



## buffer public through user (ERM)

---

- [buffer public](#), page 2
- [cpu interrupt](#), page 4
- [cpu process](#), page 6
- [cpu total](#), page 8
- [critical rising](#), page 10
- [instance \(resource group\)](#), page 16
- [major rising](#), page 18
- [memory io](#), page 24
- [memory processor](#), page 26
- [memory statistics history table](#), page 28
- [minor rising](#), page 29
- [policy \(ERM\)](#), page 35
- [policy \(resource group\)](#), page 37
- [resource policy](#), page 39
- [show resource all](#), page 41
- [show resource database](#), page 56
- [show resource owner](#), page 59
- [show resource relationship](#), page 68
- [show resource user](#), page 71
- [slot \(ERM policy\)](#), page 75
- [system \(ERM policy\)](#), page 77
- [user \(ERM\)](#), page 79

# buffer public

To enter buffer owner configuration mode to set thresholds for buffer usage, use the **bufferpublic** command in resource policy node configuration mode. To exit buffer owner configuration mode, use the **no** form of this command.

**buffer public**

**no buffer public**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Resource policy node configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines** This command allows you to enter buffer owner configuration mode to set rising and falling values for critical, major, and minor thresholds for buffer usage.

**Examples** The following example shows how to enter buffer owner configuration mode to set thresholds for buffer usage:

```
Router(config-res-policy-node)# buffer public
```

## Related Commands

Command	Description
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.

<b>Command</b>	<b>Description</b>
<b>resource policy</b>	Enters ERM configuration mode.
<b>show buffer leak</b>	Displays the buffer details.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

# cpu interrupt

To enter CPU owner configuration mode to set thresholds for interrupt level CPU utilization, use the **cpu interrupt** command in resource policy node configuration mode. To exit CPU owner configuration mode, use the **no** form of this command.

**cpu interrupt**

**no cpu interrupt**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Resource policy node configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines** This command allows you to enter CPU owner configuration mode to set rising and falling values for critical, major, and minor thresholds for interrupt level CPU utilization.

**Examples** The following example shows how to enter CPU owner configuration mode to set thresholds for interrupt level CPU utilization:

```
Router (config-res-policy-node) # cpu interrupt
```

## Related Commands

Command	Description
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.

<b>Command</b>	<b>Description</b>
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

## cpu process

To enter CPU owner configuration mode to set thresholds for process level CPU utilization, use the **cpu process** command in resource policy node configuration mode. To exit CPU owner configuration mode, use the **no** form of this command.

**cpu process**

**no cpu process**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Resource policy node configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines** This command allows you to enter CPU owner configuration mode to set rising and falling values for critical, major, and minor thresholds for process level CPU utilization.

**Examples** The following example shows how to enter CPU owner configuration mode to set thresholds for process level CPU utilization:

```
Router (config-res-policy-node) # cpu process
```

### Related Commands

Command	Description
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.

<b>Command</b>	<b>Description</b>
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

# cpu total

To enter CPU owner configuration mode to set thresholds for total CPU utilization, use the **cputotal** command in resource policy node configuration mode. To exit CPU owner configuration mode, use the **no** form of this command.

**cpu total**

**no cpu total**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Resource policy node configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines** This command allows you to enter CPU owner configuration mode to set rising and falling values for critical, major, and minor thresholds for total CPU utilization.

**Examples** The following example shows how to enter CPU owner configuration mode to set thresholds for total CPU utilization:

```
Router(config-res-policy-node)# cpu total
```

## Related Commands

Command	Description
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.



<b>Command</b>	<b>Description</b>
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

## critical rising

To set critical level threshold values for the buffer, CPU, and memory ROs, use the **criticalrising** command in buffer owner configuration mode, CPU owner configuration mode, or memory owner configuration mode. To disable this function, use the **no** form of this command.

**critical rising** *rising-threshold-value* [**interval** *interval-value*] [**falling** *falling-threshold-value* [**interval** *interval-value*]] [**global**]

**no critical rising**

### Syntax Description

<i>rising-threshold-value</i>	The rising threshold value as a percentage. Valid values are from 1 to 100.
<b>interval</b>	(Optional) Specifies the time, in seconds, during which the variation in rising or falling threshold values is not reported to the RU, resource groups, or resource user types. For example, if the buffer usage count remains above the configured threshold value for the configured interval, a notification is sent to the RU, resource group, or resource user types.
<i>interval-value</i>	The time, in seconds, during which the variation in rising or falling threshold values are not reported to the RU, resource groups, or resource user types. Valid values are from 0 to 86400. The default value is 0.
<b>falling</b>	(Optional) Specifies the falling threshold value as a percentage.
<i>falling-threshold-value</i>	(Optional) The falling threshold value as a percentage. Valid values are from 1 to 100.
<b>global</b>	(Optional) Configures a global threshold.  The <b>global</b> keyword is optional when you set critical threshold values for public buffer, processor CPU, I/O memory, and processor memory.  The <b>global</b> keyword is required when you set critical threshold values for interrupt CPU and total CPU.

**Command Default** Disabled

**Command Modes** Buffer owner configuration CPU owner configuration Memory owner configuration

**Command History**

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines**

The interval is the dampening or observation interval time, in seconds, during which the variations in the rising and falling threshold values are not reported to the RUs. That is, the interval is the time the system waits to check whether the threshold value stabilizes. The interval is set to avoid unnecessary and unwanted threshold notifications. If not configured, the system defaults to 0 seconds.

This command allows you to configure three types of thresholding:

- System Global Thresholding
- User Local Thresholding
- Per User Global Thresholding

**System Global Thresholding**

System global thresholding is used when the entire resource reaches a specified value. That is, RUs are notified when the total resource utilization goes above or below a specified threshold value. The notification order is determined by the priority of the RU. The RUs with a lower priority are notified first and expected to reduce the resource utilization. This notification order prevents the sending of unwanted notifications to high-priority RUs.

You can set rising and falling threshold values. For example, if you set a total CPU utilization threshold value of 90% as the rising critical value and 20% as falling critical value, when the total CPU utilization crosses the 90% mark, a critical Up notification is sent to all the RUs and when the total CPU utilization falls below 20%, a critical Down notification is sent to all the RUs. The same criteria also apply to buffer ROs and memory ROs.

**User Local Thresholding**

User local thresholding is used when a specified RU exceeds the configured limits. The user local thresholding method prevents a single RU from monopolizing the resources. That is, the specified RU is notified when the resource utilization of the specified RU goes above or below a configured threshold value. For example, if you set a CPU utilization threshold value of 90% as the rising critical value and 20% as falling critical value, when the CPU utilization of the specified RU crosses the 90% mark, a critical Up notification is sent to that RU only and when the CPU utilization of the specified RU falls below 20%, a critical Down notification is sent to that RU only. The same method also applies to buffer and memory ROs.

**Per User Global Thresholding**

Per user global thresholding is used when the entire resource reaches a specified value. This value is unique for each RU and notification is sent only to the specified RU. User global thresholding is similar to user local thresholding, except that the global resource usage is compared against the thresholds. That is, only the specified RU is notified when the total resource utilization goes above or below a configured threshold value. For example, if you have set a CPU utilization threshold value of 90% as the rising critical value and 20% as falling critical value, when the total CPU utilization crosses the 90% mark, a critical Up notification is sent

to the specified RU only and when the total CPU utilization falls below 20%, a critical Down notification is sent to the specified RU only. The same method also applies to buffer and memory ROs.

### Threshold Violations

The Cisco IOS device sends out error messages when a threshold is violated. The following examples help you understand the error message pattern when different threshold violations occur in buffer, CPU, and memory ROs:

#### System Global Threshold Violation in Buffer RO

The threshold violation in buffer RO for a system global threshold shows the following output:

```
System global threshold-Violation (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:15:11: %SYS-4-GLOBALBUFEXCEED: Buffer usage has gone above global buffer Critical threshold
configured <value> Current usage :<value>
```

For example:

```
00:15:11: %SYS-4-GLOBALBUFEXCEED: Buffer usage has gone above global buffer Critical threshold
configured 144 Current usage :145
System global threshold- Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:17:10: %SYS-5-GLOBALBUFRECOVER: Buffer usage has gone below global buffer Critical
threshold
configured <value> Current usage :<value>
```

For example:

```
00:17:10: %SYS-5-GLOBALBUFRECOVER: Buffer usage has gone below global buffer Critical
threshold
configured 90 Current usage :89
```

#### Per User Global Threshold Violation in Buffer RO

The threshold violation in buffer RO for a user global threshold shows the following output:

```
User global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:24:04: %SYS-4-RESGLOBALBUFEXCEED: Buffer usage has gone above buffer Critical threshold
  configured by resource user <user-name>
configured 144 Current usage :145
User global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:25:08: %SYS-4-RESGLOBALBUFRECOVER: Buffer usage has gone below buffer Critical threshold
  configured by resource user <user-name>
configured 126 Current usage :125
```

#### User Local Threshold Violation in Buffer RO

The threshold violation in buffer RO for a user local threshold shows the following output:

```
User local threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:31:15: %SYS-4-RESBUFEXCEED: Resource user user_1 has exceeded the buffer Critical
threshold. configured 108 Current usage :109
User local threshold- Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:31:05: %SYS-5-RESBUFRECOVER: Resource user user_1 has recovered after exceeding the
buffer Critical threshold. configured 90 Current usage :89
```

#### System Global Threshold Violation in CPU RO

The threshold violation in CPU RO for a system global threshold shows the following output:

```
System global threshold- Violation
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly )
=====
00:19:36: %SYS-4-CPURESRRISING: System is seeing global cpu util 19% at total level more
than the configured minor limit 11%
System global threshold - Recovery
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:20:56: %SYS-6-CPURESRRFALLING: System is no longer seeing global high cpu at total level
for the configured minor limit 10%, current value 4%
```

#### Per User Global Threshold Violation in CPU RO

The threshold violation in CPU RO for a user global threshold shows the following output:

```
User global threshold - Violation
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:14:21: %SYS-4-CPURESRRISING: Resource user <user-name> is seeing global cpu util 11% at
total level more than the configured minor limit 6 %
```

For example:

```
00:14:21: %SYS-4-CPURESRRISING: Resource user Test-proc-14:99s:1w:100n is seeing global cpu
util 11% at total level more than the configured minor limit 6%
User global threshold- Recovery
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:14:46: %SYS-6-CPURESRRFALLING: Resource user <user-name> is no longer seeing global high
cpu at total level for the configured critical limit 9%, current value 4%
```

For example:

```
00:14:46: %SYS-6-CPURESRRFALLING: Resource user Test-proc-14:99s:1w:100n is no longer seeing
global high cpu at total level for the configured critical limit 9%, current value 4%
```

#### User Local Threshold Violation in CPU RO

The threshold violation in CPU RO for a user local threshold shows the following output:

```
User local threshold - Violation (keywords Critical, Major and Minor will vary accordingly
- only process level)
=====
00:12:11: %SYS-4-CPURESRRISING: Resource user <user-name> is seeing local cpu util 15% at
process level more than the configured minor limit 6%
```

For example:

```
00:12:11: %SYS-4-CPURESRRISING: Resource user Test-proc-9:85s:15w:100n is seeing local cpu
util 15% at process level more than the configured minor limit 6%
User local threshold- Recovery (keywords Critical, Major and Minor will vary accordingly
- only process level)
=====
00:13:11: %SYS-6-CPURESRRFALLING: Resource user <user-name> is no longer seeing local high
cpu at process level for the configured critical limit 9%, current value 3%
```

#### System Global Threshold Violation in Memory RO

The threshold violation in memory RO for a system global threshold shows the following output:

```
System global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly )
(If violation happens in IO memory pool will be : I/O)
=====
13:53:22: %SYS-5-GLOBALMEMEXCEED: Global Memory has exceeded the Minor threshold
Pool: Processor Used: 422703520 Threshold: 373885200
```

For example:

```
13:54:03: %SYS-5-GLOBALMEMEXCEED: Global Memory has exceeded the Critical threshold
Pool: Processor Used: 622701556 Threshold: 467356500
System global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
(If recovery happens in IO memory pool will be : I/O)
```

```
=====
%SYS-5-GLOBALMEMRECOVER: Global Memory has recovered after exceeding Minor threshold
Pool: Processor Used: 222473448 Threshold: 355190940
```

For example:

```
13:50:41: %SYS-5-GLOBALMEMRECOVER: Global Memory has recovered after exceeding Critical
threshold
Pool: Processor Used: 222473152 Threshold: 443988675
```

### Per User Global Threshold Violation in Memory RO

The threshold violation in memory RO for a user global threshold shows the following output:

```
User global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
(If violation happens in IO memory pool will be : I/O)
```

```
=====
00:53:14: %SYS-4-RESGLOBALMEMEXCEED: Global Memory has exceeded the Minor threshold configure
by resource user <XYZ>
```

```
Pool: Processor Used: 62273916 Threshold: 62246820
User global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
(If recovery happens in IO memory pool will be : I/O)
```

```
=====
00:32:56: %SYS-4-RESGLOBALMEMRECOVER: Global Memory has recovered after exceeding the
Critical threshold configure by resource user <XYZ>
```

```
Pool: Processor Used: 329999508 Threshold: 375865440
```

### User Local Threshold Violation in Memory RO

The threshold violation in memory RO for a user local threshold shows the following output:

```
User local threshold- Violation (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
=====
01:05:42: %SYS-4-RESMEMEXCEED: Resource user <XYZ> has exceeded the Critical memory threshold
Pool: Processor Used: 103754740 Threshold: 103744700
```

```
User local threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
=====
00:44:43: %SYS-5-RESMEMRECOVER: Resource user <XYZ> has recovered after exceeding the
Critical memory threshold
```

```
Pool: Processor Used: 328892280 Threshold :375865440
```

## Examples

### Examples

The following example shows how to configure the critical threshold values for system global thresholding with a critical rising threshold of 90% at an interval of 12 seconds and a critical falling threshold of 20% at an interval of 10 seconds:

```
Router(config-owner-cpu)# critical rising 90 interval 12 falling 20 interval 10 global
Router(config-owner-buffer)# critical rising 90 interval 12 falling 20 interval 10 global
Router(config-owner-memory)# critical rising 90 interval 12 falling 20 interval 10 global
```

**Examples**

The following example shows how to configure the critical threshold values for user local thresholding with a critical rising threshold of 90% at an interval of 12 seconds and a critical falling threshold of 20% at an interval of 10 seconds:

```
Router(config-owner-cpu)# critical rising 90 interval 12 falling 20 interval 10
Router(config-owner-buffer)# critical rising 90 interval 12 falling 20 interval 10
Router(config-owner-memory)# critical rising 90 interval 12 falling 20 interval 10
```

**Examples**

The following example shows how to configure the critical threshold values for per user global thresholding with a critical rising threshold of 90% at an interval of 12 seconds and a critical falling threshold of 20% at an interval of 10 seconds:

```
Router(config-owner-cpu)# critical rising 90 interval 12 falling 20 interval 10 global
Router(config-owner-buffer)# critical rising 90 interval 12 falling 20 interval 10 global
Router(config-owner-memory)# critical rising 90 interval 12 falling 20 interval 10 global
```

**Related Commands**

Command	Description
<b>buffer public</b>	Enters the buffer owner configuration mode and sets threshold values for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets threshold values for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets threshold values for processor level CPU utilization.
<b>cpu total</b>	Enters the CPU owner configuration mode and sets threshold values for total CPU utilization.
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

## instance (resource group)

To add request/response units (RUs) to a specified resource group, use the **instance** command in resource group configuration mode. To disable this function, use the **no** form of this command.

**instance** *instance-name*

**no instance** *instance-name*

### Syntax Description

<i>instance-name</i>	Name of the RU you want to add to the resource group (for example, <b>http</b> , <b>snmp</b> ).
----------------------	---

### Command Default

Disabled

### Command Modes

Resource group configuration

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

### Usage Guidelines

Before adding RUs to a resource group, you must create a resource group using the **usergroup***resource-group-name***type***resource-user-type* command in ERM configuration mode.

For example, you have a resource group named `lowPrioUsers` with a type of `iosprocess`. You have low-priority RUs or tasks such as HTTP and Simple Network Management Protocol (SNMP), and you want to set a threshold for all the low-priority RUs as a group. You must add the RUs to the resource group using the **instance***instance-name* command and then apply a resource policy.

If the resource policy you applied sets a minor rising threshold value of 10 percent for the resource group, when the accumulated usage of both HTTP and SNMP RUs crosses 10 percent a notification is sent to the RUs in the resource group `lowPrioUsers`. For example, if HTTP usage is 4 percent and SNMP usage is 7 percent, a notification is sent to the resource group.

### Examples

The following example shows how to add an HTTP RU to a resource group named `lowPrioUsers`:

```
Router(config-erm) # user group lowPrioUsers type iosprocess
Router(config-res-group) # instance http
```



**Related Commands**

<b>Command</b>	<b>Description</b>
<b>policy (resource group)</b>	Applies a policy to all the RUs in the resource group.
<b>user (ERM)</b>	Creates a resource group.

## major rising

To set major level threshold values for the buffer, CPU, and memory resource owners (ROs), use the **majorrising** command in buffer owner configuration mode, CPU owner configuration mode, or memory owner configuration mode. To disable this function, use the **no** form of this command.

**major rising** *rising-threshold-value* [**interval** *interval-value*] [**falling** *falling-threshold-value* [**interval** *interval-value*]] [**global**]

**no major rising**

### Syntax Description

<i>rising-threshold-value</i>	The rising threshold value as a percentage. Valid values are from 1 to 100.
<b>interval</b>	(Optional) Specifies the time, in seconds, during which the variation in rising or falling threshold values are not reported to the request/response unit (RU), resource group, or resource user types. For example, if the buffer usage count remains above the configured threshold value for the configured interval, a notification is sent to the RU, resource group, or resource user types.
<i>interval-value</i>	The time, in seconds, during which the variation in rising or falling threshold values is not reported to the RU, resource group, or resource user types. Valid values are from 0 to 86400. The default value is 0.
<b>falling</b>	(Optional) Specifies the falling threshold value as a percentage.
<i>falling-threshold-value</i>	(Optional) The falling threshold value. Valid values are from 1 to 100.
<b>global</b>	(Optional) Configures a global threshold.  The <b>global</b> keyword is optional when you set major threshold values for public buffer, processor CPU, I/O memory, and processor memory.  The <b>global</b> keyword is required when you set major threshold values for interrupt CPU and total CPU.

**Command Default** Disabled

**Command Modes** Buffer owner configuration CPU owner configuration Memory owner configuration

**Command History**

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines**

The interval is the dampening or observation interval time, in seconds, during which the variations in the rising and falling threshold values are not notified to the ROs or RUs. That is, the interval is the time the system waits to check whether the threshold value stabilizes. The interval is set to avoid unnecessary and unwanted threshold notifications. If not configured, the system defaults to 0 seconds.

This command allows you to configure three types of thresholding:

- System Global Thresholding
- User Local Thresholding
- Per User Global Thresholding

**System Global Thresholding**

System global thresholding is used when the entire resource reaches a specified value. That is, RUs are notified when the total resource utilization goes above or below a specified threshold value. The notification order is determined by the priority of the RU. The RUs with a lower priority are notified first, and are expected to reduce the resource utilization. This notification order prevents the high-priority RUs from being sent unwanted notifications.

You can set rising and falling threshold values. For example, if you have set a total CPU utilization threshold value of 70% as the rising major value and 15% as the falling major value, when the total CPU utilization crosses the 70% mark, a major Up notification is sent to all the RUs and when the total CPU utilization falls below 15%, a major Down notification is sent to all the RUs. The same criteria apply to buffer ROs and memory ROs.

**User Local Thresholding**

User local thresholding is used when a specified RU exceeds the configured limits. The user local thresholding method prevents a single RU from monopolizing resources. That is, the specified RU is notified when its resource utilization exceeds or falls below a configured threshold value. For example, if you set a CPU utilization threshold value of 70% as the rising major value and 15% as the falling major value, when the CPU utilization of the specified RU crosses the 70% mark, a major Up notification is sent to that RU only and when the CPU utilization of the specified RU falls below 15%, a major Down notification is sent to only that RU. The same method also applies to buffer and memory ROs.

**Per User Global Thresholding**

Per user global thresholding is used when the entire resource reaches a specified value. This value is unique for each RU and notification is sent only to the specified RU. User global thresholding is similar to user local thresholding, except that the global resource usage is compared against the thresholds. That is, only the specified RU is notified when the total resource utilization exceeds or falls below a configured threshold value. For example, if you set a CPU utilization threshold value of 70% as the rising major value and 15% as the falling major value, when the total CPU utilization crosses the 70% mark, a major Up notification is sent

to only the specified RU and when the total CPU utilization falls below 15%, a major Down notification is sent to only the specified RU. The same method also applies to buffer and memory ROs.

### Threshold Violations

The Cisco IOS device sends out error messages when a threshold is violated. The following examples help you understand the error message pattern when different threshold violations occur in buffer, CPU, and memory ROs:

#### System Global Threshold Violation in Buffer RO

The threshold violation in buffer RO for a system global threshold shows the following output:

```
System global threshold-Violation (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:15:11: %SYS-4-GLOBALBUFEXCEED: Buffer usage has gone above global buffer Major threshold
configured <value> Current usage :<value>
```

For example:

```
00:15:11: %SYS-4-GLOBALBUFEXCEED: Buffer usage has gone above global buffer Major threshold
configured 100 Current usage :101
System global threshold- Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
00:17:10: %SYS-5-GLOBALBUFRECOVER: Buffer usage has gone below global buffer Major threshold
configured <value> Current usage :<value>
```

For example:

```
00:17:10: %SYS-5-GLOBALBUFRECOVER: Buffer usage has gone below global buffer Critical
threshold
configured 70 Current usage :69
```

#### Per User Global Threshold Violation in Buffer RO

The threshold violation in buffer RO for a user global threshold shows the following output:

```
User global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:24:04: %SYS-4-RESGLOBALBUFEXCEED: Buffer usage has gone above buffer Major threshold
configured by resource user <user-name>
configured 100 Current usage :101
User global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:25:08: %SYS-4-RESGLOBALBUFRECOVER: Buffer usage has gone below buffer Major threshold
configured by resource user <user-name>
configured 76 Current usage :75
```

#### User Local Threshold Violation in Buffer RO

The threshold violation in buffer RO for a user local threshold shows the following output:

```
User local threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:31:15: %SYS-4-RESBUFEXCEED: Resource user user_1 has exceeded the buffer Major threshold.
configured 108 Current usage :109
User local threshold- Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
=====
00:31:05: %SYS-5-RESBUFRECOVER: Resource user user_1 has recovered after exceeding the
buffer Major threshold. configured 90 Current usage :89
```

#### System Global Threshold Violation in CPU RO

The threshold violation in CPU RO for a system global threshold shows the following output:

```
System global threshold- Violation
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly )
=====
00:19:36: %SYS-4-CPURESRRISING: System is seeing global cpu util 19% at total level more
than the configured major limit 11%
System global threshold - Recovery
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly )
=====
00:20:56: %SYS-6-CPURESRRFALLING: System is no longer seeing global high cpu at total level
for the configured major limit 10%, current value 4%
```

#### Per User Global Threshold Violation in CPU RO

The threshold violation in CPU RO for a user global threshold shows the following output:

```
User global threshold - Violation
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly )
=====
00:14:21: %SYS-4-CPURESRRISING: Resource user <user-name> is seeing global cpu util 11% at
total level more than the configured major limit 6%
```

For example:

```
00:14:21: %SYS-4-CPURESRRISING: Resource user Test-proc-14:99s:1w:100n is seeing global cpu
util 11% at total level more than the configured major limit 6%
User global threshold- Recovery
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly )
=====
00:14:46: %SYS-6-CPURESRRFALLING: Resource user <user-name> is no longer seeing global high
cpu at total level for the configured critical limit 9%, current value 4%
```

For example:

```
00:14:46: %SYS-6-CPURESRRFALLING: Resource user Test-proc-14:99s:1w:100n is no longer seeing
global high cpu at total level for the configured critical limit 9%, current value 4%
```

#### User Local Threshold Violation in CPU RO

The threshold violation in CPU RO for a user local threshold shows the following output:

```
User local threshold - Violation (keywords Critical, Major and Minor will vary accordingly
- only process level)
=====
00:12:11: %SYS-4-CPURESRRISING: Resource user <user-name> is seeing local cpu util 15% at
process level more than the configured minor limit 6 %
```

For example:

```
00:12:11: %SYS-4-CPURESRRISING: Resource user Test-proc-9:85s:15w:100n is seeing local cpu
util 15% at process level more than the configured minor limit 6%
User local threshold- Recovery (keywords Critical, Major and Minor will vary accordingly
- only process level)
=====
00:13:11: %SYS-6-CPURESRRFALLING: Resource user <user-name> is no longer seeing local high
cpu at process level for the configured critical limit 9%, current value 3%
```

#### System Global Threshold Violation in Memory RO

The threshold violation in memory RO for a system global threshold shows the following output:

```
System global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
(If violation happens in IO memory pool will be : I/O)
=====
13:53:22: %SYS-5-GLOBALMEMEXCEED: Global Memory has exceeded the Minor threshold
Pool: Processor Used: 422703520 Threshold: 373885200
```

For example:

```
13:54:03: %SYS-5-GLOBALMEMEXCEED: Global Memory has exceeded the Critical threshold
Pool: Processor Used: 622701556 Threshold: 467356500
System global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
(If recovery happens in IO memory pool will be : I/O)
```

```
=====
%SYS-5-GLOBALMEMRECOVER: Global Memory has recovered after exceeding Minor threshold
Pool: Processor Used: 222473448 Threshold: 355190940
```

For example:

```
13:50:41: %SYS-5-GLOBALMEMRECOVER: Global Memory has recovered after exceeding Critical
threshold
Pool: Processor Used: 222473152 Threshold: 443988675
```

### Per User Global Threshold Violation in Memory RO

The threshold violation in memory RO for a user global threshold shows the following output:

```
User global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
(If violation happens in IO memory pool will be : I/O)
```

```
=====
00:53:14: %SYS-4-RESGLOBALMEMEXCEED: Global Memory has exceeded the Minor threshold configure
by resource user <XYZ>
```

```
Pool: Processor Used: 62273916 Threshold: 62246820
User global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
(If recovery happens in IO memory pool will be : I/O)
```

```
=====
00:32:56: %SYS-4-RESGLOBALMEMRECOVER: Global Memory has recovered after exceeding the
Critical threshold configure by resource user <XYZ>
```

```
Pool: Processor Used: 329999508 Threshold: 375865440
```

### User Local Threshold Violation in Memory RO

The threshold violation in memory RO for a user local threshold shows the following output:

```
User local threshold- Violation (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
=====
01:05:42: %SYS-4-RESMEMEXCEED: Resource user <XYZ> has exceeded the Critical memory threshold
Pool: Processor Used: 103754740 Threshold: 103744700
```

```
User local threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
=====
00:44:43: %SYS-5-RESMEMRECOVER: Resource user <XYZ> has recovered after exceeding the
Critical memory threshold
```

```
Pool: Processor Used: 328892280 Threshold :375865440
```

## Examples

### Examples

The following example shows how to configure the major threshold values for system global thresholding with a major rising threshold of 70% at an interval of 12 seconds and a major falling threshold of 15% at an interval of 10 seconds:

```
Router(config-owner-cpu)# major rising 70 interval 12 falling 15 interval 10 global
Router(config-owner-buffer)# major rising 70 interval 12 falling 15 interval 10 global
Router(config-owner-memory)# major rising 70 interval 12 falling 15 interval 10 global
```

**Examples**

The following example shows how to configure the major threshold values for user local thresholding with a major rising threshold of 70% at an interval of 12 seconds and a major falling threshold of 15% at an interval of 10 seconds:

```
Router(config-owner-cpu)# major rising 70 interval 12 falling 15 interval 10
Router(config-owner-buffer)# major rising 70 interval 12 falling 15 interval 10
Router(config-owner-memory)# major rising 70 interval 12 falling 15 interval 10
```

**Examples**

The following example shows how to configure the major threshold values for per user global thresholding with a major rising threshold of 70% at an interval of 12 seconds and a major falling threshold of 15% at an interval of 10 seconds:

```
Router(config-owner-cpu)# major rising 70 interval 12 falling 15 interval 10 global
Router(config-owner-buffer)# major rising 70 interval 12 falling 15 interval 10 global
Router(config-owner-memory)# major rising 70 interval 12 falling 15 interval 10 global
```

**Related Commands**

Command	Description
<b>buffer public</b>	Enters the buffer owner configuration mode and sets threshold values for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets threshold values for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets threshold values for processor level CPU utilization.
<b>cpu total</b>	Enters the CPU owner configuration mode and sets threshold values for total CPU utilization.
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

# memory io

To enter memory owner configuration mode to set threshold values for I/O memory, use the **memoryio** command in resource policy node configuration mode. To exit memory owner configuration mode, use the **no** form of this command.

**memory io**

**no memory io**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Resource policy node configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines** This command allows you to enter memory owner configuration mode to set rising and falling values for critical, major, and minor thresholds for I/O memory.

**Examples** The following example shows how to enter memory owner configuration mode to set threshold values for I/O memory:

```
Router(config-res-policy-node)# memory io
```

## Related Commands

Command	Description
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.



<b>Command</b>	<b>Description</b>
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

# memory processor

To enter memory owner configuration mode to set the threshold values for the processor memory, use the **memoryprocessor** command in resource policy node configuration mode. To exit memory owner configuration mode, use the **no** form of this command.

**memory processor**

**no memory processor**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Resource policy node configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines** This command allows you to enter memory owner configuration mode to set rising and falling values for critical, major, and minor thresholds for the processor memory.

**Examples** The following example shows how to enter memory owner configuration mode to set the threshold values for the processor memory:

```
Router(config-res-policy-node)# memory processor
```

## Related Commands

Command	Description
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.

<b>Command</b>	<b>Description</b>
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

# memory statistics history table

To change the number of hours for which the memory log is maintained, use the **memorystatisticshistorytable** command in global configuration mode. To return the logging to its default values, use the **no** form of this command.

**memory statistics history table** *number-of-hours*

**no memory statistics history table** *number-of-hours*

## Syntax Description

<i>number-of-hours</i>	Number of hours of history for which the log is maintained.  Valid values are from 12 to 72. The default value is 24.
------------------------	---

## Command Default

The memory log is maintained for 24 hours.

## Command Modes

Global configuration

## Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

## Usage Guidelines

This command allows you to change the number of hours for which the memory log is maintained. You cannot disable this command. The **no** form of the command only returns the logging to its default value.

## Examples

The following example shows how to change the memory log time to 48 hours of history:

```
Router(config)# memory statistics history table 48
```

## Related Commands

Command	Description
<b>show memory statistics history table</b>	Displays the history of memory consumption on the Cisco IOS router over a specified period of time.

## minor rising

To set minor level threshold values for the buffer, CPU, and memory resource owners (ROs), use the **minorrising** command in buffer owner configuration mode, CPU owner configuration mode, or memory owner configuration mode. To disable this function, use the **no** form of this command.

**minor rising** *rising-threshold-value* [**interval** *interval-value*] [**falling** *falling-threshold-value* [**interval** *interval-value*]] [**global**]

**no minor rising**

### Syntax Description

<i>rising-threshold-value</i>	The rising threshold value as a percentage. Valid values are from 1 to 100.
<b>interval</b>	(Optional) Specifies the time, in seconds, during which the variation in rising or falling threshold values are not reported to the request/response unit (RU), resource group, or resource user types. For example, if the buffer usage count has gone above the configured threshold value and if it remains longer than the configured interval, a notification is sent to the RU, resource group, or resource user types.
<i>interval-value</i>	(Optional) The time, in seconds, during which the variation in rising or falling threshold values are not reported to the RU, resource group, or resource user types. Valid values are from 0 to 86400. The default value is 0.
<b>falling</b>	(Optional) Specifies the falling threshold value as a percentage.
<i>falling-threshold-value</i>	(Optional) The falling threshold value as a percentage. Valid values are from 1 to 100.
<b>global</b>	(Optional) Configures a global threshold.  The <b>global</b> keyword is optional when you set major threshold values for public buffer, processor CPU, I/O memory, and processor memory.  The <b>global</b> keyword is required when you set major threshold values for interrupt CPU and total CPU.

**Command Default** Disabled by default.

**Command Modes**

Buffer owner configuration CPU owner configuration Memory owner configuration

**Command History**

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Usage Guidelines**

The interval is the dampening or observation interval time in seconds during which the variations in the rising and falling threshold values are not notified to the ROs or RUs. That is, the interval is the time the system waits to check whether the threshold value stabilizes or not. The interval is set to avoid unnecessary and unwanted threshold notifications. If not configured, the system defaults to 0 seconds.

This command allows you to configure three types of thresholding:

- System Global Thresholding
- User Local Thresholding
- Per User Global Thresholding

**System Global Thresholding**

System global thresholding is used when the entire resource reaches a specified value. That is, RUs are notified when the total resource utilization goes above or below a specified threshold value. The notification order is determined by the priority of the RU. The RUs with a lower priority will be notified first, so that these low-priority RUs are expected to reduce the resource utilization. This order prevents the high-priority RUs from getting affected with unwanted notifications.

You can set rising and falling threshold values. For example, if you have set a total CPU utilization threshold value of 60% as the rising minor value and 5% as falling minor value, then when the total CPU utilization crosses the 60% mark, a minor Up notification is sent to all the RUs and when the total CPU utilization falls below 5%, a minor Down notification is sent to all the RUs. The same criteria apply to buffer ROs and memory ROs.

**User Local Thresholding**

User local thresholding is used when a specified RU exceeds the configured limits. The user local thresholding method prevents a single RU from monopolizing the resources. That is, the specified RU is notified when the resource utilization of the specified RU goes above or below a configured threshold value. For example, if you have set a CPU utilization threshold value of 60% as the rising minor value and 5% as the falling minor value, when the CPU utilization of the specified RU crosses the 60% mark, a minor Up notification is sent to only that RU and when the CPU utilization of the specified RU falls below 5%, a minor Down notification is sent to only that RU. The same method also applies to buffer and memory ROs.

**Per User Global Thresholding**

Per user global thresholding is used when the entire resource reaches a specified value. This value is unique for each RU and notification is sent only to the specified RU. User global thresholding is similar to user local thresholding, except that the global resource usage is compared against the thresholds. That is, only the specified RU is notified when the total resource utilization exceeds or falls below a configured threshold

value. For example, if you have set a CPU utilization threshold value of 60% as the rising minor value and 5% as the falling minor value, when the total CPU utilization crosses the 60% mark, a minor Up notification is sent to only the specified RU and when the total CPU utilization falls below 5%, a minor Down notification is sent to only the specified RU. The same criteria also apply to buffer and memory ROs.

### Threshold Violations

The Cisco IOS device sends out error messages when a threshold is violated. The following examples help you understand the error message pattern when different threshold violations occur in buffer, CPU, and memory ROs:

#### System Global Threshold Violation in Buffer RO

The threshold violation in buffer RO for a system global threshold shows the following output:

```
System global threshold-Violation (keywords Critical, Major and Minor alone will vary accordingly)
=====
00:15:11: %SYS-4-GLOBALBUFEXCEED: Buffer usage has gone above global buffer Critical threshold
configured <value> Current usage :<value>
```

For example:

```
00:15:11: %SYS-4-GLOBALBUFEXCEED: Buffer usage has gone above global buffer Critical threshold
configured 144 Current usage :145
System global threshold- Recovery (keywords Critical, Major and Minor alone will vary accordingly)
=====
00:17:10: %SYS-5-GLOBALBUFRECOVER: Buffer usage has gone below global buffer Critical
threshold
configured <value> Current usage :<value>
```

For example:

```
00:17:10: %SYS-5-GLOBALBUFRECOVER: Buffer usage has gone below global buffer Critical
threshold
configured 90 Current usage :89
```

#### Per User Global Threshold Violation in Buffer RO

The threshold violation in buffer RO for a user global threshold shows the following output:

```
User global threshold - Violation (keywords Critical, Major and Minor alone will vary accordingly)
=====
00:24:04: %SYS-4-RESGLOBALBUFEXCEED: Buffer usage has gone above buffer Critical threshold
  configured by resource user <user-name>
  configured 144 Current usage :145
User global threshold - Recovery (keywords Critical, Major and Minor alone will vary accordingly)
=====
00:25:08: %SYS-4-RESGLOBALBUFRECOVER: Buffer usage has gone below buffer Critical threshold
  configured by resource user <user-name>
  configured 126 Current usage :125
```

#### User Local Threshold Violation in Buffer RO

The threshold violation in buffer RO for a user local threshold shows the following output:

```
User local threshold - Violation (keywords Critical, Major and Minor alone will vary accordingly)
=====
00:31:15: %SYS-4-RESBUFEXCEED: Resource user user_1 has exceeded the buffer Critical
threshold. configured 108 Current usage :109
User local threshold- Recovery (keywords Critical, Major and Minor alone will vary accordingly)
=====
```

```
00:31:05: %SYS-5-RESBUFRECOVER: Resource user user_1 has recovered after exceeding the
buffer Critical threshold. configured 90 Current usage :89
```

### System Global Threshold Violation in CPU RO

The threshold violation in CPU RO for a system global threshold shows the following output:

```
System global threshold- Violation
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:19:36: %SYS-4-CPURESRISE: System is seeing global cpu util 19% at total level more
than the configured minor limit 11%
System global threshold - Recovery
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:20:56: %SYS-6-CPURESFALLING: System is no longer seeing global high cpu at total level
for the configured minor limit 10%, current value 4%
```

### Per User Global Threshold Violation in CPU RO

The threshold violation in CPU RO for a user global threshold shows the following output:

```
User global threshold - Violation
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:14:21: %SYS-4-CPURESRISE: Resource user <user-name> is seeing global cpu util 11% at
total level more than the configured minor limit 6 %
```

For example:

```
00:14:21: %SYS-4-CPURESRISE: Resource user Test-proc-14:99s:1w:100n is seeing global cpu
util 11% at total level more than the configured minor limit 6%
User global threshold- Recovery
(1) keywords Critical, Major and Minor will vary accordingly
(2) keywords total, process and interrupt will vary accordingly
=====
00:14:46: %SYS-6-CPURESFALLING: Resource user <user-name> is no longer seeing global high
cpu at total level for the configured critical limit 9%, current value 4%
```

For example:

```
00:14:46: %SYS-6-CPURESFALLING: Resource user Test-proc-14:99s:1w:100n is no longer seeing
global high cpu at total level for the configured critical limit 9%, current value 4%
```

### User Local Threshold Violation in CPU RO

The threshold violation in CPU RO for a user local threshold shows the following output:

```
User local threshold - Violation (keywords Critical, Major and Minor will vary accordingly
- only process level)
=====
00:12:11: %SYS-4-CPURESRISE: Resource user <user-name> is seeing local cpu util 15% at
process level more than the configured minor limit 6%
```

For example:

```
00:12:11: %SYS-4-CPURESRISE: Resource user Test-proc-9:85s:15w:100n is seeing local cpu
util 15% at process level more than the configured minor limit 6%
User local threshold- Recovery (keywords Critical, Major and Minor will vary accordingly
- only process level)
=====
00:13:11: %SYS-6-CPURESFALLING: Resource user <user-name> is no longer seeing local high
cpu at process level for the configured critical limit 9%, current value 3%
```

### System Global Threshold Violation in Memory RO

The threshold violation in memory RO for a system global threshold shows the following output:

```
System global threshold - Violation (keywords Critical, Major and Minor alone will vary
accordingly)
```



(If violation happens in IO memory pool will be : I/O)

```
=====
13:53:22: %SYS-5-GLOBALMEMEXCEED: Global Memory has exceeded the Minor threshold
Pool: Processor Used: 422703520 Threshold: 373885200
```

For example:

```
13:54:03: %SYS-5-GLOBALMEMEXCEED: Global Memory has exceeded the Critical threshold
Pool: Processor Used: 622701556 Threshold: 467356500
System global threshold - Recovery ( keywords Critical, Major and Minor alone will vary
accordingly )
(If recovery happens in IO memory pool will be : I/O)
```

```
=====
%SYS-5-GLOBALMEMRECOVER: Global Memory has recovered after exceeding Minor threshold
Pool: Processor Used: 222473448 Threshold: 355190940
```

For example:

```
13:50:41: %SYS-5-GLOBALMEMRECOVER: Global Memory has recovered after exceeding Critical
threshold
Pool: Processor Used: 222473152 Threshold: 443988675
```

### Per User Global Threshold Violation in Memory RO

The threshold violation in memory RO for a user global threshold shows the following output:

User global threshold - Violation (keywords Critical, Major and Minor alone will vary accordingly)  
(If violation happens in IO memory pool will be : I/O)

```
=====
00:53:14: %SYS-4-RESGLOBALMEMEXCEED: Global Memory has exceeded the Minor threshold configure
by resource user <XYZ>
```

```
Pool: Processor Used: 62273916 Threshold: 62246820
User global threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
(If recovery happens in IO memory pool will be : I/O)
```

```
=====
00:32:56: %SYS-4-RESGLOBALMEMRECOVER: Global Memory has recovered after exceeding the
Critical threshold configure by resource user <XYZ>
```

```
Pool: Processor Used: 329999508 Threshold: 375865440
```

### User Local Threshold Violation in Memory RO

The threshold violation in memory RO for a user local threshold shows the following output:

User local threshold- Violation (keywords Critical, Major and Minor alone will vary accordingly)

```
=====
01:05:42: %SYS-4-RESMEMEXCEED: Resource user <XYZ> has exceeded the Critical memory threshold
Pool: Processor Used: 103754740 Threshold: 103744700
User local threshold - Recovery (keywords Critical, Major and Minor alone will vary
accordingly)
```

```
=====
00:44:43: %SYS-5-RESMEMRECOVER: Resource user <XYZ> has recovered after exceeding the
Critical memory threshold
Pool: Processor Used: 328892280 Threshold :375865440
```

## Examples

### Examples

The following example shows how to configure the minor threshold values for the system global thresholding with a minor rising threshold of 60% at an interval of 12 seconds and a minor falling threshold of 5% at an interval of 10 seconds:

```
Router(config-owner-cpu)# minor rising 60 interval 12 falling 5 interval 10 global
```

```
Router(config-owner-buffer)# minor rising 60 interval 12 falling 5 interval 10 global
```

```
Router(config-owner-memory)# minor rising 60 interval 12 falling 5 interval 10 global
```

**Examples**

The following example shows how to configure the minor threshold values for user local thresholding with a minor rising threshold of 60% at an interval of 12 seconds and a minor falling threshold of 5% at an interval of 10 seconds:

```
Router(config-owner-cpu)# minor rising 60 interval 12 falling 5 interval 10
Router(config-owner-buffer)# minor rising 60 interval 12 falling 5 interval 10

Router(config-owner-memory)# minor rising 60 interval 12 falling 5 interval 10
```

**Examples**

The following example shows how to configure the minor threshold values for per user global thresholding with a minor rising threshold of 60% at an interval of 12 seconds and a minor falling threshold of 5% at an interval of 10 seconds:

```
Router(config-owner-cpu)# minor rising 60 interval 12 falling 5 interval 10 global
Router(config-owner-buffer)# minor rising 60 interval 12 falling 5 interval 10 global
Router(config-owner-memory)# minor rising 60 interval 12 falling 5 interval 10 global
```

**Related Commands**

Command	Description
<b>buffer public</b>	Enters the buffer owner configuration mode and sets threshold values for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets threshold values for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets threshold values for processor level CPU utilization.
<b>cpu total</b>	Enters the CPU owner configuration mode and sets threshold values for total CPU utilization.
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level ROs.

## policy (ERM)

To configure an Embedded Resource Manager (ERM) resource policy, use the **policy** command in ERM configuration mode. To disable this function, use the **no** form of this command.

**policy** *policy-name* [**global**] **type** *resource-user-type*]

**no policy** *policy-name*

### Syntax Description

<i>policy-name</i>	Name of the policy you want to configure.
<b>global</b>	(Optional) Configures a global policy.
<b>type</b>	(Optional) Specifies a type for the policy you are configuring.
<i>resource-user-type</i>	(Optional) Name of the resource user type.

### Command Default

Disabled

### Command Modes

ERM configuration (config-erm)

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

### Usage Guidelines

You can configure a resource policy only in ERM configuration mode.

### Examples

The following example shows how to configure a resource policy with the policy name `cpu_mem_policy` and the resource user type `iosprocess`:

```
Router(config-erm)# policy cpu_mem_policy type iosprocess
```

### Related Commands

Command	Description
<b>resource policy</b>	Enters ERM configuration mode.

<b>Command</b>	<b>Description</b>
<b>show resource all</b>	Displays all the resource details.
<b>show resource database</b>	Displays the resource database details.
<b>show resource owner</b>	Displays the resource owner details.
<b>show resource relationship</b>	Displays the resource relationship details.
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level resource owners.

## policy (resource group)

To apply an already configured policy to a specified resource group, use the **policy** command in resource group configuration mode. To disable this function, use the **no** form of this command.

**policy** *policy-name*

**no policy** *policy-name*

### Syntax Description

<i>policy-name</i>	Name of the policy to apply to the resource group.
--------------------	--

### Command Default

Disabled

### Command Modes

Resource group configuration

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

### Usage Guidelines

Before applying a policy to a resource group, you must configure a resource policy using the **policy***policy-name* command in Embedded Resource Manager (ERM) configuration mode and create a resource group using the **usergroup***resource-group-name***type***resource-user-type* command in ERM configuration mode.

When you apply a policy using the **policy***policy-name* command in resource group configuration mode, you are applying a policy (which contains the thresholds) to the resource group you created using the **usergroup***resource-group-name***type***resource-user-type* command in ERM configuration mode.

For example, you create a resource group with the name `lowPrioUsers` and type `iosprocess` and have low-priority resource users (RUs) or tasks such as HTTP and Simple Network Management Protocol (SNMP) that you want to set a threshold for as a group. You must add the RUs to `lowPrioUsers` using the **instance***instance-name* command and then apply a resource policy. If the resource policy you apply sets a minor rising threshold value of 10 percent, a notification is sent to the RUs in `lowPrioUsers` when the accumulated usage of both HTTP and SNMP RUs crosses the 10 percent threshold (for example, if HTTP usage is 4 percent and SNMP usage is 7 percent).

### Examples

The following example shows how to apply a resource policy named `group-policy1` to a resource group named `lowPrioUsers`:

```
Router(config-erm)# user group lowPrioUsers type iosprocess
Router(config-res-group)# policy group-policy1
```

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>instance (resource group)</b>	Adds the RUs to the resource group.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>user (ERM)</b>	Creates a resource group.

# resource policy

To enter Embedded Resource Manager (ERM) configuration mode to configure an ERM policy, use the **resourcepolicy** command in global configuration mode. To exit ERM configuration mode, use the **no** form of this command.

**resource policy**

**no resource policy**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Disabled

**Command Modes** Global configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Examples** The following example shows how to configure an ERM policy:

```
Router(config)# resource policy
Router(config-erm)# policy memory_policy type iosprocess
Router(config-erm-policy)# system
Router(config-policy-node)# memory processor
Router(config-owner-memory)# critical rising 80
Router(config-owner-memory)# major rising 40 falling 35
```

## Related Commands

Command	Description
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>show resource all</b>	Displays all the resource details.
<b>show resource all</b>	Displays resource details for all RUs.
<b>show resource database</b>	Displays the resource database details.
<b>show resource owner</b>	Displays the resource owner details.
<b>show resource relationship</b>	Displays the resource relationship details.

<b>Command</b>	<b>Description</b>
<b>slot (ERM policy)</b>	Configures line cards.
<b>system (ERM policy)</b>	Configures system level resource owners.



# show resource all

To display the details of a Resource Owner (RO), use the **showresourceall** command in user EXEC or privileged EXEC mode.

**show resource all [brief| detailed]**

## Syntax Description

<b>brief</b>	(Optional) Displays the brief details of the ROs.
<b>detail</b>	(Optional) Displays all the details of the ROs.

## Command Modes

User EXEC (>) Privileged EXEC (#)

## Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

## Examples

The following is sample output from the **showresourceall** command:

```
Router# show resource all
Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777217           0           0         0  0.00%  0.00%  0.00% Init
  Resource User: Scheduler(ID: 0x1000002)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777218           0           0         0  0.00%  0.00%  0.00% Scheduler
  Resource User: Dead(ID: 0x1000003)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777219           0           0         0  0.00%  0.00%  0.00% Dead
  Resource User: Interrupt(ID: 0x1000004)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777220           0           0         0  0.00%  0.00%  0.00% Interrupt
  Resource User: Memory RO RU(ID: 0x1000005)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777221           0           0         0  0.00%  0.00%  0.00% Memory RO RU
  Resource User: Chunk Manager(ID: 0x1000006)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777222           0           13         0  0.00%  0.00%  0.00% Chunk Manager
  Resource User: Load Meter(ID: 0x1000007)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777223           2872        36029       79  0.00%  0.00%  0.00% Load Meter
  Resource User: Check heaps(ID: 0x1000009)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777225           352744      33446      10546  0.00%  0.20%  0.17% Check heaps
```

show resource all

```

Resource User: Pool Manager(ID: 0x100000A)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777226           0          1         0  0.00%  0.00%  0.00% Pool Manager
Resource User: Buffer RO RU(ID: 0x100000B)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777227           0          0         0  0.00%  0.00%  0.00% Buffer RO RU
Resource User: Timers(ID: 0x100000C)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777228           0          2         0  0.00%  0.00%  0.00% Timers
Resource User: Serial Background(ID: 0x100000D)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777229           0          2         0  0.00%  0.00%  0.00% Serial Backgroun
Resource User: AAA_SERVER_DEADTIME(ID: 0x100000E)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777230           0          1         0  0.00%  0.00%  0.00% AAA_SERVER_DEADT
Resource User: AAA high-capacity counters(ID: 0x100000F)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777231           0          2         0  0.00%  0.00%  0.00% AAA high-capacit
Resource User: Policy Manager(ID: 0x1000010)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777232           0          1         0  0.00%  0.00%  0.00% Policy Manager
Resource User: Crash writer(ID: 0x1000011)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777233           0          1         0  0.00%  0.00%  0.00% Crash writer
Resource User: RO Notify Timers(ID: 0x1000012)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777234           0          1         0  0.00%  0.00%  0.00% RO Notify Timers
Resource User: RMI RM Notify Watched Policy(ID: 0x1000013)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777235           0          1         0  0.00%  0.00%  0.00% RMI RM Notify Wa
Resource User: EnvMon(ID: 0x1000014)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777236          11164      92859      120  0.00%  0.00%  0.00% EnvMon
Resource User: IPC Dynamic Cache(ID: 0x1000015)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777237           0          3004         0  0.00%  0.00%  0.00% IPC Dynamic Cach
Resource User: IPC Periodic Timer(ID: 0x1000017)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777239           0          180082       0  0.00%  0.00%  0.00% IPC Periodic Tim
Resource User: IPC Managed Timer(ID: 0x1000018)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777240           572          79749         7  0.00%  0.00%  0.00% IPC Managed Time
Resource User: IPC Deferred Port Closure(ID: 0x1000019)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777241           4          180088         0  0.00%  0.00%  0.00% IPC Deferred Por
Resource User: IPC Seat Manager(ID: 0x100001A)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777242          97560      1408799         69  0.23%  0.02%  0.00% IPC Seat Manager
Resource User: IPC Session Service(ID: 0x100001B)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777243           0          1         0  0.00%  0.00%  0.00% IPC Session Serv
Resource User: ARP Input(ID: 0x100001C)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777244           20          3082         6  0.00%  0.00%  0.00% ARP Input
Resource User: EEM ED Syslog(ID: 0x100001D)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
.
.
.
Resource Owner: memory
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
Chunk Elements :
Allocated Size(b): 35152564 Count: 91901 Freed Size(b): 31793276 Count: 39159
Processor memory
Total Memory held : 46596832 bytes
pc = 0x403089D8, size = 10499724, count = 1
pc = 0x402996C8, size = 6737976, count = 8298
pc = 0x402F0C9C, size = 5821352, count = 10
pc = 0x40A25134, size = 4194324, count = 1
pc = 0x41D6D414, size = 1704144, count = 52
pc = 0x40451BE0, size = 1114180, count = 17
pc = 0x402D0DAC, size = 917600, count = 1

```

```

pc = 0x4043E5F4, size = 836076, count = 12291
pc = 0x404A276C, size = 617476, count = 1
pc = 0x41CDED1C, size = 569844, count = 125
pc = 0x4194C2D0, size = 524292, count = 1
pc = 0x405FD93C, size = 516100, count = 1
pc = 0x414D67AC, size = 473224, count = 199
pc = 0x41016294, size = 458756, count = 1
pc = 0x4046E618, size = 432096, count = 1
pc = 0x400A1134, size = 412420, count = 1
pc = 0x402ABB50, size = 317316, count = 93
pc = 0x41D53668, size = 262148, count = 1
pc = 0x4049BA04, size = 206640, count = 84
pc = 0x41E3FE30, size = 196620, count = 3
pc = 0x40B05214, size = 196612, count = 1
pc = 0x40494D94, size = 180180, count = 4095
pc = 0x402ABB6C, size = 144708, count = 93
pc = 0x41586A38, size = 144004, count = 1
pc = 0x4030B408, size = 140028, count = 7
pc = 0x415090EC, size = 131768, count = 4
pc = 0x41E37B94, size = 131088, count = 4
pc = 0x4195C348, size = 131076, count = 1
pc = 0x400A1194, size = 124420, count = 1
pc = 0x41503BC4, size = 122768, count = 1
pc = 0x404E888C, size = 114660, count = 4095
pc = 0x40494D50, size = 114660, count = 4095
pc = 0x404D99B0, size = 114660, count = 4095
pc = 0x4023F5B4, size = 98312, count = 2
pc = 0x41E45894, size = 97456, count = 626
pc = 0x41E2D4C4, size = 91584, count = 12
pc = 0x416D9768, size = 84004, count = 1
pc = 0x40452790, size = 84000, count = 3000
pc = 0x40322A74, size = 81948, count = 7
pc = 0x41D0FF4C, size = 81924, count = 1
pc = 0x40E9F7B0, size = 81364, count = 1
pc = 0x414FB1BC, size = 78740, count = 2
pc = 0x414D4A64, size = 72916, count = 2
pc = 0x40328770, size = 72144, count = 36
pc = 0x414FA938, size = 71592, count = 2
pc = 0x414EF938, size = 71096, count = 2
pc = 0x41947EEC, size = 65540, count = 1
pc = 0x41935B5C, size = 65540, count = 1
pc = 0x4193A348, size = 65540, count = 1
pc = 0x4193FF5C, size = 65540, count = 1
pc = 0x41D6E32C, size = 65540, count = 1
pc = 0x41DD534C, size = 65540, count = 1
pc = 0x414B5870, size = 65540, count = 1
pc = 0x4078521C, size = 65540, count = 1
.
.
.
I/O memory
Total Memory held : 9816224 bytes
pc = 0x4029983C, size = 9791584, count = 8290
pc = 0x403EC2A4, size = 8208, count = 1
pc = 0x403F8CD0, size = 8208, count = 1
pc = 0x403EC2E0, size = 4112, count = 1
pc = 0x403F8D0C, size = 4112, count = 1
Resource User: Scheduler(ID: 0x1000002)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 13052 bytes
pc = 0x4037BCC8, size = 12004, count = 1
pc = 0x40327110, size = 1048, count = 24
Resource User: Dead(ID: 0x1000003)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 447448 bytes
pc = 0x404A276C, size = 395636, count = 5
pc = 0x4043E5F4, size = 18676, count = 271
pc = 0x40494D94, size = 6888, count = 82
pc = 0x4044B9E4, size = 6672, count = 6

```

show resource all

```

pc = 0x40C8BAB4, size =      5780, count =   34
pc = 0x404943DC, size =      2836, count =   82
pc = 0x40494D50, size =      2796, count =   82
pc = 0x4044DAF0, size =      2224, count =    2
pc = 0x40393168, size =      1772, count =    1
pc = 0x40FF2688, size =       728, count =    6
pc = 0x40CBC5A4, size =       400, count =    4
pc = 0x40455144, size =       320, count =   10
pc = 0x40C9A8D8, size =       288, count =    8
pc = 0x40CADE10, size =       260, count =    5
pc = 0x40B19484, size =       256, count =    2
pc = 0x4052BD2C, size =       208, count =    4
pc = 0x40CADE50, size =       188, count =    5
pc = 0x4044FBD8, size =       184, count =    1
pc = 0x40A9B2F0, size =       184, count =    1
pc = 0x40CBC45C, size =       160, count =    2
pc = 0x4038BF34, size =       144, count =    2
pc = 0x40529610, size =       136, count =    2
pc = 0x405CF034, size =       104, count =    1
pc = 0x414D67AC, size =       104, count =    1
pc = 0x4038BF68, size =        88, count =    2
pc = 0x4044F078, size =        84, count =    3
pc = 0x41555624, size =        84, count =    1
pc = 0x40685250, size =        76, count =    1
pc = 0x40481AD4, size =        68, count =    1
pc = 0x4044DB18, size =        56, count =    2
pc = 0x401B6960, size =        48, count =    1
Resource User: Interrupt(ID: 0x1000004)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 39652 Count: 1070
Processor memory
Total Memory held : 0 bytes
Resource User: Memory RO RU(ID: 0x1000005)
Chunk Elements :
Allocated Size(b): 12320 Count: 120 Freed Size(b): 10164 Count: 99
Processor memory
Total Memory held : 131080 bytes
pc = 0x40357C54, size =      65540, count =    1
pc = 0x40357D98, size =      65540, count =    1
Resource User: Chunk Manager(ID: 0x1000006)
Chunk Elements :
Allocated Size(b): 124 Count: 6 Freed Size(b): 48 Count: 3
Processor memory
Total Memory held : 9788 bytes
pc = 0x4037BCC8, size =       6004, count =    1
pc = 0x40332490, size =       3008, count =    2
pc = 0x4035E160, size =        636, count =    1
pc = 0x403604BC, size =        140, count =    1
Resource User: Load Meter(ID: 0x1000007)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 44 Count: 1
Processor memory
Total Memory held : 3780 bytes
pc = 0x4037BCC8, size =       3004, count =    1
pc = 0x4035E160, size =        636, count =    1
pc = 0x403604BC, size =        140, count =    1
Resource User: Check heaps(ID: 0x1000009)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 44 Count: 1
Processor memory
Total Memory held : 7236 bytes
pc = 0x4037BCC8, size =       6004, count =    1
pc = 0x4035E160, size =        636, count =    1
pc = 0x41E2B0D0, size =       324, count =    1
pc = 0x403604BC, size =        140, count =    1
pc = 0x40351D2C, size =        76, count =    1
pc = 0x40351CF8, size =        56, count =    1
Resource User: Pool Manager(ID: 0x100000A)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 6780 bytes
pc = 0x4037BCC8, size =       6004, count =    1

```

```

pc = 0x4035E160, size =      636, count =    1
pc = 0x403604BC, size =      140, count =    1
Resource User: Buffer RO RU(ID: 0x100000B)
Chunk Elements :
Allocated Size(b): 4960 Count: 40 Freed Size(b): 4092 Count: 33
Processor memory
Total Memory held : 0 bytes
Resource User: Timers(ID: 0x100000C)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 44 Count: 1
.
.
Resource User: PF_Init Process(ID: 0x100004F)
Chunk Elements :
Allocated Size(b): 8104 Count: 126 Freed Size(b): 1400 Count: 29
Processor memory
Total Memory held : 31204 bytes
pc = 0x4027EF10, size =    21540, count =    5
pc = 0x4037BCC8, size =    6004, count =    1
pc = 0x4044DAF0, size =    1112, count =    1
pc = 0x4035E160, size =     636, count =    1
pc = 0x4038BF68, size =     308, count =    7
pc = 0x4038BF34, size =     280, count =    7
pc = 0x403604BC, size =     280, count =    2
pc = 0x41E45ED0, size =     240, count =    5
pc = 0x401FB400, size =     236, count =    5
pc = 0x40529610, size =     136, count =    2
pc = 0x4047D560, size =     108, count =    2
pc = 0x4038C114, size =      88, count =    2
pc = 0x4044DB18, size =      72, count =    1
pc = 0x40211DCC, size =      56, count =    2
pc = 0x4038E038, size =      44, count =    1
pc = 0x40402C98, size =      32, count =    1
pc = 0x40455144, size =      32, count =    1
Resource User: PF_Split Sync Process(ID: 0x1000052)
Chunk Elements :
Allocated Size(b): 6092 Count: 87 Freed Size(b): 5644 Count: 81
Processor memory
Total Memory held : 10356 bytes
pc = 0x4037BCC8, size =    6004, count =    1
pc = 0x4060364C, size =    1760, count =   10
pc = 0x41E45894, size =     960, count =    2
pc = 0x4060AE18, size =     856, count =   10
pc = 0x4035E160, size =     636, count =    1
pc = 0x403604BC, size =     140, count =    1
Resource User: RPC pf-split-rp(ID: 0x1000053)
Chunk Elements :
Allocated Size(b): 1348 Count: 20 Freed Size(b): 1304 Count: 19
Processor memory
Total Memory held : 6780 bytes
pc = 0x4037BCC8, size =    6004, count =    1
pc = 0x4035E160, size =     636, count =    1
pc = 0x403604BC, size =     140, count =    1
Resource User: RPC idprom-MP(ID: 0x1000054)
Chunk Elements :
Allocated Size(b): 4708 Count: 68 Freed Size(b): 4664 Count: 67
Processor memory
Total Memory held : 16648 bytes
pc = 0x405023D4, size =    9732, count =   18
pc = 0x4037BCC8, size =    6004, count =    1
pc = 0x4035E160, size =     636, count =    1
pc = 0x403604BC, size =     140, count =    1
pc = 0x405D000C, size =     136, count =    1
Resource User: Net Input(ID: 0x1000055)
Chunk Elements :
Allocated Size(b): 88 Count: 2 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 6780 bytes
pc = 0x4037BCC8, size =    6004, count =    1
pc = 0x4035E160, size =     636, count =    1
pc = 0x403604BC, size =     140, count =    1
Resource User: Compute load avgs(ID: 0x1000056)

```

show resource all

```

Chunk Elements :
Allocated Size(b): 11948724 Count: 215941 Freed Size(b): 11948724 Count: 215941
Processor memory
Total Memory held : 10720 bytes
pc = 0x4037BCC8, size =      6004, count =      1
pc = 0x404FC9C0, size =      3940, count =      1
pc = 0x4035E160, size =       636, count =      1
pc = 0x403604BC, size =       140, count =      1
Resource User: RTTYS Process(ID: 0x1000057)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 6780 bytes
pc = 0x4037BCC8, size =      6004, count =      1
pc = 0x4035E160, size =       636, count =      1
pc = 0x403604BC, size =       140, count =      1
Resource User: BACK CHECK(ID: 0x1000059)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 6780 bytes
pc = 0x4037BCC8, size =      6004, count =      1
pc = 0x4035E160, size =       636, count =      1
pc = 0x403604BC, size =       140, count =      1
Resource User: chkpt message handler(ID: 0x100005A)
Chunk Elements :
Allocated Size(b): 156 Count: 2 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 6780 bytes
pc = 0x4037BCC8, size =      6004, count =      1
pc = 0x4035E160, size =       636, count =      1
pc = 0x403604BC, size =       140, count =      1
Resource User: cpf_process_msg_holdq(ID: 0x100005B)
Chunk Elements :
Allocated Size(b): 152 Count: 3 Freed Size(b): 0 Count: 0
.
.
.
Resource Owner: Buffer
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
Getbufs Retbufs Holding RU Name
1367 31237 4294937426 Init
Resource User: Scheduler(ID: 0x1000002)
Getbufs Retbufs Holding RU Name
0 0 0 Scheduler
Resource User: Dead(ID: 0x1000003)
Getbufs Retbufs Holding RU Name
6 3 3 Dead
Resource User: Interrupt(ID: 0x1000004)
Getbufs Retbufs Holding RU Name
221580 221580 0 Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
Getbufs Retbufs Holding RU Name
0 0 0 Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)
Getbufs Retbufs Holding RU Name
0 0 0 Chunk Manager
Resource User: Load Meter(ID: 0x1000007)
Getbufs Retbufs Holding RU Name
0 0 0 Load Meter
Resource User: Check heaps(ID: 0x1000009)
Getbufs Retbufs Holding RU Name
0 0 0 Check heaps
Resource User: Pool Manager(ID: 0x100000A)
Getbufs Retbufs Holding RU Name
5554 0 5554 Pool Manager
Resource User: Buffer RO RU(ID: 0x100000B)
Getbufs Retbufs Holding RU Name
0 0 0 Buffer RO RU
Resource User: Timers(ID: 0x100000C)
Getbufs Retbufs Holding RU Name
0 0 0 Timers

```

```

Resource User: Serial Background(ID: 0x100000D)
Getbufs Retbufs Holding RU Name
0 0 0 Serial Backgroun
Resource User: AAA_SERVER_DEADTIME(ID: 0x100000E)
Getbufs Retbufs Holding RU Name
0 0 0 AAA_SERVER_DEADT
Resource User: AAA high-capacity counters(ID: 0x100000F)
Getbufs Retbufs Holding RU Name
0 0 0 AAA high-capacit
Resource User: Policy Manager(ID: 0x1000010)
Getbufs Retbufs Holding RU Name
0 0 0 Policy Manager
Resource User: Crash writer(ID: 0x1000011)
Getbufs Retbufs Holding RU Name
0 0 0 Crash writer
Resource User: RO Notify Timers(ID: 0x1000012)
Getbufs Retbufs Holding RU Name
0 0 0 RO Notify Timers
Resource User: RMI RM Notify Watched Policy(ID: 0x1000013)
Getbufs Retbufs Holding RU Name
0 0 0 RMI RM Notify Wa
.
.
Resource User: DHCPD Timer(ID: 0x100011B)
Getbufs Retbufs Holding RU Name
0 0 0 DHCPD Timer
Resource User: DHCPD Database(ID: 0x100011C)
Getbufs Retbufs Holding RU Name
0 0 0 DHCPD Database
Resource User: draco-oir-process:slot 2(ID: 0x100011E)
Getbufs Retbufs Holding RU Name
0 0 0 draco-oir-proces
Resource User: SCP async: Draco-LC4(ID: 0x1000125)
Getbufs Retbufs Holding RU Name
35849 243101 4294760044 SCP async: Draco
Resource User: IFCOM Msg Hdlr(ID: 0x1000127)
Getbufs Retbufs Holding RU Name
2 2 0 IFCOM Msg Hdlr
Resource User: IFCOM Msg Hdlr(ID: 0x1000128)
Getbufs Retbufs Holding RU Name
28 28 0 IFCOM Msg Hdlr
Resource User: Exec(ID: 0x100012C)
Getbufs Retbufs Holding RU Name
912 912 0 Exec
Resource Owner: test_mem
Resource User Type: test_process
Resource User Type: mem_rut
Resource Owner: test_cpu
Resource User Type: test_process
Resource User Type: cpu_rut

```

The following is a sample output from the **showresourceallbrief** command:

```

Router# show resource all brief
Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
  RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777217 0 0 0 0.00% 0.00% 0.00% Init
Resource User: Scheduler(ID: 0x1000002)
  RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777218 0 0 0 0.00% 0.00% 0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
  RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777219 0 0 0 0.00% 0.00% 0.00% Dead
Resource User: Interrupt(ID: 0x1000004)
  RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777220 0 0 0 0.00% 0.00% 0.00% Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
  RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777221 0 0 0 0.00% 0.00% 0.00% Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)

```

show resource all

```

RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777222         0         13       0    0.00% 0.00% 0.00% Chunk Manager
Resource User: Load Meter (ID: 0x1000007)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777223        2872       36069    79    0.00% 0.00% 0.00% Load Meter
Resource User: Check heaps (ID: 0x1000009)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777225       353092     33481   10546  0.00% 0.17% 0.17% Check heaps
Resource User: Pool Manager (ID: 0x100000A)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777226         0         1       0    0.00% 0.00% 0.00% Pool Manager
Resource User: Buffer RO RU (ID: 0x100000B)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777227         0         0       0    0.00% 0.00% 0.00% Buffer RO RU
Resource User: Timers (ID: 0x100000C)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777228         0         2       0    0.00% 0.00% 0.00% Timers
Resource User: Serial Background (ID: 0x100000D)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777229         0         2       0    0.00% 0.00% 0.00% Serial Backgroun
Resource User: AAA_SERVER_DEADTIME (ID: 0x100000E)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777230         0         1       0    0.00% 0.00% 0.00% AAA_SERVER_DEADT
Resource User: AAA high-capacity counters (ID: 0x100000F)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777231         0         2       0    0.00% 0.00% 0.00% AAA high-capacit
Resource User: Policy Manager (ID: 0x1000010)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777232         0         1       0    0.00% 0.00% 0.00% Policy Manager
Resource User: Crash writer (ID: 0x1000011)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777233         0         1       0    0.00% 0.00% 0.00% Crash writer
Resource User: RO Notify Timers (ID: 0x1000012)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777234         0         1       0    0.00% 0.00% 0.00% RO Notify Timers
Resource User: RMI RM Notify Watched Policy (ID: 0x1000013)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777235         0         1       0    0.00% 0.00% 0.00% RMI RM Notify Wa
Resource User: EnvMon (ID: 0x1000014)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777236       11176     92958   120    0.00% 0.00% 0.00% EnvMon
Resource User: IPC Dynamic Cache (ID: 0x1000015)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777237         0       3007     0    0.00% 0.00% 0.00% IPC Dynamic Cach
Resource User: IPC Periodic Timer (ID: 0x1000017)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777239         0     180279   0    0.00% 0.00% 0.00% IPC Periodic Tim
Resource User: IPC Managed Timer (ID: 0x1000018)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777240         572     79833    7    0.00% 0.00% 0.00% IPC Managed Time
Resource User: IPC Deferred Port Closure (ID: 0x1000019)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777241         4     180285   0    0.00% 0.00% 0.00% IPC Deferred Por
Resource User: IPC Seat Manager (ID: 0x100001A)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777242       97684   1410183  69    0.00% 0.03% 0.00% IPC Seat Manager
Resource User: IPC Session Service (ID: 0x100001B)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777243         0         1       0    0.00% 0.00% 0.00% IPC Session Serv
Resource User: ARP Input (ID: 0x100001C)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777244         20       3085     6    0.00% 0.00% 0.00% ARP Input
Resource User: EEM ED Syslog (ID: 0x100001D)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777245         0         49       0    0.00% 0.00% 0.00% EEM ED Syslog
Resource User: DDR Timers (ID: 0x100001E)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777246         0         2       0    0.00% 0.00% 0.00% DDR Timers
Resource User: Dialer event (ID: 0x100001F)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777247         0         2       0    0.00% 0.00% 0.00% Dialer event
Resource User: Entity MIB API (ID: 0x1000020)
RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr

```



```

16777248      28      16      1750  0.00%  0.00%  0.00% Entity MIB API
  Resource User: Compute SRP rates(ID: 0x1000021)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777249      0     18037      0  0.00%  0.00%  0.00% Compute SRP rate
  Resource User: SERIAL A'detect(ID: 0x1000022)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777250      0      1      0  0.00%  0.00%  0.00% SERIAL A'detect
  Resource User: GraphIt(ID: 0x1000023)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777251      0     180267      0  0.00%  0.00%  0.00% GraphIt
  Resource User: rf proxy rp agent(ID: 0x1000024)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777252     40      416      96  0.00%  0.00%  0.00% rf proxy rp agen
  Resource User: HC Counter Timers(ID: 0x1000025)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777253     60     41360      1  0.00%  0.00%  0.00% HC Counter Timer
  Resource User: Snmp ICC Process(ID: 0x1000026)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777254      0      1      0  0.00%  0.00%  0.00% Snmp ICC Process
  Resource User: Cat6k SNMP(ID: 0x1000027)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777255     20      29      689  0.00%  0.00%  0.00% Cat6k SNMP
  Resource User: Cat6k SNMP Trap handler(ID: 0x1000028)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777256      0      7      0  0.00%  0.00%  0.00% Cat6k SNMP Trap
  Resource User: Critical Bkgnd(ID: 0x1000029)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777257      0      1      0  0.00%  0.00%  0.00% Critical Bkgnd
  Resource User: Net Background(ID: 0x100002A)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777258     112     44787      2  0.00%  0.00%  0.00% Net Background
  Resource User: Logger(ID: 0x100002B)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777259      0      50      0  0.00%  0.00%  0.00% Logger
  Resource User: TTY Background(ID: 0x100002C)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777260      0     180263      0  0.00%  0.00%  0.00% TTY Background
  Resource User: Per-Second Jobs(ID: 0x100002D)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777261     52     180549      0  0.00%  0.00%  0.00% Per-Second Jobs
  Resource User: Per-minute Jobs(ID: 0x100002E)
    RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
.
.
.
Resource User: Exec(ID: 0x100012C)
  RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777516     8964      965     9289  0.39%  0.66%  1.55% Exec
Resource Owner: memory
  Resource User Type: iosprocess
  Resource User: Init(ID: 0x1000001)
Processor memory
Allocated  Freed  Holding  Blocks
55233064  8636232  46596832  48832
I/O memory
Allocated  Freed  Holding  Blocks
9816224   0  9816224  8294
  Resource User: Scheduler(ID: 0x1000002)
Processor memory
Allocated  Freed  Holding  Blocks
13052     0  13052    25
  Resource User: Dead(ID: 0x1000003)
Processor memory
Allocated  Freed  Holding  Blocks
687916   240468  447448    630
  Resource User: Interrupt(ID: 0x1000004)
Processor memory
Allocated  Freed  Holding  Blocks
0         0     0         0
  Resource User: Memory RO RU(ID: 0x1000005)
Processor memory
Allocated  Freed  Holding  Blocks
131080     0  131080    2

```

show resource all

```

Resource User: Chunk Manager(ID: 0x1000006)
Processor memory
Allocated Freed Holding Blocks
14300 4512 9788 5
Resource User: Load Meter(ID: 0x1000007)
Processor memory
Allocated Freed Holding Blocks
3920 140 3780 3
Resource User: Check heaps(ID: 0x1000009)
Processor memory
Allocated Freed Holding Blocks
7376 140 7236 6
Resource User: Pool Manager(ID: 0x100000A)
Processor memory
Allocated Freed Holding Blocks
6780 0 6780 3
Resource User: Buffer RO RU(ID: 0x100000B)
Processor memory
Allocated Freed Holding Blocks
0 0 0 0
Resource User: Timers(ID: 0x100000C)
Processor memory
Allocated Freed Holding Blocks
6920 140 6780 3
Resource User: Serial Background(ID: 0x100000D)
Processor memory
Allocated Freed Holding Blocks
6920 140 6780 3
.
.
Resource User: IFCOM Msg Hdlr(ID: 0x1000128)
Getbufs Retbufs Holding RU Name
28 28 0 IFCOM Msg Hdlr
Resource User: Exec(ID: 0x100012C)
Getbufs Retbufs Holding RU Name
1404 1404 0 Exec
Resource Owner: test_mem
Resource User Type: test_process
Resource User Type: mem_rut
Resource Owner: test_cpu
Resource User Type: test_process
Resource User Type: cpu_rut

```

The following is sample output from the **showresourcealldetailed** command:

```

Router# show resource all detailed
Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777217 0 0 0 0.00% 0.00% 0.00% Init
Resource User: Scheduler(ID: 0x1000002)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777218 0 0 0 0.00% 0.00% 0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777219 0 0 0 0.00% 0.00% 0.00% Dead
Resource User: Interrupt(ID: 0x1000004)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777220 0 0 0 0.00% 0.00% 0.00% Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777221 0 0 0 0.00% 0.00% 0.00% Memory RO RU
Resource User: Chunk Manager (ID: 0x1000006)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777222 0 13 0 0.00% 0.00% 0.00% Chunk Manager
Resource User: Load Meter (ID: 0x1000007)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777223 2872 36075 79 0.00% 0.00% 0.00% Load Meter
Resource User: Check heaps(ID: 0x1000009)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Ushr
16777225 353168 33486 10546 0.00% 0.10% 0.15% Check heaps

```

```

Resource User: Pool Manager (ID: 0x100000A)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777226          0          1          0  0.00% 0.00% 0.00% Pool Manager
Resource User: Buffer RO RU (ID: 0x100000B)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777227          0          0          0  0.00% 0.00% 0.00% Buffer RO RU
Resource User: Timers (ID: 0x100000C)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777228          0          2          0  0.00% 0.00% 0.00% Timers
Resource User: Serial Background (ID: 0x100000D)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777229          0          2          0  0.00% 0.00% 0.00% Serial Background
Resource User: AAA_SERVER_DEADTIME (ID: 0x100000E)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777230          0          1          0  0.00% 0.00% 0.00% AAA_SERVER_DEADT
Resource User: AAA high-capacity counters (ID: 0x100000F)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777231          0          2          0  0.00% 0.00% 0.00% AAA high-capacit
Resource User: Policy Manager (ID: 0x1000010)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777232          0          1          0  0.00% 0.00% 0.00% Policy Manager
Resource User: Crash writer (ID: 0x1000011)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777233          0          1          0  0.00% 0.00% 0.00% Crash writer
Resource User: RO Notify Timers (ID: 0x1000012)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777234          0          1          0  0.00% 0.00% 0.00% RO Notify Timers
Resource User: RMI RM Notify Watched Policy (ID: 0x1000013)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777235          0          1          0  0.00% 0.00% 0.00% RMI RM Notify Wa
Resource User: EnvMon (ID: 0x1000014)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777236          11176      92958      120 0.00% 0.00% 0.00% EnvMon
Resource User: IPC Dynamic Cache (ID: 0x1000015)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr
16777237          0          3008       0  0.00% 0.00% 0.00% IPC Dynamic Cach
Resource User: IPC Periodic Timer (ID: 0x1000017)
RUID Runtime(ms)  Invoked      uSecs  5Sec  1Min  5Min Res Usr

```

```

.
.
.

```

Resource Owner: memory

Resource User Type: iosprocess

Resource User: Init (ID: 0x1000001)

Chunk Elements :

Allocated Size(b): 35152564 Count: 91901 Freed Size(b): 31793276 Count: 39159

Processor memory

Address	Bytes	Prev	Next	Ref	Alloc	PC	What
4393BAA0	0010499772	00000000	4433F15C	001	513DD000		*Init*
4433F15C	0000012852	4393BAA0	44342390	001	513DD000		*Init*
44342390	0000005052	4433F15C	4434374C	001	513DD000		List Headers
4434374C	0000000096	44342390	443437AC	001	513DD000		*Init*
443437AC	0000000096	4434374C	4434380C	001	513DD000		*Init*
4434380C	0000000096	443437AC	4434386C	001	513DD000		*Init*
4434386C	0000000096	4434380C	443438CC	001	513DD000		*Init*
443438CC	0000000096	4434386C	4434392C	001	513DD000		*Init*
4434392C	0000004356	443438CC	44344A30	001	513DD000		TTY data
44344A30	0000000564	4434392C	44344C64	001	513DD000		TTY Output Buf
44344C64	0000000096	44344A30	44344CC4	001	513DD000		*Init*
44344CC4	0000001552	44344C64	443452D4	001	513DD000		Watched messages
443452D4	0000010052	44344CC4	44347A18	001	513DD000		Watched Boolean
44347A18	0000001552	443452D4	44348028	001	513DD000		Watched Semaphore
44348028	0000000380	44347A18	443481A4	001	513DD000		Watched Message Queue
443481A4	0000003052	44348028	44348D90	001	513DD000		Read/Write Locks
44348D90	0000020052	443481A4	4434DBE4	001	513DD000		RMI-RO_RU Chunks
4434DBE4	0000000116	44348D90	4434DC58	001	513DD000		Resource Owner IDs
4434DC58	0000001552	4434DBE4	4434E268	001	513DD000		String-DB entries
4434E268	0000000532	4434DC58	4434E47C	001	513DD000		String-DB handles
4434E47C	0000000076	4434E268	4434E4C8	001	513DD000		NameDB String
4434E4C8	0000000116	4434E47C	4434E53C	001	513DD000		Resource User Type IDs
4434E53C	0000000184	4434E4C8	4434E5F4	001	513DD000		*Init*
4434E5F4	0000002100	4434E53C	4434EE28	001	513DD000		Resource Owner IDs
4434EE28	0000000076	4434E5F4	4434EE74	001	513DD000		NameDB String

show resource all

```

4434EE74 0000000076 4434EE28 4434EEC0 001 513DD000 NameDB String
4434EEC0 0000065588 4434EE74 4435EEF4 001 513DD000 Buffer RU Notify Chunks
44360754 0000000076 44360698 443607A0 001 513DD000 *Init*
443607A0 0000002100 44360754 44360FD4 001 513DD000 Resource User Type IDs
44360FD4 0000004148 443607A0 44362008 001 513DD000 Resource User IDs
44362008 0000000076 44360FD4 44362054 001 513DD000 NameDB String
44362054 0000000076 44362008 443620A0 001 513DD000 NameDB String
443620A0 0000000096 44362054 44362100 001 513DD000 *Init*
443623AC 0000000076 44362100 443623F8 001 513DD000 NameDB String
443623F8 0000010052 443623AC 44364B3C 001 513DD000 List Elements
44364B3C 0000010052 443623F8 44367280 001 513DD000 List Elements
4436758C 0000001552 4436752C 44367B9C 001 513DD000 Reg Function iList
44367B9C 0000000164 4436758C 44367C40 001 513DD000 *Init*
44367C40 0000000076 44367B9C 44367C8C 001 513DD000 Parser Linkage
44367C8C 0000000076 44367C40 44367CD8 001 513DD000 Parser Linkage
44367CD8 0000000076 44367C8C 44367D24 001 513DD000 Parser Linkage
44367D70 0000000076 44367D24 44367DBC 001 513DD000 Parser Linkage
44367DBC 0000000076 44367D70 44367E08 001 513DD000 Cond Debug definition
44367E08 0000000076 44367DBC 44367E54 001 513DD000 Parser Linkage
44367E54 0000000076 44367E08 44367EA0 001 513DD000 Cond Debug definition
44367EA0 0000000076 44367E54 44367EEC 001 513DD000 Cond Debug definition
44367EEC 0000000076 44367EA0 44367F38 001 513DD000 Cond Debug definition
44367F38 0000000076 44367EEC 44367F84 001 513DD000 Cond Debug definition
44367F84 0000000384 44367F38 44368104 001 513DD000 *Init*
4436B5C8 0000000076 4436B57C 4436B614 001 513DD000 Init
4436B614 0000000076 4436B5C8 4436B660 001 513DD000 Init
4436B660 0000000076 4436B614 4436B6AC 001 513DD000 Init
4436BC04 0000000076 4436BBB8 4436BC50 001 513DD000 Init
4436BC50 0000003460 4436BC04 4436C9D4 001 513DD000 *Hardware IDB*
4436C9D4 0000000076 4436BC50 4436CA20 001 513DD000 Init
4436CA20 0000001080 4436C9D4 4436CE58 001 513DD000 Index Table Block
4436CE58 0000000076 4436CA20 4436CEA4 001 513DD000 Init
4436CEA4 0000000076 4436CE58 4436CEF0 001 513DD000 Init
4436CEF0 0000000308 4436CEA4 4436D024 001 513DD000 Init
4436D024 0000000076 4436CEF0 4436D070 001 513DD000 NameDB String
4436D070 0000000104 4436D024 4436D0D8 001 513DD000 NameDB String
4436D434 0000000096 4436D188 4436D494 001 513DD000 Init
4436D740 0000000096 4436D494 4436D7A0 001 513DD000 Init
4436D7A0 0000010052 4436D740 4436FEE4 001 513DD000 Packet Elements
4436FEE4 0000000372 4436D7A0 44370058 001 513DD000 Pool Info
44370058 0000000372 4436FEE4 443701CC 001 513DD000 Pool Info
443701CC 0000000372 44370058 44370340 001 513DD000 Pool Info
44370340 0000000860 443701CC 4437069C 001 513DD000 *Packet Header*
4437069C 0000000372 44370340 44370810 001 513DD000 Pool Info
44370810 0000000860 4437069C 44370B6C 001 513DD000 *Packet Header*
44370B6C 0000000860 44370810 44370EC8 001 513DD000 *Packet Header*
44370EC8 0000000860 44370B6C 44371224 001 513DD000 *Packet Header*
44371224 0000000860 44370EC8 44371580 001 513DD000 *Packet Header*
44371580 0000000860 44371224 443718DC 001 513DD000 *Packet Header*
443718DC 0000000860 44371580 44371C38 001 513DD000 *Packet Header*
44371C38 0000000860 443718DC 44371F94 001 513DD000 *Packet Header*
44371F94 0000000860 44371C38 443722F0 001 513DD000 *Packet Header*
443722F0 0000000860 44371F94 4437264C 001 513DD000 *Packet Header*
4437264C 0000000860 443722F0 443729A8 001 513DD000 *Packet Header*
443729A8 0000000860 4437264C 44372D04 001 513DD000 *Packet Header*
.
.
.
Resource User: Compute SRP rates(ID: 0x1000021)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
  Address      Bytes      Prev      Next Ref   Alloc PC  What
446D502C 0000006052 446D4D5C 446D67D0 001 513DD000 Init
446D67D0 0000000188 446D502C 446D688C 001 513DD000 Process Events
5055163C 0000000684 505512CC 505518E8 001 513DD000 Init
Resource User: SERIAL A'detect(ID: 0x1000022)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 0 Count: 0
Processor memory
  Address      Bytes      Prev      Next Ref   Alloc PC  What
44722FCC 0000000684 4471DE58 44723278 001 513DD000 Init
50598A4C 0000006052 505989E8 5059A1F0 001 513DD000 Init

```

```

5059A1F0 0000000188 50598A4C 5059A2AC 001 513DD000 Process Events
  Resource User: GraphIt(ID: 0x1000023)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 44 Count: 1
Processor memory
  Address      Bytes      Prev      Next Ref   Alloc PC  What
447235B8 0000000684 4472356C 44723864 001 513DD000 Init
5059A8A8 00000006052 5059A350 5059C04C 001 513DD000 Init
5059C04C 0000000188 5059A8A8 5059C108 001 513DD000 Process Events
  Resource User: rf proxy rp agent(ID: 0x1000024)
Chunk Elements :
Allocated Size(b): 39056 Count: 504 Freed Size(b): 33756 Count: 452
Processor memory
  Address      Bytes      Prev      Next Ref   Alloc PC  What
446B752C 0000000144 446B74D4 446B75BC 001 513DD000 NameDB String
44728FC0 0000000684 44728F74 4472926C 001 513DD000 Init
44B19780 0000001160 44B1867C 44B19C08 001 513DD000 IPC Port
44B204A0 0000000148 44B2042C 44B20534 001 513DD000 IPC Name String
44B220E8 0000000096 44B2202C 44B22148 001 513DD000 rf proxy rp agent
44B22148 0000001160 44B220E8 44B225D0 001 513DD000 IPC Port
44B22938 0000000076 44B2287C 44B22984 001 513DD000 NameDB String
44B22984 0000000096 44B22938 44B229E4 001 513DD000 rf proxy rp agent
44B22D4C 0000000076 44B22C90 44B22D98 001 513DD000 NameDB String
44B22D98 0000000096 44B22D4C 44B22DF8 001 513DD000 rf proxy rp agent
44B23160 0000000076 44B230A4 44B231AC 001 513DD000 NameDB String
44B231AC 0000000096 44B23160 44B2320C 001 513DD000 rf proxy rp agent
44B2320C 0000000076 44B231AC 44B23258 001 513DD000 IPC Name String
50543ABC 0000000104 50543A00 50543B24 001 513DD000 IPC Name
5061CC34 0000000188 5059EC00 5061CCF0 001 513DD000 Process Events
5061CDB4 00000006052 5061CD68 5061E558 001 513DD000 Init
50A8780C 0000000132 50A877C0 50A87890 001 513DD000 IPC Name String
50AC8094 0000065588 50AC7C0C 50AD80C8 001 513DD000 EvtMgr active chunk
50AD986C 0000001160 50AD80C8 50AD9CF4 001 513DD000 IPC Port
  Resource User: HC Counter Timers(ID: 0x1000025)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
.
.
.
Resource User: NetFlow Agg Task(ID: 0x1000114)
Getbufs  Retbufs  Holding  RU Name
0         0         0        NetFlow Agg Task
  Resource User: CWAN OIR IPC Ready Process(ID: 0x1000115)
Getbufs  Retbufs  Holding  RU Name
0         0         0        CWAN OIR IPC Rea
  Resource User: PF Clock Process(ID: 0x1000116)
Getbufs  Retbufs  Holding  RU Name
0         0         0        PF Clock Process
  Resource User: CEF IPC Background(ID: 0x1000117)
Getbufs  Retbufs  Holding  RU Name
0         0         0        CEF IPC Backgrou
  Resource User: RTTYS Process(ID: 0x1000118)
Getbufs  Retbufs  Holding  RU Name
0         0         0        RTTYS Process
  Resource User: DHCPD Timer(ID: 0x100011B)
Getbufs  Retbufs  Holding  RU Name
0         0         0        DHCPD Timer
  Resource User: DHCPD Database(ID: 0x100011C)
Getbufs  Retbufs  Holding  RU Name
0         0         0        DHCPD Database
  Resource User: draco-oir-process:slot 2(ID: 0x100011E)
Getbufs  Retbufs  Holding  RU Name
0         0         0        draco-oir-proces
  Resource User: SCP async: Draco-LC4(ID: 0x1000125)
Getbufs  Retbufs  Holding  RU Name
35908    243517    4294759687 SCP async: Draco
  Resource User: IFCOM Msg Hdrlr(ID: 0x1000127)
Getbufs  Retbufs  Holding  RU Name
2         2         0        IFCOM Msg Hdrlr
  Resource User: IFCOM Msg Hdrlr(ID: 0x1000128)
Getbufs  Retbufs  Holding  RU Name
28        28        0        IFCOM Msg Hdrlr
  Resource User: Exec(ID: 0x100012C)

```

```

Getbufs  Retbufs  Holding  RU Name
17552    17552    0        Exec
Resource Owner: test_mem
Resource User Type: test_process
Resource User Type: mem_rut
Resource Owner: test_cpu
Resource User Type: test_process
Resource User Type: cpu_rut

```

The table below describes the significant fields shown in the display.

**Table 1: show resource all Field Descriptions**

Field	Description
Runtime(ms)	The runtime of the process in milliseconds.
Invoked	The number of times a Resource User (RU) has been allowed to run.
uSecs	The amount of runtime per invocation in microseconds.
Allocated Size(b)	The number of bytes of memory that is allocated.
Freed Size(b)	The number of bytes of memory that is freed.
Count	The number of elements that are allocated or freed. For example, if two elements of 50 bytes each are allocated, then the allocated count is 2 and allocated size is 100.
pc	Displays the details of the memory that is held by a process. Each line of the output displays one or more blocks of memory. The pc is the allocator pc of a particular block of memory.
size	The total size of memory allocated to each block. The sum of the size of all blocks is equivalent to the total memory held by the process.
count	The count is the number of blocks of memory.
Getbufs	The number of buffers allocated by the RU.
Retbufs	The number of buffers freed by the RU.
Holding	The number of buffers the RU is holding currently.

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>buffer public</b>	Enters the buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets thresholds for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets thresholds for processor level CPU utilization.
<b>cpu total</b>	Enters the CPU owner configuration mode and sets thresholds for total CPU utilization.
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource database</b>	Displays the database details of ROs.
<b>show resource owner</b>	Displays the RO details.
<b>show resource relationship</b>	Displays the relationship between the RUs and the ROs.

# show resource database

To display the details of a resource owner, use the **showresourcedatabase** command in user EXEC or privileged EXEC mode.

## show resource database

**Syntax Description** This command has no arguments or keywords.

**Command Modes** User EXEC (>) Privileged EXEC (#)

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
	12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

**Examples** The following is sample output from the **showresourcedatabase** command:

```
Router# show resource database
List of all Resource Owners :
Owner: cpu Id:0x1
Owner's list of monitors is empty.
Owner: memory Id:0x2
Owner's list of monitors is empty.
Owner: Buffer Id:0x3
Owner's list of monitors is empty.
Owner: test_mem Id:0x4
Owner's list of monitors is empty.
Owner: test_cpu Id:0x5
Owner's list of monitors is empty.
Owner: test_RO0 Id:0x7
Owner's list of monitors is empty.
Owner: test_RO1 Id:0x8
Owner's list of monitors is empty.
Owner: test_RO2 Id:0x9
Owner's list of monitors is empty.
Owner: test_RO3 Id:0xA
Owner's list of monitors is empty.
Owner: test_RO4 Id:0xB
Owner's list of monitors is empty.
Owner: test_RO5 Id:0xC
Owner's list of monitors is empty.
.
.
.
List of all Resource Usertypes :
RUT: iosprocess Id:0x1
RUT: test_process Id:0x2
RUT: mem_rut Id:0x3
RUT: cpu_rut Id:0x4
RUT: test_RUT0 Id:0x5
```



```

RUT: test_RUT1           Id:0x6
RUT: test_RUT2           Id:0x7
RUT: test_RUT3           Id:0x8
RUT: test_RUT4           Id:0x9
RUT: test_RUT5           Id:0xA
.
.
.
List of all Resource User Groups :
List of all Resource Users :
usertype: iosprocess      Id:0x1
user: Init                Id:0x1000001, priority:0
user: Scheduler           Id:0x1000002, priority:0
user: Dead                Id:0x1000003, priority:0
user: Interrupt           Id:0x1000004, priority:0
user: Memory RO RU       Id:0x1000005, priority:0
user: Chunk Manager      Id:0x1000006, priority:1
user: Load Meter          Id:0x1000007, priority:1
user: Check heaps        Id:0x1000009, priority:4
user: Pool Manager       Id:0x100000A, priority:1
user: Buffer RO RU        Id:0x100000B, priority:0
user: Timers              Id:0x100000C, priority:3
user: Serial Background  Id:0x100000D, priority:3
user: ALARM TRIGGER_SCAN Id:0x100000E, priority:4
user: AAA_SERVER_DEADTIME Id:0x100000F, priority:4
user: AAA_high-capacity counter Id:0x1000010, priority:3
user: Policy Manager     Id:0x1000011, priority:3
user: Crash writer       Id:0x1000012, priority:3
user: RO Notify Timers   Id:0x1000013, priority:3
user: RMI_RM_Notify_Watched_Pol Id:0x1000014, priority:3
user: EnvMon             Id:0x1000015, priority:3
user: OIR_Handler        Id:0x1000016, priority:3
user: IPC_Dynamic_Cache  Id:0x1000017, priority:3
user: IPC_Zone_Manager   Id:0x1000018, priority:3
user: IPC_Periodic_Timer Id:0x1000019, priority:3
user: IPC_Managed_Timer  Id:0x100001A, priority:3
user: IPC_Deferred_Port_Closure Id:0x100001B, priority:3
.
.
.
Resource Monitor: test_ROM0, ID: 0x1B
  Not Watching any Relations.
  Not Watching any Policies.
Resource Monitor: test_ROM1, ID: 0x1C
  Not Watching any Relations.
  Not Watching any Policies.
Resource Monitor: test_ROM2, ID: 0x1D
  Not Watching any Relations.
  Not Watching any Policies.

```

**Related Commands**

Command	Description
<b>buffer public</b>	Enters the buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets thresholds for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets thresholds for processor level CPU utilization.
<b>cpu total</b>	Enters the CPU owner configuration mode and sets thresholds for total CPU utilization.

<b>Command</b>	<b>Description</b>
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>show resource owner</b>	Displays the RO details.
<b>show resource relationship</b>	Displays the relationship between the RUs and the ROs.

## show resource owner

To display the details of a resource owner (RO), use the **showresourceowner** command in user EXEC or privileged EXEC mode.

**show resource owner** {*resource-owner-name*| **all**} **user** {*resource-user-type-name*| **all**} [**brief** [**triggers**]| **detailed** [**triggers**]| **triggers**]

### Syntax Description

<i>resource-owner-name</i>	Name of the specified RO whose details are displayed.
<b>all</b>	Displays details of all the ROs.
<b>user</b>	Displays details of the specified resource user (RU) type.
<i>resource-user-type-name</i>	Single resource user type.
<b>all</b>	Displays details of all the resource user types.
<b>brief</b>	(Optional) Displays brief details.
<b>detailed</b>	(Optional) Displays complete details.
<b>triggers</b>	(Optional) Displays the triggers.

### Command Modes

User EXEC (>) Privileged EXEC (#)

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

### Examples

The following is sample output from the **showresourceowner** command:

```
Router# show resource owner all user all
Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
  RUID Runtime(ms)  Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777217         0         0         0  0.00%  0.00%  0.00% Init
Resource User: Scheduler(ID: 0x1000002)
```

```

RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777218         0            0        0  0.00% 0.00% 0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777219         0            0        0  0.00% 0.00% 0.00% Dead
Resource User: Interrupt(ID: 0x1000004)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777220         0            0        0  0.00% 0.00% 0.00% Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777221         0            0        0  0.00% 0.00% 0.00% Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777222         4            3       1333  0.00% 0.00% 0.00% Chunk Manager
Resource User: Load Meter(ID: 0x1000007)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777223         4           292        13  0.00% 0.00% 0.00% Load Meter
Resource User: Check heaps(ID: 0x1000009)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777225       376         192       1958  0.00% 0.02% 0.00% Check heaps
Resource User: Pool Manager(ID: 0x100000A)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777226         0            1        0  0.00% 0.00% 0.00% Pool Manager
Resource User: Buffer RO RU(ID: 0x100000B)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777227         0            0        0  0.00% 0.00% 0.00% Buffer RO RU
Resource User: Timers(ID: 0x100000C)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777228         0            2        0  0.00% 0.00% 0.00% Timers
Resource User: Serial Background(ID: 0x100000D)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
.
.
.
Resource Owner: memory
  Resource User Type: iosprocess
  Resource User: Init(ID: 0x1000001)
Chunk Elements :
Allocated Size(b): 25967632 Count: 46612 Freed Size(b): 21487684 Count: 26053
Processor memory
Total Memory held : 15250376 bytes
pc = 0x6072D840, size = 4040536, count = 6
pc = 0x6034E040, size = 1937508, count = 2
pc = 0x6070DAF0, size = 560096, count = 1
pc = 0x606D7530, size = 556220, count = 685
pc = 0x613AFA74, size = 350972, count = 25
pc = 0x60ECA4F0, size = 280004, count = 1
pc = 0x606DEC1C, size = 270600, count = 100
pc = 0x616EF268, size = 262148, count = 1
pc = 0x6085C318, size = 196620, count = 3
pc = 0x61479630, size = 144004, count = 1
pc = 0x613E1DB0, size = 131768, count = 4
.
.
.
I/O memory
Total Memory held : 4059856 bytes
pc = 0x606DEC30, size = 3408704, count = 52
pc = 0x606DEB94, size = 442464, count = 6
pc = 0x606D76A4, size = 179872, count = 146
pc = 0x600ED530, size = 16448, count = 4
pc = 0x600ED498, size = 8256, count = 4
pc = 0x6080D3F0, size = 4112, count = 1
  Resource User: Scheduler(ID: 0x1000002)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 12172 bytes
pc = 0x607B44F0, size = 12004, count = 1
pc = 0x607643B8, size = 168, count = 4
.
.
.

```

```

Resource User: Critical Bkgnd(ID: 0x1000026)
Chunk Elements :
Allocated Size(b): 44 Count: 1 Freed Size(b): 0 Count: 0
Processor memory
Total Memory held : 6780 bytes
pc = 0x607B44F0, size =      6004, count =    1
pc = 0x6079CB28, size =       636, count =    1
pc = 0x6079EE84, size =       140, count =    1
.
.
.
Resource Owner: Buffer
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
Getbufs  Retbufs  Holding  RU Name
319      51         268     Init
Resource User: Scheduler(ID: 0x1000002)
Getbufs  Retbufs  Holding  RU Name
0         0         0       Scheduler
Resource User: Dead(ID: 0x1000003)
Getbufs  Retbufs  Holding  RU Name
0         0         0       Dead
Resource User: Interrupt(ID: 0x1000004)
Getbufs  Retbufs  Holding  RU Name
1356     1356      0       Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
Getbufs  Retbufs  Holding  RU Name
0         0         0       Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)
Getbufs  Retbufs  Holding  RU Name
0         0         0       Chunk Manager
.
.
.
Resource Owner: test_mem
Resource User Type: test_process
Resource User Type: mem_rut
Resource Owner: test_cpu
Resource User Type: test_process
Resource User Type: cpu_rut
Resource User: test_RU0(ID: 0x4000001)
>>>RU: Blank
Resource User: test_RU1(ID: 0x4000002)
>>>RU: Blank
Resource User: test_RU2(ID: 0x4000003)
>>>RU: Blank
Resource User: test_RU3(ID: 0x4000004)
>>>RU: Blank
.
.
.
Resource User Type: test_RUT143
Resource User Type: test_RUT144
Resource User Type: test_RUT145
Resource User Type: test_RUT146
Resource User Type: test_RUT147

```

The following is sample output from the **showresourceowneralluserallbrief** command:

```

Router# show resource owner all user all brief

Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
RUID Runtime(ms)  Invoked    uSecs    5Sec    1Min    5Min Res Usr
16777217          0           0         0  0.00%  0.00%  0.00% Init
Resource User: Scheduler(ID: 0x1000002)
RUID Runtime(ms)  Invoked    uSecs    5Sec    1Min    5Min Res Usr
16777218          0           0         0  0.00%  0.00%  0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
RUID Runtime(ms)  Invoked    uSecs    5Sec    1Min    5Min Res Usr
16777219          0           0         0  0.00%  0.00%  0.00% Dead
Resource User: Interrupt(ID: 0x1000004)

```

## show resource owner

```

RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777220          0             0        0  0.00%  0.00%  0.00% Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777221          0             0        0  0.00%  0.00%  0.00% Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777222          4             3       1333  0.00%  0.00%  0.00% Chunk Manager
Resource User: Load Meter(ID: 0x1000007)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777223          4            322        12  0.00%  0.01%  0.00% Load Meter
Resource User: Check heaps(ID: 0x1000009)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res  Usr
16777225         424           214       1981  0.00%  0.04%  0.00% Check heaps
.
.
.
Resource Owner: memory
  Resource User Type: iosprocess
  Resource User: Init(ID: 0x1000001)
Processor memory
Allocated   Freed   Holding   Blocks
21916780   6666404 15250376    8688
I/O memory
Allocated   Freed   Holding   Blocks
4059856     0      4059856    213
Resource User: Scheduler(ID: 0x1000002)
Processor memory
Allocated   Freed   Holding   Blocks
12172       0      12172      5
.
.
.
Resource Owner: test_mem
  Resource User Type: test_process
  Resource User Type: mem_rut
Resource Owner: test_cpu
  Resource User Type: test_process
  Resource User Type: cpu_rut
Resource User: test_RU0(ID: 0x4000001)
>>>RU: Blank
Resource User: test_RU1(ID: 0x4000002)
>>>RU: Blank
Resource User: test_RU2(ID: 0x4000003)
>>>RU: Blank
Resource User: test_RU3(ID: 0x4000004)
>>>RU: Blank
Resource User: test_RU4(ID: 0x4000005)
>>>RU: Blank
.
.
.
Resource Owner: test_RO0
Resource User Type: test_RUT0
Resource User Type: test_RUT1
Resource User Type: test_RUT2
Resource User Type: test_RUT3
Resource User Type: test_RUT4
Resource User Type: test_RUT5
Resource User Type: test_RUT6
Resource User Type: test_RUT7
Resource User Type: test_RUT8
Resource User Type: test_RUT9
Resource User Type: test_RUT10
Resource User Type: test_RUT11
Resource User Type: test_RUT12
Resource User Type: test_RUT13
Resource User Type: test_RUT14
Resource User Type: test_RUT15
Resource User Type: test_RUT16

```

The following is sample output from the **showresourceowneralluserallbrieftriggers** command:

```
Router# show resource owner all user all brief triggers

Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777217          0          0        0  0.00%  0.00%  0.00% Init
Resource User: Scheduler(ID: 0x1000002)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777218          0          0        0  0.00%  0.00%  0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777219          0          0        0  0.00%  0.00%  0.00% Dead
Resource User: Interrupt(ID: 0x1000004)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777220          0          0        0  0.00%  0.00%  0.00% Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777221          0          0        0  0.00%  0.00%  0.00% Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777222          4          3       1333  0.00%  0.00%  0.00% Chunk Manager
.
.
Resource Owner: test_mem
Resource User Type: test_process
Resource User Type: mem_rut
Resource Owner: test_cpu
Resource User Type: test_process
Resource User Type: cpu_rut
Resource User: test_RU0(ID: 0x4000001)
>>>RU: Blank
Resource User: test_RU1(ID: 0x4000002)
>>>RU: Blank
Resource User: test_RU2(ID: 0x4000003)
>>>RU: Blank
Resource User: test_RU3(ID: 0x4000004)
>>>RU: Blank
Resource User: test_RU4(ID: 0x4000005)
>>>RU: Blank
Resource User: test_RU5(ID: 0x4000006)
>>>RU: Blank
```

The following is sample output from the **showresourceowneralluseralldetailed** command:

```
Router# show resource owner all user all detailed

Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777217          0          0        0  0.00%  0.00%  0.00% Init
Resource User: Scheduler(ID: 0x1000002)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777218          0          0        0  0.00%  0.00%  0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777219          0          0        0  0.00%  0.00%  0.00% Dead
Resource User: Interrupt(ID: 0x1000004)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777220          0          0        0  0.00%  0.00%  0.00% Interrupt
Resource User: Memory RO RU(ID: 0x1000005)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777221          0          0        0  0.00%  0.00%  0.00% Memory RO RU
Resource User: Chunk Manager(ID: 0x1000006)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777222          4          3       1333  0.00%  0.00%  0.00% Chunk Manager
Resource User: Load Meter(ID: 0x1000007)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
```

## show resource owner

```

16777223      4      353      11  0.00%  0.01%  0.00% Load Meter
Resource User: Check heaps(ID: 0x1000009)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777225      456      232      1965  0.00%  0.01%  0.00% Check heaps
Resource User: Pool Manager(ID: 0x100000A)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777226      0         1         0  0.00%  0.00%  0.00% Pool Manager
Resource User: Buffer RO RU(ID: 0x100000B)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777227      0         0         0  0.00%  0.00%  0.00% Buffer RO RU
Resource User: Timers(ID: 0x100000C)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777228      0         2         0  0.00%  0.00%  0.00% Timers
.
.
Resource Owner: memory
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
Chunk Elements :
Allocated Size(b): 25967632 Count: 46612 Freed Size(b): 21487684 Count: 26053
Processor memory
Address      Bytes      Prev      Next Ref   Alloc PC  What
63700E18  0000020052  636FD0D4  63705C6C  001  6412D2C0  Managed Chunk Queue Elements
63705C6C  0000012852  63700E18  63708EA0  001  6412D2C0  *Init*
63708EA0  0000010052  63705C6C  6370B5E4  001  6412D2C0  List Elements
6370B5E4  0000005052  63708EA0  6370C9A0  001  6412D2C0  List Headers
6370C9A0  0000009052  6370B5E4  6370ECFC  001  6412D2C0  Interrupt Stack
6370ECFC  0000000096  6370C9A0  6370ED5C  001  6412D2C0  *Init*
6370ED5C  0000000084  6370ECFC  6370EDB0  001  6412D2C0  *Init*
6370EDB0  0000000132  6370ED5C  6370EE34  001  6412D2C0  *Init*
6370EE34  0000000092  6370EDB0  6370EE90  001  6412D2C0  *Init*
6370EE90  0000000436  6370EE34  6370F044  001  6412D2C0  *Init*
6370F044  0000000076  6370EE90  6370F090  001  6412D2C0  *Init*
6370F090  0000000132  6370F044  6370F114  001  6412D2C0  *Init*
6370F114  0000000092  6370F090  6370F170  001  6412D2C0  *Init*
.
.
Resource User: Scheduler(ID: 0x1000002)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Address      Bytes      Prev      Next Ref   Alloc PC  What
63799F04  0000012052  63799EB8  6379CE18  001  6412D2C0  Scheduler Stack
643E9A38  0000000076  643D9A04  643E9A84  001  6412D2C0  *Sched*
644C47F0  0000000076  644C4790  644C483C  001  6412D2C0  *Sched*
645FF744  0000000096  645FF6E8  645FF7A4  001  6412D2C0  *Sched*
64904354  0000000112  649040D0  649043C4  001  6412D2C0  *Sched*
Resource User: Dead(ID: 0x1000003)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Address      Bytes      Prev      Next Ref   Alloc PC  What
63F9D328  0000000096  63F984D4  63F9D388  001  6412D2C0  AAA MI SG NAME
Resource User: Interrupt(ID: 0x1000004)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0

```

The following is sample output from the **showresourceowneralluseralldetailedtriggers** command:

```

Router# show resource owner all user all detailed triggers
Resource Owner: cpu
Resource User Type: iosprocess
Resource User: Init(ID: 0x1000001)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777217      0         0         0  0.00%  0.00%  0.00% Init
Resource User: Scheduler(ID: 0x1000002)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777218      0         0         0  0.00%  0.00%  0.00% Scheduler
Resource User: Dead(ID: 0x1000003)
RUID Runtime(ms)  Invoked      uSecs   5Sec   1Min   5Min Res Usr
16777219      0         0         0  0.00%  0.00%  0.00% Dead

```



```

Resource User: Interrupt (ID: 0x1000004)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777220 0 0 0 0.00% 0.00% 0.00% Interrupt
Resource User: Memory RO RU (ID: 0x1000005)
RUID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min Res Usr
16777221 0 0 0 0.00% 0.00% 0.00% Memory RO RU
Resource User: Chunk Manager (ID: 0x1000006)
.
.
.
Resource User: Scheduler (ID: 0x1000002)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
Processor memory
Address Bytes Prev Next Ref Alloc PC What
63799F04 0000012052 63799EB8 6379CE18 001 6412D2C0 Scheduler Stack
643E9A38 0000000076 643D9A04 643E9A84 001 6412D2C0 *Sched*
644C47F0 0000000076 644C4790 644C483C 001 6412D2C0 *Sched*
645FF744 0000000096 645FF6E8 645FF7A4 001 6412D2C0 *Sched*
64904354 0000000112 649040D0 649043C4 001 6412D2C0 *Sched*
Resource User: Dead (ID: 0x1000003)
Chunk Elements :
Allocated Size(b): 0 Count: 0 Freed Size(b): 0 Count: 0
.
.
.
Resource User Type: test_RUT142
Resource User Type: test_RUT143
Resource User Type: test_RUT144
Resource User Type: test_RUT145
Resource User Type: test_RUT146
Resource User Type: test_RUT147
Resource User Type: test_RUT148
Resource User Type: test_RUT149

```

The table below describes the significant fields shown in the display.

**Table 2: show resource owner Field Descriptions**

Field	Description
Runtime(ms)	The runtime of the process in milliseconds.
Invoked	The number of times an RU has been allowed to run.
uSecs	The amount of runtime per invocation in microseconds.
Allocated Size(b)	The number of bytes of memory that are allocated.
Freed Size(b)	The number of bytes of memory that are freed.
Count	The number of elements that are allocated or freed. For example, if two elements of 50 bytes each are allocated, the allocated count is 2 and allocated size is 100.

Field	Description
pc	Displays the details of the memory that is held by a process. Each line of the output displays one or more blocks of memory.  The pc is the allocator pc of a particular block of memory.
size	The total size of memory allocated to each block. The sum of the size of all blocks is equivalent to the total memory held by the process.
count	The count is the number of blocks of memory.
Getbufs	The number of buffers allocated by the RU.
Retbufs	The number of buffers freed by the RU.
Holding	The number of buffers the RU is holding currently.

**Related Commands**

Command	Description
<b>buffer public</b>	Enters buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters CPU owner configuration mode and sets thresholds for interrupt level CPU utilization.
<b>cpu process</b>	Enters CPU owner configuration mode and sets thresholds for processor level CPU utilization.
<b>cpu total</b>	Enters CPU owner configuration mode and sets thresholds for total CPU utilization.
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>memory io</b>	Enters memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters memory owner configuration mode and sets threshold values for processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.

<b>Command</b>	<b>Description</b>
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>show resource database</b>	Displays the entire database of all resource entry relationships.
<b>show resource relationship</b>	Displays the relationship between the RUs and the ROs.

# show resource relationship

To display the details of relationships between different resource owners, use the **showresourcerelationship** command in user EXEC or privileged EXEC mode.

**show resource relationship user** *resource-user-type*

## Syntax Description

<b>user</b>	Identifies a resource user (RU).
<i>resource-user-type</i>	Type of RU.

## Command Modes

User EXEC (>) Privileged EXEC (#)

## Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

## Examples

The following is sample output from the **showresourcerelationship** command:

```
Router# show resource relationship
Resource User Type: iosprocess (ID: 0x1)
-> Resource Owner: cpu (ID: 0x1)
-> Resource Owner: memory (ID: 0x2)
-> Resource Owner: Buffer (ID: 0x3)
-> Resource User: Init (ID: 0x1000001)
-> Resource User: Scheduler (ID: 0x1000002)
-> Resource User: Dead (ID: 0x1000003)
-> Resource User: Interrupt (ID: 0x1000004)
-> Resource User: Memory RO RU (ID: 0x1000005)
-> Resource User: Chunk Manager (ID: 0x1000006)
-> Resource User: Load Meter (ID: 0x1000007)
-> Resource User: Check heaps (ID: 0x1000009)
-> Resource User: Pool Manager (ID: 0x100000A)
-> Resource User: Buffer RO RU (ID: 0x100000B)
-> Resource User: Timers (ID: 0x100000C)
-> Resource User: Serial Background (ID: 0x100000D)
-> Resource User: ALARM_TRIGGER_SCAN (ID: 0x100000E)
-> Resource User: AAA_SERVER_DEADTIME (ID: 0x100000F)
-> Resource User: AAA_high-capacity_counters (ID: 0x1000010)
-> Resource User: Policy Manager (ID: 0x1000011)
-> Resource User: Crash writer (ID: 0x1000012)
-> Resource User: RO Notify Timers (ID: 0x1000013)
-> Resource User: RMI_RM_Notify_Watched_Policy (ID: 0x1000014)
-> Resource User: EnvMon (ID: 0x1000015)
-> Resource User: OIR_Handler (ID: 0x1000016)
-> Resource User: IPC_Dynamic_Cache (ID: 0x1000017)
```

```

-> Resource User: IPC Zone Manager (ID: 0x1000018)
-> Resource User: IPC Periodic Timer (ID: 0x1000019)
-> Resource User: IPC Managed Timer (ID: 0x100001A)
-> Resource User: IPC Deferred Port Closure (ID: 0x100001B)
-> Resource User: IPC Seat Manager (ID: 0x100001C)
-> Resource User: IPC Session Service (ID: 0x100001D)
-> Resource User: Compute SRP rates (ID: 0x100001E)
-> Resource User: ARP Input (ID: 0x100001F)
-> Resource User: DDR Timers (ID: 0x1000020)
-> Resource User: Dialer event (ID: 0x1000021)
-> Resource User: Entity MIB API (ID: 0x1000022)
-> Resource User: SERIAL A'detect (ID: 0x1000023)
-> Resource User: GraphIt (ID: 0x1000024)
-> Resource User: HC Counter Timers (ID: 0x1000025)
.
.
Resource User Type: test_RUT141 (ID: 0x92)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT142 (ID: 0x93)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT143 (ID: 0x94)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT144 (ID: 0x95)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT145 (ID: 0x96)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT146 (ID: 0x97)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT147 (ID: 0x98)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT148 (ID: 0x99)
-> Resource Owner: test_ROO (ID: 0x7)
Resource User Type: test_RUT149 (ID: 0x9A)
-> Resource Owner: test_ROO (ID: 0x7)

```

## Related Commands

Command	Description
<b>buffer public</b>	Enters buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters CPU owner configuration mode and sets thresholds for interrupt level CPU utilization.
<b>cpu process</b>	Enters CPU owner configuration mode and sets thresholds for processor level CPU utilization.
<b>cpu total</b>	Enters CPU owner configuration mode and sets thresholds for total CPU utilization.
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>memory io</b>	Enters memory owner configuration mode and sets threshold values for the I/O memory.

<b>Command</b>	<b>Description</b>
<b>memory processor</b>	Enters memory owner configuration mode and sets threshold values for the processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>show resource database</b>	Displays the entire database of all resource entry relationships.
<b>show resource owner</b>	Displays the RO details.

## show resource user

To display the policy details or Resource User (RU) template details of a resource user, use the **showresourceuser** command in user EXEC or privileged EXEC mode.

**show resource user** {**all**| *resource-user-type*} [**brief**| **detailed**]

### Syntax Description

<b>all</b>	Displays the policy details of all the RUs.
<i>resource-user-type</i>	Type of RU. For example, iosprocess.
<b>brief</b>	(Optional) Displays a short description of the policy details.
<b>detailed</b>	(Optional) Displays all details of a policy.

### Command Modes

User EXEC (>) Privileged EXEC (#)

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

### Examples

The following is sample output from the **showresourceuser** command:

```
Router# show resource user all
Resource User Type: iosprocess
Resource Grp: Init
Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
27197780   8950144 18247636   6552
I/O memory
Allocated   Freed   Holding   Blocks
7296000     9504    7286496   196
Resource Owner: cpu
  RUID Runtime(ms)   Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777224     14408     116      124206 100.40% 8.20% 1.70% Init
Resource Owner: Buffer
Getbufs Retbufs Holding RU Name
332     60     272    Init
Resource User: Init
Resource User: Scheduler
Resource Owner: memory
Processor memory
```

## show resource user

```

Allocated   Freed   Holding   Blocks
 77544      0     77544      2
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777218    0         0         0     0.00% 0.00% 0.00% Scheduler
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
 0       0       0         Scheduler
Resource User: Dead
Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
1780540    260   1780280    125
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777219    0         0         0     0.00% 0.00% 0.00% Dead
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
 9       8       1         Dead
Resource User: Interrupt
Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
 0       0       0         0
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777220    0         0         0     0.00% 0.00% 0.00% Interrupt
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
14128    14128    0         Interrupt
Resource User: Memory RO RU
Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
132560    1480   131080      2
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777221    0         0         0     0.00% 0.00% 0.00% Memory RO RU
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
64       64       0         Memory RO RU
.
.
.
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777401   7124    4250     1676   0.00% 0.03% 0.01% Exec
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
38       38       0         Exec
Resource User: BGP Router
Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
43380    26556   16824      8
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777404   12    19705         0     0.00% 0.00% 0.00% BGP Router
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
0       0       0         BGP Router
Resource User: BGP I/O
Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
6892     6892     0         0
Resource Owner: cpu
  RUID Runtime(ms)   Invoked     uSecs   5Sec   1Min   5Min Res Usr
16777405    0         1         0     0.00% 0.00% 0.00% BGP I/O
Resource Owner: Buffer
Getbufs Retbufs Holding   RU Name
0       0       0         BGP I/O
Resource User: BGP Scanner

```



```

Resource Owner: memory
Processor memory
Allocated   Freed   Holding   Blocks
  9828     9828         0         0
Resource Owner: cpu
RUID Runtime(ms)   Invoked    uSecs   5Sec   1Min   5Min Res Usr
16777406          660        659     1001  0.00% 0.00% 0.00% BGP Scanner
Resource Owner: Buffer
Getbufs   Retbufs   Holding   RU Name
  0         0         0         BGP Scanner
Resource User Type: test_process
Resource User Type: mem_rut
Resource User Type: cpu_rut

```

The table below describes the significant fields shown in the display.

**Table 3: show resource user Field Descriptions**

Field	Description
Allocated	The number of bytes of memory that is allocated.
Freed	The number of bytes of memory that is freed.
Count	The number of elements that are allocated or freed. For example, if two elements of 50 bytes each are allocated, the allocated count is 2 and allocated size is 100.
Runtime(ms)	The runtime of the process in milliseconds.
Invoked	The number of times an RU has been allowed to run.
uSecs	The amount of runtime per invocation in microseconds.
Getbufs	The number of buffers allocated by the RU.
Retbufs	The number of buffers freed by the RU.
Holding	The number of buffers the RU is holding currently.

#### Related Commands

Command	Description
<b>buffer public</b>	Enters buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters CPU owner configuration mode and sets thresholds for interrupt-level CPU utilization.
<b>cpu process</b>	Enters CPU owner configuration mode and sets thresholds for processor-level CPU utilization.

<b>Command</b>	<b>Description</b>
<b>cpu total</b>	Enters CPU owner configuration mode and sets thresholds for total CPU utilization.
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>memory io</b>	Enters memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters memory owner configuration mode and sets threshold values for processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.
<b>show resource database</b>	Displays entire database of all resource entry relationships.
<b>show resource owner</b>	Displays the RO details.

## slot (ERM policy)

To configure line cards, use the **slot** command in ERM policy configuration mode.

**slot** *slot-number*

### Syntax Description

<i>slot-number</i>	Integer that identifies a slot number or the start of a range of slots.
--------------------	---

### Command Default

Disabled.

### Command Modes

ERM policy configuration

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

### Usage Guidelines

You can configure line cards using the **slotslot-number** command in ERM policy configuration mode. This command is available only in distributed platforms such as the Route Switch Processor (RSP). You must use a Cisco 7500 router with a line card for executing this command.

### Examples

The following example shows how to configure the line card 0:

```
Router(config-erm-policy)# slot 0
```

### Related Commands

Command	Description
<b>buffer public</b>	Enters the buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets thresholds for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets thresholds for processor level CPU utilization.

<b>Command</b>	<b>Description</b>
<b>cpu total</b>	Enters the CPU owner configuration mode and sets thresholds for total CPU utilization.
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.

## system (ERM policy)

To configure system level resource owners (ROs), use the **system** command in Embedded Resource Manager (ERM) policy configuration mode.

**system**

**Syntax Description** This command has no arguments or keywords.

**Command Default** No system level ROs are configured.

**Command Modes** ERM policy configuration

Command History	Release	Modification
	12.3(14)T	This command was introduced.
	12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

**Examples** The following example shows how to configure system level ROs:

```
Router(config-erm-policy)# system
```

### Related Commands

Command	Description
<b>buffer public</b>	Enters the buffer owner configuration mode and sets thresholds for buffer usage.
<b>cpu interrupt</b>	Enters the CPU owner configuration mode and sets thresholds for interrupt level CPU utilization.
<b>cpu process</b>	Enters the CPU owner configuration mode and sets thresholds for processor level CPU utilization.
<b>cpu total</b>	Enters the CPU owner configuration mode and sets thresholds for total CPU utilization.
<b>critical rising</b>	Sets the critical level threshold values for the buffer, CPU, and memory ROs.
<b>major rising</b>	Sets the major level threshold values for the buffer, CPU, and memory ROs.

<b>Command</b>	<b>Description</b>
<b>memory io</b>	Enters the memory owner configuration mode and sets threshold values for I/O memory.
<b>memory processor</b>	Enters the memory owner configuration mode and sets threshold values for processor memory.
<b>minor rising</b>	Sets the minor level threshold values for the buffer, CPU, and memory ROs.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays all the resource details.

## user (ERM)

To apply a global policy, create a resource group, or add resource users (RUs) to a resource group, use the **user** command in Embedded Resource Manager (ERM) configuration mode. To disable applying the policy, use the **no** form of this command.

**user** {*resource-instance-name resource-user-type resource-policy-name*| **global** *global-policy-name*| **group** *resource-group-name type resource-user-type*}

**no user** {*resource-instance-name resource-user-type resource-policy-name*| **global** *global-policy-name*| **group** *resource-group-name type resource-user-type*}

### Syntax Description

<i>resource-instance-name</i>	Name of the RU to which you are applying a policy.
<i>resource-user-type</i>	Name of the RU type.
<i>resource-policy-name</i>	Name of the policy you are applying to the specified RU.
<b>global</b>	Applies a global policy.
<i>global-policy-name</i>	Name of the global policy you are applying.
<b>group</b>	Specifies a resource group to which the policy is being applied.
<i>resource-group-name</i>	Name of the resource group to which the policy is being applied.
<b>type</b>	Specifies the type of the RU to which the policy is being applied.
<i>resource-user-type</i>	Name of the RU type to which the policy is being applied.

**Command Default** No policy is configured.

**Command Modes** ERM configuration (config-erm)

### Command History

Release	Modification
12.3(14)T	This command was introduced.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.

Release	Modification
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

### Usage Guidelines

This command helps you to apply the various policies (system global, per-user local, and user global) to resource owners (ROs), RUs, or a group of RUs.

Use the **userresource-instance-nameresource-user-typeresource-policy-name** command to apply a specified policy to a RU. This policy is also known as a per-user local policy or per-user template.

Use the **userglobalglobal-policy-name** command to apply a global thresholding policy to all the users.

Use the **usergroupresource-group-nametyperesource-user-type** command to create a resource group and to enter resource group configuration mode. After you create the resource group, you can add RUs using the **instanceinstance-name** command and apply the same thresholding policy to all the RUs against the resource group using the **policypolicy-name** command in resource group configuration mode.

For example, you created a resource group named lowPrioUsers with a type of iosprocess. You have low-priority RUs or tasks such as HTTP and Simple Network Management Protocol (SNMP), and you want to set a threshold for all the low-priority RUs as a group. You must add the RUs to the resource group using the **instanceinstance-name** command and then apply a resource policy. If the resource policy you apply sets a minor rising threshold value of 10 percent for the resource group, when the accumulated usage of both HTTP and SNMP RUs crosses the 10 percent mark, a notification is sent to the RUs in the resource group lowPrioUsers. That is, if HTTP usage is 4 percent and SNMP usage is 7 percent, a notification is sent to lowPrioUsers.

### Examples

The following example shows how to apply a per-user thresholding policy for the resource instance EXEC, resource user type iosprocess, and resource policy name policy-test1:

```
Router(config-erm) # user EXEC iosprocess policy-test1
```

The following example shows how to apply a global thresholding policy with policy name global-global-test1:

```
Router(config-erm) # user global global-global-test1
```

The following example shows how to create a resource group with the resource group name lowPrioUsers and RU type as iosprocess, and how to add the RU HTTP to the resource group and apply a thresholding policy group-policy1:

```
Router(config-erm) # user group lowPrioUsers type iosprocess
Router(config-res-group) # instance http
Router(config-res-group) # policy group-policy1
```

### Related Commands

Command	Description
<b>instance (resource group)</b>	Adds RUs to a resource group.
<b>policy (ERM)</b>	Configures an ERM resource policy.
<b>policy (resource group)</b>	Applies the same policy to all the RUs in a resource group.



<b>Command</b>	<b>Description</b>
<b>resource policy</b>	Enters ERM configuration mode.
<b>show resource all</b>	Displays resource details for all RUs.

