了解 eStreamer 应用协议

Cisco Secure Firewall 系统 Event Streamer (eStreamer) 使用面向消息的协议来将事件和主机配置文件信息通过流传输发送到您的客户端应用。您的客户端可以从管理中心请求事件数据和主机配置文件数据，从受管设备只能请求入侵事件数据。您的客户端应用可以通过提交请求消息（指定要发送的数据）启动数据流，然后在流传输开始后控制来自管理中心或受管设备的消息流。

在本文中，管理中心或受管设备上的 eStreamer 服务可能会称为 eStreamer 服务器或 eStreamer。

以下部分描述连接到 eStreamer 服务的要求，并介绍 eStreamer 协议中使用的命令和数据格式：

- 连接规格，第 2-1 页介绍了 eStreamer 服务与您的客户端之间的通信流，并且介绍了客户端是如何与其进行交互的。
- 了解 eStreamer 通信阶段，第 2-1 页介绍了用于客户端应用向 eStreamer 服务器提交数据请求以及 eStreamer 向客户端传送所请求的信息的通信协议。
- 了解 eStreamer 消息类型，第 2-8 页介绍了 eStreamer 协议中使用的消息类型；讨论了 eStreamer 用于向客户端返回入侵事件数据、发现事件数据、元数据和主机数据的数据包的基本结构；并提供了其他信息来帮助您编写能够解释 eStreamer 消息的客户端。

连接规格

eStreamer 服务：

- 使用 TCP 通过 SSL 连接进行通信（客户端应用必须支持基于 SSL 的身份验证）。
- 接受端口 8302 上的连接请求。
- 等待客户端启动所有通信会话。
- 按网络字节顺序（大端字节）编写所有消息字段。
- 以 UTF-8 格式进行文本编码。

了解 eStreamer 通信阶段

客户端与 eStreamer 服务之间的通信有四个主要阶段：

1. 客户端与 eStreamer 服务器建立连接，并且双方对连接进行身份验证。

   有关详细信息，请参阅建立经过身份验证的连接，第 2-2 页。

2. 客户端向 eStreamer 服务请求数据，并指定要流传输的数据类型。单个事件请求消息可以指定可用事件数据的任意组合，包括事件元数据。单个主机配置文件请求可以指定单个主机或多个主机。

   请求事件数据可采用两种请求模式：

   - 事件流请求 - 客户端提交包含请求标志（指定请求的事件类型和每个类型的版本）的消息，eStreamer 服务器则通过流传输所请求的数据作出响应。
   - 扩展请求 - 客户端提交请求（其消息格式与事件流请求相同），但是设置了扩展请求的标志。这将启动客户端与 eStreamer 服务器之间的消息交互，客户端可向此服务器请求通过事件流请求无法获得的额外信息和版本组合。

   有关请求数据的信息，请参阅向 eStreamer 请求数据，第 2-2 页。

3. eStreamer 与客户端建立请求的数据流。

   有关详细信息，请参阅接受来自 eStreamer 的数据，第 2-7 页。

4. 连接终止。
   有关详细信息， 请参阅终止连接，第 2-7 页。

建立经过身份验证的连接

客户端必须先与eStreamer服务建立支持SSL的TCP连接，才能向eStreamer请求数据。客户端可以在管理中心或受管设备上已配置的任何管理接口上发出请求。客户端连接对管理接口不实施流量信道配置，所以在选择您的连接接口时，可以忽略该配置。当客户端发起连接时，eStreamer 服务器响应，发起与客户端的SSL握手。作为SSL握手的一部分，eStreamer服务器请求客户端身份验证证书，并确认证书是有效的（由 eStreamer 服务器上的内部认证机构 [内部 CA] 签署）。

注释 思科建议您还要求您的客户端确认 eStreamer 服务器提供的证书已由受信任的认证机构签署。这是您向管理中心或受管设备注册新的eStreamer客户端时，思科提供的PKCS#12文件中包含的内部 CA 证书。有关详细信息，请参阅为 eStreamer 客户端添加身份验证，第 6-3 页。

在建立 SSL 会话后，eStreamer 服务器会另外对证书进行一次连接后验证。验证内容包括确认客户端连接是从证书中指定的主机发起的，并且证书的持有者名称包含适当的值。如果任一项连接后检查失败，则eStreamer服务器将关闭连接。必要时，您可以配置eStreamer服务，使其不执行客户端主机名称检查（有关更多信息，请参阅 eStreamer 服务选项，第 6-4 页）。

虽然不要求客户端执行连接后验证，但是，思科仍然建议客户端执行此验证步骤。身份验证证书的持有者名称包含以下字段值：

表 2-1　　证书持有者名称字段

字段	值
title	eStreamer
generationQualifier	server

完成连接后验证之后，eStreamer 服务器会等待客户端发出数据请求。

向 eStreamer 请求数据

您的客户端在管理数据请求时会执行以下高级任务：

- 初始化请求会话 - 请参阅建立会话，第 2-3 页。
- 从 eStreamer 事件存档请求事件 - 使用事件流请求和扩展请求启动事件流传输，第 2-3 页。
- 请求主机数据 - 请参阅请求主机数据，第 2-6 页。
- 更改请求 - 请参阅更改请求，第 2-6 页。
- 请求完全限定事件 - 请参阅 请求完全限定事件，第 2-4 页

建立会话

客户端通过向 eStreamer 服务发送初始事件流请求建立会话。

在此初始消息中，您可以添加数据请求标志，也可以在后续消息中提交数据请求。此初始事件流请求消息本身是所有 eStreamer 请求的前提条件，不管是请求事件数据还是主机数据，都是如此。有关使用事件流请求消息的信息，请参阅事件流请求消息格式，第 2-11 页。

注释 eStreamer 客户端可以在管理中心或受管设备上已配置的任何管理接口上发出请求。客户端连接对管理接口不实施流量信道配置，所以在选择您的连接接口时，可以忽略该配置。

使用事件流请求和扩展请求启动事件流传输

eStreamer 服务提供两种事件流传输的请求模式。您的请求可以组合不同模式。在两种模式中，您的客户端均利用一条事件流请求消息发起请求，但是对请求标志位进行的设置不同。有关事件流消息格式的详细信息，请参阅事件流请求消息格式，第 2-11 页。

当 eStreamer 收到一个事件流请求消息时，它对该客户端请求的处理如下：

- 如果该请求消息**未**设置请求标志字段中的位 30，则 eStreamer 开始流传输该请求标志字段中其他已设置的位请求的任何事件。有关信息，请参阅提交事件流请求，第 2-3 页。
- 如果事件流请求中的位 30 **已**设置，则 eStreamer 提供扩展的请求处理。如果此位已设置，则必须发送扩展请求标志。有关信息，请参阅提交扩展请求，第 2-3 页。请注意，eStreamer 会解决所有重复请求。如果您请求相同数据的多个版本，不管是通过多个标志还是多个扩展请求，系统都会采用最高版本。例如，如果 eStreamer 收到对发现事件版本 1 和 6 的标志请求，以及对版本 3 的扩展请求，则发送版本 6。

提交事件流请求

事件流请求的流程很简单：

- 您的客户端向 eStreamer 服务发送一条请求消息。该消息带有起始日期和时间，以及一个指定要在数据流中包含的事件及其版本级别的请求标志字段。
- eStreamer 通过流传输发送开始于指定时间的事件。有关流传输协议的信息，请参阅接受来自 eStreamer 的数据，第 2-7 页。

有关客户端的事件流请求消息格式和内容的信息，请参阅事件流请求消息格式，第 2-11 页。

有关客户端可以请求的事件的事件类型和版本的信息，请参阅表 2-6，第 2-13 页。

提交扩展请求

如果您在事件流请求消息的请求标志字段中设置了位 30，则您发起一个扩展请求，该请求启动与服务器之间的协商。如果此位已设置，则必须发送扩展请求标志。有关扩展请求可用的事件类型，请参阅表 2-22，第 2-34 页。

扩展请求的步骤如下：

- 您的客户端向 eStreamer 发送一条事件流请求消息，其中的请求标志位 30 设置为 1，表示这是扩展请求。有关消息格式详细信息，请参阅事件流请求消息格式，第 2-11 页。
- eStreamer 回复一条流传输信息消息，通告客户端可用的服务列表。有关流传输信息消息的详细信息，请参阅流传输信息消息格式，第 2-29 页。
- 客户端返回一条流传输请求消息。该消息指明其希望使用的服务，并且包含该服务中可用的事件类型和版本的请求列表。该请求列表对应于进行标准事件流请求时请求标志字段中的设置位。有关如何使用流传输请求消息来请求事件的详细信息，请参阅“扩展请求消息示例”节，第 2-36 页。
- eStreamer 处理客户端的流传输请求消息，并在消息中指定的时间开始流传输数据。有关流传输协议的信息，请参阅接受来自 eStreamer 的数据，第 2-7 页。

请求完全限定事件

我们建议您的客户端使用此选项来请求文本格式（例如 JSON 或 CSV）的完全限定事件，而不是以复杂的二进制格式接收事件。使用此选项时，本文档中介绍二进制格式的大部分内容都无关紧要。在 SDK 包中，python_client 子目录提供使用此选项的示例代码。

此选项当前仅支持请求几种事件类型的信息：连接事件、入侵事件、入侵数据包和文件事件。如果需要以二进制格式接收其他事件类型，则必须为完全限定和二进制事件格式使用单独的客户端连接。

要请求完全限定事件，请使用记录的“事件流请求消息”，并在消息末尾附加 JSON 格式的配置块。该请求将包括如下所示的五个二进制整数，后跟 JSON 格式的配置详细信息，例如：

```
<报头版本 (1)>
<消息类型 (2)>
<消息长度>
<初始时间戳>
<请求标志>
<JSON 格式配置块>
```

二进制消息长度字段必须包括二进制报头的长度以及JSON块的长度。在JSON块之后可以选择空字符终止，但如果包含空字符，则消息长度必须将空字符考虑在内。对于Request Flags字段，仅支持第 23 位（扩展事件报头）；所有其他位应为零，特别是位 30（扩展请求）必须为零。

在客户端发送请求消息后，如果已在服务器端UI eStreamer配置页面上启用了请求的事件类型，则 eStreamer 服务将立即开始发送事件数据。

JSON 文件的格式

此示例也可以在 eStreamer SDK 的 `json_request.json` 文件中找到。

```
{
"Events":
{
"ConnectionEvent":
{
"FieldSetDef":
{
"OutputFieldSet": ["HeaderFieldSet", "ConnectionKeySet", "DetailFieldSet"]
},
"Fields": ["OutputFieldSet"]
},
"IntrusionEvent":
{
"FieldSetDef":
{
"OutputFieldSet": ["HeaderFieldSet", "ConnectionKeySet", "DetailFieldSet", "Impact"]
},
"Fields": ["OutputFieldSet"]
},
"IntrusionPacket":
{
"FieldSetDef":
{
"OutputFieldSet": ["HeaderFieldSet", "DetailFieldSet"]
},
"Fields": ["OutputFieldSet"]
},
"FileEvent":
{
"FieldSetDef":
{
"OutputFieldSet": ["HeaderFieldSet", "ConnectionKeySet", "DetailFieldSet"]
},
"Fields": ["OutputFieldSet"]
}
},
"OutputFormat":
{
"Transform": "Text",
"TransformConfig": "JSON"
}
}
```

在`事件 (Events)`部分中，为您希望客户端接收的每种事件类型指定一个块（仅支持三种示例类型：`ConnectionEvent`、`IntrusionEvent`、 `IntrusionPacket` 和 `FileEvent`）。每个事件的 `FieldSetDef` 部分必须指定一个 `OutputFieldSet`，其中列出将包含在该事件类型的事件中的字段或字段集。示例文件仅指定字段集，但您可以使用字段名称和字段集的任意组合。

每种事件类型的可用字段列表以及预定义字段集可在 Firepower 管理中心的文件 `/etc/sf/EventHandler/EventCatalog/EventCatalog.json` 中找到。在文件末尾的字段部分中，查找所需的事件类型（例如 `IntrusionEvent`），然后查看 `Fields` 和 `FieldSetDef` 块，以查看可用于该事件类型的内容。

OutputFormat 部分包含输出设置。转换 (Transform) 字段始终为文本 (Text)，您可以使用 TransformConfig 字段指定输出转换格式。该示例显示 JSON，但您也可以指定 CSV。其他文本格式以及 FlatBuffer 均可用，但您需要请求这些格式的文档。

在 TransformConfig 中指定 JSON 输出时，输出将包含每个请求的字段的名称-值对，与事件无关的任何字段都将被跳过（例如，如果您请求了 SSL 字段，并且事件未使用 SSL，则输出将不包含这些字段）。

在 TransformConfig 中指定 CSV 输出时，输出将包含按配置中列出的顺序排列的所需字段。如果某个字段与事件无关，则CSV只会包含该字段的逗号。在请求CSV时，不要使用预定义的字段集，因为版本之间的字段集可能会发生变化，从而导致 CSV 不兼容。

完全限定事件消息

事件消息包含在捆绑包中，如 eStreamer 文档中“消息捆绑包格式”消息类型 4002 中所述。

如文档所述，客户端必须通过向 eStreamer 服务器发送一条空消息来确认每个接收到的数据包，从而表明已准备好接受更多数据。

对于所有受支持的事件类型，事件数据消息以二进制报头开头，如“关联记录报头”等 eStreamer文档中所述。唯一的区别在于数据块格式是请求的格式（JSON、CSV等）。基本结构的快速参考如下：

```
<报头版本 (1)>
<消息类型 (3)>
<消息长度>
<记录类型（可应要求包括可选的 Netmap ID）>
<记录长度>
<时间戳（在指定请求位 23 时）>
<保留（在指定请求位 23 时）>
<数据>
```

请求主机数据

建立会话后，您可以随时提交主机数据请求。eStreamer 从 Cisco Secure Firewall 系统网络映射生成有关所请求主机的信息。

更改请求

要更改已建立会话的请求参数，客户端必须断开连接，并请求建立新会话。

接受来自 eStreamer 的数据

注释 eStreamer 服务器不保留其发送的事件的历史记录。您的客户端应用必须检查是否存在重复事件。重复事件可能由于各种原因而意外出现。例如，启动新的流传输会话时，客户端指定为新会话起点的时间可能有多条消息，其中一些消息可能已经在前一个会话中发送，有些则没有。eStreamer 会发送所有符合指定请求条件的消息。您的应用应能够检测出产生的任何重复。

在非活动时段，eStreamer 会定期向客户端发送空消息，使连接保持开启状态。如果它从客户端或中间主机收到错误消息，则会关闭连接。

根据请求模式，eStreamer 以不同方式将请求的数据传输给客户端。

事件流请求

如果客户端提交事件流请求，eStreamer 会逐条消息返回数据。它可以接连发送多条消息，无需等待客户端确认。在某个点，它会暂停并等待客户端。客户端操作系统会缓存收到的数据，让客户端按照自己的节奏处理这些数据。

如果客户端请求中包含元数据请求，则 eStreamer 会先发送元数据。客户端应该将元数据存储在内存中，以便在处理后续事件记录时使用。

扩展请求

如果客户端提交扩展请求，eStreamer 会将消息排队，并捆绑发送。eStreamer 可以接连发送多个捆绑包，无需等待客户端确认。在某个点，它会暂停并等待客户端。客户端操作系统会缓存收到的数据，让客户端按照自己的节奏读取这些数据。

客户端会逐条消息地解开每个捆绑包，然后根据记录和块的长度解析每条消息。可以根据每个消息报头中的总消息长度计算出到达每条消息末尾的时间，根据总捆绑包长度确定到达捆绑包末尾的时间。捆绑包的正确解析并不需要其内容的索引。

有关消息捆绑机制的信息，请参阅消息捆绑包格式，第 2-37 页。

有关客户端能够用于额外流控制的空消息的信息，请参阅空消息格式，第 2-9 页。

终止连接

在关闭连接之前，eStreamer服务器会尝试发送一条错误消息。有关错误消息的信息，请参阅错误消息格式，第 2-10 页。

eStreamer 服务器可出于以下原因关闭客户端连接：

- 在任何时候发送消息产生错误时。这包括事件数据消息以及 eStreamer 在非活动期间发送的空的保持连接消息。
- 处理客户端请求时出错。
- 客户端身份验证失败（不发送错误消息）。
- eStreamer 服务将要关闭（不发送错误消息）。

您的客户端可随时关闭与eStreamer服务器的连接，且应尝试使用错误消息格式告知eStreamer服务器原因。

了解 eStreamer 消息类型

eStreamer 应用协议采用的消息格式很简单：包含标准消息报头和各种子报头字段，后接包含消息负载的记录数据。所有 eStreamer 消息类型都采用相同的消息报头；有关更多信息，请参阅eStreamer 消息报头，第 2-9 页。

表 2-2　　eStreamer 消息类型

消息类型	名称	说明
0	“空消息” (Null message)	eStreamer 服务器和客户端都通过发送空消息来控制数据流。有关信息，请参阅空消息格式，第 2-9 页。
1	“错误消息” (Error message)	eStreamer 服务器和客户端使用错误消息来说明关闭连接的原因。有关信息，请参阅错误消息格式，第 2-10 页。
2	“事件流请求” (Event Stream Request)	客户端向 eStreamer 服务发送此消息类型以启动新的流传输会话和请求数据。有关信息，请参阅事件流请求消息格式，第 2-11 页。
4	“事件数据” (Event Data)	eStreamer 服务使用此消息类型向客户端发送事件数据和元数据。有关信息，请参阅事件数据消息格式，第 2-16 页。
5	“主机数据请求” (Host Data Request)	客户端向 eStreamer 服务发送此消息类型来请求主机数据。必须已经通过事件流请求消息开始会话。有关信息，请参阅主机请求消息格式，第 2-24 页。
6	“单主机数据” (Single Host Data)	eStreamer 服务使用此消息类型发送客户端请求的单主机数据。有关信息，请参阅主机数据和多主机数据消息格式，第 2-28 页。
7	“多主机数据” (Multiple Host Data)	eStreamer 服务使用此消息类型发送客户端请求的多主机数据。有关信息，请参阅主机数据和多主机数据消息格式，第 2-28 页。
2049	“流传输请求” (Streaming Request)	客户端在扩展请求中使用此消息类型来指定其需要流信息消息中的哪些通告事件。有关信息，请参阅扩展请求消息示例，第 2-36 页。
2051	“流传输信息” (Streaming Information)	eStreamer 在扩展请求中使用此消息类型来向客户端通告可用服务列表。有关信息，请参阅流传输信息消息格式，第 2-29 页。
4002	“消息捆绑包” (Message Bundle)	eStreamer 服务使用此消息类型对要通过流传输发送给客户端的消息进行打包。有关信息，请参阅消息捆绑包格式，第 2-37 页。

eStreamer 消息报头

所有 eStreamer 消息都以下图所示的消息报头开始。下表对这些字段进行了说明。

表 2-3　　标准eStreamer消息报头字段

字段	数据类型	说明 (Description)
报头版本 (Header Version)	uint16	表示消息使用的报头的版本。对于当前版本的 eStreamer，此值始终为。
消息类型 (Message Type)	uint16	表示传输的消息的类型。有关当前值的列表，请参阅表 2-2，第 2-8 页。
消息长度 (Message Length)	uint32	表示后续内容的长度，不包括消息报头本身的字节。带有报头但是没有数据的消息的消息长度为零。

空消息格式

客户端应用和 eStreamer 服务都会发送空信息。空消息的类型为 0，在消息报头之后不含数据。

客户端向 eStreamer 服务器发送空消息，表明自己已准备好接受更多数据。不传输数据时，eStreamer 服务向客户端发送空消息，使连接保持活动状态。空消息的消息长度值始终设置为 0。

提示　在本文的数据结构图中，括号内的整数（例如 (1) 或 (115)）表示恒定字段值。例如，报头版本 (1) 表示这里讨论的数据结构中的字段值始终为 1。

空消息格式如下所示。空消息中唯一的非零值为报头版本。

空消息的二进制格式示例如下。请注意，唯一的非零值位于第二个字节，表示报头版本值为 1。消息类型和长度字段（加阴影部分）的值都是 0。

字节	0								1								2								3							
位	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

提示 本指南中的示例以二进制格式显示，以清楚地展示设置了哪些位。这对于某些消息非常重要，例如事件请求消息和事件影响字段。

错误消息格式

客户端应用和 eStreamer 服务都会使用错误信息。错误消息的消息类型为 1，并且包含报头、错误代码、错误文本长度和实际错误文本。错误文本可包含 0 至 65535 个字节。

为客户端应用创建自定义错误消息时，思科建议使用 -1 作为错误代码。

下图说明基本的错误消息格式。加阴影的字段是错误消息所特有的。

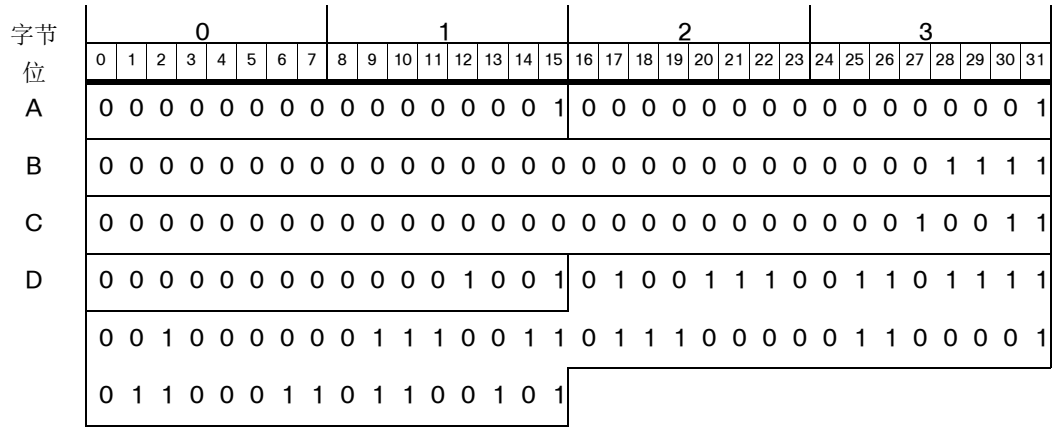
字节	0								1								2								3							
位	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	报头版本 (1) (Header Version (1))																消息类型 (1) (Message Type (1))															
	消息长度 (Message Length)																															
	错误代码 (Error Code)																															
	错误文本长度 (Error Text Length)																错误文本... (Error Text...)															

下表介绍错误代码消息中的每个字段。

表 2-4　　错误消息字段

字段	数据类型	说明 (Description)
错误代码	int32	表示错误的号码。
错误文本长度 (Error Text Length)	uint16	错误文本字段中包含的字节数。
错误文本 (Error Text)	变量	错误消息。最多 65535 字节。

下图显示了一个错误消息示例：



在上一个实例中，出现以下信息：

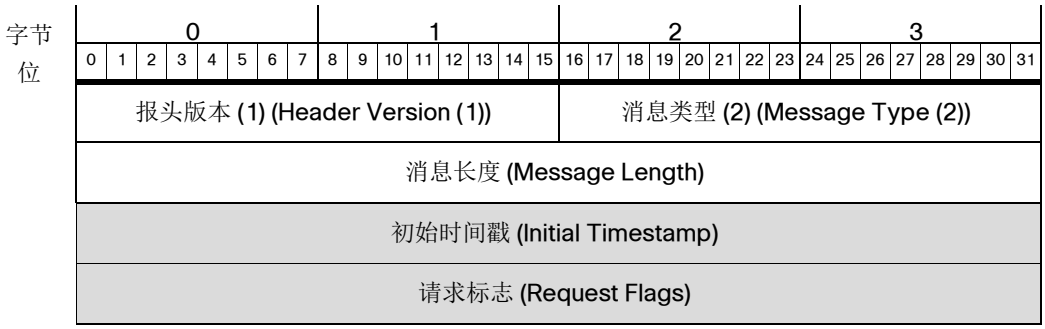
字母	说明
A	前两个字节表示标准报头值 1。接下来的两个字节显示值 1，说明传输的是错误消息。
B	此行表示后面的消息数据量。在本例中，后面是 15个字节（二进制为 1111）的数据。
C	此行显示错误代码。在本例中，该消息包含的值为 19 (10011)。因此，在消息中传输的是错误代码 19。
D	此行包含错误消息中的字节数（1001，或 9 个字节），错误消息本身在接下来的 9 个字节中。错误消息值，转换为 ASCII 文本时，等于“无空间”，这是错误代码为 19 的错误消息。

事件流请求消息格式

eStreamer 客户端使用事件流请求消息开始流传输会话。该请求消息包括开始时间和位标志字段，以此指定 eStreamer 服务应该包含的数据，这些数据可以是事件的任意组合，以及入侵事件额外数据和元数据。事件流请求消息可以发起事件流请求和扩展请求。消息类型为 2。

您必须为所有数据请求（包括专用于主机配置文件信息的请求）提交事件流请求消息。在这种情况下，您首先要提交事件流请求消息，然后提交主机请求消息（类型 5）来指定主机数据。

下图说明事件流请求消息格式。该消息使用标准报头。加阴影的字段是请求消息所特有的，后面的表格对此进行了介绍。



下表介绍事件流请求消息中的每个字段。

表 2-5　事件流请求消息字段

字段	数据类型	说明 (Description)
初始时间戳 (Initial Timestamp)	uint32	定义会话的开始。若要： ▪ 在客户端连接到 eStreamer 时开始，请将所有时间戳位设置为 1。 ▪ 从最早的可用数据开始，请将所有时间戳位设置为 0。 ▪ 在给定日期和时间开始，请指定UNIX时间戳（自 1970年1 月 1 日起经过的秒数）。 有关重要信息，请参阅下文的初始时间戳，第 2-12 页。
请求标志 (Request Flags)	bits[32]	指定要在事件流请求中返回的事件和元数据的类型和版本。有关标志定义，请参阅请求标志，第 2-12 页。 设置位 30 会发起一个扩展请求，该请求可以与事件流请求共存于同一个消息中。

初始时间戳

注释 提交事件流请求时，您的客户端应用应使用“初始时间戳”(Initial Timestamp) 字段中的存档时间戳，如下所述。这可以确保您不会意外地排除事件。设备使用具有传输延迟的“存储和转发”机制将数据传输到管理中心。如果您根据检测到事件的设备分配的生成时间戳来请求事件，则可能会漏掉延迟的事件。

开始会话时，最佳做法是从上一个会话的最后记录的存档时间戳（也称为“服务器时间戳”）开始。这并不是技术要求，但强烈建议这样做。 通过使用上次会话中最后一条记录的存档时间戳，eStreamer 服务不会重新发送之前的记录或元数据。在某些情况下，如果您使用生成时间戳，则您可能意外地将事件排除在新的流传输会话之外。

要将存档时间戳添加到您的流传输事件中，您必须在请求标志字段中设置位 23。

请注意，只有基于时间的事件才带有存档时间戳。如果在请求扩展事件报头时设置了位 23，则 eStreamer 生成的事件（例如元数据）的这个字段值为 0。

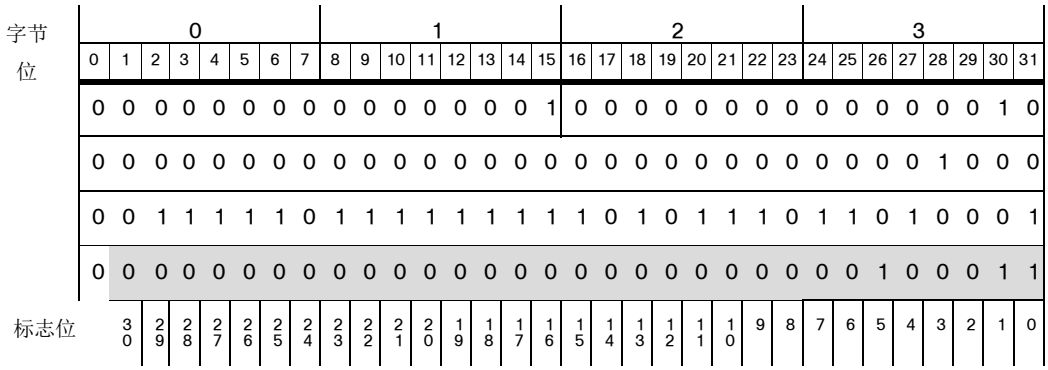
请求标志

在事件数据请求标志字段中设置位0至29，以选择您希望eStreamer发送的事件的类型。设置位 30 可激活扩展请求模式。设置位 30 并不会直接请求任何数据。如果此位已设置，则必须发送扩展请求标志。您的客户端会在提交事件流请求消息后进行的服务器与客户端消息对话中请求数据。有关扩展请求的信息，请参阅向 eStreamer 请求数据，第 2-2 页。

有关请求标志字段中的位设置的定义，请参阅表 2-6，第 2-13页。不同的标志请求不同版本的事件数据。例如，要获取 Cisco Secure Firewall 系统 4.9 格式而不是 4.10 格式的数据，您需要设置不同的标志位。有关在请求特定产品版本的数据时需使用的标志的具体信息，请参阅表 2-7，第 2-15 页。

请注意，您根据版本请求元数据，而不是根据具体元数据记录。有关每个受支持的元数据版本的信息，请参阅请求标志，第 2-12 页。

下图加阴影部分显示当前使用的标志字段中的位：



有关每个请求标志位的信息，请参阅下表。

表 2-6　　请求标志

位字段	说明 (Description)
位 0	请求传输与入侵事件相关的数据包数据。如果设置为 1，数据包数据随入侵事件传输。如果设置为 0，数据包数据不传输。
位 1	请求传输与入侵事件、发现事件、关联事件和连接事件相关的版本 1 元数据。如果设置为 1，版本 1 元数据随事件传输。如果设置为 0，版本 1 元数据不传输。 可以使用元数据解析事件中的编码字段和数字字段。有关 向客户端传输元数据的方式 （续）客户端如何利用元数据的一般信息，请参阅了解元数据，第 2-38 页。
位 2	请求传输入侵事件。如果位 2 和位 6 其中之一或两者都设置为 1，但是扩展请求标志（位 30）设置为 0，则系统将其解释为来自版本 4.x 客户端的请求，并发送记录类型 104/105。如果位 2 和位 6 其中之一或两者都设置为 1 时没有指定事件类型，而位 30 设置为 1，则系统将其解释为来自版本 5.0-5.1 客户端的请求，并发送记录类型 207/208。如果位 30 设置为 1，并且请求了特定的事件类型，则不管位 2 和 6 如何设置，都会发送入侵事件。 有关请求记录类型的详细信息，请参阅提交扩展请求，第 2-3 页。 如果位 2、位 6 和位 30 都设置为 0，则不发送入侵事件。 位 6 的使用方式与位 2 相同。可以将任意一个位设置为请求入侵事件。将这两个位中的一个设置为 0 不会覆盖另一个位；将位 2 设置为 0、位 6 设置为 1，或者将位 2 设置为 1、位 6 设置为 0，都会被解释为请求入侵事件。
位 3	请求传输发现数据版本 1（管理中心 3.2）。如果设置为 0，则不传输发现数据版本 1。 有关发现事件的详细信息，请参阅了解发现和连接数据结构，第 4-1 页。
位 4	请求传输关联数据版本 1（管理中心 3.2）。如果设置为 0，则不传输关联数据版本 1。
位 5	请求传输影响关联事件（入侵影响警报）。如果设置为 1，则传输入侵影响警报。如果设置为 0，则不传输入侵影响警报。 有关入侵影响警报的详细信息，请参阅入侵影响警报数据 5.3+，第 3-19 页。
位 6	位 6 的使用方式与位 2 相同。请参阅位 2，第 2-13 页。
位 7	如果设置为 1，则请求传输发现数据版本 2（管理中心 4.0 - 4.1）。如果设置为 0，则不传输发现数据版本 2。
位 8	如果设置为 1，则请求传输连接数据版本 1（管理中心 4.0 - 4.1）。如果设置为 0，则不发送连接数据版本 1。
位 9	如果设置为 1，则请求传输关联数据版本 2（管理中心 4.0 - 4.1.x）。如果设置为 0，则不传输关联策略数据版本 2。
位 10	如果设置为 1，则请求传输发现数据版本 3（管理中心 4.5 - 4.6.1）。如果设置为 0，则不传输发现数据版本 3。 有关旧版发现事件的详细信息，请参阅旧版发现数据结构，第 B-121 页。
位 11	禁用事件传输。
位 12	如果设置为 1，则请求传输连接数据版本 3（管理中心 4.5 - 4.6.1）。如果设置为 0，则不发送连接数据版本 3。
位 13	请求传输关联数据版本 3（管理中心 4.5 - 4.6.1）。如果设置为 0，则不传输关联数据版本 3。

表 2-6　请求标志（续）

位字段	说明 (Description)
位 14	请求传输与入侵事件、发现事件、关联事件和连接事件相关的版本 2 元数据。如果设置为 1，版本 2 元数据随事件传输。如果设置为 0，版本 2 元数据不传输。 有关 向客户端传输元数据的方式管理中心客户端如何利用元数据的一般信息，请参阅了解元数据，第 2-38 页。
位 15	请求传输与入侵事件、关联事件、发现事件和连接事件相关的版本 3 元数据。如果设置为 1，版本 3 元数据随事件传输。如果设置为 0，版本 3 元数据不传输。 有关 向客户端传输元数据的方式客户端如何利用元数据的一般信息，请参阅了解元数据，第 2-38 页。
位 16	未使用
位 17	请求传输发现数据版本 4（管理中心 4.7-4.8.x）。如果设置为 0，则不传输发现数据版本 4。
位 18	如果设置为 1，则请求传输连接数据版本 4（管理中心 4.7 - 4.9.0.x）。如果设置为 0，则不发送连接数据版本 4。 有关详细信息，请参阅连接区块消息，第 4-52 页。
位 19	请求传输关联数据版本 4（管理中心 4.7）。如果设置为 0，则不传输关联数据版本 4。 有关以管理中心 4.7 格式传输的关联事件的信息，请参阅旧版关联事件数据结构，第 B-347 页。
位 20	请求传输与入侵事件、发现事件、用户活动事件、关联事件和连接事件相关的版本 4 元数据。如果设置为 1，版本 4 元数据随事件传输。如果设置为 0，版本 4 元数据不传输。 版本 4 元数据包括： ▪ 关联（合规性）规则信息 ▪ 关联（合规性）策略信息 ▪ 指纹记录 ▪ 客户端应用记录 ▪ 客户端应用类型记录 ▪ 漏洞记录 ▪ 主机重要性记录 ▪ 网络协议记录 ▪ 主机属性记录 ▪ 扫描类型记录 ▪ 用户记录 ▪ 服务检测设备（版本 2）记录 ▪ 事件分类（版本 2）记录 ▪ 优先级记录 ▪ 规则信息（版本 2） ▪ 恶意软件信息 如果同时请求位 20 和位 22，则也会发送用户元数据。 有关 向客户端传输元数据的方式客户端如何利用元数据的一般信息，请参阅了解元数据，第 2-38 页。
位 21	请求传输版本 1 用户事件。有关用户事件的详细信息，请参阅用户记录，第 4-19 页。
位 22	请求传输关联数据版本 5（管理中心 4.8.0.2 - 4.9.1）。如果设置为 0，则不传输关联数据版本 5。 如果同时请求位 20 和位 22，则也会发送用户元数据。 有关旧版关联（合规性）事件的详细信息，请参阅旧版关联事件数据结构，第 B-347 页。
位 23	请求扩展事件报头。如果设置为 1，则随事件一起传输事件被存档（以便 eStreamer 服务器处理）时的时间戳，并且保留四个字节供未来使用。如果此字段设置为 0，则随事件一起发送仅包含记录类型和记录长度的标准事件报头。 有关事件消息报头的信息，请参阅eStreamer 消息报头，第 2-9 页。
位 24	请求传输发现数据版本 5（管理中心 4.9.0.x）。如果设置为 0，则不传输发现数据版本 5。 有关发现事件的详细信息，请参阅了解发现和连接数据结构，第 4-1 页。
位 25	请求传输发现数据版本 6（管理中心 4.9.1+）。如果设置为 0，则不传输发现数据版本 6。 有关发现事件的详细信息，请参阅了解发现和连接数据结构，第 4-1 页。

表 2-6　　请求标志 （续）

位字段	说明 (Description)
位 26	如果设置为 1，则请求传输连接数据版本 5（管理中心 4.9.1 - 4.10.x）。如果设置为 0，则不发送连接数据版本 5。 有关详细信息，请参阅连接区块消息，第 4-52 页。
位 27	请求额外数据记录中与入侵事件相关联的事件额外数据。 有关事件数据的详细信息，请参阅表 B-11入侵事件额外数据数据块字段，第 B-67 页。
位 28	请求传输发现数据版本 7（管理中心 4.10.0+）。如果设置为 0，则不传输发现数据版本 7。 有关发现事件的详细信息，请参阅了解发现和连接数据结构，第 4-1 页。
位 29	请求传输关联数据版本 6（管理中心 4.10 - 4.10.x）。如果设置为 0，则不传输关联策略数据版本 6。 如果同时请求位 20 和位 29，则也会发送用户元数据。 有关关联事件的详细信息，请参阅该产品的早期版本。
位 30	表示向 eStreamer 发出的一个扩展请求。如果此位已设置，则必须发送扩展请求标志。有关扩展请求的信息，请参阅提交扩展请求，第 2-3 页。

为了帮助您决定使用哪些标志来请求特定版本的数据，请参阅下表。对于版本 5.0 及更高版本，请参阅提交扩展请求，第 2-3 页了解有关使用位 30 的更多信息。

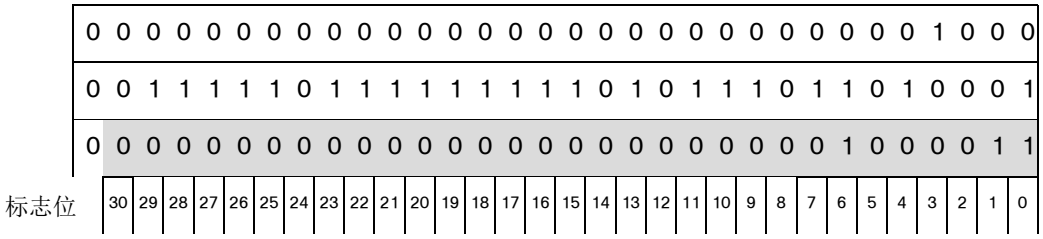
表 2-7　　按产品版本划分的事件请求标志

请求的数据的类型	4.9.0.x	4.9.1.x	4.10.x	5.0+	5.1	5.1.1+
数据包数据	位 0	位 0	位 0	位 0	位 0	位 0
入侵事件	位 2	位 2	位 2	位 2	位 2	位 30
元数据	位 20	位 20	位 20	位 20	位 20	位 20
发现事件	位 24	位 25	位 28	位 30	位 30	位 30
关联事件	位 22	位 22	位 29	位 30	位 30	位 30
事件额外数据	-	-	位 27	位 27	位 27	位 27
影响事件警报	位 5	位 5	位 5	位 5	位 5	位 5
连接数据	位 18	位 26	位 26	位 30	位 30	位 30
用户事件	位 21	位 21	位 21	位 30	位 30	位 30
恶意软件事件	-	-	-	-	-	位 30
文件事件	-	-	-	-	-	位 30

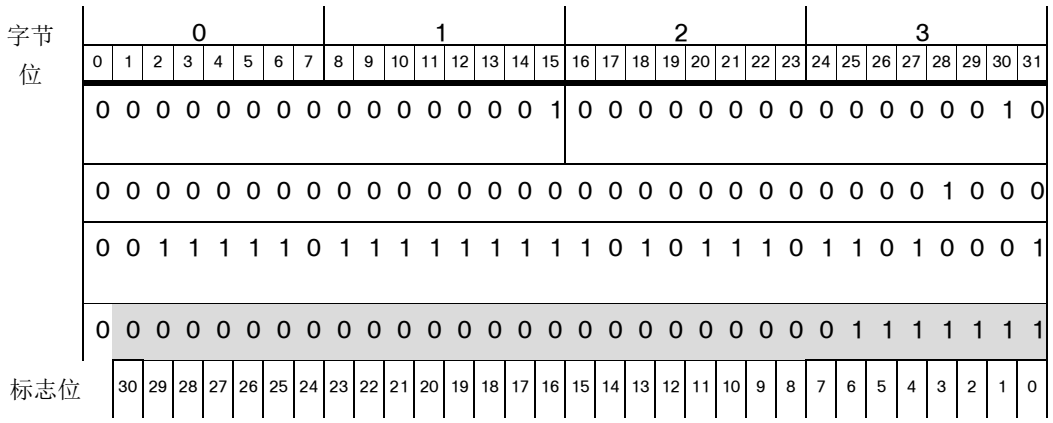
小心　在所有事件类型中，在版本 5.x 之前，标准客户端都将 `detection engine ID` 字段标记为 `sensor ID`。

以下示例请求类型为 7 的入侵事件（与 Cisco Secure Firewall 系统 3.2+ 兼容）以及版本 1 元数据和数据包标志：

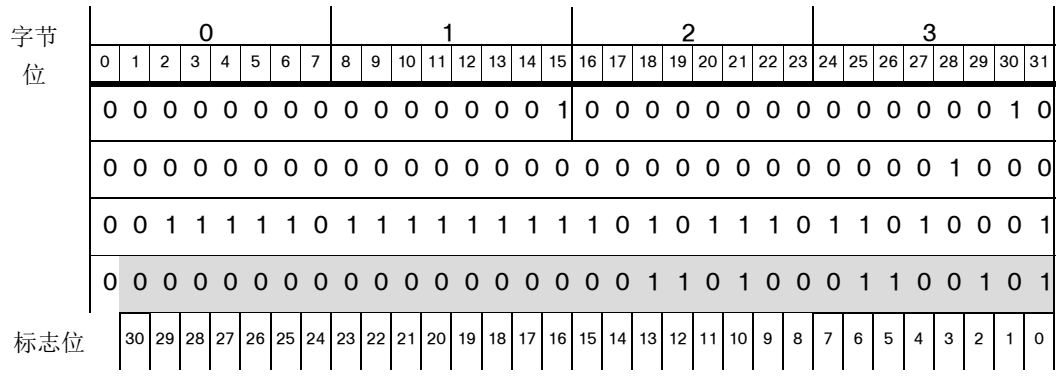
字节	0								1								2								3							
位	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0



要仅请求与 Cisco Secure Firewall 系统 3.2 兼容的数据（包括入侵事件、数据包、元数据、影响警报、策略违规事件和版本 2.0 事件），则使用：



要请求类型为 7 的入侵影响警报、关联事件、发现事件、连接事件和入侵事件，以及数据包和管理中心 4.6.1+ 格式的版本 3 元数据，则使用：



事件数据消息格式

在收到事件请求时，eStreamer 服务会将事件数据及相关元数据传输到客户端。事件数据消息的消息类型为 3。每条消息都包含单个带有事件数据或元数据的数据记录。

请注意，类型 3 消息只传送事件数据和元数据。eStreamer 以类型 6（单主机）和类型 7（多主机）消息传输主机信息。有关主机消息格式的信息，请参阅主机数据和多主机数据消息格式，第 2-28 页。

了解事件数据消息的组织

eStreamer 发送的事件数据和元数据消息包含以下部分：

- eStreamer 消息报头 - eStreamer 消息报头，第 2-9 页上定义的标准消息报头。
- 特定于事件的子报头 - 因事件类型而异的字段组，带有描述额外事件详细信息并确定后续负载数据结构的代码。
- 数据记录 - 多个固定长度的字段和一个数据块。

注释　客户端应根据字段长度对所有消息进行解包。

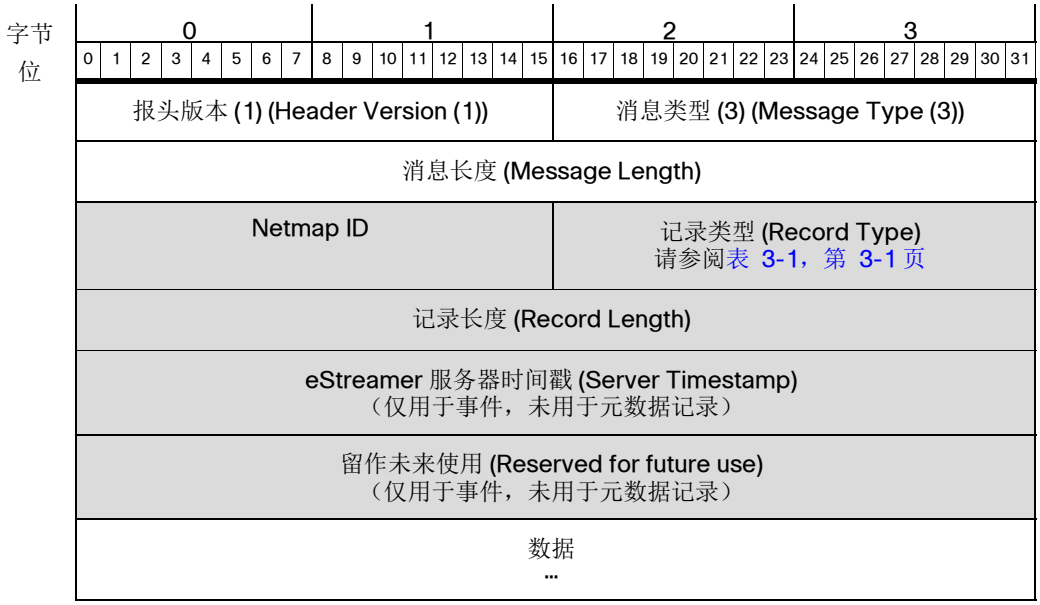
有关根据事件类型划分的事件消息格式，请参阅：

- 入侵事件和元数据消息格式，第 2-17页，了解入侵事件数据记录和所有元数据记录。这些消息含有固定长度字段。
- 发现事件消息格式，第 2-19 页，了解含有发现事件或用户事件数据的消息。除了有标准 eStreamer 消息报头和与入侵事件消息相似的记录报头外，发现消息还有一个与众不同的含有事件类型和子类型字段的发现事件报头。发现事件消息中的数据记录打包在系列 1 数据块中。该数据块可包含可变长度字段和多层封装数据块。
- 连接事件消息格式，第 2-20页，了解带有连接统计信息的消息。连接事件消息的一般结构与发现事件消息相同。但是它们的数据块类型是特定于连接统计信息的。
- 关联事件消息格式，第 2-20 页，了解带有关联（合规性）事件数据的消息。此类消息中的报头与入侵事件消息中的报头相同，但数据块是系列1 数据块。
- 事件额外数据消息格式，第 2-22 页，了解传输带有可变长度字段和多层嵌套数据块（例如入侵事件额外数据）的入侵相关记录类型的一系列消息。有关此消息系列的结构的一般信息，请参阅事件额外数据消息格式，第 2-22 页。有关此系列数据块结构（类似于系列 1 数据块，但是单独编号）的信息，请参阅数据块报头，第 2-23 页。

入侵事件和元数据消息格式

下图显示了入侵事件和元数据消息的一般结构。

下图显示了入侵事件和元数据消息格式的记录报头部分的详细信息。加阴影部分为记录报头字段。后面的表格定义这些字段。



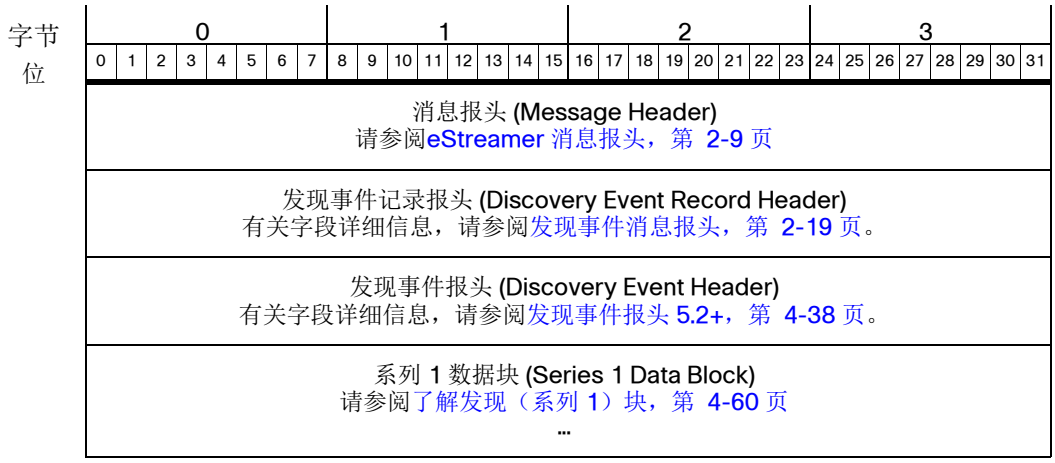
下表介绍入侵事件和元数据消息报头中的每个字段。

表 2-8　　入侵事件和元数据记录报头字段

字段	数据类型	说明 (Description)
Netmap ID	uint16	此字段的第一位是一个标志，表示该报头是否为含有存档时间戳的扩展报头。其余 15 位是一个可选字段，包含在其上检测到事件的域的NetmapID。如果不使用此字段，则字段显示为空。NetmapID 根据元数据中的规定映射到域。
记录类型 (Record Type)	uint16	确定数据记录内容类型。有关记录类型的列表，请参阅表 3-1入侵事件与一般元数据记录类型，第 3-1 页。
记录长度 (Record Length)	uint32	记录报头后面的消息内容长度。不包括记录报头的 8 或 16 个字节。（记录长度加上记录报头的长度等于消息长度。）
eStreamer 服务器时间戳 (eStreamer Server Timestamp)	uint32	表示事件被 eStreamer 服务器存档时的时间戳。也称为存档时间戳。 只有设置了请求消息标志中的位 23，才存在此字段。
留作未来使用 (Reserved for future use)	uint32	已保留供将来使用。 只有设置了请求消息标志中的位 23，才存在此字段。

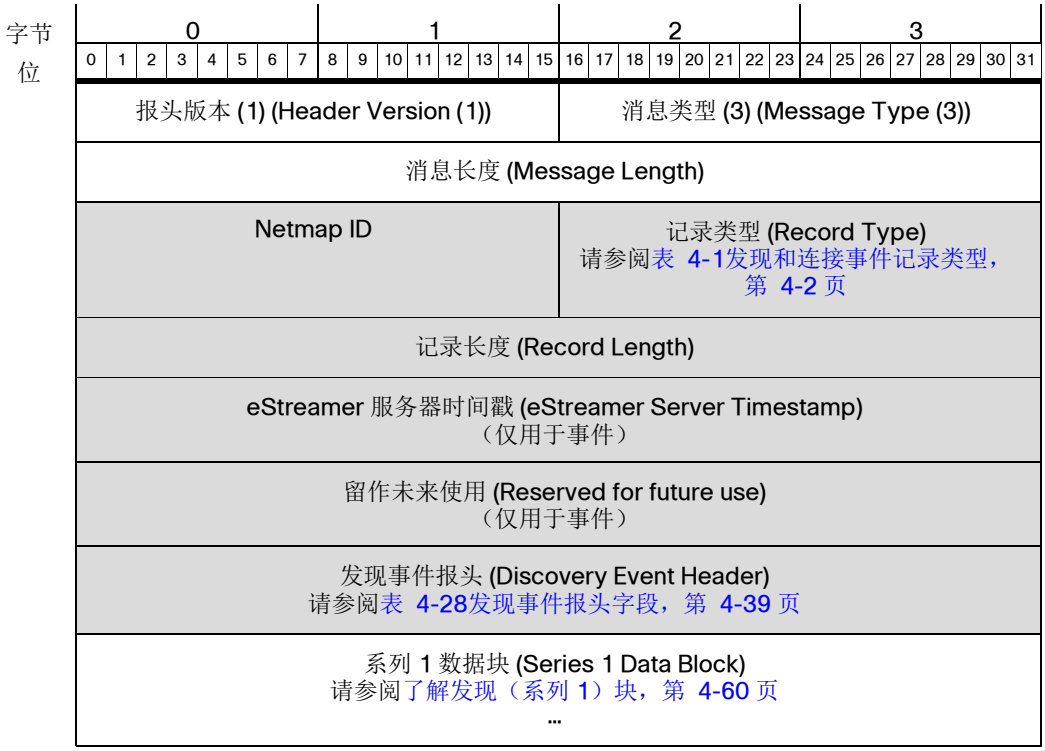
发现事件消息格式

下图显示了发现事件消息的结构。标准 eStreamer 消息报头和事件记录报头后面是仅在发现和用户事件消息中使用的发现事件报头。消息的发现事件报头部分包含发现事件类型和子类型字段。这些字段在一起构成后面的数据块的密钥。有关当前发现事件类型和子类型的信息，请参阅表 4-29按类型和子类型划分的发现与连接事件，第 4-40 页。



发现事件消息报头

下图中的加阴影部分显示了发现事件数据消息格式中的记录报头的字段，并且显示了后面的事件报头的位置。下表定义发现事件消息报头中的字段。



下表介绍发现事件消息的记录报头和事件报头中的字段。

表 2-9　　发现事件消息报头字段

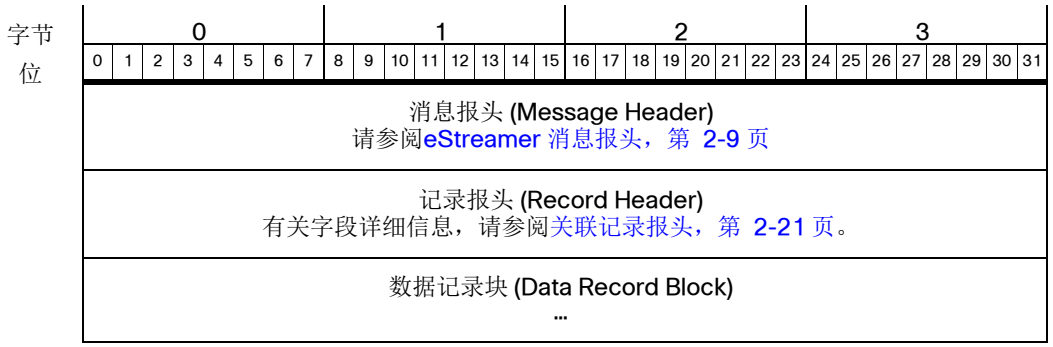
字段	数据类型	说明 (Description)
Netmap ID	uint16	此字段的第一位是一个标志，表示该报头是否为含有存档时间戳的扩展报头。其余 15 位是一个可选字段，包含在其上检测到事件的域的 Netmap ID。如果不使用此字段，则字段显示为空。Netmap ID 根据元数据中的规定映射到域。
记录类型 (Record Type)	uint16	确定数据记录内容类型。有关记录类型的列表，请参阅表 4-1发现和连接事件记录类型，第 4-2 页。
记录长度 (Record Length)	uint32	记录报头后面的消息内容长度。不包括记录报头的 8 或 16 个字节。（记录长度加上记录报头的长度等于消息长度。）
eStreamer 服务器时间戳 (eStreamer Server Timestamp)	uint32	表示事件被 eStreamer 服务器存档时的时间戳。也称为存档时间戳。只有在事件流请求的请求标志字段中设置了位 23，才存在此字段。
留作未来使用 (Reserved for future use)	uint32	已保留供将来使用。只有设置了请求消息标志中的位 23，才存在此字段。
发现事件报头 (Discovery Event Header)	视情况而定	包含大量字段，其中包括事件类型和子类型字段，这些字段在一起构成后面的数据结构的密钥。有关发现事件报头中的字段的定义，请参阅发现事件报头 5.2+，第 4-38 页。

连接事件消息格式

带有连接统计信息的消息的结构与发现事件消息相同。有关一般消息格式信息，请参阅发现事件消息格式，第 2-19 页。连接事件消息具有不同的数据块类型。

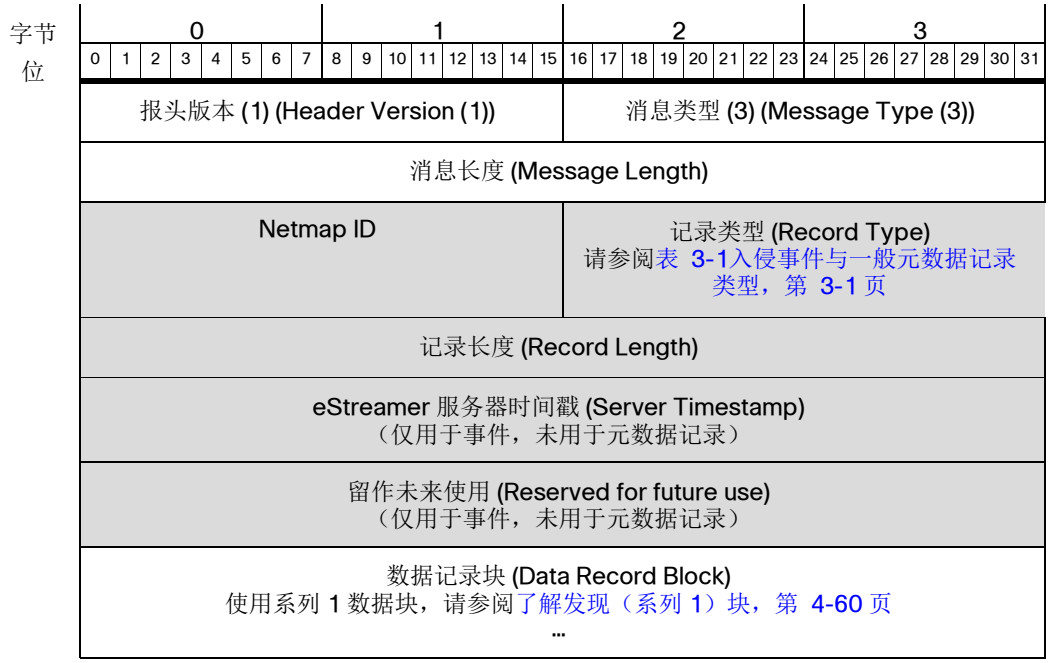
关联事件消息格式

下图显示了关联（合规性）事件消息的一般结构。标准 eStreamer 消息报头和记录报头后面紧跟着消息的数据记录部分中的数据块。关联消息使用系列 1 数据块。



关联记录报头

下图中的加阴影部分显示了关联事件消息中记录报头的字段。请注意，关联消息使用系列 1 数据块；但是它们没有发现事件消息中存在的发现报头。它们的报头字段与入侵事件消息的相似。下图后面的表格定义关联事件的记录报头字段。



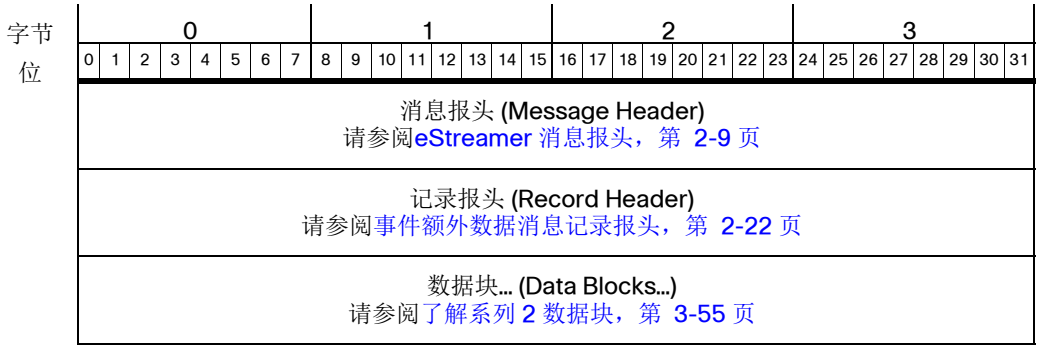
下表介绍关联事件消息的记录报头中的每个字段。

表 2-10　　关联事件消息记录报头字段

字段	数据类型	说明 (Description)
Netmap ID	uint16	此字段的第一位是一个标志，表示该报头是否为含有存档时间戳的扩展报头。其余 15 位是一个可选字段，包含在其上检测到事件的域的Netmap ID。如果不使用此字段，则字段显示为空。Netmap ID 根据元数据中的规定映射到域。
记录类型 (Record Type)	uint16	确定数据记录内容类型。有关入侵、关联和元数据记录类型的列表，请参阅表 3-1，第 3-1 页。
记录长度 (Record Length)	uint32	记录报头后面的消息内容长度。不包括记录报头的 8 或 16 个字节。（记录长度加上记录报头的长度等于消息长度。）
eStreamer 服务器时间戳 (eStreamer Server Timestamp)	uint32	表示事件被 eStreamer 服务器存档时的时间戳。也称为存档时间戳。 只有设置了请求消息标志中的位 23，才存在此字段。 对于管理中心生成的数据（例如主机配置文件和元数据），此字段为零。
留作未来使用 (Reserved for future use)	uint32	已保留供将来使用。 只有设置了请求消息标志中的位 23，才存在此字段。

事件额外数据消息格式

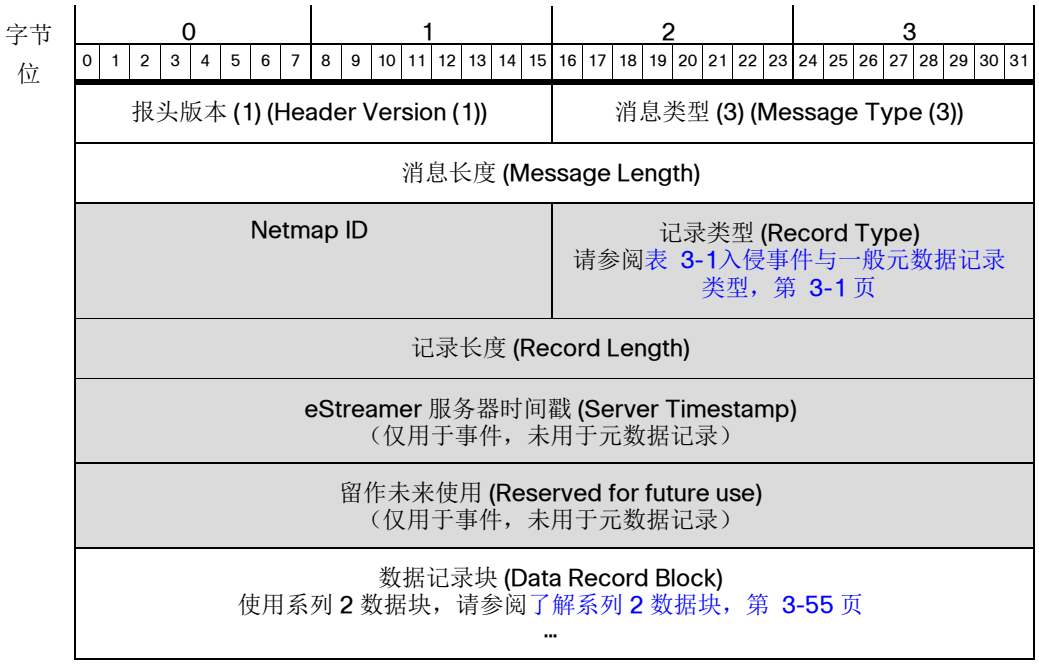
下图显示了事件额外数据消息的结构。此消息组的一个示例是入侵事件额外数据消息。



事件额外数据消息的格式与关联事件消息相同，在记录报头后面直接跟着数据块。和关联消息不同的是，它们使用具有单独编号顺序的系列 2 数据块而不是系列 1 数据块。有关系列 2 数据块类型的信息，请参阅了解系列 2 数据块，第 3-55 页。

事件额外数据消息记录报头

下图中的加阴影部分显示了事件额外数据消息中的记录报头的字段。后面的表格定义事件额外数据消息的记录报头字段。



下表介绍事件额外数据消息的记录报头中的每个字段。

表 2-11　事件额外数据消息记录报头字段

字段	数据类型	说明 (Description)
Netmap ID	uint16	此字段的第一位是一个标志，表示该报头是否为含有存档时间戳的扩展报头。其余 15 位是一个可选字段，包含在其上检测到事件的域的 Netmap ID。如果不使用此字段，则字段显示为空。Netmap ID 根据元数据中的规定映射到域。
记录类型 (Record Type)	uint16	确定数据记录内容类型。有关事件额外数据记录类型的列表，请参阅表 3-1入侵事件与一般元数据记录类型，第 3-1 页。
记录长度 (Record Length)	uint32	记录报头后面的消息内容长度。不包括记录报头的 8 或 16 个字节。（记录长度加上记录报头的长度等于消息长度。）
eStreamer 服务器时间戳 (eStreamer Server Timestamp)	uint32	表示事件被 eStreamer 服务器存档时的时间戳。也称为存档时间戳。 只有设置了请求消息标志中的位 23，才存在此字段。对于管理中心生成的事件，不存在此字段。
留作未来使用 (Reserved for future use)	uint32	已保留供将来使用。 只有设置了请求消息标志中的位 23，才存在此字段。对于管理中心生成的事件，不存在此字段。

数据块报头

系列 1 数据块和系列 2 数据块具有类似结构，但编号不同。这些数据块可以出现在发现、关联、连接或事件额外数据消息的数据部分中的任何位置。这些数据块在多个嵌套层级上封装其他数据块。

第一和第二系列的数据块都以下图中显示的报头结构开始。后面的表格提供有关报头字段的信息。报头后面紧跟着与数据块类型相关的数据结构。

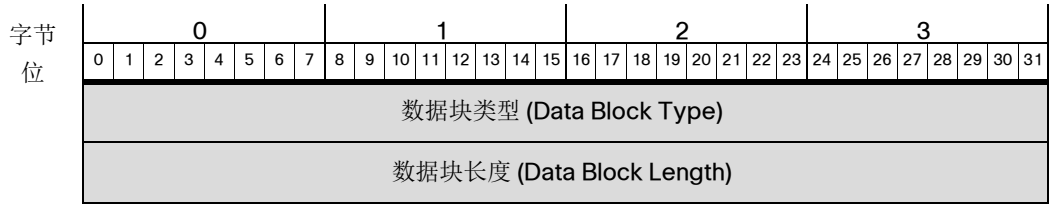


表 2-12

字段	数据类型	说明 (Description)
数据块类型 (Data Block Type)	uint32	对于系列 1 数据块类型，请参阅了解发现（系列 1）块，第 4-60 页。 对于系列 2 数据块类型，请参阅表 3-24系列 2 块类型，第 3-55 页。
数据块长度 (Data Block Length)	uint32	数据块的长度。包括数据字节数加上两个数据块报头字段中的

主机请求消息格式

要接收主机配置文件，您需提交主机请求消息。可以请求单个主机或 IP 地址范围定义的多个主机的数据。

请注意，对于所有数据请求（包括主机配置文件信息请求），都必须先通过提交事件流请求消息来初始化会话。要设置仅流传输主机数据，您可以在您的初始事件流请求消息中使用以下任意一个请求标志设置：

- 设置表示适当版本的元数据的位（这可能有助于流传输主机数据）
- 设置无请求标志
- 设置位 11（在使用旧版 eStreamer 时禁用任何默认事件流传输）

在初始消息后，使用主机请求消息（类型 5）以指定主机。

注释 对于带有默认事件流传输的旧版 eStreamer，如果想仅传输主机配置文件数据，则需要禁用默认事件消息。首先向服务器发送事件流请求消息（请求标志字段中的位 11 设置为 1）；然后，发送主机请求消息。

下图显示了主机请求消息的格式。加阴影的字段是主机请求消息格式特有的，后面的表格给出了这些字段的定义。前面的三个字段是标准消息报头。

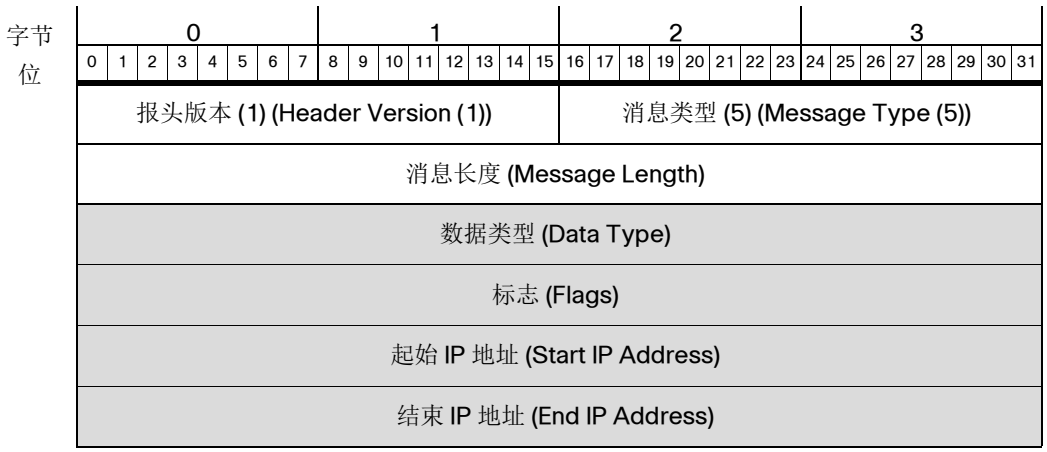
字节	0								1								2								3							
位	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
报头版本 (1) (Header Version (1))																	消息类型 (5) (Message Type (5))															
消息长度 (Message Length)																																
数据类型 (Data Type)																																
标志 (Flags)																																
起始 IP 地址 (Start IP Address)																																
起始 IP 地址 (Start IP Address)（续）																																
起始 IP 地址 (Start IP Address)（续）																																
起始 IP 地址 (Start IP Address)（续）																																
结束 IP 地址 (End IP Address)																																
结束 IP 地址 (End IP Address)（续）																																
结束 IP 地址 (End IP Address)（续）																																
结束 IP 地址 (End IP Address)（续）																																

下表对消息字段进行了说明。

表 2-13　主机请求消息字段

字段	数据类型	说明 (Description)
数据类型 (Data Type)	uint32	使用下列代码请求单个主机或多个主机的数据： ▪ 0 — 版本 3.5-4.6 数据，单主机。 ▪ 1 — 版本 3.5-4.6 数据，多主机（使用数据块 34）。 ▪ 2 — 版本 4.7-4.8 数据，单主机（使用数据块 47）。 ▪ 3 — 版本 4.7-4.8 数据，多主机（使用数据块 47）。 ▪ 4 — 版本 4.9 - 4.10 数据，单主机（使用数据块 92）。 ▪ 5 — 版本 4.9 - 4.10 数据，多主机（使用数据块 92）。 ▪ 6 — 版本 5.0.x .x 数据，单主机（使用数据块 111，请参阅完整主机配置文件数据块 5.0 - 5.0.2，第 B-363 页）。 ▪ 7 — 版本 5.0.x 数据，多主机（使用数据块 111，请参阅完整主机配置文件数据块 5.0 - 5.0.2，第 B-363 页）。 ▪ 8 — 版本 5.1.x 数据，多主机（使用数据块 111，请参阅完整主机配置文件数据块 5.1.1，第 B-372 页）。 ▪ 9 — 版本 5.1.x 数据，多主机（使用数据块 111，请参阅完整主机配置文件数据块 5.1.1，第 B-372 页）。 ▪ 10 — 规则文档数据（使用数据块 27，请参阅规则文档消息格式，第 2-27 页） ▪ 11 — 版本 5.2x 数据，多主机（使用数据块 111，请参阅完整主机配置文件数据块 5.2.x，第 B-381 页）。 ▪ 12 — 版本 5.2.x 数据，多主机（使用数据块 111，参阅完整主机配置文件数据块 5.2.x，第 B-381 页）。 ▪ 13 — 版本 5.3+ 数据，多主机（使用数据块 111，参阅完整主机配置文件数据块 5.3+，第 5-1 页）。 ▪ 14 — 版本 5.3+ 数据，多主机（使用数据块 111，参阅完整主机配置文件数据块 5.3+，第 5-1 页）。
标志 (Flags)	32 位字段	▪ 0x00000001 - 使主机配置文件的“注释”字段填充（有关 Cisco Secure Firewall 系统中存储的主机的用户定义信息）。 ▪ 0x00000002 - 使服务块的“横幅”字段填充（为服务检测到的第一个数据包的前 256 个字节）。默认禁用横幅，只有配置后才能使用。
起始 IP 地址 (Start IP Address)	uint8[16]	应返回其数据的主机的 IP 地址（如果请求针对单主机），或 IP 地址范围的起始地址（如果请求针对多主机）。可以是 IPv4 或 IPv6 地址。
结束 IP 地址 (End IP Address)	uint8[16]	IP 地址范围的结束地址（如果请求针对多主机），或起始 IP 地址的值（如果请求针对单主机）。可以是 IPv4 或 IPv6 地址。

下图显示了旧版主机请求消息的格式。eStreamer 仍会响应此请求。与当前请求的唯一区别是 IPv4 地址字段较小。加阴影的字段是主机请求消息格式特有的，后面的表格给出了这些字段的定义。前面的三个字段是标准消息报头。



下表对消息字段进行了说明。

表 2-14 主机请求消息字段

字段	数据类型	说明 (Description)
数据类型 (Data Type)	uint32	使用下列代码请求单个主机或多个主机的数据： ▪ 0 - 版本 3.5-4.6 数据，单主机。 ▪ 1 - 版本 3.5-4.6 数据，多主机（使用数据块 34）。 ▪ 2 - 版本 4.7-4.8 数据，单主机（使用数据块 47）。 ▪ 3 - 版本 4.7-4.8 数据，多主机（使用数据块 47）。 ▪ 4 - 版本 4.9 - 4.10 数据，单主机（使用数据块 92）。 ▪ 5 - 版本 4.9 - 4.10 数据，多主机（使用数据块 92）。 ▪ 6 - 版本 5.0+ 数据，单主机（使用数据块 111，参阅完整主机配置文件数据块 5.3+，第 5-1 页）。 ▪ 7 - 版本 5.0+ 数据，多主机（使用数据块 111，参阅完整主机配置文件数据块 5.3+，第 5-1 页）。
标志 (Flags)	32 位字段	▪ 0x00000001 - 使主机配置文件的“注释”字段填充（有关 Cisco Secure Firewall 系统中存储的主机的用户定义信息）。 ▪ 0x00000002 - 使服务块的“横幅”字段填充（为服务检测到的第一个数据包的前 256 个字节）。默认禁用横幅，只有配置后才能使用。
起始 IP 地址 (Start IP Address)	uint8[4]	应返回其数据的主机的 IP 地址（如果请求针对单主机），或 IP 地址范围的起始地址（如果请求针对多主机）。以 IP 地址八位字节指定地址。
结束 IP 地址 (End IP Address)	uint8[4]	IP 地址范围的结束地址（如果请求针对多主机），或起始 IP 地址的值（如果请求针对单主机）。

规则文档消息格式

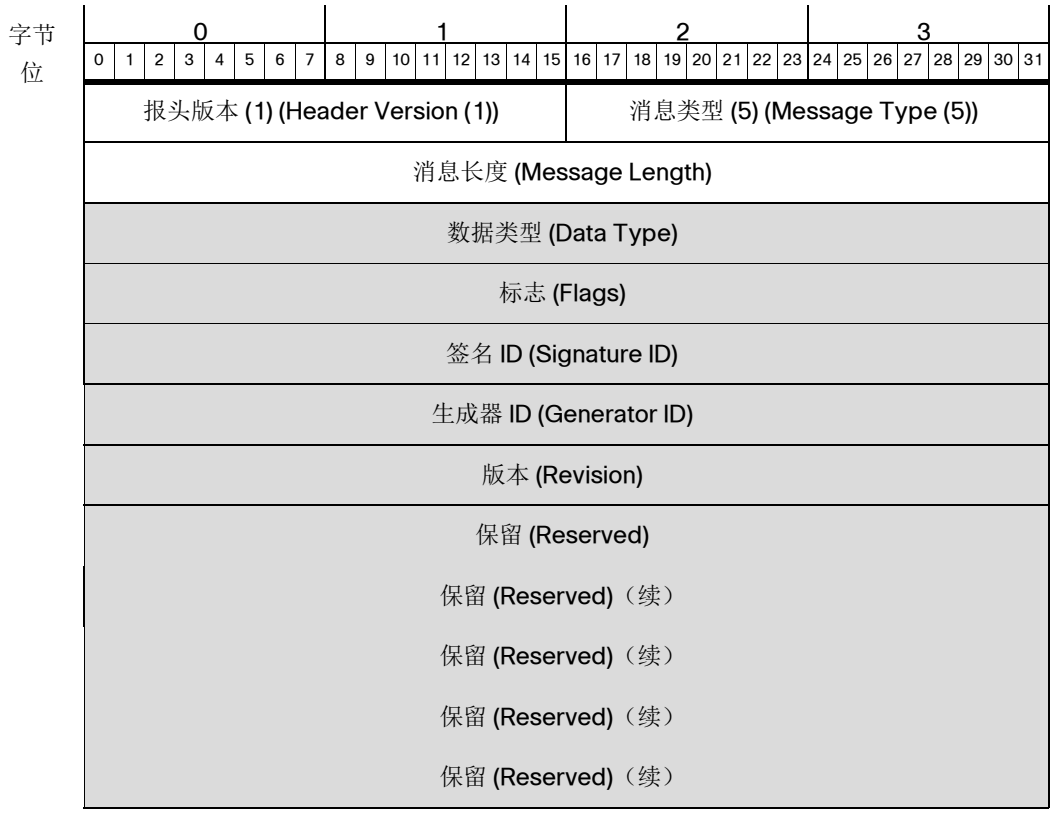
要接收规则文档配置文件，您需要提交规则文档消息。您可以按生成器 和版本请求这些规则文档配置文件。

请注意，对于所有数据请求（包括规则文档信息请求），都必须先通过提交事件流请求消息来初始化会话。要设置仅流传输主机数据，您可以在您的初始事件流请求消息中使用以下任意一个请求标志设置：

- 设置表示适当版本的元数据的位（这可能有助于流传输主机数据）
- 设置无请求标志
- 设置位 11（在使用旧版 eStreamer 时禁用任何默认事件流传输）

在初始消息后，使用规则文档消息（类型 10）来指定规则。

下图显示了规则文档消息的格式。加阴影的字段是规则文档消息格式特有的，后面的表格给出了这些字段的定义。前面的三个字段是标准消息报头。



下表对消息字段进行了说明。

表 2-15 规则文档消息字段

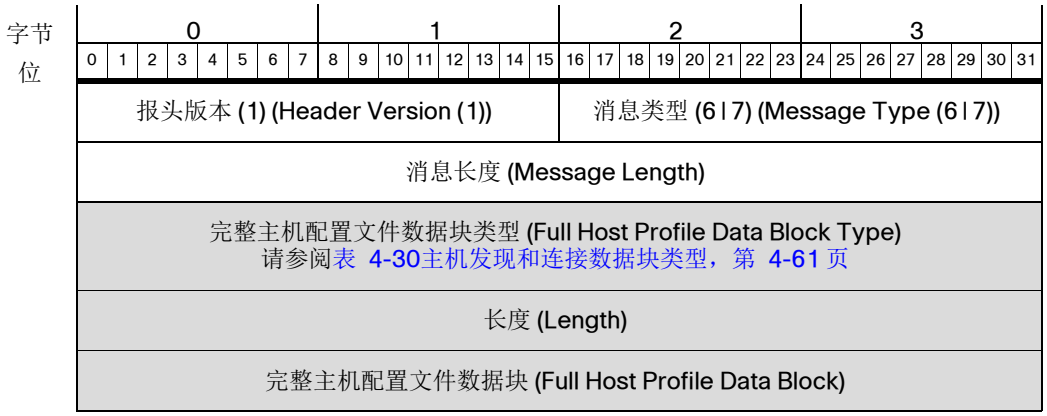
字段	数据类型	说明 (Description)
数据类型 (Data Type)	uint32	请求规则文档数据块的数据。值始终为 10。请参阅用于 5.2+ 的规则文档数据块，第 3-105 页。
标志 (Flags)	32 位字段	▪ 0x00000001 - 使规则文档数据块的“注释”(Notes) 字段被填充（有关 Cisco Secure Firewall 系统中存储的主机的用户定义信息）。 ▪ 0x00000002 - 使服务块的“横幅”字段填充（为服务检测到的第一个数据包的前 256 个字节）。默认禁用横幅，只有配置后才能使用。
签名 ID (Signature ID)	uint32	所请求规则的标识号。
生成器 ID (Generator ID)	uint32	所请求规则的 Cisco Secure Firewall 系统预处理器的标识号。
规则修订 (Rule Revision)	uint32	规则版本号。
已预留	uint8[20]	当前不使用此字段。

主机数据和多主机数据消息格式

eStreamer 通过发送主机数据消息响应主机请求，每条消息都带有完整的主机配置文件数据块。eStreamer 为请求中指定的每个主机发送一条主机数据消息。eStreamer 使用类型 6 消息响应单主机配置文件请求，并使用类型 7 消息响应多主机请求。类型 6 和类型 7 消息的格式相同，只是消息类型不同。

主机数据消息没有记录类型字段。消息的结构通过消息中包含的完整主机配置文件的消息类型和数据块类型传输。完整主机配置文件数据块为一组数据块系列。

下图显示了主机数据消息的格式，后面的表格定义加阴影的字段：



主机请求消息特有的字段如下：

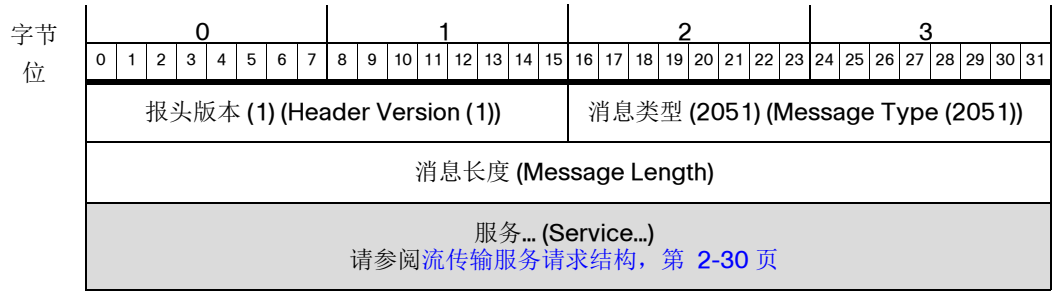
表 2-16

字段	数据类型	说明 (Description)
完整主机配置文件数据块类型 (Full Host Profile Data Block Type)	uint32	为消息中包含的完整主机配置文件数据指定数据块类型。请参阅表 4-30主机发现和连接数据块类型，第 4-61 页。
长度 (Length)	uint32	消息中完整主机配置文件数据的长度。
完整主机配置文件数据块 (Full Host Profile Data Block)	变量	主机数据。有关当前完整主机配置文件数据块的定义的链接，请参阅表 4-30主机发现和连接数据块类型，第 4-61 页。

流传输信息消息格式

当 eStreamer 服务收到需要一个扩展请求时，它会向客户端发送流传输信息消息，如下所述。此消息通告服务器的可用服务列表。目前，唯一的相关选项是 eStreamer 服务 (6667)，不过该消息可能会列出其他服务（应忽略这些服务）。通告的每项服务都由一个流传输服务请求结构（在流传输服务请求结构，第 2-30 页中介绍）表示。

下图说明流传输信息消息的格式。加阴影的字段是此消息类型所特有的。前面的三个字段是标准消息报头。



流传输信息消息的字段如下：

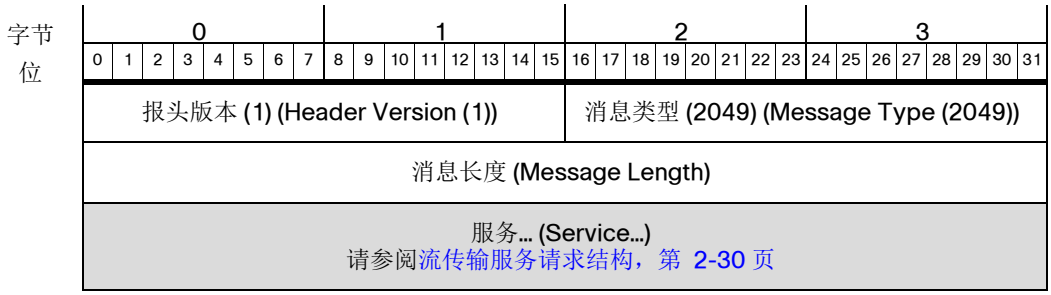
表 2-17　流传输信息消息字段

字段	数据类型	说明 (Description)
报头版本 (Header Version)	uint16	设置为 1。
消息类型 (Message Type)	uint16	eStreamer 消息类型。对于流传输请求消息，设置为 2051。
消息长度 (Message Length)	uint32	在消息报头后面的消息内容长度。不包括报头版本、消息类型和消息长度字段中的字节。
服务[] (Service[])	数组	可用服务的列表。请参阅流传输服务请求结构，第 2-30 页。

流传输请求消息格式

客户端使用流传输请求消息向 eStreamer 指定其要使用的流传输信息消息中的服务，后面跟着一组对要进行流传输的事件类型和版本的请求。下图显示该消息结构，后面的表格给出了字段的定义。请求的服务由一个流传输服务请求结构（在流传输服务请求结构，第 2-30页中介绍）表示。

下图说明流传输信息消息的格式。加阴影的字段是此消息类型所特有的。前面的三个字段是标准消息报头。



流传输请求消息的字段如下：

表 2-18　　流传输请求消息字段

字段	数据类型	说明 (Description)
报头版本 (Header Version)	uint16	设置为 1。
消息类型 (Message Type)	uint16	eStreamer 消息类型。对于流传输请求消息，设置为 2049。
消息长度 (Message Length)	uint32	在消息报头后面的消息内容长度。不包括报头版本、消息类型和消息长度字段中的字节。
服务[] (Service[])	数组	请求的服务结构的列表。请参阅流传输服务请求结构，第 2-30 页。

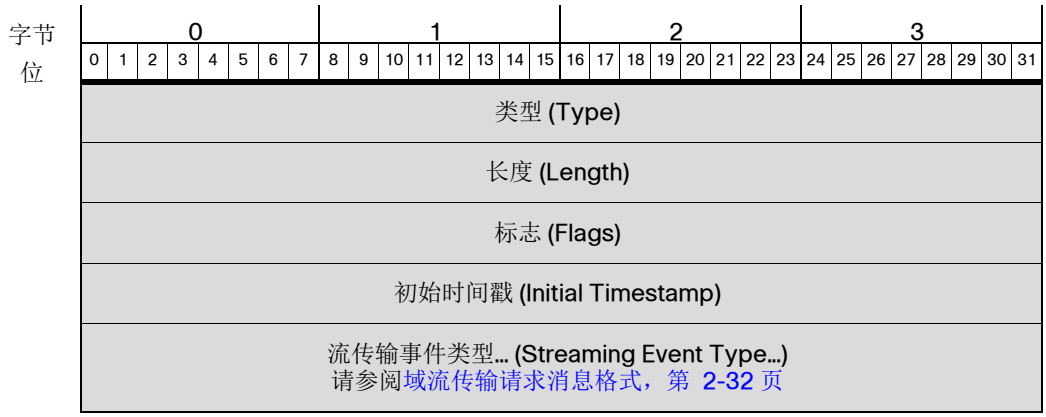
流传输服务请求结构

对于其通告的每项服务，eStreamer 服务都会在流传输信息消息中发送一个流传输服务请求数据结构。eStreamer 服务不使用流传输服务请求的最后一个字段。该字段用于要包含的事件类型列表。

客户端会处理来自 eStreamer 的流传输服务请求结构，并在其返回给服务器的响应中使用相同的结构。客户端向服务器发送的流传输服务请求首先包含一个对 eStreamer 通告的服务的请求，其次包含一个流传输事件类型结构列表（指定客户端希望接收的请求的事件类型）。

每个流传输事件类型结构包含两个字段，指定每个请求的事件类型的事件类型和版本。有关流传输事件类型结构的信息，请参阅域流传输请求消息格式，第 2-32 页。

下图显示了流传输服务请求结构的字段。后面的表格定义这些字段。



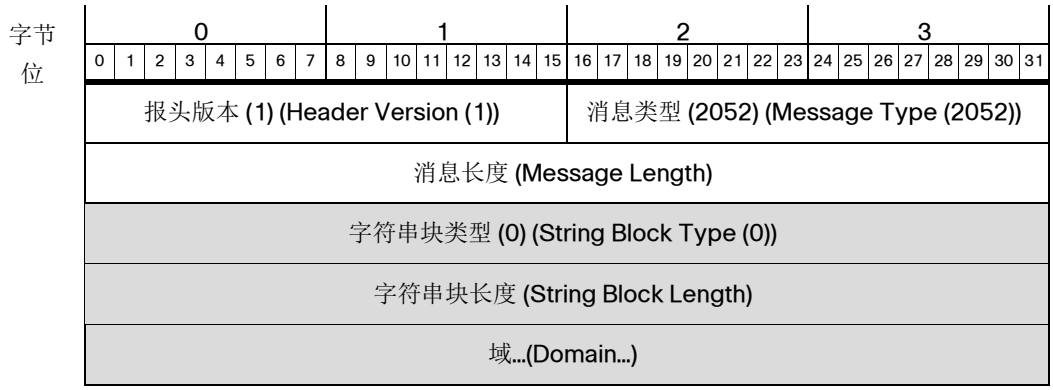
流传输服务请求结构的字段如下：

表 2-19　　流传输服务请求字段

字段	数据类型	说明
类型 (Type)	uint32	服务 ID。 在 eStreamer 服务器消息中，此字段通告一项可用服务。 在客户端消息中，此字段指定一项请求的服务。 当前有效选项： ▪ 6667（用于 eStreamer 服务）
长度 (Length)	uint32	服务请求长度。描述服务请求的长度，包括类型和长度。 请注意，长度必须包括消息中的所有流传输事件类型记录，加上终止记录。
标志 (Flags)	uint32	在 eStreamer 的流传输信息消息中：始终为 0。 在客户端的流传输请求消息中：复制原始事件流请求消息中的标志设置。
初始时间戳 (Initial Timestamp)	uint32	在 eStreamer 的流传输信息消息中：始终为 0。 在客户端的流传输请求消息中：复制原始事件流请求消息中的时间戳。
流传输事件类型 (Streaming Event Type)	数组	在 eStreamer 的流传输信息消息中： ▪ 已保留供将来使用。长度为 0。 在客户端的流传输请求消息中： ▪ 每个请求的事件类型各有一个流传输事件类型条目。请参阅域流传输请求消息格式，第 2-32 页。 ▪ 以 0 事件类型条目终止请求列表，事件类型和版本都设置为 0。 请参阅域流传输请求消息格式，第 2-32 页。

域流传输请求消息格式

客户端使用域流传输请求消息向 eStreamer 请求来自特定域的事件。下图显示该消息结构，后面的表格给出了字段的定义。加阴影的字段是此消息类型所特有的。前面的三个字段是标准消息报头。



域流传输请求消息的字段如下：

表 2-20 域流传输请求消息字段

字段	数据类型	说明 (Description)
报头版本 (Header Version)	uint16	设置为 1。
消息类型 (Message Type)	uint16	eStreamer 消息类型。对于域流传输请求消息，设置为 2052。
消息长度 (Message Length)	uint32	在消息报头后面的消息内容长度。不包括报头版本、消息类型和消息长度字段中的字节。
字符串块类型 (String Block Type)	uint32	启动包含域的字符串数据块。值始终为 0。
字符串块长度 (String Block Length)	uint32	域字符串数据块包含的字节数，包括数据块类型和报头字段的八个字节，加上域中的字节数。
域 (Domain)	字符串	请求流传输事件的域。如果留空，则服务将向客户端具有访问权限的所有域进行事件流传输。

流传输事件类型结构

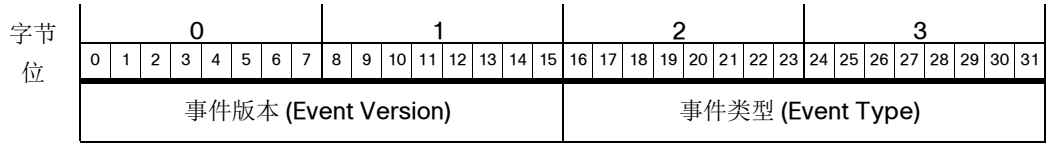
eStreamer 客户端使用流传输事件类型结构来指定事件的版本和类型。每个事件版本/类型组合构成一个事件流请求。

流传输事件类型结构的列表必须以一个所有字段都设置为零的结构终止。特点：

`Event Version = 0`

`事件类型 = 0`

下图说明流传输事件类型结构的格式。



流传输事件类型结构的字段如下：

表 2-21　　流传输事件类型字段

字段	数据类型	说明 (Description)
事件版本 (Event Version)	uint16	事件类型的版本号。有关每种事件类型的受支持版本的列表，请参阅表 2-22扩展请求的事件类型和版本，第 2-34 页。
事件类型 (Event Type)	uint16	请求的事件类型的代码。有关有效事件类型和版本代码的最新列表，请参阅表 2-22扩展请求的事件类型和版本，第 2-34 页。 要终止事件类型列表，应将事件类型和事件版本均设置为 0。

下表列出了客户端可以在扩展请求中指定的事件类型和版本。该表指出了与每种事件类型版本对应的管理中心软件版本。例如，要请求版本 4.8.0.2 - 4.9.1 中管理中心支持的关联事件，您应该请求事件类型 31、版本 5。如果事件被记录为不同的事件类型，它将被升级或降级，以符合请求的事件类型的格式。

表 2-22　　扩展请求的事件类型和版本

要请求...	使用此事件版本号...	以及此事件代码
入侵事件	1 — 4.8.x 及更早版本 2 — 4.9 - 4.10.x 3 — 5.0 - 5.1 4 — 5.1.1.x 5 — 5.2.x 6 — 5.3 7 — 5.3.1 8 — 5.4.x 9— 6.x 10— 7.0+	12
元数据	1 — 3.2 - 4.5.x 2 — 4.6.0.x 3 — 4.6.1 - 4.6.x 4 - 4.7+	21
关联和合规性允许列表事件	1 — 3.2 及更早版本 2 — 4.0 - 4.4.x 3 — 4.5 - 4.6.1 4 — 4.7 - 4.8.0.1 5 — 4.8.0.2 - 4.9.1.x 6 — 4.10.0 - 4.10.x 7 — 5.0 - 5.0.2 8 — 5.1 - 5.3.x 9 - 5.4+	31
发现事件	1 — 3.2 及更早版本 2 — 3.0 - 3.4.x 3 — 3.5 - 4.6.x 4 — 4.7 - 4.8.x 5 — 4.9.0.x 6 — 4.9.1 - 4.9.x.x 7 — 4.10.0 - 4.10.x 8 — 5.0.x 9 — 5.1.x 10 — 5.2 - 5.3 11 — 5.3.1+	61

表 2-22　　扩展请求的事件类型和版本 （续）

要请求...	使用此事件版本号...	以及此事件代码
连接事件	1 — 4.0 - 4.1 3 — 4.5 - 4.6.1 4 — 4.7 - 4.9.0.x 5 — 4.9.1 - 4.10.x 6 — 5.0.x 7 — 5.1.0.x 8 — 5.1.1.x 9 — 5.2.x 10 — 5.3 11 — 5.3.1 12 — 5.4 13 — 5.4.0.1-5.4.0.2 14 — 6.0.x 15 — 6.1.x 16 — 7.0.x 17 — 7.1+	71
用户事件	1 — 4.7 - 4.10.x 2 — 5.0.x 3 — 5.1-5.1.x 4 — 5.2 5— 6.0 6— 6.1 7— 6.2+	91
恶意软件事件	1 — 5.1.0.x 2 — 5.1.1.x 3 — 5.2.x 4 — 5.3 5 — 5.3.1 6 — 5.4.x 7— 6.x 8— 7.0+	101
文件事件	1 — 5.1.1 - 5.1.x 2 — 5.2.x 3 — 5.3 4 — 5.3.1 5 — 5.4.x 6 — 6.x 7 — 7.0+	111
影响关联事件	1 — 5.2.x 及更早版本 2 - 5.3+	131
终止列表中的事件类型	0	0

扩展请求消息示例

流传输信息消息

在以下示例中，服务器通告两项服务，一是类型 6667 (eStreamer)，二是类型 5000。在来自服务器的流传输信息消息中，标志字段和初始时间戳字段均为 0，该消息不指定事件类型。

表 2-23

报头版本：	1	/*始终为 1*/
消息类型：	2051	/*流传输信息消息*/
消息长度	32	/*消息内容字节数*/
服务[1].类型	6667	/*eStreamer 服务 ID*/
服务[1].长度	8	
服务[1].标志	0	/*没有来自服务器的标志 */
服务[1].初始时间戳	0	/*始终为 0*/
服务[2].类型	5000	/*服务-2 ID*/
服务[2].长度	8	
服务[2].标志	0	/*没有来自服务器的标志 */
服务[2].初始时间戳	0	/*始终为 0*/
报头版本：	1	/*始终为 1*/
消息类型：	2051	/*流传输信息消息*/

流传输请求消息

下面是一个流传输请求消息，其中客户端请求服务类型 6667 (eStreamer)，并指定了两个事件类型：版本 6 的连接事件（事件类型 71）和版本 4 的元数据（事件类型 21）。

表 2-24

报头版本：	1	/*始终为 1*/
消息类型：	2049	/*流请求消息*/
消息长度	28	/*负载字节*/
服务[1].类型	6667	/*eStreamer 服务 ID*/
服务[1].长度	20	
服务[1].标志	30	/*原始标志值*/
服务[1].初始时间戳	0	/*原始时间戳*/
服务[1].事件[1].版本	6	/*版本 6*/
服务[1].事件[1].类型	71	/*连接事件*/
服务[1].事件[2].版本	4	/* 版本 4*/

表 2-24

服务[1].事件[2].类型	21	/*元数据*/
服务[1].事件[3].版本	0	/*终止事件列表*/
服务[1].事件[3].类型	0	/*终止事件列表*/

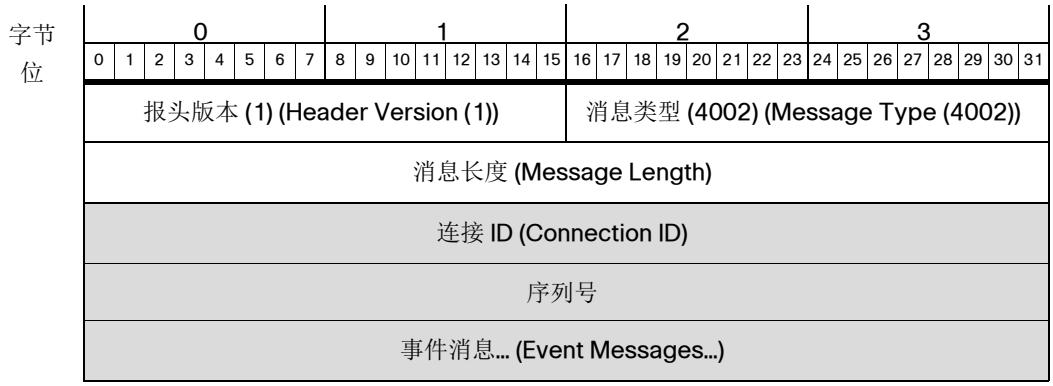
消息捆绑包格式

当客户端提交扩展请求时，eStreamer 服务器以捆绑包格式发送消息。

客户端回复空消息，确认收到整个捆绑包。客户端不应确认收到捆绑包内的单个消息。

消息捆绑包的消息类型应该为 4002。

下图显示消息捆绑包的结构。加阴影的字段是捆绑包消息类型所特有的。后面的表格介绍字段和数据结构的内容。



消息捆绑包消息的字段如下：

表 2-25　　消息捆绑包消息字段

字段	数据类型	说明 (Description)
报头版本 (Header Version)	uint16	始终为 1。
消息类型 (Message Type)	uint16	始终为 4002。
消息长度 (Message Length)	uint32	在消息报头后面的消息内容长度。不包括捆绑包的报头版本、消息类型和消息长度字段中的字节。 当客户端从捆绑包加载一条消息时，它会从此字段显示的长度中减去该消息的总长度（包括报头）。只要余数为正值，就有更多的消息要处理。
连接 ID (Connection ID)	uint32	与服务器建立的连接的唯一标识符。

表 2-25 消息捆绑包消息字段 （续）

字段	数据类型	说明 (Description)
序列号 (Serial Number)	uint32	从 1 开始，eStreamer 服务器每发送一个捆绑包，增加 1。
事件消息 [] (Event Messages [])	数组	服务器以捆绑包格式进行流传输的事件。每个消息都有全套报头，包括消息版本号 (1)、存档时间戳（如有请求）等等。

了解元数据

eStreamer 服务器可以将元数据与请求的事件记录一起提供。要接收元数据，您必须明确提出请求。有关如何请求给定版本的元数据的信息，请参阅表 2-6请求标志，第 2-13 页。元数据为事件记录中的代码和数字标识符提供情景信息。例如，入侵事件仅包含检测设备的内部标识符，而元数据则提供设备名称。

根据请求的元数据和环境，发送的元数据量可能有很大差异。

元数据传输

如果请求消息指定元数据，则 eStreamer 在发送任何相关的事件记录之前，先发送相关的元数据记录。

eStreamer 会记录已发送给客户端的元数据，不会重复发送相同的元数据记录。客户端应缓存收到的每个元数据记录。 如果客户端应用使用有限的缓存大小，则当缓存已满时，客户端应刷新缓存并重新连接到 eStreamer 服务，以确保客户端接收正在流传输的事件的所有数据值。从一个会话进入下一个会话后，eStreamer 不保留元数据传输历史记录，因此，当开始一个新会话，并且请求消息指定元数据时，eStreamer 会从头开始重新进行元数据流传输。 重新连接时，客户端可以在请求消息中指定“初始时间戳”，以避免事件重复或丢失事件。

当地语言翻译版本说明

思科可能会在某些地方提供本内容的当地语言翻译版本。请注意，翻译版本仅供参考，如有任何不一致之处，以本内容的英文版本为准。