

System Load Threshold Reached, Load Average too High

IPWorks

OPERATING INSTRUCTIONS

Copyright

© Ericsson AB 2017, 2018. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
1.1	Alarm Description	1
1.2	Prerequisites	2
2	Procedure	2
2.1	Analyzing Alarm	2
2.2	Actions	4



System Load Threshold Reached, Load Average too High



1 Introduction

This instruction concerns alarm handling.

1.1 Alarm Description

The alarm is raised when the total CPU load exceeds the threshold value.

The possible alarm causes, fault locations and impact are explained in Table 1.

Table 1 Alarm Causes

Alarm Cause	Description	Fault Reason	Fault Location	Impact
System overload	The total CPU load exceeds the threshold value.	The system is overloaded.	System dimensioning	Service downtime or reduced service capacity when the problem is too often detected or when it persists.
One or more processes occupy the CPU	The total CPU load exceeds the threshold value.	Some processes are suspended, occupying large CPU load.	Software component	Software crash leading to service downtime or reduced service capacity depending on redundancy.

Note: This alarm can appear as a result of a maintenance activity.

The alarm attributes are listed and explained in Table 2.

Table 2 Alarm Attributes

Attribute Name	Attribute Value
Major Type	193
Minor Type	868353
Source	One of the following: <ul style="list-style-type: none"> • ManagedElement=1, SystemFunctions=1, Pm=1, PmJob=OSDefaultPmThresholdJob, MeasurementReader=OSDefaultPmThresholdMr:OSProcessingUnit=PL-3 me>, ERIC-LINUX_PAYLOAD-*



Attribute Name	Attribute Value
Specific Problem	System Load Threshold Reached, Load Average too high
Event Type	environmentalAlarm (6)
Probable Cause	x733ThresholdCrossed(351)
Additional Text	System Load threshold limit exceeded
Perceived Severity	critical (3)

1.2 Prerequisites

This section provides information on the documents, tools, and conditions that apply to the procedure.

1.2.1 Documents

This instruction references the following document:

- *IPWorks Measurement List*

1.2.2 Tools

Not applicable.

1.2.3 Conditions

Before starting this procedure, ensure that the following condition is met:

- A CPU Load Exceed Threshold alarm is raised.

2 Procedure

This section describes the procedure to follow when this alarm is received.

2.1 Analyzing Alarm

Do the following:



1. Make sure if the alarm severity is `Major` or `Critical`.

- Yes: Continue with the next step.
- No: The alarm severity is `Minor`. No further immediate action is needed from this procedure. If the alarm severity level rises, re-enter this procedure.

2. Log on to the host to access a Linux® shell.

```
#ssh <user>@<hostname> -p 22
```

The `hostname` is a part of the alarm attribute `Source`.

3. Show the current CPU usage.

```
#grep cpu /proc/stat
```

Example output:

```
cpu 3230142 178606 3965500 1170305669 205972 197 581121 0 0 0
cpu0 236802 8986 353998 58205108 45254 16 24624 0 0 0
cpu1 254928 18629 308858 58301106 6186 15 15427 0 0 0
cpu2 251278 8929 275700 58317394 4748 7 44985 0 0 0
cpu3 294922 10750 280483 58285279 5967 15 20157 0 0 0
cpu4 287733 11930 280030 57870496 2097 27 77899 0 0 0
cpu5 264204 14193 307790 58071692 4259 19 64210 0 0 0
cpu6 291224 9060 268330 58322866 2276 16 14126 0 0 0
cpu7 299320 9677 289972 58256060 5591 9 21184 0 0 0
cpu8 297335 6545 276951 58249189 3899 10 22685 0 0 0
cpu9 282245 20588 301752 58215901 44119 12 34439 0 0 0
cpu10 23398 16257 87483 58936207 10058 1 7040 0 0 0
cpu11 45559 9957 71927 58955657 6982 2 3394 0 0 0
cpu12 75008 4053 73203 58907246 3878 2 18167 0 0 0
cpu13 16413 3538 51189 59026597 4173 0 3418 0 0 0
cpu14 230939 4411 482998 57347890 561 35 190985 0 0 0
cpu15 15589 4182 50157 58961391 1800 0 3564 0 0 0
cpu16 16142 6333 56052 59017656 9756 1 3811 0 0 0
cpu17 14702 4071 42035 59057502 1815 0 1190 0 0 0
cpu18 16966 3207 51344 59033053 5518 0 2792 0 0 0
cpu19 15426 3303 55242 58967372 37028 4 7016 0 0 0
```

In the output, the very first row `cpu` line aggregates the numbers of all the other `cpuX` lines. These numbers identify the amount of time the CPU has spent performing different kinds of work. Time units are in `USER_HZ` or Jiffies (typically hundredths of a second).

Different meanings of the columns from left to right are as following

Column	Name	Description
1	user	Time spent in user mode.
2	nice	Time spent in user mode with low priority (nice).
3	system	Time spent in system mode.



Column	Name	Description
4	idle	Time spent in the idle task. This value should be USER_HZ times the second entry in the <code>/proc/uptime</code> pseudo-file.
5	iowait (since Linux 2.5.41)	Time waiting for I/O to complete.
6	irq (since Linux 2.6.0-test4)	Time servicing interrupts.
7	softirq (since Linux 2.6.0-test4)	Time servicing softirqs.
8	steal (since Linux 2.6.11)	Stolen time, which is the time spent in other operating systems when running in a virtualized environment.
9	guest (since Linux 2.6.24)	Time spent running a virtual CPU for guest operating systems under the control of the Linux kernel.
10	guest_nice (since Linux 2.6.33)	Time spent running a niced guest (virtual CPU for guest operating systems under the control of the Linux kernel).

4. Calculate the total CPU load using the following formula:

Average total CPU load percentage X % = ((user + nice + system + iowait + irq + softirq) * 100) / (user + nice + system + idle + iowait + irq + softirq)

2.2 Actions

If any of the CPU average loads exceed the associated maximum value, try to locate the reason. If you fail to identify the reason, consult the next level of maintenance support for help.