取以下HTTP特定的替代措施之一：
- 通过响应HTTP 503状态码（服务不可用）来拒绝每个连接请求。在这种情况下，CPE不应在响应中包含HTTP Retry-After头部。
- 在当前CWMP端点的会话完成后，一次性仅发起一个新的会话（无论在之前的会话期间收到了多少个连接请求），并在Inform中包含“6 CONNECTION REQUEST”事件代码。未被接受的连接请求必须被拒绝（使用HTTP 503状态码）。如果新会话是为当前正在进行会话的CWMP端点准备的，CPE必须在现有会话完成并且该会话中的所有更改都已应用后立即发起新的会话。
- 如果连接请求不是针对任何当前处于会话中的CWMP端点，CPE可以在现有会话仍然活跃时与请求的CWMP端点发起新的会话。
  此要求适用于CPE认为自己与至少一个CWMP端点处于会话期间任何时候收到的连接请求，包括CPE正在建立会话的过程中。
- 如果ACS通过将Device.ManagementServer.HTTPConnectionRequestEnable参数设置为“False”来禁用HTTP连接请求机制，则CPE必须关闭用于HTTP连接请求的端口。

3.3 使用TLS 和 TCP

3.4 HTTP的使用

SOAP消息在CPE（客户驻地设备）和ACS（自动配置服务器）之间通过HTTP 1.1 [6]进行传输，其中CPE充当HTTP客户端，而ACS充当HTTP服务器。

注意 – *CPE*广域网管理协议也使用*HTTP*来进行连接请求，在这种情况下*ACS*充当*HTTP*客户端，而*CPE*充当*HTTP*服务器。这种*HTTP*的使用方式在第*3.2.2*节中有描述。

3.4.1 基于http的soap封装

SOAP over HTTP的编码扩展了[12]第6节中定义的SOAP的HTTP绑定，具体如下：

- 从ACS到CPE的SOAP请求通过HTTP响应发送，而CPE对ACS请求的SOAP响应则通过随后的HTTP POST发送。
- 当HTTP请求中包含SOAP响应时，或者当HTTP请求中包含SOAP错误响应时，HTTP请求中的SOAPAction头必须没有值（不带引号），表示该头不提供关于消息意图的信息。也就是说，它必须如下所示：

  ```
  SOAPAction:
  ```

- 当HTTP请求或响应包含一个SOAP信封时，HTTP的Content-Type头必须有"type/subtype"为"text/xml"。
- 空的HTTP POST不得包含SOAPAction头。
- 空的HTTP POST不得包含Content-Type头。
- 包含任何CPE WAN管理协议负载的HTTP响应（即向CPE发出的SOAP请求、对CPE的成功SOAP响应，或包含在第3.5节中定义的Fault元素的SOAP错误响应）必须使用HTTP状态码200（OK）。

当在XML文档中传输字符串值时，所有对于XML来说特殊的字符都必须按照XML规范[7]的规定进行转义。此外，除了可打印的ASCII字符外，即那些十进制ASCII表示不在（包括）9-10, 13和32-126范围内的字符，也应当按照XML规范的规定进行转义。

下面是一个来自ACS并包含SOAP请求的HTTP响应示例：

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: xyz
<soap:Envelope
 xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:cwmp="urn:dslforum-org:cwmp-1-2">
 <soap:Body>
 <cwmp:Request>
 <argument>value</argument>
 </cwmp:Request>
 </soap:Body>
</soap:Envelope>
```

注意 – 在上面的例子中，所使用的*XML*命名空间前缀仅作为示例。实际的命名空间前缀值是任意的，仅用于引用命名空间声明。

注意 – 在上面的例子中，*CWMP*（*CPE*广域网管理协议）命名空间标识符*"urn:dslforum-org:cwmp-1-0"*只是一个示例，并不一定是由本规范定义的命名空间。

3.4.2 会话

对于形成单一会话的一系列事务，CPE应当在整个会话期间维持一个持久的TCP连接。具体来说，CPE在会话期间不得主动发起TCP连接的关闭。然而，如果在一个HTTP请求/响应往返后干净地关闭了TCP连接，并且在最后一个HTTP响应时会话没有以其他方式终止（无论是成功还是不成功），CPE必须通过新的TCP连接发送下一个HTTP请求来继续该会话。

注意 – 每个会话使用单一的TCP连接是非常理想的，因为重新协商连接会消耗额外资源（网络套接字、网络带宽、处理能力），这增加了系统过载和会话丢失的风险，从而增加了负载。

在接受到身份验证挑战后，除非ACS通过"Connection: close"HTTP头明确要求关闭TCP连接，否则CPE必须在同一TCP连接中发送下一个带有"Authorization"HTTP头的HTTP请求。在这种情况下，CPE必须遵守ACS的要求，关闭TCP连接，并在新的TCP连接中发送下一个带有"Authorization"HTTP头的HTTP请求。

如果出于任何原因，CPE未能建立TCP连接、未能发送HTTP消息或未能接收HTTP响应，CPE必须认为会话未成功终止。CPE在宣告无法建立TCP连接或未收到HTTP响应之前，至少应等待30秒。

ACS应该使用会话cookie来维护如[11]中所述的会话状态。ACS应该只使用标记为Discard的cookie，并且不应假设CPE会在会话之外的时间内保留cookie。

为了确保ACS可以使用会话cookie，CPE必须支持如[11]中定义的cookie使用，包括在后续每个HTTP POST中返回cookie值，但CPE不需要支持超出会话持续时间的cookie存储。CPE必须支持ACS使用多个cookie，并且必须提供至少512字节用于存储cookie。

注意 - 当与使用本规范旧版本的*ACS*打交道时，可能会使用老式的*"Netscape"cookie*。

当会话成功完成或不成功终止时，CPE必须关闭与ACS相关的TCP连接。

CPE必须支持ACS使用的HTTP重定向。CPE和ACS与使用HTTP重定向相关的要求如下：

当然，以下是按照格式要求翻译的内容：

- CPE必须支持302（Found）和307（Temporary Redirect）HTTP状态码。
- CPE也可以支持301（Moved Permanently）HTTP状态码用于重定向。
- CPE必须允许在会话的任何时候（包括InformResponse时）发生重定向，并且ACS可以在会话的任何时间点发出重定向。

- 如果CPE被重定向，它必须尝试使用HTTP重定向响应中提供的URL继续会话。具体来说，CPE必须将导致重定向响应的HTTP POST重新发送到重定向后的URL，并且CPE随后必须像没有发生重定向一样尝试继续会话。
- 如果CPE被重定向，重定向的URL仅应适用于当前会话的剩余部分或直到同一会话中稍后发生另一次重定向为止。CPE不得保存重定向的URL（即，不作为[27]、[34]和[35]中定义的ManagementServer.URL值），以供后续会话或会话的任何后续重试使用。即使使用了301（Moved Permanently）HTTP状态码进行重定向，这一要求也必须得到遵守。
- CPE必须允许最多5次连续重定向。如果CPE连续重定向超过5次，它可以认为会话未成功终止。
- HTTP重定向中提供的URL可以是HTTP或HTTPS URL。无论重定向前使用的传输方式如何，都必须使用适当的传输机制（TCP或TLS）与新的目标进行通信。
- 如果重定向会话使用TLS，要求CPE对ACS进行身份验证，则认证必须基于重定向后的URL而不是预配置的ACS URL（见第3.3节）。
- 在由ACS发送的包含重定向状态码的HTTP响应中，HTTP实体主体的长度（参见3.3/RFC7230 [6]）必须为零。如果CPE收到一个带有非空实体主体的HTTP重定向响应，它必须忽略实体主体的内容。
- 当被重定向时，CPE必须在随后对重定向后的ACS的HTTP请求中包含与会话相关的所有cookie。CPE必须将从ACS的重定向视为[11]中定义的"可验证交易"，因此它必须在不执行每个cookie的域名验证的情况下向重定向后的ACS发送cookie。

3.4.3 文件传输

如果CPE被ACS指示通过Download、ScheduleDownload、Upload或ChangeDUState（安装或更新操作）请求执行文件传输，并且文件位置被指定为与ACS相同的主机名的HTTP URL，那么CPE必须选择以下方法之一来执行传输：

- CPE可以通过已经建立的连接发送HTTP GET/PUT。一旦文件传输完成，CPE可以继续在同一连接上向ACS发送其他消息（此选项不适用于ScheduleDownload或ChangeDUState（安装或更新操作））。
- CPE可以打开第二个连接以进行文件传输，同时保持与ACS的会话，以便它能够继续发送消息。
- CPE可以终止与ACS的会话，然后执行文件传输。

如果文件位置不是一个HTTP URL，或者不在ACS的同一个域内，或者需要使用不同的端口，那么只有后两种选项是可用的。

CPE必须支持第3.3节中规定的TLS，用于建立单独的TCP连接以通过HTTP传输文件。当文件位置被指定为HTTPS URL时，CPE必须使用TLS。

对于文件传输，CPE必须同时支持HTTP基本认证和摘要认证。具体的认证方法由文件服务器提供基本或摘要认证挑战来决定。如果文件服务器使用了认证，那么ACS必须使用发起传输的具体RPC方法（即Download、ScheduleDownload、Upload、ChangeDUState（安装或更新操作））来指定凭证。

3.4.4 认证

如果CPE没有使用TLS进行身份验证，ACS必须使用HTTP身份验证[8]来对CPE进行身份验证。如果正在使用TLS进行加密，ACS应该使用[10]中定义的基本身份验证方案。如果不使用TLS，则ACS必须使用[9]中定义的摘要身份验证方案。

CPE必须同时支持HTTP基本身份验证和摘要身份验证方案。ACS通过提供基本或摘要身份验证挑战来选择身份验证方案。如果正在使用TLS进行加密，CPE应该预先发送[10]中定义的基本身份验证凭证。

注意 – 使用身份验证挑战需要首先发送初始消息（通常是Inform RPC方法请求）；使用带有TLS的预基本身份验证是安全的，并且避免了额外请求的需求。

如果CPE从ACS收到了身份验证挑战（无论是基本还是摘要），CPE应在TCP连接持续期间的所有后续HTTP请求中发送Authorization头。无论CPE是否这样做，ACS可以在单个或多个TCP连接中的会话任何阶段发出后续的身份验证挑战。

如果使用任何形式的HTTP身份验证来验证CPE，CPE应使用在所有CPE制造商中全球唯一的用户名/用户ID。具体来说，CPE的用户名/用户ID应采用以下两种格式之一：

`<OUI> "-" <ProductClass> "-" <SerialNumber>`

`<OUI> "-" <SerialNumber>`

如果使用上述格式的用户名/用户ID，则、和字段必须与Inform消息中定义的DeviceIdStruct所包含的相应参数完全匹配，如附录A中定义的那样。不过，为了确保可以从用户名/用户ID中提取参数值，和中的任何非0-9、A-Z、a-z或下划线（“_”）字符必须按照RFC 3986 [15]的规定使用URI百分比编码进行转义。

百分比编码必须通过将每个字符转换为UTF-8，然后对每个字节进行百分比编码来执行。例如，字符é（带尖音符的小写拉丁字母E）在UTF-8中表示为两个字节0xC3 0xA9，因此会被百分比编码为“%C3%A9”。

注意 – 在明确指出在百分比编码之前需要转换为UTF-8之前，转义后的用户名/用户ID是模糊不清的。例如，实现可能会将字符é视为ISO Latin-1字节0xE9，这将被百分比编码为“%E9”。

如果使用上述格式的用户名/用户ID，并且只有当ProductClass参数的值为空时，才必须使用第二种形式。

示例：

```
012345-0123456789
012345-STB-0123456789
012345-Set%2DTop%2DBox-0123456789
```

无论使用哪种形式的HTTP身份验证，密码应该是每个CPE的唯一值。也就是说，多个CPE不应共享相同的密码。这个密码是一个共享密钥，因此必须由CPE和ACS双方都知道。初始安装CPE时，使共享密钥为双方所知的方法不在本规范的范围内。CPE和ACS都应采取适当的措施防止未经授权访问密码，对于ACS来说则是防止未经授权访问密码列表。

3.4.5 摘要认证

本节概述了在CPE广域网管理协议中使用摘要认证的要求。这些要求适用于RPC交换连接的身份验证以及文件传输的身份验证。请注意，对于不同类型连接，ACS和CPE交替扮演HTTP客户端和服务器的角色。当发出连接请求时，ACS扮演HTTP客户端的角色；当向ACS发起连接时，CPE扮演HTTP客户端的角色。

CPE和ACS必须支持RFC 7616 [9]中的“qop”选项，并且该选项包含值“auth”。

在使用摘要认证时，对于每个新打开的TCP连接，ACS应该使用一个新的nonce值，而CPE应该使用一个新的cnonce值。

注意 – 如果*CPE*广域网管理协议会话不使用*TLS*，则*ACS*用于*HTTP*身份验证的*nonce*值重用策略可以显著影响会话的安全性。特别是，如果*ACS*在跨多个*TCP*连接重新认证时重用了*nonce*值，那么*ACS*可能会受到重放攻击。然而，如果会话使用了*TLS*，则这种风险将大大降低。

出于向后兼容的原因，CPE和ACS必须支持MD5摘要算法。同样出于向后兼容的原因，CPE还必须支持MD5-sess摘要算法。此外，SHA-256和SHA-256-sess算法（根据RFC 7616 [9]是必须实现的）可由ACS选择性地使用，但这可能导致对旧版CPE进行身份验证尝试失败，并需要回退到MD5或MD5-sess算法。

3.4.6 额外的http要求

以下指定了额外的HTTP相关要求：

- 每当ACS发送一个空的HTTP响应时，它必须使用“204（无内容）”HTTP状态码。
- CPE不得使用HTTP 1.1 [6]中定义的流水线技术。

3.4.7 http压缩

本节概述了在CPE广域网管理协议中使用HTTP协议交换内容编码的要求，这些内容编码的定义见RFC 7230 [6]第4节“传输编码”。这些要求适用于RPC交换以及文件传输的压缩。

以下是额外指定的HTTP相关要求：

- ACS应该支持RFC 7230 [6]第4节“传输编码”中定义的内容编码交换。
- CPE应该支持RFC 7230 [6]第4节“传输编码”中定义的内容编码交换。

为了使CPE和ACS能够高效地交换压缩消息，除非ManagementServer.HTTPCompression参数设置为“禁用”，否则CPE必须发送带有由该参数定义的Content-Encoding头的压缩消息。

如果ACS不支持Content-Encoding头或其值，ACS必须响应“415 - 不支持的媒体类型”HTTP状态码。当接收到“415 - 不支持的媒体类型”HTTP状态码时，CPE必须移除压缩并按照3.2.1.1节会话重试策略的规定重试会话。

ACS可以通过将ManagementServer.HTTPCompression参数设置为CPE和ACS都支持的值来启用HTTP压缩。ACS可以通过将ManagementServer.HTTPCompression参数设置为“禁用”来关闭HTTP压缩。CPE在ManagementServer.HTTPCompressionSupported参数中列出所支持的HTTP压缩机制。

3.5 SOAP的使用

CPE广域网管理协议定义了SOAP 1.1 [12]作为传输附录A中定义的RPC方法调用和响应的编码语法。

以下描述了RPC方法到SOAP编码的映射：

- 编码必须使用标准的SOAP 1.1信封和序列化命名空间：
  - 信封命名空间标识符 "http://schemas.xmlsoap.org/soap/envelope/"
  - 序列化命名空间标识符 "http://schemas.xmlsoap.org/soap/encoding/"
- 附录A中使用的数据类型直接对应于SOAP 1.1序列化命名空间中定义的数据类型。（通常，附录A中使用的是相应SOAP类型的受限子集。）
- 根据SOAP规范[12]，指定为“anySimpleType”类型的所有元素必须包含一个type属性来指示元素的实际类型。
- 非“anySimpleType”类型的元素只有在该元素使用附录A中的RPC方法XML模式中定义的命名数据类型时，才可以包含type属性。如果包含了type属性，那么type属性的值必须与模式中指定的命名数据类型完全匹配。
- 对于数组参数，数组定义表中指定的参数名称必须用作整个数组元素的名称。数组成员元素的名称必须是数组在定义表中指定的数据类型（不包括方括号和任何圆括号内的长度限制），并且不得有命名空间限定。例如，名为ParameterList的参数，它是一个ParameterValueStruct结构的数组，应该被编码为：

```
<ParameterList soap-enc:arrayType="cwmp:ParameterValueStruct[2]">
 <ParameterValueStruct>
 <name>Parameter1</name>
 <value xsi:type="someType">1234</value>
 </ParameterValueStruct>
 <ParameterValueStruct>
 <name>Parameter2</name>
 <value xsi:type="someType">5678</value>
 </ParameterValueStruct>
</ParameterList>
```

作为第二个例子，GetRPCMethodsResponse中的MethodList数组将被编码为：

```
<MethodList soap-enc:arrayType="xsd:string[3]">
 <string>GetRPCMethods</string>
 <string>Inform</string>
 <string>TransferComplete</string>
</MethodList>
```

注意 – 在上面的例子中，所使用的*XML*命名空间前缀仅作为示例。实际的命名空间前缀值是任意的，仅用于引用命名空间声明。

注意 – 始终需要为*arrayType*属性指定一个*XML*命名空间前缀。对于*CPE*广域网管理协议（*CWMP*）特定类型的数组，这始终将是*CWMP*命名空间前缀；而对于其他类型的数组，则始终是*XML Schema*命名空间前缀或*SOAP*编码命名空间前缀。

- 关于RPC方法的SOAP编码规范（[12]的第7节），对于附录A中定义的每个方法，方法调用中列出的每个参数代表[in]参数，而方法响应中列出的每个参数代表[out]参数。没有使用[in/out]参数。
- 定义的RPC方法使用标准的SOAP命名约定，即与给定方法对应的响应消息是通过在方法名称后加上“Response”后缀来命名的。
- 一个SOAP信封必须恰好包含一个Body元素。
- CPE必须能够接受总信封大小至少为32千字节（32,768字节）的SOAP请求，而不导致“资源超出”响应。
- CPE必须能够生成任意所需长度的SOAP响应，而不导致“资源超出”响应，即CPE SOAP响应长度没有最大限制。
- ACS必须能够接受总信封大小至少为32千字节（32,768字节）的SOAP请求，而不导致“资源超出”响应。
- ACS必须能够生成任意所需长度的SOAP响应，而不导致“资源超出”响应，即ACS SOAP响应长度没有最大限制。
- 故障响应必须使用以下约定的SOAP Fault元素：
    - SOAP faultcode元素必须指示故障源，根据具体的故障情况应为“Client”或“Server”。在这种用法中，“Client”表示SOAP请求的发起者，“Server”表示SOAP响应者。SOAP faultcode元素的值必须是未限定的，或者使用SOAP信封命名空间前缀进行限定。故障响应的接收方不需要使用该元素的值，并可以完全忽略SOAP faultcode元素。
    - SOAP faultstring子元素必须包含字符串“CWMP fault”。
    - SOAP detail元素必须包含一个Fault结构。附录A中的RPC方法XML模式正式定义了这个结构。此结构包含以下元素：
        - **FaultCode** 元素，其中包含附录A中定义的单个数字故障代码。
        - **FaultString** 元素，其中包含可读的人类描述的故障信息。
        - **SetParameterValuesFault** 元素，仅用于SetParameterValues方法的错误响应中，它包含一个或多个结构列表，指出每个出错参数的具体故障。此结构包含以下元素：
            - **ParameterName** 元素，其中包含出错参数的完整路径名。
            - **FaultCode** 元素，其中包含附录A中定义的单个数字故障代码，指示特定参数的故障。
            - **FaultString** 元素，其中包含特定参数故障的可读人类描述。

下面是包含了一个错误响应的示例信封：

```
<soap:Envelope
 xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:cwmp="urn:dslforum-org:cwmp-1-0">
     <soap:Header>
```

```
        <cwmp:ID soap:mustUnderstand="1">1234</cwmp:ID>
    </soap:Header>
    <soap:Body>
        <soap:Fault>
            <faultcode>Client</faultcode>
            <faultstring>CWMP fault</faultstring>
            <detail>
                <cwmp:Fault>
                    <FaultCode>9000</FaultCode>
                    <FaultString>Upload method not supported</FaultString>
                </cwmp:Fault>
            </detail>
        </soap:Fault>
    </soap:Body>
</soap:Envelope>
```

下面是一个包含了对应SetParameterValues 方法调用的错误响应的示例信封：

```
<soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:cwmp="urn:dslforum-org:cwmp-1-0">
    <soap:Header>
        <cwmp:ID soap:mustUnderstand="1">1234</cwmp:ID>
    </soap:Header>
    <soap:Body>
        <soap:Fault>
            <faultcode>Client</faultcode>
            <faultstring>CWMP fault</faultstring>
            <detail>
                <cwmp:Fault>
                    <FaultCode>9003</FaultCode>
                    <FaultString>Invalid arguments</FaultString>
                    <SetParameterValuesFault>
                        <ParameterName>Device.Time.NTPServer1</ParameterName>
                        <FaultCode>9007</FaultCode>
                        <FaultString>Invalid IP Address</FaultString>
                    </SetParameterValuesFault>
                    <SetParameterValuesFault>

 <ParameterName>Device.Time.LocalTimeZoneName</ParameterName>
                        <FaultCode>9007</FaultCode>
                        <FaultString>String too long</FaultString>
                    </SetParameterValuesFault>
                </cwmp:Fault>
            </detail>
        </soap:Fault>
    </soap:Body>
</soap:Envelope>
```

注意 – 在上面的例子中，使用的*XML*命名空间前缀仅作为示例。实际的命名空间前缀值是任意的，仅用于引用命名空间声明。
注意 – 在上面的例子中，*CWMP*命名空间标识符*“urn:dslforum-org:cwmp-1-0”*仅作为示例，并不一定是本规范定义的命名空间。

故障响应必须只在响应SOAP请求时发送。故障响应不得在响应SOAP响应或其他故障响应时发送。

如果故障响应不符合上述所有要求，接收方必须认为该SOAP消息无效。无效的SOAP对CPE广域网管理协议会话的影响在第3.7节中描述。

- PE可以忽略：(a) SOAP Body中的任何未知XML元素及其子元素或内容，(b) 任何未知XML属性及其值，(c) 任何嵌入的XML注释，以及(d) 任何XML处理指令。或者，ACS和CPE可以明确验证接收到的XML，并拒绝包含未知元素的信封。请注意，这排除了通过添加额外参数而不改变消息名称来扩展现有消息的可能性。
- 如果RPC方法需要引用XML Schema命名空间（例如对于"type"属性，或对XML Schema数据类型的引用），这些引用必须指向这些命名空间定义的2001版本，具体来说是http://www.w3.org/2001/XMLSchema-instance 和 http://www.w3.org/2001/XMLSchema。接收方可以拒绝引用这两个命名空间不同版本的RPC方法。

作为一个按照上述编码方式的RPC方法示例，GetParameterNames请求将被编码为：

```
<soap-env:Envelope
    xmlns:soap-enc="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:soap-env="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:cwmp="urn:dslforum-org:cwmp-1-0">
    <soap-env:Header>
        <cwmp:ID soap-env:mustUnderstand="1">0</cwmp:ID>
    </soap-env:Header>
    <soap-env:Body>
        <cwmp:GetParameterNames>
            <ParameterPath>Object.</ParameterPath>
            <NextLevel>0</NextLevel>
        </cwmp:GetParameterNames>
    </soap-env:Body>
</soap-env:Envelope>
```

注意 – 在上面的例子中，使用的*XML*命名空间前缀仅作为示例。实际的命名空间前缀值是任意的，仅用于引用命名空间声明。

注意 – *CWMP*命名空间前缀仅指定给*CWMP*模式顶层定义的元素（如上例中的*ID*和*GetParameterNames*）。在这些元素内部包含的元素（如上例中的*ParameterPath*和*NextLevel*）上指定命名空间是不正确的。这是因为*CWMP*模式指定了一个*"unqualified"*（未限定）的*elementFormDefault*值。

注意 – 在上面的例子中，*CWMP*命名空间标识符*"urn:dslforum-org:cwmp-1-0"*仅作为示例，并不一定是由本规范定义的版本。

*CPE*广域网管理协议定义了一系列*SOAP Header*元素，如表*4*中所规定。

Table 4 – SOAP Header Elements

Tag Name	Description
ID	此头部元素可用于通过为每个请求使用唯一标识符来关联SOAP请求和响应，相应的响应中包含匹配的标识符。标识符的值是一个任意字符串，由请求者自行决定设置。 如果在SOAP请求中使用了ID头部，那么该ID头部必须出现在对应的响应中（无论响应是成功还是失败）。 由于支持这个头部是必需的，mustUnderstand属性必须设置为“1”（即true）以表示这个头部。
HoldRequests	此头部可以包含在从ACS发送到CPE的SOAP信封中，以调节从CPE发出的请求的传输。此头部不得出现在从CPE发送到ACS的信封中。 该标签具有“0”（假）或“1”（真）的布尔值。如果标签不存在，则解释为等同于“0”（假）。 CPE接收到此头部时的行为在第3.7.1.3节中定义。CPE对此头部的支持是必需的。 由于支持这个头部是必需的，mustUnderstand属性必须设置为“1”（即true）。 此头部已被弃用，因为它不必要地使协议和CWMP会话流程变得复杂。
SessionTimeout	此头部可以在CPE向ACS发送的SOAP信封中包含，仅用于在CWMP会话初始化期间提供一个可接受的CWMP会话超时持续时间的建议。此头部不得出现在从ACS发送到CPE的信封中。此外，如果SOAP体中不包含CWMP Inform请求，则此头部也不得出现。 SessionTimeout是一个整数，表示ACS应等待的时间（秒数），直到由于CPE未响应而使CWMP会话超时。建议的SessionTimeout必须为30秒或更长。 由于对此头部的支持是可选的，mustUnderstand属性必须为此头部设置为“0”（即false）。
pportedCWMPVersions	此头部可以包含在从ACS发送到CPE的SOAP信封中，以调节从CPE发出的请求的传输。此头部不得出现在从CPE发送到ACS的信封中。 该标签具有“0”（假）或“1”（真）的布尔值。如果标签不存在，则解释为等同于“0”（假）。 CPE接收到此头部时的行为在第3.7.1.3节中定义。CPE对此头部的支持是必需的。 由于支持这个头部是必需的，mustUnderstand属性必须设置为“1”（即true）。 此头部已被弃用，因为它不必要地使协议和CWMP会话流程变得复杂。
UseCWMPVersion	此头部必须在ACS向CPE发送的SOAP信封中包含，仅用于在CWMP会话初始化期间向CPE提供所选的CWMP版本，前提是且仅当Inform请求中包含了SupportedCWMPVersions头部并且ACS支持该SupportedCWMPVersions头部时。此头部不得出现在从CPE发送到ACS的信封中。此外，如果SOAP体中不包含CWMP InformResponse，则此头部也不得出现。 UseCWMPVersion值必须是从Inform请求中发送的SupportedCWMPVersions列表中选择的一个版本。由于此头部仅在CPE发送了SupportedCWMPVersions时才会发送，因此mustUnderstand属性必须为此头部设置为“1”（即true）。

下面是一个两个信息的样例，这个样例展示了一些无DEPRECATED头部的用法：

- CPE到ACS的SOAP头部

```
<soap:Envelope
               xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
               xmlns:cwmp="urn:dslforum-org:cwmp-1-2">
    <soap:Header>
        <cwmp:ID soap:mustUnderstand="1">1234</cwmp:ID>
        <cwmp:SessionTimeout
soap:mustUnderstand="0">40</cwmp:SessionTimeout>
    </soap:Header>
    <soap:Body>
        <cwmp:Action>
            <argument>value</argument>
        </cwmp:Action>
    </soap:Body>
</soap:Envelope>
```

- ACS到CPE的SOAP头部

```
<soap:Envelope
               xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
               xmlns:cwmp="urn:dslforum-org:cwmp-1-2">
    <soap:Header>
        <cwmp:ID soap:mustUnderstand="1">1234</cwmp:ID>
    </soap:Header>
    <soap:Body>
        <cwmp:Action>
            <argument>value</argument>
        </cwmp:Action>
    </soap:Body>
</soap:Envelope>
```

注意 – 在上面的例子中，使用的*XML*命名空间前缀仅作为示例。实际的命名空间前缀值是任意的，仅用于引用命名空间声明。

注意 – 在上面的例子中，*CWMP*命名空间标识符*“urn:dslforum-org:cwmp-1-2”*仅作为示例，并不一定是由本规范定义的版本。

3.6 RPC支持要求

3.6.1 基于别名的寻址机制要求

可选的基于别名的寻址机制使用在A.2.2.2中定义并在附录II中描述的实例别名标识符。

支持基于别名寻址机制的ACS必须完全遵守以下所有要求：

1. 如果CPE在Inform参数中没有包含ManagementServer.AliasBasedAddressing参数（设置为true），ACS不得使用实例别名标识符。

支持基于别名寻址机制的CPE必须完全遵守以下所有要求：

1. CPE必须支持实例别名标识符作为多实例对象的寻址方法，除了实例编号标识符之外。

2. 在创建一个多实例对象的实例时，CPE必须分配一个唯一的实例别名（使用“cpe-”前缀），除非在来自ACS的CWMP RPC中提供了实例别名值。由于任何其他操作或包含在CPE出厂默认设置中的实例所创建的别名必须带有“cpe-”前缀。对于同一硬件型号和软件版本的所有CPE实例，CPE必须对出厂默认对象使用相同的实例别名值。
3. CPE必须支持ManagementServer.AliasBasedAddressing参数作为强制Inform参数，并且在所有的Inform消息中包含它（设置为true）。
4. CPE必须支持对ManagementServer.AutoCreateInstances参数的写访问权限，该参数用于由ACS启用或禁用CPE自动创建实例机制（定义于A.3.2.1）。
5. CPE必须支持对ManagementServer.InstanceMode参数的写访问权限。此参数用于ACS控制CPE在返回的路径名称中使用实例编号还是实例别名，详细要求见6、7、8和9条。
6. 在接收到请求时，CPE必须支持参数路径名称中对象的一致或混合实例标识符。混合参数路径名称在不同的节点级别具有不同类型的实例标识符（实例编号或实例别名）。当发出响应时，CPE必须使参数路径名称中每个节点级别的每个对象与ACS请求中相同类型的实例标识符（实例编号或实例别名）相匹配。下表中呈现的所有排列组合（以任意顺序）都是有效的，并且必须被支持：

Path Type	Message	Path Name Example
Uniform Instance Number Identifier	Request	TopGroup.Lev1Obj.1.Lev2Obj.1.
	Response	TopGroup.Lev1Obj.1.Lev2Obj.1.Parameter
Uniform Instance Alias Identifier	Request	TopGroup.Lev1Obj.[a].Lev2Obj.[b].
	Response	TopGroup.Lev1Obj.[a].Lev2Obj.[b].Parameter
Mixed Instance Identifier	Request	TopGroup.Lev1Obj.1.Lev2Obj.[b].
	Response	TopGroup.Lev1Obj.1.Lev2Obj.[b].Parameter

7. 如果CPE需要发出的响应中包含参数路径名称中的对象实例，并且这些实例位于ACS请求中接收到的路径节点之下的节点层级，那么它必须使用ManagementServer.InstanceMode参数来选择如何在响应中提供路径名称：
    - 如果ManagementServer.InstanceMode参数设置为InstanceNumber，则所有位于接收到的部分路径名称之下的对象必须仅使用实例编号标识符返回。下表中呈现的所有排列组合（以任意顺序）都是有效的，并且必须被支持：

Path Type	Message	Path Name Example
Uniform Instance Number identifier	Request	TopGroup.Lev1Obj.1.Lev2Obj.1.
	Response	TopGroup.Lev1Obj.1.Lev2Obj.1.Lev3Obj.1.Parameter
Uniform Instance Alias identifier	Request	TopGroup.Lev1Obj.[a].Lev2Obj.[b].
	Response	TopGroup.Lev1Obj.[a].Lev2Obj.[b]. Lev3Obj.1.Parameter
Mixed Instance identifier	Request	TopGroup.Lev1Obj.1.Lev2Obj.[b].
	Response	TopGroup.Lev1Obj.1.Lev2Obj.[b]. Lev3Obj.1.Parameter

8. ManagementServer.InstanceMode 参数通过CPE返回参数路径名称的方式，影响CPE的Inform RPC中的ParameterList参数。
   - 如果ManagementServer.InstanceMode参数设置为InstanceNumber，则参数路径名称中的所有对象必须仅使用实例编号标识符。例如：

Path Type	Path Name Example
Uniform Instance Number identifier	TopGroup.Lev1Obj.1.Lev2Obj.1.Parameter

   - 如果ManagementServer.InstanceMode参数设置为InstanceAlias，则在存在实例别名标识符的情况下，参数路径名称中的所有对象必须使用实例别名标识符。例如：

Path Type	Path Name Example
Uniform Instance Alias identifier	TopGroup.Lev1Obj.[a].Lev2Obj.[b].

9. ManagementServer.InstanceMode 参数还影响CPE返回作为路径名称或路径名称列表的Parameter15值的方式。
   - 如果ManagementServer.InstanceMode参数设置为InstanceNumber，则所有作为路径名称或路径名称列表的参数值必须仅使用实例编号标识符返回。例如：

Path Type	Path Name Example
Uniform Instance Number identifier	TopGroup.Lev1Obj.1.Lev2Obj.1.

   - 如果ManagementServer.InstanceMode参数设置为InstanceAlias，则所有作为路径名称或路径名称列表的参数值必须按以下方式返回：
     - 对于那些不是通过SetParameterValues或AddObject由ACS生成的参数值，使用存在的实例别名标识符。例如：

- 对于那些通过SetParameterValues或AddObject由ACS生成的参数值，使用它们被设置时相同的实例标识符类型。例如：

Path Type	Path Name Example
Uniform Instance Alias identifier	TopGroup.Lev1Obj.[cpe-1].Lev2Obj.[cpe-2].

10. CPE在遇到任何需要发出BOOTSTRAP事件的情况时，必须将其ManagementServer.InstanceMode参数更改为其出厂默认值。这样翻译后的内容清晰地说明了根据ManagementServer.InstanceMode参数的不同设置，CPE如何处理和返回路径名称及路径名称列表的参数值，并且指出了在特定事件发生时，CPE应如何重置该参数到出厂默认值。

3.6.2 对象实例通配符要求

在参数名称中使用对象实例通配符对ACS和CPE来说是可选的。它利用了A.2.4中定义的实例通配符。

支持对象实例通配符的ACS必须完全遵守以下所有要求：

- ACS只能与已通过设置“ManagementServer.InstanceWildcardsSupported”参数（设为true）表明支持通配符的CPE一起使用对象实例通配符。
- 在“0 BOOTSTRAP”事件后首次使用对象实例通配符之前，ACS必须查询CPE以获取该参数的值。如果ACS不存储此参数的结果，则在会话中使用通配符之前必须再次查询CPE。

支持对象实例通配符的CPE必须完全遵守以下所有要求：

- CPE必须支持在参数名称中使用实例通配符作为多实例对象的寻址方法，除了实例编号标识符之外。
- CPE必须通过实现“ManagementServer.InstanceWildcardsSupported”参数并将其设置为true来表明其支持。

这样翻译后的文本明确了ACS和CPE在处理对象实例通配符时各自需要遵循的要求，并且指出了如何通过特定参数来表明对通配符的支持。

3.6.3 引用参数要求

以下要求与引用类型及其相关的CPE行为有关：

- CPE必须拒绝尝试设置强引用参数，如果新值不引用现有的参数或对象。
- CPE不得因为新值不引用现有的参数或对象而拒绝设置弱引用参数。
- 当被引用的参数或对象被删除时，CPE必须将非列表值的强引用参数的值更改为null引用。
- 当被引用的参数或对象被删除时，CPE必须从列表值的强引用参数中移除相应的列表项。
- 当被引用的参数或对象被删除时，CPE不得更改弱引用参数的值。

当引用参数包含实例别名（如A.2.2.2节所定义）时，适用以下要求：

- 强引用参数指向实际实例。当某个实例的别名改变，并且存在强引用参数引用了路径中包含该实例的参数或对象时，CPE必须在别名改变后保持这些强引用参数仍然引用相同的实际参数或对象。
- 弱引用参数值以路径名称的形式存储。因此，弱引用参数总是引用当前由存储的路径名称所指向的任何参数或对象（如果有）。这意味着，如果存储的路径名称包括别名，那么任何这些别名的改变都会导致弱引用参数引用不同的参数或对象（或者不引用任何东西）。

3.7 会话流程

所有会话必须以CPE在初始HTTP POST中包含的Inform消息开始。这用于启动一系列事务并传达CPE关于消息编码的限制。在一个会话中，Inform消息不得出现超过一次（此限制不适用于由于作为HTTP认证过程一部分收到的HTTP“401 Unauthorized”状态码或作为HTTP重定向一部分收到的HTTP 3xx状态码而可能需要重新发送Inform请求的情况）。

当ACS和CPE都没有更多请求要发送，并且双方都不再有待响应的消息时，会话结束。此时，CPE必须关闭连接。

CWMP端点与其关联的ACS之间同时只能存在一个会话。

注意——会话仅应在有消息需要在任一方向传输时持续。一旦特定的信息交换完成，会话及其相关的*TCP*连接不应保持打开状态。

3.7.1 CPE 操作

3.7.1.1 会话初始化

CPE将根据3.2.1节列出的条件向ACS发起会话。一旦与ACS的连接成功建立，CPE通过向ACS发送初始Inform请求来启动会话。这表明了CPE当前的状态，并且CPE已经准备好接受来自ACS的请求。

支持CWMP 1.4（或更高版本）的CPE将在每个Inform请求中（在SOAP SupportedCWMPVersions头部）包含一个它所支持的所有CWMP版本号的逗号分隔列表。

- 如果一个支持CMWP 1.4（或更新版本）的CPE在InformResponse中收到了UseCWMPVersion头部，并且该头部包含了CPE在SupportedCWMPVersions头部发送的版本之一，那么CPE必须使用UseCWMPVersion返回的CWMP版本。
- 如果一个支持CWMP 1.4（或更新版本）的CPE在InformResponse中收到了UseCWMPVersion头部，并且该头部包含了CPE没有在SupportedCWMPVersions头部发送的版本，那么CPE必须通过关闭TCP连接来中止会话初始化。
- 如果支持任何CWMP版本的CPE没有收到UseCWMPVersion，CPE必须从ACS返回的InformResponse中的CWMP命名空间推断CWMP版本。表6给出了CPE使用的CWMP命名空间和CWMP版本之间的映射关系。

Table 6 – Inferring ACS CWMP Version 1.0-1.3 from CWMP Namespace

CWMP Namespace	CWMP Version
urn:dslforum-org:cwmp-1-0	1.0
urn:dslforum-org:cwmp-1-1	1.1
urn:dslforum-org:cwmp-1-2	1.2 (there is no way for the CPE to infer 1.3)

在CWMP会话的剩余时间内，CPE不得使用任何与CPE选择的CWMP版本不兼容的功能。

CPE只有在接收到成功的InformResponse时，才应认为会话已成功启动。

从会话开始到会话终止期间，CPE必须确保通过CPE广域网管理协议可访问的所有参数的事务完整性。在整个会话过程中，CPE的所有可配置参数对于ACS来说必须表现为一个一致的集合，且仅由ACS修改。在整个会话中，CPE必须防止ACS看到由其他实体对参数进行的任何更新。这包括可配置参数的值以及可配置参数和对象的存在与否。CPE实现这种事务完整性的方法是本地问题。

CPE必须采取任何必要的步骤以确保会话的事务完整性。例如，在特殊情况下，CPE可能需要终止局域网侧的管理会话，以便满足CWMP会话建立的要求。

3.7.1.2 入站请求

在会话期间（即在会话成功启动后，但在满足3.7.1.4节中描述的会话终止条件之前），当从ACS接收到一个SOAP请求时，CPE必须在其发送给ACS的下一个HTTP POST中响应该请求。

3.7.1.3 出站请求

在会话期间（即在会话成功启动后，但在满足3.7.1.4节中描述的会话终止条件之前），如果CPE有一个或多个请求要发送给ACS，CPE只有在以下所有条件都满足时才可以在下一个HTTP POST中发送其中一个请求：

1. 最近从ACS收到的HTTP响应中不包含SOAP请求。
2. ACS已表明HoldRequests为false（见第3.4.7节）。这个条件仅当最近从ACS收到的HTTP响应中包含以下之一时才满足：
   - 一个带有HoldRequests头部且值为false的SOAP信封。
   - 一个没有HoldRequests头部的SOAP信封。
   - 没有SOAP信封（空的HTTP响应）。
     注意 – HoldRequests SOAP头部元素已被弃用（见第3.4.7节），因此预期ACS不会发送它。但是，ACS仍可能发送它，所以CPE仍然需要支持它。
3. 在当前会话的任何先前时间点，当ACS表明HoldRequests为false时（如上所述），CPE没有发送过空的HTTP POST。

如果在上述条件满足时CPE有待发送的多个请求，除非另有规定，否则选择哪个请求发送由CPE自行决定。

在会话期间，如果上述任一条件不满足，或者CPE没有任何请求要发送给ACS，并且最近从ACS收到的HTTP响应中不包含SOAP请求，那么CPE必须发送一个空的HTTP POST。

一旦CPE在最近的HoldRequests为false的情况下发送了一个空的HTTP POST（见第3.4.7节），CPE在会话剩余时间内不得再发送任何进一步的请求。在这种情况下，如果CPE还有其他请求要发送给ACS，CPE必须等到下一次会话来发送这些请求。

表7总结了在会话进行中（即在会话成功启动后，但在满足3.7.1.4节中描述的会话终止条件之前）CPE必须发送给ACS的内容。

Table 7 – CPE Message Transmission Constraints

	HoldRequests	**ACS request outstanding**	**No ACS request outstanding**
CPE requests pending	**false**	Response	Request
	true	Response	Empty HTTP POST
No CPE requests pending	**-**	Response	Empty HTTP POST

3.7.1.4 会话终止

当满足以下所有条件时，CPE必须终止会话：

1. ACS没有更多的请求要发送给CPE。CPE只有在最近从ACS收到的HTTP响应为空时才能得出这个结论。
2. CPE没有更多的请求要发送给ACS，并且CPE在HoldRequests为false的情况下已经向ACS发送了一个空的HTTP POST（这表明CPE在会话剩余时间内没有更多请求）。根据表7中的定义，如果此条件未满足但CPE没有更多的请求或响应，则它必须发送一个空的HTTP POST，从而满足这一条件。
   注意 – HoldRequests SOAP头部元素已被弃用（见第3.5节），因此预期ACS不会发送它。但是，ACS仍可能发送它，所以CPE仍然需要支持它。
3. CPE已接收到来自ACS的所有未完成的响应消息。
4. CPE已发送了由于先前请求而产生的所有未完成的响应消息到ACS。

如果CPE在本地确定的时间段内（不少于30秒）没有收到来自ACS的HTTP响应，CPE也必须认为会话未成功终止。如果CPE未能接收到HTTP响应，CPE不得尝试在同一会话中重传相应的HTTP请求。

如果CPE在响应Inform请求时收到一个SOAP层故障，且故障代码不是“重试请求”（故障代码8005），CPE必须认为会话未成功终止。

如果CPE从ACS收到的HTTP响应中XML格式不正确、SOAP结构被认为是无效的、包含的SOAP故障不符合第3.5节中指定的形式，或者CPE认为协议被违反，CPE必须认为会话未成功终止。

如果CPE从ACS接收到一个带有故障状态码（4xx或5xx状态码）的HTTP响应，并且这个响应没有被CPE以其他方式处理，CPE必须认为会话未成功终止。需要注意的是，虽然CPE会在正常认证过程中接受带有“401 Unauthorized”状态码的HTTP响应，但当CPE随后尝试认证时，如果结果HTTP响应中包含“401 Unauthorized”状态码，CPE必须认为会话未成功终止。

如果上述条件不满足，CPE必须继续会话。

如果CPE在对除Inform以外的任何方法的响应中收到了第3.5节定义的SOAP层故障响应，且故障代码不是“重试请求”（故障代码8005），CPE必须继续会话的剩余部分。也就是说，这种类型的故障响应不得导致会话未成功终止。

注意 – 在故障情况下，完全由ACS自行决定其故障响应是SOAP层故障（这将导致会话继续）还是HTTP层故障（这将导致会话未成功终止）。

如果会话期间交换的一个或多个消息导致CPE需要重启来完成请求的操作，CPE必须等到根据上述标准会话干净地终止之后再执行重启。

如果会话意外终止，CPE必须按照第3.2.1.1节的规定重试会话。在这种情况下，CPE可以自行规定尝试重新建立会话次数的本地限制。

3.7.1.5 事件

事件是指发生了需要CPE通过A.3.3.1节定义的Inform请求通知ACS的有趣事情的指示。CPE必须至少尝试传递每个事件一次。如果CPE当前没有与ACS处于会话中，它必须立即尝试传递事件；否则，它必须在当前会话终止后尝试传递事件。CPE必须从ACS收到确认才能认为事件已成功传递。一旦CPE成功传递了某个事件，CPE不得再次发送相同的事件。另一方面，ACS必须准备好接收同一个事件多次，因为可能存在ACS发送了响应但CPE从未收到的情况。许多类型的事件（例如，周期性、值变化）即使在之前的会话中已成功传递，也可以合法地出现在后续会话中。在这种情况下，后期会话中的事件表示同类型事件的再次发生，而不是尝试重新传递失败的事件。

对于每种类型的事件，都有一个策略规定如果之前的传递尝试失败，CPE何时以及是否必须重试事件传递。当重试事件传递时，必须在紧随其后的下一个会话中进行；在一个会话中未能传递的事件不能在接下来的会话中被忽略，然后在更晚的时候重新传递。

对于大多数事件，当CPE收到成功的InformResponse时，即确认了事件的传递。六种标准事件类型（KICKED18、TRANSFER COMPLETE、AUTONOMOUS TRANSFER COMPLETE、REQUEST DOWNLOAD、DU STATE CHANGE COMPLETE 和 AUTONOMOUS DU STATE CHANGE COMPLETE）表示在会话稍后将调用一个或多个方法（分别对应于 Kicked [第A.4.2.1节]、TransferComplete [第A.3.3.2节]、AutonomousTransferComplete [第A.3.3.3节]、RequestDownload [第A.4.2.2节]、DUStateChangeComplete [第A.4.2.3节] 和 AutonomousDUStateChangeComplete [第A.4.2.4节]），并且是这些方法的成功响应指示了事件的传递。CPE也可以发送厂商特定的事件（使用表8中指定的语法），在这种情况下，成功传递、重试和丢弃策略遵循厂商定义。

如果在CPE有待重新传递的事件期间没有新的事件发生，CPE必须根据第3.2.1.1节中定义的会话重试策略尝试重新传递这些事件。

下表列出了事件类型、它们在Inform请求中的代码、它们的累积行为、CPE必须收到的响应以认为事件已成功传递，以及如果传递不成功时的重试和/或丢弃策略。

Table 8 – Event Types

【此表请参见 原文档】

上表中的“累积行为”列区分了非累积（“单次”）和累积（“多次”）的事件类型。例如，如果CPE在前一个“1 BOOT”事件尚未传递时重启，那么下一个Inform中包含两个“1 BOOT”事件数组条目是没有意义的。相反，如果在前一个“M Download”事件尚未传递时下载完成，那么下一个Inform将包含两个“M Download”事件数组条目，因为每个条目都与不同的ACS请求相关。"单次"和"多次"累积行为定义如下：

- 如果发生具有“单次”累积行为的事件，则下次Inform中的事件列表必须只包含该EventCode的一个实例，无论是否存在相同类型的未传递事件。
- 如果发生具有“多次”累积行为的事件，则新的EventCode必须被包含在事件列表中，这独立于任何相同类型的未传递事件，并且不应影响这些未传递事件。

当一个或多个事件直接关联于同一个根本原因时，所有此类事件都必须包含在事件数组中。以下是一些这样的情况的例子（此列表并不详尽）：

- 由Reboot RPC方法引起的重启。在这种情况下，Inform必须至少包括以下EventCode值：
  - "1 BOOT"
  - "M Reboot"
- 因先前的Download请求而在新会话中发送TransferComplete，其中没有与传输完成相关的重启：
  - "7 TRANSFER COMPLETE"
  - "M Download"
- 自最近一次Inform以来，设置了被动通知的一个或多个参数值发生了变化，并且发生了周期性Inform（在这种情况下，事件必须包含在同一Inform中，因为对于被动通知，包含“4 VALUE CHANGE”事件的Inform必须是由于其他原因触发的——在这个例子中，是一个周期性Inform）：
  - "2 PERIODIC"
  - "4 VALUE CHANGE"

对于由于不相关原因导致的事件，如果它们同时发生，CPE应该将所有这些事件包含在同一个Inform消息中，但也可以为每个这样的事件发送单独的Inform消息。不相关事件的一个例子是：

- "2 PERIODIC"
- "7 TRANSFER COMPLETE"

3.7.1.6 方法重试行为

如果CPE在响应来自ACS的请求时收到“重试请求”响应（故障代码8005），CPE必须在当前会话中的下一个HTTP POST中重新发送相同的请求。在当前会话中的重试次数必须限制为3次，如果超过了会话重试限制，CPE必须认为会话失败并终止它。此行为适用于所有ACS方法（包括Inform）。

相反，如果CPE在对除Inform以外的任何方法的响应中收到了故障代码不是8005的故障响应，CPE必须继续进行会话，并且不得尝试重试该方法（如在Inform情况下收到这样的响应将终止会话，如3.7.1.4节所述）。