

Network Working Group  
Request for Comments: 3619  
Category: Informational

S. Shah  
M. Yip  
Extreme Networks  
October 2003

Extreme Networks'  
Ethernet Automatic Protection Switching (EAPS)  
Version 1

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

This document describes the Ethernet Automatic Protection Switching (EAPS) (tm) technology invented by Extreme Networks to increase the availability and robustness of Ethernet rings. An Ethernet ring built using EAPS can have resilience comparable to that provided by SONET rings, at a lower cost and with fewer constraints (e.g., ring size).

1. Introduction

Many Metropolitan Area Networks (MANs) and some Local Area Networks (LANs) have a ring topology, as the fibre runs. The Ethernet Automatic Protection Switching (EAPS) technology described here works well in ring topologies for MANs or LANs.

Most MAN operators want to minimise the recovery time in the event that a fibre cut occurs. The Ethernet Automatic Protection Switching (EAPS) technology described here converges in less than one second, often in less than 50 milliseconds. EAPS technology does not limit the number of nodes in the ring, and the convergence time is independent of the number of nodes in the ring.

## 2. Concept of Operation

An EAPS Domain exists on a single Ethernet ring. Any Ethernet Virtual Local Area Network (VLAN) that is to be protected is configured on all ports in the ring for the given EAPS Domain. Each EAPS Domain has a single designated "master node". All other nodes on that ring are referred to as "transit nodes".

Of course, each node on the ring will have 2 ports connected to the ring. One port of the master node is designated as the "primary port" to the ring, while the other port is designated as the "secondary port".

In normal operation, the master node blocks the secondary port for all non-control Ethernet frames belonging to the given EAPS Domain, thereby avoiding a loop in the ring. Existing Ethernet switching and learning mechanisms operate per existing standards on this ring. This is possible because the master node makes the ring appear as though there is no loop from the perspective of the Ethernet standard algorithms used for switching and learning. If the master node detects a ring fault, it unblocks its secondary port and allows Ethernet data frames to pass through that port. There is a special "Control VLAN" that can always pass through all ports in the EAPS Domain, including the secondary port of the master node.

EAPS uses both a polling mechanism and an alert mechanism, described below, to verify the connectivity of the ring and quickly detect any faults.

### 2.1. Link Down Alert

When a transit node detects a link-down on any of its ports in the EAPS Domain, that transit node immediately sends a "link down" control frame on the Control VLAN to the master node.

When the master node receives this "link down" control frame, the master node moves from the "normal" state to the ring-fault state and unblocks its secondary port. The master node also flushes its bridging table, and the master node also sends a control frame to all other ring nodes, instructing them to flush their bridging tables as well. Immediately after flushing its bridging table, each node begins learning the new topology.

## 2.2. Ring Polling

The master node sends a health-check frame on the Control VLAN at a user-configurable interval. If the ring is complete, the health-check frame will be received on its secondary port, where the master node will reset its fail-period timer and continue normal operation.

If the master node does not receive the health-check frame before the fail-period timer expires, the master node moves from the normal state to the "ring-fault" state and unblocks its secondary port. The master node also flushes its bridging table and sends a control frame to all other nodes, instructing them to also flush their bridging tables. Immediately after flushing its bridge table, each node starts learning the new topology. This ring polling mechanism provides a backup in the event that the Link Down Alert frame should get lost for some unforeseen reason.

## 2.3. Ring Restoration

The master node continues sending periodic health-check frames out its primary port even when operating in the ring-fault state. Once the ring is restored, the next health-check frame will be received on the master node's secondary port. This will cause the master node to transition back to the normal state, logically block non-control frames on the secondary port, flush its own bridge table, and send a control frame to the transit nodes, instructing them to flush their bridging tables and re-learn the topology.

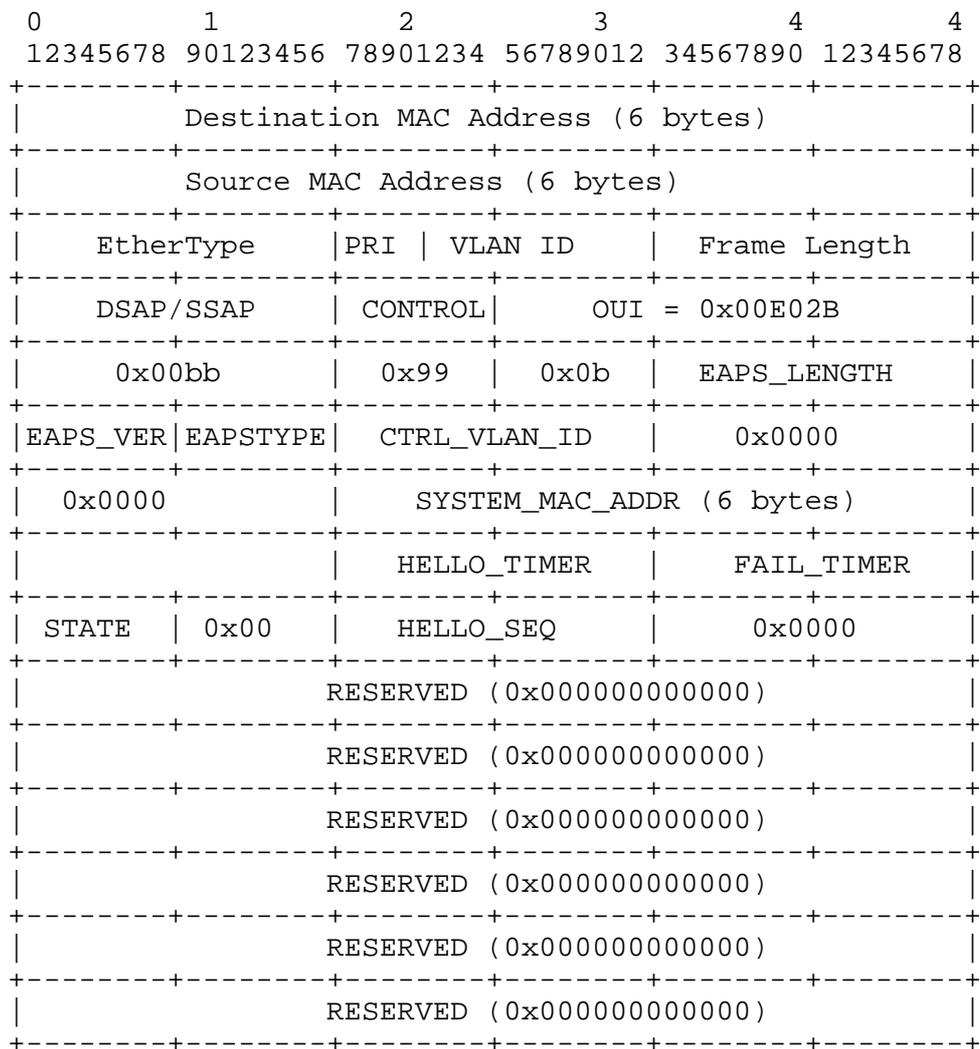
During the time between the transit node detecting that its link is restored and the master node detecting that the ring is restored, the secondary port of the master node is still open -- creating the possibility of a temporary loop in the topology. To prevent this, the transit node will place all the protected VLANs transiting the newly restored port into a temporary blocked state, remember which port has been temporarily blocked, and transition into the "pre-forwarding" state. When the transit node in the "pre-forwarding" state receives a control frame instructing it to flush its bridging table, it will flush the bridging table, unblock the previously blocked protected VLANs on the newly restored port, and transition to the "normal" state.

### 3. Multiple EAPS Domains

An EAPS-enabled switch can be part of more than one ring. Hence, an EAPS-enabled switch can belong to more than one EAPS Domain at the same time. Each EAPS Domain on a switch requires a separate instance of the EAPS protocol on that same switch, one instance per EAPS-protected ring.

One can also have more than one EAPS domain running on the same ring at the same time. Each EAPS Domain has its own unique master node and its own set of protected VLANs. This facilitates spatial reuse of the ring's bandwidth.

#### EAPS Frame Format



Where:

Destination MAC Address is always 0x00e02b000004.  
PRI contains 3 bits of priority, with 1 other bit reserved.  
EtherType is always 0x8100.  
DSAP/SSAP is always 0xAAAA.  
CONTROL is always 0x03.  
EAPS\_LENGTH is 0x40.  
EAPS\_VERS is 0x0001.  
CTRL\_VLAN\_ID is the VLAN ID for the Control VLAN in use.  
SYSTEM\_MAC\_ADDR is the System MAC Address of the sending node.  
HELLO\_TIMER is the value set by the Master Node.  
FAIL\_TIMER is the value set by the Master Node.  
HELLO\_SEQ is the sequence number of the Hello Frame.

EAPS Type (EAPSTYPE) values:

HEALTH = 5  
RING-UP-FLUSH-FDB = 6  
RING-DOWN-FLUSH-FDB = 7  
LINK-DOWN = 8  
All other values are reserved.

STATE values:

IDLE = 0  
COMPLETE = 1  
FAILED = 2  
LINKS-UP = 3  
LINK-DOWN = 4  
PRE-FORWARDING = 5  
All other values are reserved.

#### 4. Security Considerations

Anyone with physical access to the physical layer connections could forge any sort of Ethernet frame they wished, including but not limited to Bridge frames or EAPS frames. Such forgeries could be used to disrupt an Ethernet network in various ways, including methods that are specific to EAPS or other unrelated methods, such as forged Ethernet bridge frames.

As such, it is recommended that users not deploy Ethernet without some form of encryption in environments where such active attacks are considered a significant operational risk. IEEE standards already exist for link-layer encryption. Those IEEE standards could be used to protect an Ethernet's links. Alternately, upper-layer security mechanisms could be used if it is more appropriate to the local threat model.

## 5. Intellectual Property Rights Notice

The IETF has been notified of intellectual property rights claimed in regard to some or all of the specification contained in this document. For more information, consult the online list of claimed rights.

## 6. Acknowledgement

This document was edited together and put into RFC format by R.J. Atkinson from internal documents created by the authors below. The Editor is solely responsible for any errors made during redaction.

## 7. Editor's Address

R. Atkinson  
Extreme Networks  
3585 Monroe Street  
Santa Clara, CA, 95051 USA

Phone: +1 (408)579-2800  
EMail: rja@extremenetworks.com

## 8. Authors' Addresses

S. Shah  
Extreme Networks  
3585 Monroe Street  
Santa Clara, CA, 95051

Phone: +1 (408)579-2800  
EMail: sshah@extremenetworks.com

M. Yip  
Extreme Networks  
3585 Monroe Street  
Santa Clara, CA, 95051

Phone: +1 (408)579-2800  
EMail: my@extremenetworks.com

## 9. Full Copyright Statement

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assignees.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

