ThinkServer User Guide: Configuring BMC LAN access and SNMP Traps using IPMI Tools commands

Note: Before using the information provided in this document, as well as the products that it supports, be sure to read and understand Appendix A: Notices.

Configuring BMC LAN access and SNMP Traps with IPMI Tools commands

1.0 Introduction

This user's guide provides instructions that help configure BMC LAN access and SNMP Trap function remotely for ThinkServer systems using IPMI Tools commands on a Linux environment. The document applies to **RD350, RD450, RD550 and RD650**.

The basic IPMI Tools Command format used in this document is [2]:

```
ipmitool -I lanplus -H <BMC IP Address> -U <BMC Account Username> -P <BMC Account
Password> <COMMAND>.
```

The <**COMMAND**> to be executed can have <**PARAMETER**> and <**DATA**> associated with it.

Some of the settings required to configure SNMP Traps will be stated in raw format; these instructions can be compared to machine code (commands, parameters and other values) in that they are represented by HEX numbers (example: 0xa1).

When executing raw commands remotely, the format used will be:

```
ipmitool -I lanplus -H <BMC IP Address> -U <BMC Account Username> -P <BMC Account
Password> RAW <RAW COMMAND>.
```

RAW indicates the use of raw commands and the <**RAW COMMAND**> to be executed can have <**RAW PARAMETER**> and <**RAW DATA**> associated with it.

For those who want to run IPMI Tools commands locally on the server (instead of doing so from a remote system), it is required that the local system have the OpenIPMI driver installed. In this case, the IPMI Tools command format to use would be:

```
ipmitool -I open <COMMAND>.
```

The <**COMMAND**> to be executed can have <**PARAMETER**> and <**DATA**> associated with it. When executing raw commands, the format would be:

```
ipmitool -I open RAW <RAW COMMAND>.
```

RAW indicates the use of raw commands and the <**RAW COMMAND**> to be executed can have <**RAW PARAMETER**> and <**RAW DATA**> associated with it.

Finally, IPMI Tools allows executing IPMI Tools commands in an IPMI Shell, which can help reduce most of the repetitive parts associated with the execution of an IPMI Tools command. For example, instead of having to use command line

> ipmitool -I lanplus -H <**BMC IP Address**> -U <**BMC Account Username**> -P <**BMC Account Password**> <**COMMAND**> <**PARAMETER**> <**DATA**>

every time we want to execute a specific command, open an IPMI shell using the **shell** command.

> ipmitool -I lanplus -H <**BMC IP Address**> -U <**BMC Account Username**> -P <**BMC Account Password**> **shell**.

From this point on, IPMI commands can be executed just following the format:

> <**COMMAND**> <**PARAMETER**> <**DATA**>.

This is true as long as the commands are executed in the same terminal running the IPMI shell just opened. Please, note that sometimes executing IPMI Tools commands using an IPMI shell can result in slow execution time. Also, if changing system parameters, such as **BMC IP address** or **BMC username** and/or **password**, the shell will not work anymore to contact the remote system; another shell will have to be open using the new parameters.

Please, notice that throughout this document we will be assuming that the IP address for the BMC is **10.240.59.159** and the account **username** and **password** are: lenovo2 and len0v022, respectively. **Remember to change these values according to your system settings (example: a system such as ThinkServer RD650 will have default username = lenovo & password = len0vO).**

1.1 Installing IPMI Tools

In order to execute IPMI Tools Commands locally or from a remote system, IPMI Tools has to be installed. Install IPMI Tools by executing the following command from a Linux terminal:

```
yum install ipmitool
```

1.2 Suggested Approach

When configuring ThinkSerevers for SNMP Trap function, the easier approach is to make use of the ThinkServer System Manager (TSM). However, for customers that prefer the command line approach or plan to create a script to configure many servers remotely, IPMI Tools is an option.

For those interested in creating scripts to program many servers, we advise combining the TSM web interface with IPMI Tool Commands to reduce the possibility of errors; this is especially important when IPMI RAW commands are required.

As a first step, use TSM web interface on a base/test system to explore the configuration settings required to make Platform Events and SNMP Traps work, as well as enable/disable or configure the different settings required for these functions.

Second, use the GET IPMI commands provided in this document to obtain and explore the values that each of the different parameters/settings is assigned when they are set, enabled or disabled. **Label and keep** these results.

Third, use the values obtained with GET IPMI commands to prepare the corresponding SET IPMI commands (explained later). Eventually, these SET IPMI Commands will be used on other remote servers to program them.

Fourth, execute the generated SET IPMI commands on the base system or another test server to set, enable or disable each one of these configuration settings.

Fifth, use TSM web interface to confirm that the SET IPMI Commands have assigned the desired state (enabled/disabled) or value.

Finally, use the tested and validated SET IPMI commands (as suggested here) to generate a script to configure a ThinkServer system remotely.

1.3 References

More information can be obtained on these references:

[1] Lenovo ThinkServer Management Module (TSM):
https://download.lenovo.com/ibmdl/pub/pc/pccbbs/thinkservers/tsmug_en.pdf

[2] Ubuntu Manpage IPMI Tool - Ubuntu Manuals:
http://manpages.ubuntu.com/manpages/lucid/man1/ipmitool.1.html

[3] Intel Platform Management Interface (IPMI):
http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-home.html

[4] Sourceforge – IPMITool: http://sourceforge.net/projects/ipmitool/

2.0 Setting BMC LAN configuration settings

2.1 Determining current BMC LAN settings

Use the **lan print** command to determine the current configuration settings that allow remote management access to the TSM (formerly known as BMC), including the web user interface or Command Line Interface, through IPMI Tool Commands.

1. ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan print
2. For this example, some of the results are:
   a. IP Address Source: DHCP Address
   b. IP Address: 10.240.59.159
   c. Subnet Mask: 255.255.255.000
3. This command provides much more information that is not included in this document, but can be reviewed locally, if desired.

2.2 Source of BMC IP Address

Use the **lan set ipsrc** command to set the source of the BMC IP address to DHCP or Static.

1. Changing source to dhcp: **ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan set 1 ipsrc dhcp**
2. Changing source to static: **ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan set 1 ipsrc static**

Notice **lan set 1**, where 1 = channel 1. Channel information is required for this instruction and can be obtained using the **channel info** command: **ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 channel info**. Part of the information displayed by this command will start with string → Channel 0x1 info. In this case, the channel number is 1 (0x1), as expected.

2.3 Changing BMC IP Address

For this specific example, let us assume that 10.240.59.159 is a **static** address that we want to change to 10.240.59.160. Use **lan set ipaddr** command to change the BMC IP address from 10.240.59.159 to 10.240.59.160. Please, note that this new IP address will be used only for this example and we will continue using DHCP IP address 10.240.59.159 for the rest of this document.

Also notice that once the IP Address is changed, the Linux terminal that executed the instruction will appear to be hung waiting for a response and may return some error messages. This happens because it is expecting a response from the original IP Address, which for this example was 10.240.59.159. However, communicating to the new IP Address (for this example is 10.240.59.160) should work without problems.

1. **ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan set 1 ipaddr 10.240.59.160**
2. Now use lan print command to confirm that the new settings have been established (note that the IP address used after lanplus -H in this instruction line is now 10.240.59.160): **ipmitool -I lanplus -H 10.240.59.160 -U lenovo2 -P len0vO22 lan print**.

3.0 Configuring PEF Filters and enabling PET Event Traps

Log on to the TSM web interface, go to the second page and double click on **PEF Management** to access **Event Filters**, **Alert Policies** and **LAN Destination** settings. In order to configure these settings correctly, follow the steps provided below, starting with **LAN Destination**.

3.1 LAN Destination

LAN Destination is the IP Address of the Manager System or systems that will be notified when an alert is triggered. More than one LAN Destination can be added in this section.

3.1.1 Assigning IPv4 address to Destination1

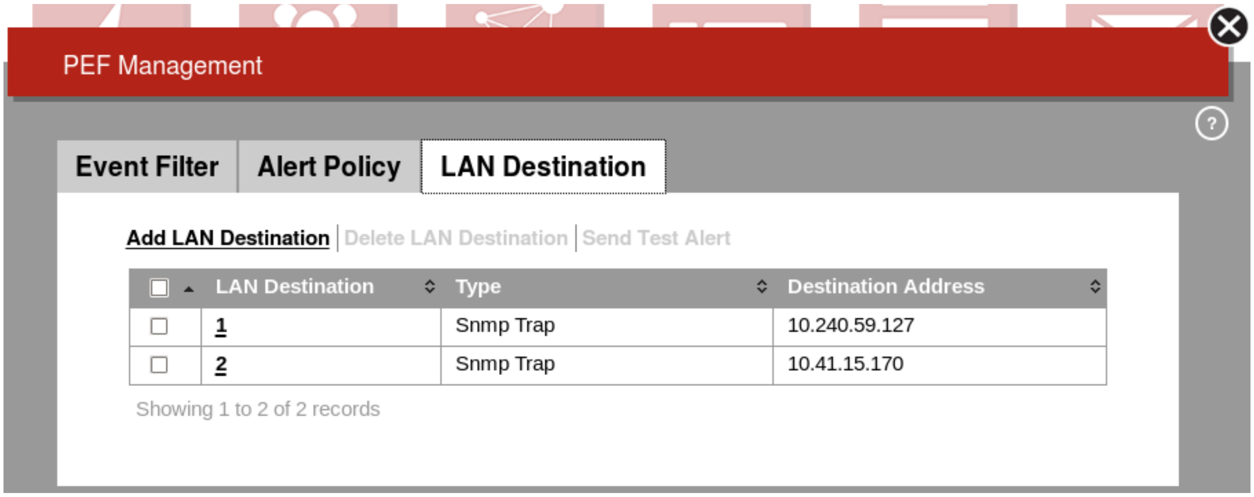
1. Use **lan alert set ipaddr** command to configure the IP address of the destination system (Manager System). For this example, we will assume that 10.240.59.127 is the IP address of destination 1:
    a. ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan alert set 1 1 ipaddr 10.240.59.127
        i. **alert set <channel number> <alert destination>** - For this example, the channel number is 1 and the destination is 1 → alert set 1 1 (make sure to keep a space between these two values; putting the two together will be equivalent to number 11, which will result in an error for this instruction).
    b. If adding a second IP address destination (example: 10.240.59.128 for Destination2), the instruction will look like this: ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan alert set 1 2 ipaddr 10.240.59.128 ← see the location of number 2, indicating destination 2.
2. Use **lan alert print** command to confirm that this configuration has been set. This instruction will provide the settings for all the IPv4 destinations available.
    a. ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan alert print
    b. **Note**: Sometimes, a Destination0 (zero) can be seen when using this command. It is advised not to use this destination because it is not supported by the TSM web interface; this destination is usually not displayed by TSM web interface.
3. If Channel number is different than 1, use **channel info** command **ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 channel info** to determine the correct value. Please, be advised that for these systems the channel number will always be 1.
    a. Channel information displayed will start with this string → Channel 0x1 info. In this case, the channel number is 1 (0x1), as expected.

3.1.2 Confirming settings with TSM web interface

In addition to using IPMI command **lan alert print** to confirm LAN destination settings, use the TSM web interface.

1. Access TSM web interface and log in. For this example, the username is lenovo2 and the password is len0vO22.
2. On the second page, click on PEF Management.
3. Select the **LAN Destination** tab

4. Notice the LAN destination settings:
    a. LAN Destination: 1
    b. Type: SNMP Trap
    c. Destination Address: 10.240.59.127



☐	LAN Destination	Type	Destination Address
☐	1	Snmp Trap	10.240.59.127
☐	2	Snmp Trap	10.41.15.170

Showing 1 to 2 of 2 records

Figure 1: LAN Destination Settings

3.2 Alert Policy

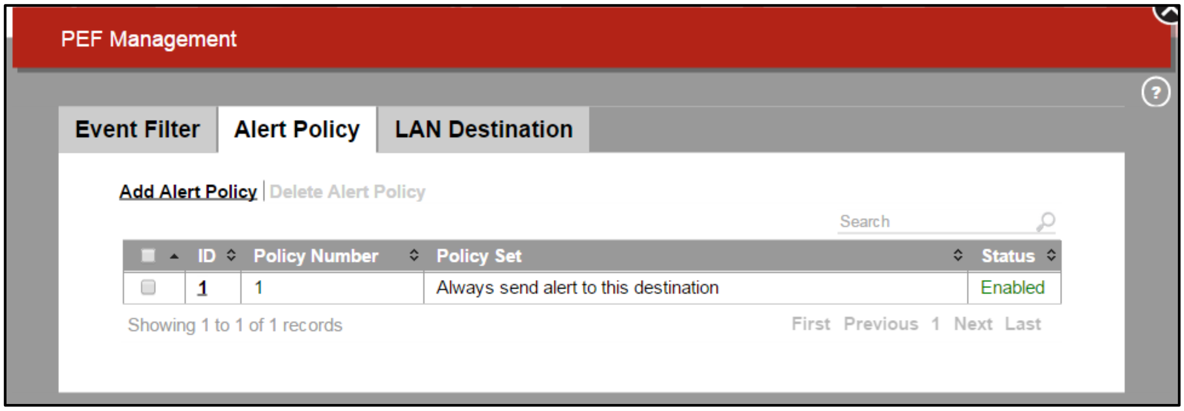
Alert Policy can be used to establish how an alert is handled for a specified destination. This is especially important if multiple destinations have been specified (each one having their own ID number). Because configuring Alert Policy settings requires the use of IPMI RAW commands, we will use a different approach.

We will start by configuring Alert Policies using the TSM web interface. Second, we will use IPMI GET command to retrieve the settings that were established with TSM web interface. Third, we will use results obtained with IPMI GET command to establish the proper IPMI SET command. Finally, we can use the IPMI SET command to program the Alert Policies on a desired system. (Keep in mind that we are using this approach because we are assuming that multiple systems have to be programmed this way.)

1. Log in to TSM web interface. (Username for this example is lenovo2 and password is len0vO22.)
2. On the second page, click on PEF Management.
3. Select the **Alert Policy** tab.
4. Click on **Add Alert Policy**, the Add Alert Policy window opens.
5. The alert will be assigned a **Policy Number**, but this number can be changed, if desired. (Note: this policy number will be used later when setting Event Filters, allowing event filters to determine how to execute an action.)
6. Change the **Status** to **ON** to enable the policy.
7. Under **Policy Set**, choose one of the options available. Choosing ***Always send alert to this destination*** will send an SNMP Trap to the destination every time the specified event occurs.
8. Scroll down to **LAN Destination** and choose the IP address of the Manager system that will be receiving the **Alert**.
9. The final option available under **Alert Policy** is ***Alert String***. The ***Alert String*** can be specific to an event (change setting to ON) or not related to a specific event. If choosing to make the Alert

String specific to an event, choose also an Alert String Key from the dropdown menu. This is useful for setting many alert strings for different trigger criteria. However, for the most part, the default value (OFF) will be enough.

10. Click the **Apply** button to save the settings and close the window.



ID	Policy Number	Policy Set	Status
1	1	Always send alert to this destination	Enabled

Showing 1 to 1 of 1 records

Figure 2: Alert Policy Settings

3.2.1 Use IPMI GET to determine configuration settings for Alert Policy 1 and Destination 1

Once the Alert Policy has been established with TSM web interface, we can use IPMI GET command to determine the values for each alert policy set. If multiple Alert Policies were set, keep each individual value that is obtained with the GET command, as we will need it to put together the equivalent IPMI SET command that can be used to program these values on other systems.

1. When Destination1 is configured, running command: ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 raw 0x04 0x13 0x09 **0x01** 0x00 should yield these results: 11 **01 18** 11 00. See the tables below for details about the instructions and results.

Table 1: Get Command				
0x04	0x13	0x09	**0x01**	0x00
			Destination 1	

Table 2: Result GET Command (Destination 1)				
11	01	18	11	00
Discard	ID Number	1 = Alert Policy 8 = Alert Policy Status	1 = Destination 1	Alert String

a. Discarding 11 (first hex data set or hex word from the left), **01** = ID number given to an Alert Policy,

b. **18** => each digit of this hex number represents various settings, see below.

   i. **1 = Policy Number** and can have values between 1 and F (F is the hex equivalent to decimal value 15)

   ii. **8 = Alert Policy Status** is **ON** and its **Policy Set Action** is ***"Always send alert to this destination"***. The tables below explain the meaning of all possible values for this digit; the tables are only provided as reference, because our approach will be to configure these settings using the TSM web user interface and extract the values with the appropriate IPMI GET command.

   iii. Please, note that for our example, we will use the hex value 18 when putting together the IPMI SET command later on (this is the value that we obtained with IPMI GET command).

Table 3: Alert Policy – Status ON

IPMI HEX Value	Equivalent Binary Representation	Set Policy Action
1**8**	0001 1000	Always send alert to this destination.
1**9**	0001 1001	If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set.
1**A**	0001 1010	If alert to previous destination was successful, do not send alert to this destination. Do not process any more entries in this policy set.
1**B**	0001 1011	If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different channel.
1**C**	0001 1100	If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different destination type.

Table 4: Alert Policy – Status OFF		
IPMI HEX Value	Equivalent Binary Representation	Set Policy Action
1**0**	0001 0000	Always send alert to this destination.
1**1**	0001 0001	If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set.
1**2**	0001 0010	If alert to previous destination was successful, do not send alert to this destination. Do not process any more entries in this policy set.
1**3**	0001 0011	If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different channel.
1**4**	0001 0100	If alert to previous destination was successful, do not send alert to this destination. Proceed to next entry in this policy set that is to a different destination type.

c. 11 is the fourth hex data set and the highlighted second digit represents the LAN destination that will receive the notification (in our example it is LAN Destination 1).

d. 00 is the fifth hex data set, which represents settings related to the **Alert String**. For this example, 00 = ***Event Specific*** is **OFF** and ***Alert String Key*** is **0**. However, these two digits will vary if either one of these two settings is changed using TSM web user interface.

2. If configuring an **Alert Policy** for any other LAN Destination, remember to run these commands for each new destination. For example, if **Alert Policy** 2 has been configured for LAN Destination 2:
   a. Running command: ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 raw 0x04 0x13 0x09 **0x02** 0x00 yields these results: 11 02 **18** 12 00

3.2.2 Use IPMI SET command to Configure Alert Policy 1 for LAN Destination1

After executing the Get PEF Configuration Parameters Command for Alert Policy 1 and LAN Destination 1, we obtained the results 11 **01 18** 11 00. Discarding the first hex data set on the left (11) we can use the SET PEF command to configure the same Alert Policy on any other system (as long as it is the same type of system) with these values.

1. ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 raw 0x04 0x12 0x09 **0x01 0x18 0x11 0x00**

Table 5: Values used with SET PEF Command

11	0x01	0x18	0x11	0x00
Discard	ID Number	Alert Policy and Status	Destination 1	Alert String

NOTE: Keep in mind that this process will have to be executed for every single **Alert Policy** that we choose to configure, example Alert Policy 2 for LAN Destination 2, etc. Also, it is important to **notice that the alert policy is required to make a PEF filter send an SNMP alert**, when triggered.

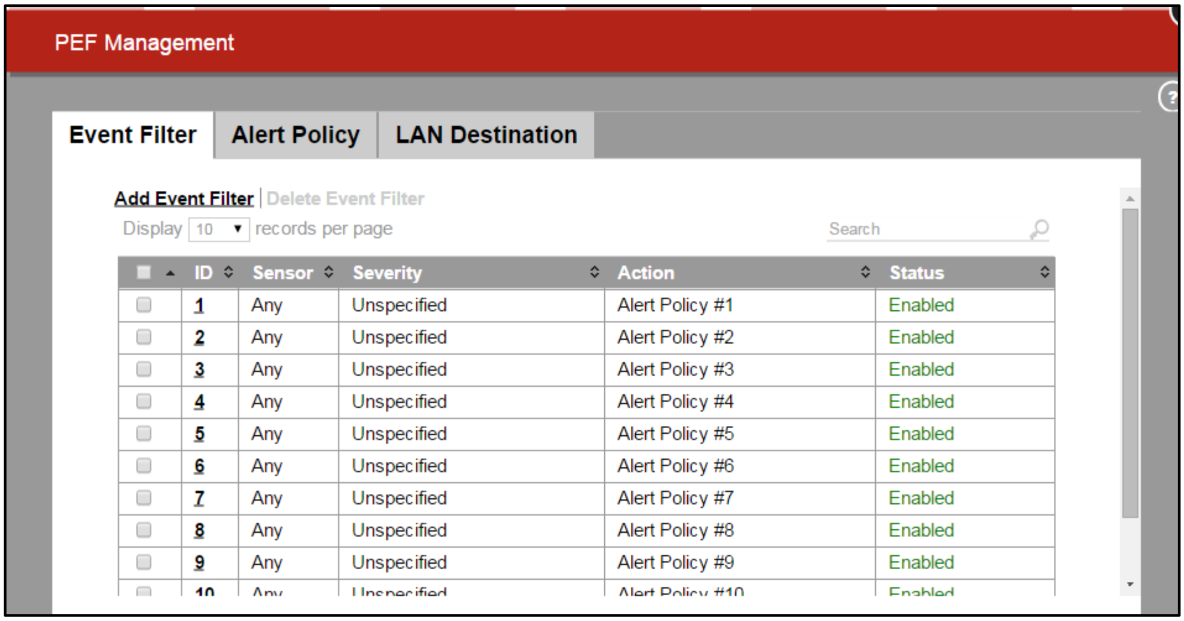
3.3 Configuring Platform Event Filters (PEF)

The Event Filter tab displays a table containing a list of event and filter options that have been predefined but should be configured to suit individual needs. The table has five columns: ID, Sensor, Severity, Action and Status.

The **Status** column indicates whether an event filter is enabled or disabled. Event filters with an enabled status are considered active and those with a disabled status are considered inactive. The **ID** column shows the number assigned to the defined filter. **Severity** highlights the significance level assigned to the event and **Action** establishes the alert policy that will be executed when the event occurs; it has to be defined for every single PEF Filter. **Note that without a defined Alert Policy, PEF filters do not execute any action**.

Defining and enabling Event Filters enables the monitoring of system sensors and establishes the conditions that can trigger an event alert (SNMP Alert). Because the settings for the Event Filters (PEF – Platform Event Filters) can be cumbersome, we strongly suggest using the TSM web interface to configure them. Then use the IPMI **GET PEF Configuration Parameters** command shown below to retrieve the specific IPMI configuration settings for each filter.

IMPORTANT NOTE: Remember to retrieve the **IPMI GET PEF Configuration data** for each individual filter and make sure to label and keep the data separately, as it will be used later to configure, enable or disable each individual filter using **IPMI SET PEF Configuration** commands. Compromising this data could result on drastic changes to these predefined filters, potentially affecting their expected behavior and function. If the filters are misconfigured or compromised using the wrong **IPMI SET PEF** command and cannot be recovered, use the TSM web user interface to reset the TSM to factory defaults. Keep in mind that after a factory reset, everything in the TSM will have to be configured from scratch.


PEF Management
Event Filter
Alert Policy
LAN Destination
Add Event Filter | Delete Event Filter
Display 10 records per page
Search

□	ID	Sensor	Severity	Action	Status
□	1	Any	Unspecified	Alert Policy #1	Enabled
□	2	Any	Unspecified	Alert Policy #2	Enabled
□	3	Any	Unspecified	Alert Policy #3	Enabled
□	4	Any	Unspecified	Alert Policy #4	Enabled
□	5	Any	Unspecified	Alert Policy #5	Enabled
□	6	Any	Unspecified	Alert Policy #6	Enabled
□	7	Any	Unspecified	Alert Policy #7	Enabled
□	8	Any	Unspecified	Alert Policy #8	Enabled
□	9	Any	Unspecified	Alert Policy #9	Enabled
□	10	Any	Unspecified	Alert Policy #10	Enabled

Figure 3: Platform Event Filter Settings

3.3.1 Adding or Configuring a New Event Filter

Using the TSM web user interface, configure every PEF Filter, as needed. The instructions below will take you through the process.

1. Click on **Add Event Filter** or choose from the list of predefined event filters. To choose from the list of predefined filters, click on any of the ID numbers (link) to edit any existing filter. **Since the TSM already has 15 predefined filters, we recommend editing and configuring those filters**. If you need more than 15 PEF Filters, then go ahead and create additional ones.
2. The "Add Event Filter" window opens (If editing a filter, the Edit Event Filter window is displayed; ***both windows have the same configuration settings***).
3. This settings window is divided into five categories: **Event Filter Configuration**, **Sensor Configuration**, **Filter Action**, **Generator ID**, and **Event Data**.
    a. **Event Filter Configuration** – choose whether this event filter will be enabled (Status ON) or disabled (Status OFF). Another setting available is ***Severity***, which has seven available options: ***unspecified***, ***monitor***, ***information***, ***normal***, ***non-critical***, ***critical*** and ***non-recoverable***. Choose the severity setting required for this event. (This is the severity of the event that will trigger the alert notification.) Refer to the IPMI spec for more information about severity settings.
    b. **Sensor Configuration**– Has three available settings: ***Sensor Type***, ***Sensor Name*** and ***Events***. These settings are used to define (filter or narrow down) the sensor(s) and/or event(s) that will trigger an action or an alert (or both).
        i. **Sensor Type** – Can be used to choose a specific sensor category (example: Temperature, Voltage, Fans, etc) to help narrow down the criteria that can trigger an event. **Choosing <u>All Events</u> from the list (drop down menu) allows the event filter to trigger on any sensor category whose sensor has generated an event**. This is true if Sensor Name and Events are also set to All Events.

ii. **Sensor Name** – Allows choosing a specific sensor available from the selected Sensor Type category. **Choosing <u>All Events</u> allows the event filter to trigger on any event generated by any sensor in the selected category** (***Sensor Type***).
   - As an example, let us assume that ***Sensor Type*** was selected to be *Processor*. For this category, **Sensor Name** has ***CPU Fault*** and ***CPU Usage*** as available options. **Choosing <u>All Events</u> for this Sensor Name would trigger a notification on any of these two options**.

iii. **Events** – Allows choosing between **<u>All Events</u>** or **<u>Specific Sensor Events</u>** related to **Sensor Name**. **We advise choosing <u>All Events</u> for this setting**, as sensor-specific events generate another tab, called **Sensor Events**, which requires some expert consideration to configure. Also, keep in mind that option **Specific Sensor Events** will be available for choosing only if a specific **Sensor Name** is chosen. If **Sensor Name** is All Events, **Specific Sensor Events** will not be available in the menu.

c. **Filter Action** – Choose the policy (***Alert Policy Number***) that will be executed with this event filter and whether a ***Power Action*** is required. The default Power Action (None) is **usually desired**, because we are trying to receive a notification; but the other options available are ***Power Down***, ***Power Reset*** and ***Power Cycle***. If choosing one of the available actions, keep in mind that the selected action will be executed when an event triggers this filter.

d. **Generator ID** – Used to identify the device that has generated the event:

   i. **Important Note:** 0xFF – allows more events – This is the default value for Generator ID 1 and Generator ID 2; we recommend keeping these values. Other values are 0x00 for Channel number (BMC or IPMB) and 0x02 for Software ID. Refer to the IPMI spec for more information.

   ii. Other settings such as Event Generator (Slave Type/Software Type), Slave Address/Software ID, Channel Number and IPMB Device LUN are also available. In order to change these settings, turn OFF setting **Enter Generator ID with Raw Data**. We recommend keeping the default value. Refer to IPMI spec for more information.

e. **Event Data** – We recommend keeping the default values for these settings. However, if changing these settings is required, we recommend referring to the IPMI spec for this information; expert knowledge of these settings is required.

f. Click the **Apply** button to save the settings and close the window.

g. The PEF Filter should be configured. Repeat this process to configure any other PEF Filter.

3.3.2 Determining configuration settings for each individual filter

Having configured the PEF Filters using the TSM web user interface, follow the steps provided below to retrieve the specific IPMI parameters for each configured PEF Filter using the **GET PEF Configuration Parameters** command shown below.

1. Use **Get PEF Configuration Parameters** command to retrieve the current configuration settings for each of the predefined PEF filters. For this example, assume Default PEF Filter values for PEF Filter 1 and Alert Policy Number = 1.

   a. Filter 1 (**0x01**): ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 raw 0x04 0x13 0x06 **0x01** 0x00

    i. The results could look like this: 11 01 80 01 01 10 ff ff ff ff ff ff 00 00 00 00 00 00 00 00 00, up to 22 possible values. The spec allows for 21 possible data values (when using SET PEF Command), which means that the first two bytes (11) can be discarded or ignored later. See the tables below for details.

Table 6: Results GET PEF Configuration Parameters (1 to 6)

1	2	3	4	5	6
0x11	0x01	0x80	0x01	0x01	0x10
Discard	ID	Status	Power Action	Alert Policy Number	Severity

Table 7: Results GET PEF Configuration Parameters (7 to 12)

7	8	9	10	11	12
0xff	0xff	0xff	0xff	0xff	0xff
Generator ID 1	Generator ID 2	Sensor Type	Sensor Name	Event Trigger (Event Data Configuration)	Sensor Events

Table 8: Results GET PEF Configuration Parameters (13 to 17)

0x13	0x14	0x15	0x16	0x17
ff	00	00	00	00
Events (Sensor Configuration)	Event Data Configuration 1	Event Data Configuration 1	Event Data Configuration 1	Event Data Configuration 2

Table 9: Results GET PEF Configuration Parameters (18 to 22)

0x18	0x19	0x20	0x21	0x22
00	00	00	00	00
Event Data Configuration 2	Event Data Configuration 2	Event Data Configuration 3	Event Data Configuration 3	Event Data Configuration 3

b. Filter 2 (**0x02**): ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 raw 0x04 0x13 0x06 **0x02** 0x00 (For this example, assume Default PEF Filter values for PEF Filter 2 and Alert Policy Number = 1.)

    i. The results could look like this: 11 02 80 01 01 00 etc., up to 22 possible values. The spec allows for 21 possible data values, which means that the first two bytes (11) can be discarded or ignored later.

c. Follow this process to retrieve the parameters for any other filter configured with TSM web user interface.

d. Note: If the result of these instructions shows fewer than 22 sets of data (1 set = 2 bytes = 1 word), we will have to add zeros later at the time of setting each individual filter, but this rarely happens.

3.3.3 Configuring Individual PEF Filters Using SET PEF Configuration Parameters Commands

Once the configuration settings have been obtained for each filter using the **GET PEF Configuration Parameters** command, we can use these results to configure PEF Filters on other systems (same model or similar TSM) using **SET PEF Configuration Parameters** command.

1. ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 raw 04 0x12 0x06 **0x01 0x80 0x01** 0x01 0x10 0xff 0xff 0xff 0xff 0xff 0xff 0xff 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
   i. For this example, we used the results obtained for PEF Filter 1 (0x01) on the previous section.

Notice that we added 0x to the front of each array value to denote HEX format; not doing so could result on failure to execute the command correctly.

Note: Another way to show the list of Filters available for configuration is to run IPMI command PEF List: ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 pef list. The results obtained from running this command can be compared to the list of Event Filters shown on the TSM web user interface.

3.4 Establishing Community Name (String)

The Community Name is not displayed on the TSM web interface for these systems. Instead, we have to use IPMI commands to retrieve the value and change it. The first step is to determine the channel that has the connection.

1. Determine the channel that has the connection – Usually the active channel for the dedicated management port is 1 (0x01); in case the channel has a different number assigned to it, do the following:
   a. Use **channel info** command: ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 channel info
      i. Channel information displayed will start with this string → Channel 0x1 info. In this case, the channel number is 1 (0x1), as expected.
      ii. If the channel number is different than 1, change this value on the next instruction set.
2. Retrieving SNMP community name:
   a. Use **lan print** command to retrieve information about the current community name (referred as SNMP Community String): ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan print
3. Changing SNMP community name:
   a. Use **lan set snmp** command to change community name from community1 to community2: ipmitool -I lanplus -H 10.240.59.159 -U lenovo2 -P len0vO22 lan set 1 snmp community2

Appendix A: Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.

1009 Think Place - Building One

Morrisville, NC 27560

U.S.A.

Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT,MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

Lenovo, the Lenovo logo, and ThinkServer are trademarks of Lenovo in the United States, other countries, or both.

Windows is a trademark of the Microsoft group of companies.

Other company, product, or service names may be trademarks or service marks of others.