

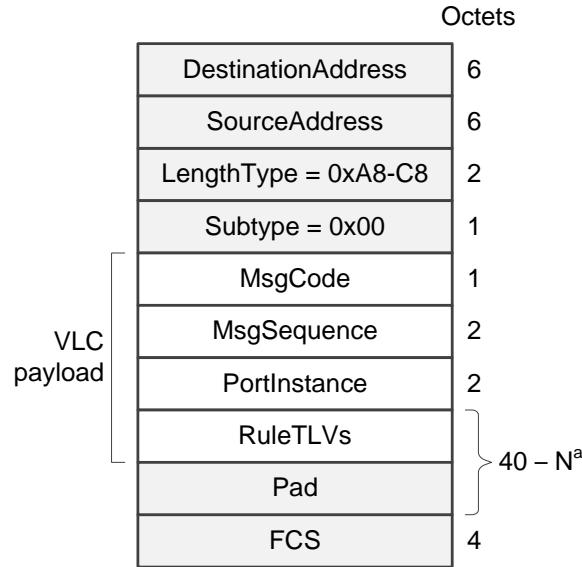
1 **8 VLC Management**

2 **8.1 VLC Configuration**

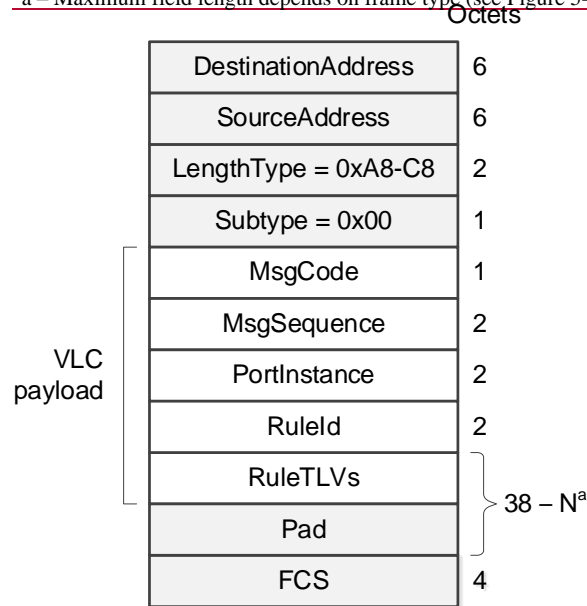
3 The tunnels originate and terminate in the VLC-aware devices. The tunnels are configured by means of
4 provisioning specific *CTE rules* for the tunnel entry and exit points. These rules are provisioned by the
5 operator using the *VLC_CONFIG* VLCPDUs, which carry a set of *condition-encoding* TLVs and a set of
6 *action-encoding* TLVs.

7 **8.1.1 Configuration VLCPDU**

8 The *VLC_CONFIG* VLCPDU format shall be as depicted in Figure 8-1. The *VLC_CONFIG* VLCPDU is
9 used as both a request to configure a *CTE rule* as well as a response containing the result of the configuration
10 request.



a – Maximum field length depends on frame type (see Figure 5-1).



a – Maximum field length depends on frame type (see Figure 5-1).

Figure 8-1—VLC_CONFIG VLCPDU format

The *VLC_CONFIG* VLCPDU is an instantiation of the generic VLCPDU (see Figure 5-1). It is identified by the *Subtype* field value of 0x00. The structure of the *VLC payload* is defined as follows:

—*MsgCode*:

The *MsgCode* field identifies whether the *VLC_CONFIG* message is a request message or a response. If the VLCPDU is a request, this field encodes the requested action. If the VLCPDU is a response, this field echoes the requested action and encodes the result code for this action. The format of the *MsgCode* field is shown in Table 8-1.

1

Table 8-1—Format of the *MsgCode* field

Bits	Field name	Value	Description
3:0	<i>MsgType</i>	0x0	The message is a request
		0x1	The message is a response indicating successful action
		0x2	The message is a response indicating failed action
		0x3	The message is a response indicating that no action was necessary
		0x4	The message is a response indicating invalid request
		0x5 to 0xF	Reserved, ignored on reception
7:4	<i>RequestCode</i>	0x0	Query all rules
		0x1	Add a rule
		0x2	Remove a rule
		0x4 to 0xF	Reserved, ignored on reception

2 —*MsgSequence*:

3 In situations when a VLC configuration request or a response consists of multiple messages, this field
 4 identifies the message sequence number. The format of the *MsgSequence* field is shown in Table 8-2.

5

Table 8-2—Format of the *MsgSequence* field

Bits	Field name	Value	Description
14:0	<i>MsgCounter</i>	0x00-01 to 0x7F-FF	A counter that increments by one for each message in a sequence. In the first message in a sequence, the <i>MsgCounter</i> is equal to 1.
15	<i>EndOfSequence</i>	0	This message is not the last message in a sequence
		1	This message is the last message in a sequence

6

7 When a request or a response consists of a single VLCPDU, the *MsgCounter* subfield is equal to 0x00-
 8 01 and the *EndOfSequence* flag is equal to 1.

9 Note that even when a VLC configuration request or a response consists of multiple messages, a single
 10 rule is not split across multiple messages and as such – no reassembly mechanism is necessary to
 11 reconstruct any rule. An example scenario where the response consists of multiple messages would be a
 12 VLC configuration response to a ‘Query all rules’ request, where multiple rules are being reported.

13 —*PortInstance*:

14 This field identifies a port instance in the VLC-aware device to which the given *VLC_CONFIG* VLCPDU
 15 applies. The format of the *PortInstance* field is shown in Table 8-3.

16

Table 8-3—Format of the *PortInstance* field

Bits	Field name	Value	Description
14:0	<i>PortIndex</i>	0x00-00 to 0x7F-FF	Index of a port (VLC sublayer) to which the requested action is to be applied.

15	<i>Direction</i>	0	The rule is to be applied to the transmit path of VLC sublayer (i.e., an egress rule)
		1	The rule is to be applied to the receive path of VLC sublayer (i.e., an ingress rule)

1 In the VLC response message, this field reflects the *PortInstance* field value from the corresponding VLC
2 request message.

3 —*RuleId*

4 This field contains a 15-bit rule identification number, positioned at the least-significant end of the field
5 (i.e., in *RuleId*[14:0]). Bit 15 shall be set to zero. The use of this field is defined in 8.1.2.

6 —*RuleTLVs*:

7 This field includes one or more *CTE rule* TLV(s) as defined in 8.1.2. The combined size of the *RuleTLV*
8 and *Pad* fields ranges between ~~40-38~~ and *N*, where *N* is defined in Figure 5-1.

9 **8.1.2 Rule identification**

10 CTE rules are identified by a 15-bit number. The rule identification number is chosen by a device when
11 it adds a new rule to its CTE rule table. The selection criteria for the rule identification number is vendor-
12 specific and outside the scope of this standard. However, each rule identification number shall be unique
13 per CTE table (i.e., unique per port and per direction) and it shall not be equal to zero.

14 The use of rule identification numbers in VLC configuration protocol is explained in 8.1.4. The rule
15 identification numbers are also used as *leaf* values under branch 0xA8 to query VLC sublayer statistics,
16 such as a count of frames and a count of octets matched by each rule (see 8.2.2).

17 **8.1.28.1.3 CTE rule TLV structure**

18 The structure of a *CTE rule* TLV is shown in Table 8-4. Each *VLC_CONFIG* VLCPDU shall contain at least
19 one *CTE rule* TLV.

1

Table 8-4—CTE rule TLV structure

Field Size (octets)	Field Name	Value	Description
1	Type	0xC0	Type code identifying the condition-encoding TLV
		0xAC	Type code identifying the action-encoding TLV
		0x00	Type code indicating <u>the terminating TLV. The terminating TLV signals to the VLC_CONFIG VLCPDU parser</u> that there are no more TLVs to process. The Length field and other fields (if present) are ignored. The <u>terminating TLV with Type = 0x00</u> shall be the last TLV in every VLC_CONFIG VLCPDU and it may be the only TLV in the VLC_CONFIG VLCPDU.
1	Length	V+M+4	The Length field encompasses the entire TLV, including the Type and Length fields. A TLV with length of 0x00 through 0x03 is invalid.
1	Operation ^a	per Table 6-1	Comparison operator code, if the TLV Type = 0xC0
		per Table 6-3	Action code, if the TLV Type = 0xAC
1	FieldCode ^a	per Table 6-2	Identifies a field to be used in a comparison, or to be modified by an action.
V	Value	Various	The value to be used in a comparison or by an Add/Change action. Some TLVs may omit this field.
M ^b	Mask	various	The mask pattern to be used in a comparison condition. The mask pattern is applied as a bitwise-AND operation to both the value to be used in a comparison (see the Value field above) as well the value of the field identified by the FieldCode parameter of this TLV. Some TLVs may omit this field ^c . When Mask is omitted, the comparison applies to the entire field.

2 ^{a)} Fields *Operation* and *FieldCode* shall be present in all TLVs, even if they are not used. When these fields
3 are not used, they should be set to the value of zero.

4 ^{b)} The length *M* of *Mask* field shall be the same as the length of *Value* field, if mask field is present. Otherwise,
5 the length *M* is considered to be equal to zero.

6 ^{c)} If a CTE rule TLV omits the *Value* field, the *Mask* field shall also be omitted.

7 **8.1.4 VLC configuration protocol**

8 The VLC configuration protocol defines the exchange of VLC_CONFIG VLCPDUs between the device that
9 issues a VLC_CONFIG request (the Requestor) and the device that issues a VLC_CONFIG response (the
10 Responder). For each VLC_CONFIG request received from the Requester, the Responder provides one of
11 following VLC_CONFIG responses that reflect different outcomes of the requested action:

12 — Response indicating successful action – the requested action was performed successfully. For
13 example, a new rule was added to the CTE table.

14 — Response indicating failed action – the requested action has failed. For example, a new rule was
15 not added due to insufficient memory, i.e., due to CTE table being full.

1 — Response indicating that no action was necessary – the requested action, if performed, would result
 2 in no changes to the CTE table. For example, a rule being added already exists in the CTE table, or
 3 a rule being removed does not exist in the CTE table.

4 — Response indicating invalid request – typically, a response to a malformed *VLC_CONFIG* request
 5 message.

6 Subclauses 8.1.4.1 through 8.1.4.3 describe the contents of VLC requests and responses for the operations of
 7 querying all rules, adding a rule, and removing a rule.

8 **8.1.4.1 Request and Response for the ‘Query all rules’ operation**

9 The reception of the ‘Query all rules’ request message causes the Responder to report the contents of the
 10 entire CTE rule table identified by the *PortInstance* field. If multiple rules are present, the Responder issues
 11 a bulk response message, with each rule being reported by a separate *VLC_CONFIG* VLCPDU with an
 12 incrementing *MsgSequence* value.

13 A response ‘no action necessary’ indicates that the CTE rule table contains no rules. A ‘failed action’
 14 response signals the Responder’s failure to retrieve all or some of provisioned rules from the CTE rule table,
 15 e.g., due to memory corruption or other unspecified local failure.

16 The contents of *RuleID* and *RuleTLVs* fields for ‘Query all rules’ request and responses are shown in Table
 17 8-5.

18 **Table 8-5 — Contents of *VLC_CONFIG* VLCPDU with *RequestCode* = 0x00**
 19 **(‘Query all rules’ operation)**

<u><i>MsgType</i> value</u>	<u>Message type</u>	<u>Content of <i>RuleID</i> field</u>	<u>Content of <i>RuleTLVs</i> field</u>
<u>0x0</u>	<u>Request</u>	<u>0x00</u>	<u>terminating TLV only</u>
<u>0x1</u>	<u>Response — successful action</u>	<u>Unique rule identification number</u>	<u>1. condition-encoding TLV(s) 2. action-encoding TLV(s) 3. terminating TLV</u>
<u>0x2</u>	<u>Response — failed action</u>	<u>0x00</u>	<u>terminating TLV only</u>
<u>0x3</u>	<u>Response — no action necessary</u>	<u>0x00</u>	<u>terminating TLV only</u>
<u>0x4</u>	<u>Response — invalid request</u>	<u>0x00</u>	<u>terminating TLV only</u>

20
 21 **8.1.4.2 Request and Response for the ‘Add a rule’ operation**

22 The reception of the ‘Add a rule’ request message causes the Responder to add the specified rule to the CTE
 23 rule table identified by the *PortInstance* field and to assign to this rule a rule identification number that is
 24 unique per CTE table (see 8.1.4.1). The Responder then reports this rule identification number together with
 25 a copy of the *RuleTLVs* field back to the Requestor using the response indicating a successful action
 26 (*MsgType* = 0x1).

27 A bulk ‘Add a rule’ request message consists of multiple *VLC_CONFIG* VLCPDUs, as indicated by
 28 incrementing *MsgSequence* values. When such a bulk request is received, the Responder shall provision all
 29 the requested rules before issuing a response. If all the rules were provisioned successfully, the bulk response

1 message is generated, also consisting of multiple *VLC CONFIG* VLCPDUs (identified by incrementing
 2 *MsgSequence* numbers). In a bulk response, each VLCPDU reports a unique rule identification number and
 3 a copy of *RuleTLVs* field for one of the provisioned rules. The order of rules in a bulk response shall match
 4 the order of rules in the corresponding bulk request.

5 A bulk response message may combine *VLC CONFIG* VLCPDUs indicating successful action with
 6 VLCPDUs indicating that no action was necessary. If some of the rules that are being provisioned by a bulk
 7 request already exist in the target CTE rule table, the VLCPDUs in the bulk response message that report the
 8 already-existing rules use the *MsgType* 0x3 (no action necessary), while those that report newly-added rules,
 9 use the *MsgType* 0x1 (successful action).

10 If the responder is unable to configure one or more rules from a bulk request, is shall not configure any of
 11 the rules from this bulk request and instead shall issue a single response *VLC CONFIG* VLCPDU with
 12 *MsgType* value equal to 0x2 (response indicating a failed action).

13 A bulk request containing gaps in the *MsgSequence* values, or missing an *EndOfSequence* flag with value of
 14 1 is considered malformed. The Responder shall not configure any rules received in a malformed bulk request
 15 and shall issue a single response *VLC CONFIG* VLCPDU containing *MsgType* 0x4 (invalid request).

16 The contents of *RuleId* and *RuleTLVs* fields for ‘Add a rule’ request and responses are shown in Table 8-6.

17 **Table 8-6— Contents of *VLC CONFIG* VLCPDU with *RequestCode* = 0x01**
 18 **(‘Add a rule’ operation)**

<u><i>MsgType</i> value</u>	<u>Message type</u>	<u>Content of <i>RuleId</i> field</u>	<u>Content of <i>RuleTLVs</i> field</u>
<u>0x0</u>	<u>Request</u>	<u>0x00</u>	<u>1. condition-encoding TLV(s) 2. action-encoding TLV(s) 3. terminating TLV</u>
<u>0x1</u>	<u>Response — successful action</u>	<u>Unique rule identification number assigned to the newly-added rule</u>	<u>Copy of the <i>RuleTLVs</i> field from the Request message</u>
<u>0x2</u>	<u>Response — failed action</u>	<u>0x00</u>	<u>Copy of the <i>RuleTLVs</i> field from the Request message</u>
<u>0x3</u>	<u>Response — no action necessary</u>	<u>Unique rule identification number assigned to the existing rule</u>	<u>Copy of the <i>RuleTLVs</i> field from the Request message</u>
<u>0x4</u>	<u>Response — invalid request</u>	<u>0x00</u>	<u>Copy of the <i>RuleTLVs</i> field from the Request message</u>

19
 20 **8.1.4.3 Request and Response for the ‘Remove a rule’ operation**

21 The reception of the ‘Remove a rule’ request message causes the Responder to remove the rule identified by
 22 *RuleId* field from the CTE rule table identified by the *PortInstance* field. The Responder then reports this
 23 rule’s identification number together with the rule conditions and actions in the *RuleTLVs* field back to the
 24 Requestor using the response indicating a successful action (*MsgType* = 0x1).

25 A bulk ‘Remove a rule’ VLCPDU includes multiple *RuleID* fields. In case the number of rules being removed
 26 exceeds the capacity of a single VLCPDU, a bulk ‘Remove a rule’ request may consist of multiple
 27 *VLC CONFIG* VLCPDUs, identified by incrementing *MsgSequence* values. When a bulk request is received,
 28 the Responder shall remove all the requested rules before issuing a response. If all the rules were removed
 29 successfully, the bulk response message is generated. In a bulk response, each removed rule is confirmed by
 30 a separate VLCPDU that contains the removed rule’s identification number and the *RuleTLVs* field

1 containing that rule. The order of rules in a bulk response shall match the order of rules in the corresponding
 2 bulk request.

3 A bulk response message may combine VLC CONFIG VLCPDUs indicating successful action with
 4 VLCPDUs indicating that no action was necessary. If some of the rules that are being removed by a bulk
 5 request do not exist in the target CTE rule table, the VLCPDUs in the bulk response message that report these
 6 rules use the MsgType 0x3 (no action necessary), while those that report newly-deleted rules, use the
 7 MsgType 0x1 (successful action).

8 If the responder is unable to remove one or more rules from a bulk request, is shall not remove any of the
 9 rules from this bulk request and instead shall issue a single response VLC CONFIG VLCPDU with MsgType
 10 value equal to 0x2 (response indicating a failed action).

11 A bulk request containing gaps in the MsgSequence values, or missing an EndOfSequence flag with value of
 12 1 is considered malformed. The Responder shall not remove any rules indicated by a malformed bulk request
 13 and shall issue a single response VLC CONFIG VLCPDU containing MsgType 0x4 (invalid request).

14 A ‘Remove a rule’ request with the RuleId value equal to 0x00 is treated as a ‘Remove all rules’ request.
 15 Upon receiving such request, the Responder shall remove all existing rules and reset the CTE table to its
 16 initialization state. The ‘Remove all rules’ response contains a single VLCPDU with the value of RuleId field
 17 equal to 0x00 (i.e., individual deleted rules are not reported).

18 The contents of RuleId and RuleTLVs fields for ‘Remove a rule’ request and responses are shown in Table
 19 8-7.

20 **Table 8-7 — Contents of VLC CONFIG VLCPDU with RequestCode = 0x02**
 21 **(‘Remove a rule’ operation)**

<u>MsgType value</u>	<u>Message type</u>	<u>Content of RuleID field</u>	<u>Content of RuleTLVs field</u>
<u>0x0</u>	<u>Request</u>	<u>Unique identification number of the rule that is to be removed</u> <u>—or</u> <u>0x00-00</u>	<u>terminating TLV only</u>
<u>0x1</u>	<u>Response — successful action</u>	<u>Copy of the RuleId field from the Request message</u>	<u>1. condition-encoding TLV(s)</u> <u>2. action-encoding TLV(s)</u> <u>3. terminating TLV</u>
<u>0x2</u>	<u>Response — failed action</u>	<u>Copy of the RuleId field from the Request message</u>	<u>terminating TLV only</u>
<u>0x3</u>	<u>Response — no action necessary</u>	<u>Copy of the RuleId field from the Request message</u>	<u>terminating TLV only</u>
<u>0x4</u>	<u>Response — invalid request</u>	<u>Copy of the RuleId field from the Request message</u>	<u>terminating TLV only</u>

22
 23

1 **8.2 Management Attributes**

2 **8.2.1 Introduction**

3 This subclause defines a set of extended management attributes for querying statistics in the VLC sublayer.
 4 In general, attributes are defined to be independent of any particular management application or management
 5 protocol. Such definitions of attributes and actions are focused on the associated device characteristics and
 6 behaviors. Within the constraints imposed by the described characteristics and behaviors, the internal
 7 representations of the attributes remain implementation dependent and outside the scope of this standard.

8 NOTE — where no default value is specified for an attribute, the attribute is assumed to initialize to a vendor-
 9 specific value.

10 To address the interoperability between multiple VLC clients, precise definitions of the TLV structures and
 11 encodings of individual attributes into TLV structure are also provided.

12 The protocol used to set and query these management attributes is outside the scope of this standard, although
 13 the OAM protocol defined in IEEE Std 802.3 and extended in IEEE Std 1904.1 is generally assumed.

14 **8.2.2 Branch 0xA8 VLC Counters**

15 This subclause lists extended management attributes, which are not part of the definitions in
 16 IEEE Std 802.3, Clause 30. The extended attributes shown in Table 8-5 shall be supported.

17 **Table 8-5—VLC Counter attributes defined in branch 0xA8**

Leaf	Attribute	Defined in
0x00-00	<i>aVlcFramesUnmatched</i>	8.2.2.1
0x00-01	<i>aVlcFramesMatchedByRule1</i>	8.2.2.2
...	...	
0x7F-FF	<i>aVlcFramesMatchedByRule32767</i>	
0x10-00	<i>aVlcOctetsUnmatched</i>	8.2.2.3
0x10-01	<i>aVlcOctetsMatchedByRule1</i>	8.2.2.4
...	...	
0xFF-FF	<i>aVlcOctetsMatchedByRule32767</i>	

18 **8.2.2.1 Attribute *aVlcFramesUnmatched* (0xA8/0x00-00)**

19 This attribute represents the current number of frames that are not matched by any rules in a port identified
 20 by the *Object Context* TLV.

21 Attribute *aVlcFramesUnmatched*:

- 22 **Syntax:** Counter, Resettable, Wrap-around
- 23 **Range:** 0x00 to 0xFF-FF-FF-FF-FF-FF-FF-FF
- 24 **Default value:** 0x00
- 25 **Remote access:** Read/Write

1 **Description:** This attribute indicates current number of frames not matched by any of the
 2 rules provisioned at the port identified by the *Object Context* TLV. On write of
 3 any value to this attribute, the counter shall reset to the value of 0x00.

4 The *aVlcFramesUnmatched* attribute is associated with a port. In an EPON ONU or OLT, this can be a
 5 UNI/NNI port, or an LLID.

6 The variable container TLV for the *aVlcFramesUnmatched* attribute shall be as specified in Table 8-6

7 **Table 8-6—VLC counter of frames unmatched by any rule (0xA8/0x00-00)**

Size (octets)	Field name	Value	Description
1	<i>Branch</i>	0xA8	VLC attribute branch identified
2	<i>Leaf</i>	0x00-00	Leaf identifier
1	<i>Length</i>	8	The size of TLV fields following the <i>Length</i> field
8	<i>VlcFramesUnmatched</i>	varies	Value of <i>aVlcFramesUnmatched</i> attribute

8

9 **8.2.2.2 Attribute *aVlcFramesMatchedByRuleN* (0xA8/0x00-01 to 0xA8/0x7F-FF)**

10 This attribute represents the current number of frames matched by a rule with ~~*RuleID*~~ *RuleId* equal to *N*, in a
 11 port identified by the *Object Context* TLV.

12 Attribute *aVlcFramesMatchedByRuleN*:

13 **Syntax:** Counter, Resettable, Wrap-around

14 **Range:** 0x00 to 0xFF-FF-FF-FF-FF-FF-FF-FF

15 **Default value:** 0x00

16 **Remote access:** Read/Write

17 **Description:** This attribute indicates current number of frames matched by the rule with the
 18 ~~*RuleID*~~ *RuleId* = *N* at the port identified by the *Object Context* TLV. On write of
 19 any value to this attribute, the counter shall reset to the value of 0x00.

20 The *aVlcFramesMatchedByRuleN* attribute is associated with a port. In an EPON ONU or OLT, this can be
 21 a UNI/NNI port, or an LLID.

22 The variable container TLV for the *aVlcFramesMatchedByRuleN* attribute shall be as specified in Table 8-7.

23 **Table 8-7—VLC counter of frames matched by rule *N* TLV**
 24 **(0xA8/0x00-01 to 0xA8/0x7F-FF)**

Size (octets)	Field name	Value	Description
1	<i>Branch</i>	0xA8	VLC attribute branch identified

2	<i>Leaf</i>	<i>N</i>	Leaf identifier. <i>aVlcFramesMatchedByRule1</i> through <i>aVlcFramesMatchedByRule32767</i> are represented by <i>Leaf</i> values ranging from 0x00-01 through 0x7F-FF.
1	<i>Length</i>	8	The size of TLV fields following the <i>Length</i> field
8	<i>VlcFramesMatchedByRuleN</i>	varies	Value of <i>aVlcFramesMatchedByRuleN</i> attribute

1 **8.2.2.3 Attribute *aVlcOctetsUnmatched* (0xA8/0x10-00)**

2 This attribute represents the total number of octets in the current number of frames that are not matched by
3 any rules in a port identified by the *Object Context* TLV.

4 Attribute *aVlcOctetsUnmatched*:

5 **Syntax:** Counter, Resettable, Wrap-around

6 **Range:** 0x00 to 0xFF-FF-FF-FF-FF-FF-FF-FF

7 **Default value:** 0x00

8 **Remote access:** Read/Write

9 **Description:** This attribute indicates the total number of octets in the current number of
10 frames not matched by any of the rules provisioned at the port identified by the
11 *Object Context* TLV. On write of any value to this attribute, the counter shall
12 reset to the value of 0x00.

13 The *aVlcOctetsUnmatched* attribute is associated with a port. In an EPON ONU or OLT, this can be a
14 UNI/NNI port, or an LLID.

15 The variable container TLV for the *aVlcOctetsUnmatched* attribute shall be as specified in Table 8-8.

16 **Table 8-8—VLC counter of octets unmatched by any rule (0xA8/0x10-00)**

Size (octets)	Field name	Value	Description
1	<i>Branch</i>	0xA8	VLC attribute branch identified
2	<i>Leaf</i>	0x10-00	Leaf identifier
1	<i>Length</i>	8	The size of TLV fields following the <i>Length</i> field
8	<i>VlcOctetsUnmatched</i>	varies	Value of <i>aVlcOctetsUnmatched</i> attribute

17 **8.2.2.4 Attribute *aVlcOctetsMatchedByRuleN* (0xA8/0x10-01 to 0xA8/0xFF-FF)**

18 This attribute represents the total number of octets in the current number of frames matched by a rule with
19 ~~*RuleID*~~ *RuleId* equal to *N*, in a port identified by the *Object Context* TLV.

20 Attribute *aVlcOctetsMatchedByRuleN*:

21 **Syntax:** Counter, Resettable, Wrap-around

22 **Range:** 0x00 to 0xFF-FF-FF-FF-FF-FF-FF-FF

1 **Default value:** 0x00
 2 **Remote access:** Read/Write
 3 **Description:** This attribute indicates current number of octets in frames matched by the rule
 4 with the ~~*RuleID*~~ *RuleId* = *N* at the port identified by the *Object Context* TLV. On
 5 write of any value to this attribute, the counter shall reset to the value of 0x00.

6 The *aVlcOctetsMatchedByRuleN* attribute is associated with a port. In an EPON ONU or OLT, this can be a
 7 UNI/NNI port, or an LLID.

8 The variable container TLV for the *aVlcOctetsMatchedByRuleN* attribute shall be as specified in Table 8-9.

9 **Table 8-9—VLC counter of octets matched by rule *N* TLV**
 10 **(0xA8/0x10-01 to 0xA8/0xFF-FF)**

Size (octets)	Field name	Value	Description
1	<i>Branch</i>	0xA8	VLC attribute branch identified
2	<i>Leaf</i>	<i>N</i>	Leaf identifier. <i>aVlcOctetsMatchedByRule0</i> through <i>aVlcOctetsMatchedByRule32767</i> are represented by <i>Leaf</i> values ranging from 0x10-01 through 0xFF-FF.
1	<i>Length</i>	8	The size of TLV fields following the <i>Length</i> field
8	<i>VlcOctetsMatchedByRuleN</i>	varies	Value of <i>aVlcOctetsMatchedByRuleN</i> attribute

11

1 **Annex 8A**
 2 **(informative)**
 3 **VLC configuration examples**

4 **Table 8A-102—Contents of VLC_CONFIG message**

Field	Subfield	Value	Description
<u>DstAddress</u>	n/a	X	VLC_CONFIG VLCPDU directed to bridge X
<u>SrcAddress</u>	n/a	any	Source address of a device that issued the VLC_CONFIG VLCPDU
<i>LengthType</i>	n/a	0xA8-C8	Ethertype value identifying VLCPDUs (see 5.1)
<i>Subtype</i>	n/a	0x00	VLCPDU carrying VLC_CONFIG message
<i>MsgCode</i>	<i>MsgType</i>	0x0	This message is a Request (see Table 8-1)
	<i>RequestCode</i>	0x1	Request to add a rule (see Table 8-1)
<i>MsgSequence</i>	<i>MsgCounter</i>	0x00-01	This request consists of a single message
	<i>EndOfSequence</i>	1	
<i>PortInstance</i>	<i>PortIndex</i>	3	The rule is to be provisioned for port #3
	<i>Direction</i>	1	The rule is to be provisioned for the receive path (i.e., an ingress rule)
<u>RuleId</u>	<u>n/a</u>	<u>0x00-00</u>	<u>In a request to add a rule, the RuleId field is set to zero (see 8.1.3.2)</u>
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x01	Compare <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	0x01-80-C2-00-00-02	IEEE 802.3 Slow_Protocols_Multicast address (see IEEE Std 802.3, 57A.3)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x03	Compare <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0x88-09	Slow Protocol EtherType value (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x05	TLV length is 5 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x06	Compare <i>FID_SUBTYPE</i> field (see Table 6-2)

Field	Subfield	Value	Description
	<i>Value</i>	0x03	Slow Protocol Subtype value for OAM (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x01	Modify <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	S	Set Station S MAC address as the destination for resulting VLCPDUs.
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x03	Modify <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0xA8-C8	Set Ethertype to be equal to <i>ETHERTYPE_VLC</i> in the resulting VLCPDUs.
<i>RuleTLV</i> (termination)	<i>Type</i>	0x00	This is a termination (end-of-rule) TLV (see Table 8-4)
	<i>Length</i>	0x04	TLV length is 4 octets
	<i>Operation</i>	0x00	Filled with zeros when not used (see Table 8-4, note)
	<i>FieldCode</i>	0x00	

1

2

Table 8A-114—Contents of *VLC_CONFIG* message

Field	Subfield	Value	Description
<u><i>DstAddress</i></u>	n/a	Y	<i>VLC_CONFIG</i> VLCPDU directed to bridge Y
<u><i>SrcAddress</i></u>	n/a	any	Source address of a device that issued the <i>VLC_CONFIG</i> VLCPDU
<i>LengthType</i>	n/a	0xA8-C8	Ethertype value identifying VLCPDUs (see 5.1)
<i>Subtype</i>	n/a	0x00	VLCPDU carrying <i>VLC_CONFIG</i> message
<i>MsgCode</i>	<i>MsgType</i>	0x0	This message is a Request (see Table 8-1)
	<i>RequestCode</i>	0x1	Request to add a rule (see Table 8-1)
<i>MsgSequence</i>	<i>MsgCounter</i>	0x00-01	This request consists of a single message
	<i>EndOfSequence</i>	1	
<i>PortInstance</i>	<i>PortIndex</i>	0	The rule is to be provisioned for port #0

	<i>Direction</i>	0	The rule is to be provisioned for the transmit path (i.e., an egress rule)
<i>RuleId</i>	<i>n/a</i>	<u>0x00-00</u>	<u>In a request to add a rule, the <i>RuleId</i> field is set to zero (see 5.1.3.2)</u>
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x01	Compare <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	S	The destination address is equal to MAC address of Station S.
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x03	Compare <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0xA8-C8	VLC Ethertype value (see 5.1)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x05	TLV length is 5 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x06	Compare <i>FID_SUBTYPE</i> field (see Table 6-2)
	<i>Value</i>	0x03	VLC Subtype identifying OAM payload (see Table 5-1)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x01	Modify <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	0x01-80-C2-00-00-02	IEEE 802.3 Slow_Protocols_Multicast address (see IEEE Std 802.3, 57A.3)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x03	Modify <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0x88-09	Slow Protocol Ethertype value (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i>	<i>Type</i>	0x00	This is a termination (end-of-rule) TLV (see Table 8-4)

(termination)	<i>Length</i>	0x04	TLV length is 4 octets
	<i>Operation</i>	0x00	Filled with zeros when not used (see Table 8-4, note)
	<i>FieldCode</i>	0x00	

1
2

Table 8A-126—Contents of *VLC_CONFIG* message

Field	Subfield	Value	Description
<u><i>DstAddress</i></u>	n/a	Y	<i>VLC_CONFIG</i> VLCPDU directed to bridge Y
<u><i>SrcAddress</i></u>	n/a	any	Source address of a device that issued the <i>VLC_CONFIG</i> VLCPDU
<i>LengthType</i>	n/a	0xA8-C8	Ethertype value identifying VLCPDUs (see 5.1)
<i>Subtype</i>	n/a	0x00	VLCPDU carrying <i>VLC_CONFIG</i> message
<i>MsgCode</i>	<i>MsgType</i>	0x0	This message is a Request (see Table 8-1)
	<i>RequestCode</i>	0x1	Request to add a rule (see Table 8-1)
<i>MsgSequence</i>	<i>MsgCounter</i>	0x00-01	This request consists of a single message
	<i>EndOfSequence</i>	1	
<i>PortInstance</i>	<i>PortIndex</i>	3	The rule is to be provisioned for port #3
	<i>Direction</i>	1	The rule is to be provisioned for the receive path (i.e., an ingress rule)
<u><i>RuleId</i></u>	<u>n/a</u>	<u>0x00-00</u>	<u>In a request to add a rule, the <i>RuleId</i> field is set to zero (see 8.1.3.3)</u>
<i>RuleTLV</i> (condition)	<i>Type</i>	0xC0	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x01	Compare <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	0x01-80-C2-00-00-02	IEEE 802.3 Slow_Protocols_Multicast address (see IEEE Std 802.3, 57A.3)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xC0	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x03	Compare <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0x88-09	Slow Protocol Ether type value (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i>	<i>Type</i>	0xC0	This is a condition TLV (see Table 8-4)

Field	Subfield	Value	Description
(condition)	<i>Length</i>	0x05	TLV length is 5 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x06	Compare <i>FID_SUBTYPE</i> field (see Table 6-2)
	<i>Value</i>	0x03	Slow Protocol Subtype value for OAM (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x01	Modify <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	M	Set manager M MAC address as the destination for resulting VLCPDUs.
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x03	Modify <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0xA8-C8	Set Ethertype to be equal to <i>ETHERTYPE_VLC</i> in the resulting VLCPDUs.
<i>RuleTLV</i> (termination)	<i>Type</i>	0x00	This is a termination (end-of-rule) TLV (see Table 8-4)
	<i>Length</i>	0x04	TLV length is 4 octets
	<i>Operation</i>	0x00	Filled with zeros when not used (see Table 8-4, note)
	<i>FieldCode</i>	0x00	Filled with zeros when not used (see Table 8-4, note)

1

2

Table 8A-138—Contents of *VLC_CONFIG* message

Field	Subfield	Value	Description
<u><i>DstAddress</i></u>	n/a	X	<i>VLC_CONFIG</i> VLCPDU directed to bridge X
<u><i>SrcAddress</i></u>	n/a	any	Source address of a device that issued the <i>VLC_CONFIG</i> VLCPDU
<i>LengthType</i>	n/a	0xA8-C8	Ethertype value identifying VLCPDUs (see 5.1)
<i>Subtype</i>	n/a	0x00	VLCPDU carrying <i>VLC_CONFIG</i> message
<i>MsgCode</i>	<i>MsgType</i>	0x0	This message is a Request (see Table 8-1)
	<i>RequestCode</i>	0x1	Request to add a rule (see Table 8-1)
<i>MsgSequence</i>	<i>MsgCounter</i>	0x00-01	This request consists of a single message
	<i>EndOfSequence</i>	1	

Field	Subfield	Value	Description
<i>PortInstance</i>	<i>PortIndex</i>	3	The rule is to be provisioned for port #3
	<i>Direction</i>	0	The rule is to be provisioned for the transmit path (i.e., an egress rule)
<i>RuleId</i>	<u>n/a</u>	<u>0x00-00</u>	<u>In a request to add a rule, the <i>RuleId</i> field is set to zero (see § 3.3.2)</u>
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x01	Compare <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	M	The destination address is equal to MAC address of Manager M.
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x03	Compare <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0xA8-C8	VLC Ethertype value (see 5.1)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x05	TLV length is 5 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x06	Compare <i>FID_SUBTYPE</i> field (see Table 6-2)
	<i>Value</i>	0x03	VLC Subtype identifying OAM payload (see Table 5-1)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x01	Modify <i>FID_DST_ADDR</i> field (see Table 6-2)
	<i>Value</i>	0x01-80-C2-00-00-02	IEEE 802.3 Slow_Protocols_Multicast address (see IEEE Std 802.3, 57A.3)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x03	Modify <i>FID_LEN_TYPE</i> field (see Table 6-2)
	<i>Value</i>	0x88-09	Slow Protocol Ethertype value (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i>	<i>Type</i>	0x00	This is a termination (end-of-rule) TLV (see Table 8-4)

Field	Subfield	Value	Description
(termination)	<i>Length</i>	0x04	TLV length is 4 octets
	<i>Operation</i>	0x00	Filled with zeros when not used (see Table 8-4, note)
	<i>FieldCode</i>	0x00	

1

2

Table 8A-1410—Contents of *VLC_CONFIG* message

Field	Subfield	Value	Description
<u><i>DstAddress</i></u>	n/a	M	<i>VLC_CONFIG</i> VLCPDU directed to Manager M
<u><i>SrcAddress</i></u>	n/a	any	Source address of the device that issued the <i>VLC_CONFIG</i> VLCPDU
<i>LengthType</i>	n/a	0xA8-C8	Ethertype value identifying VLCPDUs (see 5.1)
<i>Subtype</i>	n/a	0x00	VLCPDU carrying <i>VLC_CONFIG</i> message
<i>MsgCode</i>	<i>MsgType</i>	0x0	This message is a Request (see Table 8-1)
	<i>RequestCode</i>	0x1	Request to add a rule (see Table 8-1)
<i>MsgSequence</i>	<i>MsgCounter</i>	0x00-01	This request consists of a single message
	<i>EndOfSequence</i>	1	
<i>PortInstance</i>	<i>PortIndex</i>	1	The rule is to be provisioned for port #1
	<i>Direction</i>	0	The rule is to be provisioned for the transmit path (i.e., an egress rule)
<u><i>RuleId</i></u>	<u>n/a</u>	<u>0x00-00</u>	<u>In a request to add a rule, the <i>RuleId</i> field is set to zero (see 6.1.4.2)</u>
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x01	Compare FID_DST_ADDR field (see Table 6-2)
	<i>Value</i>	0x01-80-C2-00-00-02	IEEE 802.3 Slow_Protocols_Multicast address (see IEEE Std 802.3, 57A.3)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x03	Compare FID_LEN_TYPE field (see Table 6-2)

Field	Subfield	Value	Description
	<i>Value</i>	0x88-09	Slow Protocol Ethertype value (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x05	TLV length is 5 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x06	Compare FID_SUBTYPE field (see Table 6-2)
	<i>Value</i>	0x03	Slow Protocol Subtype value for OAM (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x01	Modify FID_DST_ADDR field (see Table 6-2)
	<i>Value</i>	S	Set Station S MAC address as the destination for resulting VLCPDUs.
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x03	Modify FID_LEN_TYPE field (see Table 6-2)
	<i>Value</i>	0xA8-C8	Set Ethertype to be equal to VLC Ethertype (ETHERTYPE_VLC) in the resulting VLCPDUs.
<i>RuleTLV</i> (termination)	<i>Type</i>	0x00	This is a termination (end-of-rule) TLV (see Table 8-4)
	<i>Length</i>	0x04	TLV length is 4 octets
	<i>Operation</i>	0x00	Filled with zeros when not used (see Table 8-4 note)
	<i>FieldCode</i>	0x00	

1

2

Table 8A-1512—Contents of VLC_CONFIG message

Field	Subfield	Value	Description
<u><i>DstAddress</i></u>	n/a	S	VLC_CONFIG VLCPDU directed to Station S
<u><i>SrcAddress</i></u>	n/a	any	Source address of the device that issued the VLC_CONFIG VLCPDU
<i>LengthType</i>	n/a	0xA8-C8	Ethertype value identifying VLCPDUs (see 5.1)
<i>Subtype</i>	n/a	0x00	VLCPDU carrying VLC_CONFIG message

Field	Subfield	Value	Description
<i>MsgCode</i>	<i>MsgType</i>	0x0	This message is a Request (see Table 8-1)
	<i>RequestCode</i>	0x1	Request to add a rule (see Table 8-1)
<i>MsgSequence</i>	<i>MsgCounter</i>	0x00-01	This request consists of a single message
	<i>EndOfSequence</i>	1	
<i>PortInstance</i>	<i>PortIndex</i>	0	The rule is to be provisioned for port #0
	<i>Direction</i>	0	The rule is to be provisioned for the transmit path (i.e., an egress rule)
<i>RuleId</i>	<i>n/a</i>	<u>0x00-00</u>	<u>In a request to add a rule, the <i>RuleId</i> field is set to zero (see 8.1.3.2)</u>
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x01	Compare FID_DST_ADDR field (see Table 6-2)
	<i>Value</i>	0x01-80-C2-00-00-02	IEEE 802.3 Slow_Protocols_Multicast address (see IEEE Std 802.3, 57A.3)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x03	Compare FID_LEN_TYPE field (see Table 6-2)
	<i>Value</i>	0x88-09	Slow Protocol Ethertype value (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (condition)	<i>Type</i>	0xCO	This is a condition TLV (see Table 8-4)
	<i>Length</i>	0x05	TLV length is 5 octets
	<i>Operation</i>	0x11	Comparison for equality (see Table 6-1)
	<i>FieldCode</i>	0x06	Compare FID_SUBTYPE field (see Table 6-2)
	<i>Value</i>	0x03	Slow Protocol Subtype value for OAM (see IEEE Std 802.3, 57A.4)
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x0A	TLV length is 10 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)
	<i>FieldCode</i>	0x01	Modify FID_DST_ADDR field (see Table 6-2)
	<i>Value</i>	M	Set Manager M MAC address as the destination for resulting VLCPDUs.
<i>RuleTLV</i> (action)	<i>Type</i>	0xAC	This is an action TLV (see Table 8-4)
	<i>Length</i>	0x06	TLV length is 6 octets
	<i>Operation</i>	0xCE	Change (replacement) of a field (see Table 6-3)

Field	Subfield	Value	Description
	<i>FieldCode</i>	0x03	Modify FID_LEN_TYPE field (see Table 6-2)
	<i>Value</i>	0xA8-C8	Set Ethertype to be equal to VLC Ethertype (ETHERTYPE_VLC) in the resulting VLCPDUs.
<i>RuleTLV</i> (termination)	<i>Type</i>	0x00	This is a termination (end-of-rule) TLV (see Table 8-4)
	<i>Length</i>	0x04	TLV length is 4 octets
	<i>Operation</i>	0x00	Filled with zeros when not used (see Table 8-4 note)
	<i>FieldCode</i>	0x00	

1

2