dagfilter-loader Software Guide

EDM04-30

Protection Against Harmful Interference

When present on equipment this manual pertains to, the statement "This device complies with part 15 of the FCC rules" specifies the equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the Federal Communications Commission [FCC] Rules.

These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment.

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications.

Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at their own expense.

Extra Components and Materials

The product that this manual pertains to may include extra components and materials that are not essential to its basic operation, but are necessary to ensure compliance to the product standards required by the United States Federal Communications Commission, and the European EMC Directive. Modification or removal of these components and/or materials, is liable to cause non compliance to these standards, and in doing so invalidate the user’s right to operate this equipment in a Class A industrial environment.

Disclaimer

Whilst every effort has been made to ensure accuracy, neither Endace Technology Limited nor any employee of the company, shall be liable on any ground whatsoever to any party in respect of decisions or actions they may make as a result of using this information.

Endace Technology Limited has taken great effort to verify the accuracy of this manual, but nothing herein should be construed as a warranty and Endace shall not be liable for technical or editorial errors or omissions contained herein.

In accordance with the Endace Technology Limited policy of continuing development, the information contained herein is subject to change without notice.

Website

http://www.endace.com

Copyright 2009-2010 Endace Technology Ltd. All Rights reserved

Use of the Endace products described in this document is subject to the Endace Terms of Trade and the Endace End User License Agreement (EULA).

Contents

Introduction

The `dagfilter-loader` software is used to configure packet filtering and allows you to assign a color value to packets using user-defined filter rules sets. Filter rule sets are stored in a TCAM. The color value is used later by the CAT (Color Association Table) to either steer the packets into stream buffer(s) or to drop the packet. For further details on the CAT, see *EDM04-35 dagcat-setup Software Guide v2*.

Note: This version of `dagfilter-loader` operates differently from previous versions of `filter_loader` and `dagfilter-loader`.

This new version of `dagfilter-loader` implements filtering on the following DAG cards:

- 8.1SX
- 7.5G4
- 9.2X2

Please read *EDM04-31 Enhanced Packet Processing v2* before programming the `dagfilter-loader`. It explains the underlying architecture of the Enhanced Packet Processing.

How does filtering work?

The filtering module provides the ability for the user to define a number of filter rules that match one or more of the fields in the packet header. These rules are written one rule per line and each rule has an associated identifying number known as the color. Any packets that match the filter are then colored. This color value is subsequently used by the CAT to help steer the packet to specific stream buffer(s) or to drop the packet.

A user configurable TCAM (Ternary Content Addressable Memory) is used to simultaneously compare each bit of the packet's extracted fields against a series of bit masks (comprised of 0/1/don't care) that have been loaded into the TCAM by the application. These bit masks are defined in the filter rules specified by the user. The width of the TCAM is 389 bits and this provides for matching on the 14 header fields detailed in the following table. The maximum number of filter rules the TCAM at any one time is DAG card dependent, see the appropriate DAG Card User Guide for details.

When filtering each packet, the extracted header fields are presented to the TCAM, which simultaneously compares the extracted fields against each filter rule in the TCAM. Depending on the filter rules loaded into the TCAM, more than one match is possible per packet. No indication of multiple matches is provided – only the lowest numbered address that matches is returned (which equates to the first matching rule in the user defined rule file used to program the TCAM). If no match is found the TCAM returns the highest address supported by that TCAM (i.e. the last filter rule loaded). Typically an explicit "match everything" rule is included at the end of the filter record file loaded into the TCAM in order to ensure a rule match always occurs at a configured location. Alternatively you can set the `sram_miss_value` attribute, if a rule other than "match everything" is desired at the end. This would be used to specify a specific color value for packets that do not match any rules in the TCAM.

The rules in the TCAM are indexed by row and when a match is returned the row index is used as a lookup in the filter module SRAM that contains the user defined color values associated with each rule. The packet is then said to have been colored and this color value is subsequently used in the steering module as mentioned above.

In order to program the filter module, the user writes a number of filter rules in the filter rule set and tags each rule with a color (see Filter rules (page 9)). The rules are then loaded into the TCAM and the rule index and associated color (or tag) is stored in the color table. For historical reasons the filter module provides a 16 bit table for color values, however not all of these can be used (see Number of color bits (page 4)).

In order to provide a hotswap capability, the TCAM is split into two banks, each containing a filter rule set. Only the active bank is used for filtering the incoming packets. The inactive bank can be independently programmed and is toggled active automatically after programming. Only the inactive bank can be written to by `dagfilter-loader`. An error displays if the user attempts to program the active bank.

The attribute `activate_bank` can be queried to determine the currently active bank (see `dagconfig` attributes (page 13)). The active bank can also be toggled at any time by `dagfilter-loader`, allowing the user to switch between the filter rule sets on demand.

Filterable and Classifiable Packet Protocol Header Fields	
Physical Interface	Interface – 2 bits
Layer 2: Ethernet	Ethernet Type – 2 bytes
Layer 2: Packet-over-SONET	PoS Protocol Field – 2 bytes
Layer 2/3: VLAN (Ethernet only)	VLAN Tag – 2 bytes
Layer 2/3: Q-in-Q (Ethernet only)	Both VLAN Tags – 2 bytes 1st VLAN Tag; 2 bytes 2nd VLAN Tag
Layer 2/3: MPLS	First and last MPLS headers – 4 bytes top MPLS header; 4 bytes bottom MPLS header
Layer 3: IPv4	Source and Destination IP addresses – 4 bytes source address; 4 bytes destination address Next Protocol – 1 byte
Layer 3: IPv6	Source and Destination IP addresses – 16 bytes source IP address; 16 bytes destination IP address Next Header – 1 byte
Layer 4: TCP	Source and Destination Ports – 2 bytes source port; 2 bytes destination port TCP Flags – 6 bits
Layer 4: UDP	Source and Destination Ports – 2 bytes source port; 2 bytes destination port

Number of color bits

The CAT lookup table in the steering module has a maximum width of 12 bits. These bits can be either all used to contain color values or combined with Hash Load Balancing bits and/or interface / port number bits. As a result of this 12 bit width, only the 12 least significant bits in the color table is used in any CAT lookups.

It is up to the user to decide how these bits in the CAT are assigned (see *EDM04-35 dagcat-setup Software Guide v2* for further details). This decision determines the maximum value of the color number used in the filter rule.

If **no** Hash Load Balancing or interface / port filtering is used, all 12 bits in the CAT can be assigned for color, in which case the color value range is 0-4095. Conversely, if the HAT is configured for Hash Load Balancing across 32 bins (see *EDM04-31 Enhanced Packet Processing v2*) then 5 bits (of the 12) are required to store the Hash values. If interface / port filtering is also enabled, a further 2 bits are used in the CAT to store the interface / port values, leaving only 5 bits for color. This would impose a color range of 0-63.

Note: The interface / port filter bits are always the top 2 bits in the CAT table.
The Hash Load Balancing bits are always the least significant bits.

Note: If you reconfigure the CAT and change the Hash Load Balancing and/or interface / port bit settings, the number of color bits available to you in the CAT will change. You may then need to alter your color values to match the number of color bits available, which would also reprogramming the filter module for this change to take effect. If you do not check the color values, then the CAT will be filtering on a subset of the bits you have allocated and you will get unexpected results.

Example

Example 1

The following filter rules are defined:

1. Filter on destination port 80 - color 50.
2. Match all other packets - color 100.

This filter rule set is then programmed into the TCAM. In this configuration the TCAM has two rows used, 0 and 1.

Any packets that match row 0 (destination port 80) have row index 0. When looked up in the color table it returns color 50.

All other packets have row index 1 and the lookup returns color 100.

These color values are used by the CAT when steering these packets.

If the CAT was originally configured to use 16 bins for Hash Load Balancing (4 bits) this leaves 8 bits for color. This gives a maximum color value of 255. The color values 50 and 100 in the filter rule set are valid in this scenario.

Example 2

The following rule is added to the above example:

- Per interface / port filtering.

With this additional rule the CAT is reconfigured. This causes the top 2 bits to be used for interface / port filtering, leaving only 6 bits for color.

Since the top 2 bits previously used for color are now used for interface / port filtering, the maximum value of the color range is now 63. The effective color of the second filter rule is now 36 (as the top two bits have been lost). The CAT rule that steers packets with color 100 (previously all packets without destination port 80) will now not match anything and these packets will be dropped at the CAT.

Writing filter rules

Before you start writing filter rules you need to know:

- what you want to filter.
- how you want to allocate the 12 bits in the CAT (i.e. number of Hash Load Balancing bins to use and whether interface / port filtering is to be used).

With these details, you can write the filter rules with appropriate color values so that the steering will give you the required / expected output.

If possible, use color values between 0 and 63 - that way you will never have to reassign the color values.

If you use color values greater than 63, there will be situations where you will need to reassign color values. For further details see Number of color bits (page 4). Remember, before programming the CAT, check the number of color bits used in each situation and alter the color values if required.

Note: *The number of filter rules that can be programmed into the TCAM is different to the maximum color range.*
- The number of filter rules available is determined by the DAG card (and is the number of rows in the TCAM).
- The color range is determined by the color/Hash Load Balancing/interface /port bit allocation in the CAT and user configurable.

Requirements

The requirements for using the `dagfilter-loader` are:

- An DAG card with the appropriate filtering firmware image.
- `dagfilter-loader` version 1.11 or greater.
- DAG software (4.0.1 or greater).
  Customers with a current support contract can download this from the secure Endace website: https://support.endace.com/.

  Refer to *EDM04-01 DAG Software Installation Guide* for details on how to install and compile the DAG software.

Data capture

The following applications can capture data from DAG cards:

- DAG API (for example, `dagconvert, dagsnap` etc)
- Libpcap (0.9.7 or higher). You can download a copy from the following location:

  http://www.tcpdump.org

Related documents

The following is a list of documents referred to in this document. These are available from the Support section of the Endace website at https://support.endace.com/:

- *EDM04-01 DAG Software Installation Guide*
- *EDM04-22 InfiniBand Filter Software Guide*
- *EDM04-31 Enhanced Packet Processing v2*
- *EDM04-35 dagcat-setup Software Guide v2*
- *EDM11-01 ERF Types*

Glossary of terms

Term	Definition
CAT	Color Association Table
DMA	Direct Memory Access
ERF	Extended Record Format
GMR	Global Mask Register
HAT	Hash Association Table
HLB	Hash Load Balancing
TCAM	Ternary Content Addressable memory
WAN	Wide Area Network

Keywords

The following keywords can be used in a rule set. All of these keywords are optional, with the default action taken if the keyword is omitted.

General actions	Description
`dst-ip`	A 32-bit or 128-bit binary representation of an IP address. • In IPv6 mode the `dagfilter-loader` can either take IPv6 or IPv4 addresses. • In IPv4 mode the `dagfilter-loader` will only accept addresses of length 32-bits. A *IPv4 address* is always specified by using 32 -bits. A field length that is greater than 32-bits but less than 128-bits is assumed to be an error. A *IPv6 address* is always specified using 128-bits. **Note:** *If the l2-proto type is IPv4, yet IPv6 length fields are specified in the filter, the* `dagfilter-loader` *will generate an error.*
`dst-port`	A 16-bit binary representation of the destination port.

`iface {xx}`
`iface <num>`
	A binary ({xx}) or integer (<num>) representation of the interface. Binary range {00} - {11}, integer range 0 - 3. e.g. iface {-1} means iface 1 and iface 3.

`ip-prot`	An 8-bit binary representation of the IP protocol field.
`l2-proto{xx...xx}`	16 bits. Specifies the layer2 protocol. Eg, 0x0800 for IP V4, 0x86DD for IPv6.
`mpls-btm`	32-bit binary representation of a MPLS field to match the bottom MPLS label on the MPLS label stack. Cannot be specified with a filter rule set that includes VLAN filtering.
`mpls-top`	32-bit binary representation of a MPLS field to match the first MPLS label on the MPLS label stack. Cannot be specified with a rule-set that includes VLAN filtering.
`src-ip`	A 32-bit or 128-bit binary representation of the IP address. The src-ip field specifier follows that for the dst-ip.
`src-port`	A 16-bit binary representation of the source port.

`tcp`
`udp`
`icmp`
`igrp`
	All layer 3 protocol values will translate to the appropriate ip-prot value. For example:
	• `accept red tcp`
	is equivalent in IPFv2.0, to:
	• `accept red ip-prot {00000110}`
	however, the IPFv1.1 syntax will also be accepted.

`tcp-flags`	An 8-bit binary value representation of the TCP flags. Includes the 6 control bits and 2 most significant ECN bits.
`vlan-1`	16-bit binary representation of a VLAN tag to match the top VLAN tag. Cannot be specified with a filter rule set that includes MPLS filtering.
`vlan-2`	16-bit binary representation of a VLAN tag to match the bottom VLAN tag. Cannot be specified with a filter rule set that includes MPLS filtering.
`label-cnt {xxx}`	VLAN / MPLS label count. 3-bit binary representation of MPLS/VLAN label count. For MPLS the maximum supported count is 7 and for VLAN the maximum supported count is 2.

Filter rule syntax

Filter rules

Filter rules are one-line specifications used to describe characteristics of packets considered to be a "match", together with an action to take for matching packets. A filter rule set contains multiple filter rules and is manually written in the Endace filter format.

Syntax

Syntax	Description
`\`	Next line is part of the same filter rule.
`{xxx}`	Specifies a bit pattern. x can be replaced with 0,1,x. If replaced with 0 or 1 exact match for that bit (the mask will be set to 1). If x used the mask will be set to 0 (used as a wild card or don't care) for the rule.
`//`	Comment from the symbol to the end of the line. Comments must be on a separate line otherwise the filter rule will not function as expected.
`-`	Wild card

Filter rule processing

- Filters are evaluated from the top of the filter rule set to the bottom.
- Missing tokens are assumed to be wildcards.
- The first matching filter is the one selected.
- Associated RAM information behavior if rule matches:

  [tag] - an Integer value from 0-65535 . User specified.

Filter examples

*Note: The following examples have been wrapped onto several lines for best presentation. In the filter rule set each rule **must** be on a single line.*
Comments must be on a separate line otherwise the filter rule will not function as expected.

Example 1

The following filter rule set captures IP packets with:

- a source IP address (128-bit) (xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxe3) sent from any source port to the destination IP address (128-bit) (xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxb0) on port 50.
- all other packets will be assigned the default colour
(set by `dagconfig -S sram_miss_value=default_colour`)

```
93  src-ip{-----------------------------------------------------------------------
-------------------------------------------------11100011}
   src-port{----------------}
   dst-ip{------------------------------------------------------------------------
------------------------------------------------10110000}
   dest-port{0000000001010000}
```

Example 2

The following filter rule set captures IP packets with:

- a source IP address (128-bit) (xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxe3),
- MPLS-BOTTOM - (32-bit) (xxxx fxxx), and
- MPLS-TOP - (32-bit) (xxxx fxxx).
- all other packets will be assigned the default colour
(set by `dagconfig -S sram_miss_value=default_colour`)

```
89  src-ip{-----------------------------------------------------------------------
-------------------------------------------------11100011}
   mpls-btm{----  ---- ---- ---- 1111 ---- ---- ---- }
   mpls-top{----  ---- ---- ---- 1111 ---- ---- ---- }
```

Example 3

The following filter rule set illustrates filtering on a mutli-port DAG card. The following filter rule set captures IP packets:

- received on Port C (`iface {10}`), and
- with a top VLAN tag (16-bit) (xfxx),
- all other packets will be assigned the default colour
(set by `dagconfig -S sram_miss_value=default_colour`)

```
76  iface {10}
   vlan-1{---- 1111 ---- ----}
```

Processing packets

When a packet matches a filter rule it is deemed to be a **hit**. Otherwise it is considered a **miss**.

- If a **hit** occurs, the packet is colorized with SRAM VALUE. The packet is then passed on for further processing. **Note:** *Ensure you map these colors to a stream in the CAT (Color Association Table).*
- If a **miss** occurs, the packet is colorized as per the setting of `sram_miss_value` attribute (page 13).
Note: *To drop these packets ensure that no stream is mapped for this color in the CAT.*

```
dagconfig -S sram_miss_value=z
```

For further details, see *EDM04-35 dagcat-setup Software Guide v2*.

Using dagfilter-loader

The following lists the steps required to configure the `dagfilter-loader`:

Note: This assumes the DAG card is physically installed, has the appropriate filtering firmware installed and is configured. See the appropriate DAG Card User Guide for further information.

1. Write the file rules and create filter rule set(s).
2. Load filter rule set into the inactive bank.
   The inactive bank automatically becomes active. For further information see dagfilter-loader options (page 12).
3. Set the appropriate Bank to be active - optional.
   For further information see `dagconfig` attributes (page 13).
4. Set the default filter rule.
   For further information see `dagconfig` attributes (page 13).

Using previous filter rule sets

You may use filter rule sets created for previous versions of filtering, however:

- the `accept`, `reject pass and drop` keywords will be ignored.
- the `red` and `blue` keywords will be ignored.

This version of `dagfilter-loader` attempts to colorize the packet based on the filter rule - ignoring the above keywords.

dagfilter-loader options

The following table explains the `dagfilter-loader` command line options.

Short Option	Long Option	Explanation
	`--db <number>`	Defines which Bank (0-1) to use. Default = 0. If the `--db` option is not specified, the file rule set is loaded into the inactive bank.
`-d`	`--device`	DAG device to use. Default is `d0`.
`-f`	`--rules <filename>`	Name of the file containing the filter rule set to load.

`-h,`
`-?`
	`--usage`
	`--help`
		Displays the help pages for this application.

	`--iface <iface>`	Use to set or overwrite the port/interface information in a filter rule at the time the filter rule is being written to the TCAM. Applicable to multi-port DAG cards only.
	`--initialize` `--initialise`	Reset the filter rules sets in the TCAM - as opposed to hot-swapping the filter rule sets.

	`--init-ports <int>`
	`--init-ifaces <int>`
		Sets the number of the section per Bank. The number must be less than or equal to the number of ports/interfaces on the DAG card. The number of sections may different in each Bank.
		Only valid with the `--initialise` or `--initialize` option.

`-l`	`--link <linktype>`	Link type: valid values are 'ethernet', 'pos4chdlc', and 'pos4ppp'. Valid with `–parse-only` option.
	`--no-drop`	Don't drop packets routed to a full receive stream. The current release does not support packet dropping at the TCAM. To drop packets correctly configure the CAT. See *EDM04-31 Enhanced Packet Processing v2*.
`-p`	`--parse-as <parsetype>`	Filter type based on which the file has to be parsed. Only valid with `–parse-only` option.
`-r`	`--verify`	Verify (Read back) rules after loading. (Default is no verification).
`-V`	`--version`	Display the version information.
`-v`	`--verbose <level>`	Sets the verbosity level, from 0 (basic) to 3 (full).
`-w`	`--width <width>`	No configurable. The TCAM width. Width is 389 bits.
`-y`	`--parse-only`	Parse the file only, do not load into hardware. You need to specify filter type based on which file is to be parsed using `--parse-as`.

dagfilter-loader example

The following is an example of how the `dagfilter-loader` options can be used:

1. Initialize the TCAM:

```
dagfilter-loader -d0 --initialize
```

This operation must be performed after power up or after switching firmware images.

Note: *This operation clears all filter rule sets.*

2. Loading filter rule sets into two Banks;

```
dagfilter-loader -d0 --db 0 -f test1.rule
dagfilter-loader -d0 --db 1 -f test2.rule
```

Overwriting source interface/port

There are two bits that indicate the source interface/port in the bit pattern extracted from the packet by the classifier. To overwrite the interface/port information in a rule set use the `–iface` command.

Example

To overwrite the interface/port information in the `rule_set_0.rule` filter rule set and change the interface/port from `0` to `1`, use the following command:

```
dagfilter_loader –d0 --db 0 --iface 1 –f rule_set_0.rule
```

You may also use the `--iface` key word in a filter rule.

dagconfig attributes

The following `dagconfig` attributes are applicable to the DAG cards. Using the `dagconfig -S` and `-G` options you can set and get values of the listed attributes.

Option	Description
`ipf_enable`	Enables (on) or disables (off) IP Filtering on the DAG card. Default is On.
`sram_miss_value`	Used to set the color of the packets (as a decimal integer) that does not match any current filter rule.
`activate_bank`	`-S` - Sets which Bank is active, either Bank 0 (off) or Bank 1 (on). `-G` - Get the number of the currently active Bank.

For more details about `dagconfig`, see your DAG Card User Guide.

dagconfig examples

The following are examples of how the `dagconfig` attributes can be used:

- Determining which Bank is active:
  ```
  dagconfig –d0 -G activate_bank
  ```
- Swap which Bank is active - from Bank 0 (off) to Bank 1 (on) or vice versa:
  ```
  dagconfig –d0 -S activate_bank=on
  ```
- Setting the default filter rule in case miss to catch any packets not matched by any filter rule: The format is specified according to the standard color. **Note:** *This can be changed in the future.*
  ```
  dagconfig –d0 -S sram_miss_value=z
  ```
  Where `z` is a decimal integer.

Version History

Version	Date	Reason
1	August 2009	First release. Based on EDM04-22 and EDM04-28.
2	December 2009	3.4.2 release. Corrected software name to dagfilter-loader. Added new introduction. Updated dagfilter-loader options, dagconfig attributes. Removed information about multi-port DAG Cards. Altered document flow.
3	February 2010	3.4.3 release. Added 7.5G4 and multi-port. Added Filter example 3.
4	September 2010	4.0.1 release. Added 9.2X2 to list of supported DAG cards. Added label-cnt keyword. Updated --db option description. Added Overwriting source interface/port section.