



# Configuring Event Notifications

---

Event notification is a feature that enables you to define conditions that cause the location server to send notifications to the listeners that you have specified in Cisco WCS. This chapter describes how to define events and event groups, and how to configure event notification parameters. It also describes how to view event notification summaries. This chapter contains the following sections:

- [“Working with Event Groups” on page 6-1](#)
- [“Working with Event Definitions” on page 6-2](#)
- [“Viewing Event Notification Summary” on page 6-6](#)
- [“Configuring Notification Parameters” on page 6-7](#)
- [“Notification Message Formats” on page 6-9](#)

## Working with Event Groups

This section describes how to add and delete event groups.

### Adding Event Groups

To manage events more efficiently, you can use Cisco WCS to create event groups. Event groups help you organize your event definitions.

To add an event group, follow these steps:

- 
- Step 1** In Cisco WCS, choose **Location > Notifications**.
  - Step 2** From the drop-down menu (right-hand side), choose **Add Event Group**, and click **Go**.
  - Step 3** Enter the name of the group in the Group Name field.
  - Step 4** Click **Save**.

The new event group appears in the Event Settings page.

---

## Deleting Event Groups

To manage events more efficiently, you can use Cisco WCS to create event groups. Event groups help you organize your event definitions.

To add an event group, follow these steps:

- 
- Step 1** In Cisco WCS, choose **Location > Notifications**.
  - Step 2** Select the event groups to delete by checking their corresponding check boxes.
  - Step 3** From the drop-down menu (right-hand side), choose **Delete Event Group(s)**, and click **Go**.
  - Step 4** Click **Save**.
- 

## Working with Event Definitions

An event definition contains information about the condition that caused the event, the assets to which the event applies, and the event notification destinations. This section describes how to add, delete, and test event definitions.

### Adding an Event Definition

Cisco WCS enables you to add definitions on a per-group basis. Any new event definition must belong to a particular group.

To add an event definition, follow these steps:

- 
- Step 1** In Cisco WCS, choose **Location > Notifications**.
  - Step 2** Click **Settings** (left).
  - Step 3** Click the name of the group to which you want to add the event.
  - Step 4** From the drop-down menu (right-hand side), choose **Add Event Definition**, and click **Go**.
  - Step 5** Enter the name of the event definition in the Event Definition Name field.



---

**Note** The event definition name must be unique within the event group.

---

- Step 6** Click **Save**.
- Step 7** In the Conditions tab, add one or more conditions. For each condition you add, specify the rules for triggering events notifications.

For example, to keep track of heart monitors in a hospital, you can add three rules to generate an event notification if the heart monitor is missing for two hours, if the heart monitor moves out of the second floor, or if the heart monitor enters a specific coverage area within a floor.

To add a condition, follow these steps:

- a. Click **Add** to add a condition that triggers this event.
- b. In the Add/Edit Condition dialog box, follow these steps:
  1. Choose a condition type from the Condition Type drop-down menu.
  2. In the Trigger If field, follow these steps:

If you chose **Missing** from the Condition Type drop-down menu, enter the number of minutes after which a missing asset event is generated. For example, if you enter 10 in this field, the location server generates a missing asset event if the location server has not located the asset for more than 10 minutes. Proceed to Step c.

If you chose **In/Out** from the Condition Type drop-down menu, select **Inside of** or **Outside of**, then click **Select Area** to select the area to monitor for assets going into it or out of it. In the Select dialog box, choose the area to monitor, then click **Select**. The area to monitor could be an entire campus, building within a campus, a floor in a building, or a coverage area (you can define a coverage area using the map editor). For example, to monitor part of a floor in a building, choose a campus from the Campus drop-down menu, choose a building from the Building drop-down menu, and choose the area to monitor from the Floor Area drop-down menu. Then click **Select**. Proceed to Step c.

If you chose **Distance** from the Condition Type drop-down menu, enter the distance in feet that will trigger an event notification if the monitored asset moves beyond the specified distance from a designated marker, then click **Select Marker**. In the Select dialog box, select the campus, building, floor, and marker from the corresponding drop-down menus and click **Select**. For example, if you add a marker to a floor plan and set the distance in the Trigger If field to 60 feet, an event notification will be generated if the monitored asset moves 73 feet away from the marker. Proceed to Step c.




---

**Note** You can create markers and coverage areas using the Map Editor. When you create marker names, make sure they are unique across the entire system.

---

If you chose **Location Change** from the Condition Type drop-down menu, proceed to Step c.

If you chose **Battery Level** from the Condition Type drop-down menu, check the box next to the appropriate battery level (low, medium, normal) that will trigger an event. Proceed to Step c.

- c. Choose the type of asset (Any, Clients, Tags, Rogue APs, or Rogue Clients) that will trigger event notifications if the triggering condition is met from the Apply To drop-down menu.
- d. Choose the asset matching criteria for generating event notifications from the Match By drop-down menu, choose the operator (**Equals** or **Like**) from the drop-down menu, and enter the information for matching the elements.

Following are examples of asset matching criteria that you can specify:

- If you choose **MAC Address** from the Match By drop-down menu, choose **Equals** from the Operator drop-down menu, and enter **12:12:12:12:12:12**, the event condition applies to the element whose MAC address is 12:12:12:12:12:12 (exact match).
  - If you choose **MAC Address** from the Match By drop-down menu, choose **Like** from the Operator drop-down menu, and enter **12:12**, the event condition applies to elements whose MAC address starts with 12:12.
- e. Click **Add** to add the condition you have just defined.

To delete a condition, select the condition and click **Delete**.

- Step 8** Under the General tab, follow these steps:
- Enable event generation (disabled by default) by checking the **Enabled** check box for the Admin Status field.
  - Set the event priority by choosing a number from the Priority drop-down menu. Zero is highest.




---

**Note** An event definition with higher priority is serviced before event definitions with lower priority.

---

- Set the date and time when you want to activate event notification using the hour, minute and AM/PM menus next to the Apply From heading.




---

**Note** If you want to continuously report events, you would select the **All the Time** option. In this case, there is no need to set start and end ranges for event notification. These options are not displayed.

---

- Set the date and time for ending notification for this event using the hour, minute and AM/PM menus next to the Apply Until heading.
- Click **Save**.

- Step 9** In the Destination and Transport tab, follow these steps to add one or more destinations to receive event notifications and configure the transport settings:

- To add a new destination, click **Add New**.
- Enter the IP address of the system that will receive event notifications, and click **OK**.  
The recipient system must have an event listener running to process notifications. By default, when you create an event definition, Cisco WCS adds its IP address as the a destination.
- To select a destination to send event notifications to, highlight one or more IP addresses in the box on the right, and click **Select** to add the IP addresses to the box on the left.
- In the Message Format field, select **XML** or **Plain Text** to specify the message format.  
If you select WCS as the destination of event notifications, you must select the XML format.
- Choose one of the following transport types from the Transport Type drop-down menu:
  - **SOAP**—Specifies Simple Object Access Protocol, a simple XML protocol, as the transport type for sending event notifications. Use SOAP to send notifications over HTTP/HTTPS and to be processed by web services on the destination.  
If you choose **SOAP**, specify whether to send notifications over HTTPS by checking its corresponding check box. If you don't, HTTP is used. Also, enter the destination port number in the Port Number field.
  - **Mail**—Use this option to send notifications via email.  
If you choose **Mail**, you need to choose the protocol for sending the mail from the Mail Type drop-down menu. You also need to enter the following information: username and password (if Authentication is enabled), name of the sender, prefix to add to the subject line, email address of recipient, and a port number if necessary.
  - **SNMP**—Use Simple Network Management Protocol, a very common technology for network monitoring used to send notifications to SNMP-capable devices.  
If you choose **SNMP**, enter the SNMP community string in the SNMP Community field and the port number to send notifications to in the Port Number field.

- **SysLog**—Specifies the system log on the destination system as the recipient of event notifications.

If you choose **SysLog**, enter the notification priority in the Priority field, the name of the facility in the Facility field, and the port number on the destination system in the Port Number field.

f. Click **Add**.

**Step 10** Go back to the page listing event definitions to verify that the new event has been added to the list.

---

## Deleting an Event Definition

To delete one or more event definitions from WCS, follow these steps:

- 
- Step 1** In Cisco WCS, choose **Location > Notifications**.
- Step 2** Click **Settings** (left).
- Step 3** Click the name of the group from which you want to delete the event definitions.
- Step 4** Select the event definitions that you want to delete by checking their corresponding check boxes.
- Step 5** From the drop-down menu (right-hand side), choose **Delete Event Definition(s)**, and click **Go**.
- Step 6** Click **OK** to confirm that you want to delete the selected event definitions.
- 

Deleting event definitions as described above removes them from only WCS.

To remove definitions from the location server, follow these steps:

- 
- Step 1** In Cisco WCS, choose **Location > Synchronize Servers**.
- Step 2** From the **Synchronize** drop-down menu, choose **Event Groups**.
- Step 3** To remove an event definition, click **Unassign** for the event group to which the event belongs.
- Step 4** Click **Synchronize**.
- 

## Testing Event Definitions

To verify that the location server is sending event definitions over the transport protocol you have specified in the event definition, use Cisco WCS to test-fire event notifications. The location server sends 3 fictitious event notifications (absence, containment, and distance) to the destinations you have specified in the event definition. The messages contain dummy MAC addresses.

To test one or more event definitions, follow these steps:

- 
- Step 1** In Cisco WCS, choose **Location > Notifications**.
- Step 2** Click **Settings** (left).
- Step 3** Click the name of the group containing the event definitions that you want to test.

- Step 4** Select the event definitions that you want to test by checking their corresponding check boxes.
  - Step 5** From the drop-down menu (right-hand side), choose **Test-Fire Event Definition(s)**, and click **GO**.
  - Step 6** Click **OK** to confirm that you want to test-fire event notifications.
  - Step 7** Check to make sure that notifications were sent to the designated recipient.
- 

## Viewing Event Notification Summary

The location server sends event notifications and does not store them (fire and forget). However, if WCS is a destination of notification events, it stores the notifications it receives and groups them into the following five categories:

- **Absence (Missing)**—The location server generates absence events when the monitored assets go missing. In other words, the location server cannot see the asset in the WLAN for the specified time.
- **In/Out Area (Containment)**—The location server generates containment events when an asset is moved inside or outside a designated area.




---

**Note** You define a containment area (campus, building, or floor) in the Maps section of Cisco WCS (**Monitor > Maps**). You can define a coverage area using the Map Editor.

---

- **Movement from Marker (Movement/Distance)**—The location server generates movement events when an asset is moved beyond a specified distance from a designated marker you define on a map.
- **Location Changes**—The location server generates location change events when client stations, asset tags, rogue clients and rogue access points move from their previous location.
- **Battery Level**—The location server generates battery level events for all tracked asset tags.

To view event notifications, follow these steps:




---

**Note** All element events are summarized hourly and daily.

---

- Step 1** In Cisco WCS, choose **Location > Notifications**.  
Cisco WCS displays a summary of event notifications for each of the five event notification categories.
  - Step 2** To view event notifications for a monitored asset, click one of its corresponding links.  
For example, to view absence events for client stations generated in the last hour, click the link in the Last Hour column for the Client Stations entry in the Absence (Missing) list.  
Clicking one of these links searches for location notifications of all severities.
-

## Notifications Cleared

A location server sends event notifications when it clears an event condition in one of the following scenarios:

- **Missing (Absence)**—Elements reappear.
- **In/Out Area (Containment)**—Elements move back in or out of the containment area.
- **Distance**—Elements move back within the specified distance from a marker.
- **Location Changes**—Clear state is not applicable to this condition.
- **Battery Level**—Tags are detected again operating with Normal battery level.



### Note

In Cisco WCS, the Notifications Summary page reflects whether notifications for cleared event conditions have been received.

## Configuring Notification Parameters




### Note

Tweak notification parameters only if you expect the location server to send a large number of notifications or if notifications are not being received.

You can use Cisco WCS to configure location server event notification parameters.

To configure notification parameters, follow these steps:

- Step 1** In Cisco WCS, choose **Location > Location Servers**.
- Step 2** Click the name of the server you want to configure.
- Step 3** Click **Administration** (left) to display the administrative configuration options.
- Step 4** Click **Notification Parameters**.
- Step 5** Configure the following parameters in the Notification Parameters page:

Parameter	Description
<b>Rate Limit</b>	Enter the rate in milliseconds at which the location server will generate notifications. A value of 0 (default) means that the server will generate event notifications as fast as possible.
<b>Queue Limit</b>	The event queue limit for sending notifications. The server will drop any event above this limit. Default value is 500.
<b>Retry Limit</b>	Enter the number of times to generate an event notification before the refresh time expires. This value ensures, to some extent, that the events that the location server generated will eventually reach WCS. Default value is 1.
	 <p><b>Note</b> The location server does not store events in its database. It just fires events and forgets about them (fire and forget).</p>

Parameter	Description
<b>Refresh Time</b>	Enter the wait time in minutes before restarting the event refresh cycle if an event notification needs to be resent. For example, suppose you enter 30 in this field. If a monitored element goes out of a specified area, the location server sends an event notification. Then, until the event is cleared, the location server resends event notifications every 30 minutes.
<b>Notifications Dropped</b>	(Read only). The number of event notifications dropped from the queue since startup.

**Step 6** Click **Save** to store your updates in the Cisco WCS and location server databases.

---

# Notification Message Formats

This section describes the notification message formats.

## Notification Formats in XML

This section describes the XML format of notification messages.

**Note**

---

The XML format is part of a supported API and Cisco will provide change notification as part of the Location Server API program, whenever the API is updated in the future.

---

### Missing (Absence) Condition

Message format for element absence:

```
<AbsenceTrackEvent
missingFor="<time in secs entity has been missing>"
lastSeen="time last seen"
trackDefn="<name of track definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for the clear state:

```
<AbsenceTrackEvent
state="clear"
trackDefn="<name of track definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Following are examples:

```
<AbsenceTrackEvent state="set" missingFor="34" lastSeen="15:00:20 28 May 2006"
trackDefn="absenceDef1" entityType="Mobile Station"
entityID="00:0c:f1:53:9e:c0"/>
```

```
<AbsenceTrackEvent state="clear" entityType="Tag"
trackDefn="absenceDef1" entityID="00:0c:cc:5b:fc:da"/>
```

## In/Out (Containment) Condition

Message format for element containment:

```
<ContainmentTrackEvent
in="true | false"
trackDefn="<name of track definition>"
containerType="Floor | Area | Network Design | Building"
containerID="<fully qualified name of container>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for the clear state:

```
<ContainmentTrackEvent
state="clear"
trackDefn="<name of track definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Following are examples:

```
<ContainmentTrackEvent in="true" trackDefn="myContainerRule1"
containerType="Area"
containerID="wcsDevArea,4th Floor,Bldg-14,WNBU_Group,WNBU,"
entityType="Tag" entityID="00:0c:cc:5b:fa:44"/>
```



**Note** The containerID string represents a coverage area called `wcsDevArea`, located in the floor 4th floor of Bldg-14 of the campus WNBU.

```
<ContainmentTrackEvent state="clear" entityType="Tag"
trackDefn="myContainerRule1" entityID="00:0c:cc:5b:f8:ab"/>
```

## Distance Condition

Message format for elements in the same floor:

```
<MovementTrackEvent
distance="<distance in feet at which the element was located>"
triggerDistance="<the distance specified on the condition>"
reference="<name of the marker specified on the condition>"
trackDefn="<name of event definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for elements located in a different floor:

```
<MovementTrackEvent optionMsg="has moved beyond original floor"
reference="<name of the marker specified on the condition>"
trackDefn="<name of event definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Message format for clear state:

```
<MovementTrackEvent
state="clear"
trackDefn="<name of event definition>"
entityType="Mobile Station | Tag | Rogue AP | Rogue Client"
entityID="<mac address"/>
```

Following are examples:

```
<MovementTrackEvent distance="115.73819627990147" triggerDistance="60.0"
reference="marker2" trackDefn="distance2" entityType="Mobile Station"
entityID="00:0c:41:15:99:92" />
```

```
<MovementTrackEvent optionMsg="has moved beyond original floor"
reference="marker2" entityType="Tag"
trackDefn="distance2"
entityID="00:0c:cc:5b:fa:4c" />
```

```
<MovementTrackEvent state="clear" entityType="Tag"
```

## Battery Level

An example:

```
<BatteryLifeTrackEvent lastSeen="10:28:52 23 May 2006" batteryStatus="medium"
trackDefn="defn1" entityType="Tag" entityID="00:01:02:03:04:06" />
```

## Location Change

An example:

```
<MovementTrackEvent distance="158.11388300841898" triggerDistance="5.0"
reference="marker1" referenceObjectID="1" trackDefn="defn1" entityType="Mobile Station"
entityID="00:01:02:03:04:05" />
```

## Notification Formats in Text

When you specify that notification be sent in Text format, the location server uses a plain-text string to indicate the condition. Following are examples:

```
Tag 00:02:02:03:03:04 is in Floor <floorName>
Tag 00:02:02:03:03:04 is outside Floor <floorName>
Client 00:02:02:03:09:09 is in Area <areaName>
RogueClient 00:02:02:08:08:08 is outside Building <buildingName>
Tag 00:02:02:03:03:06 has moved 105 feet where the trigger distance was 90 feet.
Tag 00:02:02:03:03:20 missing for 14 mins, last seen <timestamp>.
```



---

**Note**

Cisco maintains the right to modify the Text notification Format, without notice, at any time in the future.

---



---

**Note**

XML is the recommended format if systems need to parse or analyze the notification contents.

---

## WCS as a Notification Listener

WCS acts as a notification listener. WCS receives the notifications from location servers in the form of the trap `locationNotifyTrap` as part of the MIB file `bsnwras.my`. The location server stores the content of the notification message in XML format in the variable `locationNotifyContent` (see [“Notification Formats in XML” on page 6-9](#)).

```
locationNotifyTrap NOTIFICATION-TYPE
  OBJECTS { locationNotifyContent}
  STATUS current
  DESCRIPTION
    "This trap will be generated by the location server
    for notifications of location events."
  ::= { bsnTraps 89 }

locationNotifyContent OBJECT-TYPE
  SYNTAX OCTET STRING(SIZE(0..512))
  MAX-ACCESS accessible-for-notify
  STATUS current
  DESCRIPTION
    "This is the content of the notification."
  ::= { bsnTrapVariable 72 }
```

WCS translates the traps into UI alerts and displays them in the following formats:

- **Missing (Absence)**  
Absence of Tag with MAC 00:0c:cc:5b:e4:1b, last seen at 16:19:45 13 Oct 2005.
- **In/Out (Containment)**  
Tag with MAC 00:0c:cc:5b:fa:44 is In the Area 'WNBU > WNBU > 4th Floor > wcsDevArea'
- **Distance**  
Tag with MAC 00:0c:cc:5b:fa:47 has moved beyond the distance configured for the marker 'marker2'.  
Tag with MAC 00:0c:cc:5b:f9:b9 has moved beyond 46.0 ft. of marker 'marker2', located at a range of 136.74526528595058 ft.
- **Battery Level**  
Tag 00:01:02:03:04:06 has medium battery, last seen 11:06:01 23 May 2006
- **Location Change**  
Mobile Station 00:01:02:03:04:05 has moved  
158.11388300841898ft, where the trigger distance was 5.0