CHAPTER 6

Configuring Event Notifications

With Cisco WCS, you can define conditions that cause the mobility service engine to send notifications to specific listeners. This chapter describes how to define events and event groups and how to view event notification summaries.

This chapter contains the following sections:

- Adding and Deleting Event Groups, page 6-2
- Adding, Deleting, and Testing Event Definitions, page 6-2
- Viewing Event Notification Summary, page 6-7
- Clearing Notifications, page 6-8
- Notification Message Formats, page 6-8

Adding and Deleting Event Groups

This section describes how to add and delete event groups. Event groups help you organize your event notifications.

Adding Event Groups

To add an event group, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2 Choose **Notification Settings** (left panel).

Step 3 From the Select a command drop-down menu, select **Add Event Group**. Click **Go**.

Step 4 Enter the name of the group in the Group Name field.

Step 5 Click **Save**.

The new event group appears in the Event Settings window.

Deleting Event Groups

To delete an event group, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2 Choose **Notification Settings** (left panel).

Step 3 Select the event group to delete by checking its corresponding check box.

Step 4 From the Select a command drop-down menu, select **Delete Event Group(s)**. Click **Go**.

Step 5 In the panel that appears, click **OK** to confirm deletion.

Step 6 Click **Save**.

Adding, Deleting, and Testing 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 destination.

This section describes how to add, delete, and test event definitions.

Adding an Event Definition

Cisco WCS enables you to add an event definition to a group. An event definition must belong to a particular group.

To add an event definition, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2 Choose **Notification Settings** (left panel).

Step 3 Click the name of the group to which you want to add an event definition. An event definition summary window appears showing existing event definitions for the event group.

Step 4 From the Select a command drop-down menu, select **Add Event Definition**. Click **Go**.

Step 5 At the Conditions tab, add one or more conditions. For each condition you add, specify the rules for triggering event notifications.

Tip For example, to keep track of heart monitors in a hospital, you might add rules to generate notifications when the following occur: (1) the heart monitor is missing for one hour, (2) the heart monitor moves off its assigned floor, or (3) the heart monitor enters a specific coverage area within a floor. In this example, we would add three separate rules to address these occurrences.

To add a condition, follow these steps:

a. Click **Add** to add a condition that triggers a notification.

b. In the Add/Edit Condition dialog box, follow these steps:

1. Choose a condition type from the Condition Type drop-down menu.

- If you chose **Missing** from the Condition Type drop-down menu, enter the number of minutes after which a missing asset generates a notification. For example, if you enter 10 in this field, the mobility service engine generates a missing asset notification if the mobility service engine 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**. Entry and exit of assets from the selected area is then monitored. 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 from a designated marker beyond which an asset triggers an event notification. 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 farther than 60 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 **Battery Level** from the Condition Type drop-down menu, check the box next to the appropriate battery level (low, medium, normal) that will trigger a notification. Proceed to Step c.
- If you chose **Location Change** from the Condition Type drop-down menu, proceed to Step c.

- If you chose **Emergency** from the Condition Type drop-down menu, click the button next to the appropriate emergency (any, panic button, tampered, detached) that will trigger a notification. Proceed to Step c.
- If you chose **Chokepoint** from the Condition Type drop-down menu, proceed to Step c. There is only one trigger condition and it is displayed by default. No configuration required.

c. From the Apply To drop-down menu, choose the type of asset (Any, Clients, Tags, Rogue APs, Rogue Clients, or Interferers) for which a notification will be generated if the trigger condition is met.

Note If you select the *Any* option from the Apply to drop-down menu, the battery condition is applied to all tags, clients, and rogue access points, and rogue clients.

Note Emergency and chokepoint notifications apply only to Cisco compatible extension (CX) tags version 1 (or later).

d. For the Match By option there are three entries, left to right:

- Choose the matching criteria (MAC Address, Asset Name, Asset Group, or Asset Category) from the first drop-down menu.
- Choose the operator (Equals or Like) from the second drop-down menu.
- Enter the relevant text into the field base on the selected Match By element.

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

  – If you choose **MAC Address** from the first drop-down menu, choose **Equals** from the second drop-down menu, and enter a MAC address (for example **12:12:12:12:12:12)** in the field, 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 first drop-down menu, choose **Like** from the second drop-down menu, and enter **12:12** in the field, the event condition applies to elements whose MAC address starts with 12:12.

e. Click **Add** to add the condition you have just defined.

Note If you are defining a chokepoint, you must select the chokepoint after you add the condition.

To select a chokepoint, do the following:

1. Click **Select Chokepoint**. An entry panel appears.
2. Select Campus, Building, and Floor from the appropriate drop-down menus.
3. Select a Chokepoint from the menu that appears.

The Add/Edit Condition panel reappears and the location path (*Campus* > *Building* > *Floor*) for the chokepoint auto-populates the entry field next to the Select Checkpoint button.

Step 6 At the Destination and Transport tab, follow these steps to add one or more destinations to receive event notifications and to configure the transport settings:

a. To add a new destination, click **Add**. The Add/Edit Destination configuration panel appears.

b. Click **Add New**.

c. In the pop up that appears, enter the IP address of the system that will receive event notifications, and click **OK**.

The new entry is placed in the right column.

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 destination.

d. To select a destination for notifications, 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.

e. Select **XML** or **Plain Text** as the message format.

Note If you select WCS as the destination for notifications, you must select the XML format.

f. Choose one of the following transport types from the Transport Type drop-down menu:

- **SOAP**—Simple Object Access Protocol, a simple XML protocol. Use SOAP to send notifications over either HTTP (default) or HTTPS for process by web services at the destination.

  Be sure to select HTTPS in step g if you do not want to send notifications over HTTP.

  Also, enter a destination port number in step h if a value other than the auto-populated value is required.

- **Mail**—Use this option to send notifications by 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.

g. To enable HTTPS, check the **Enable** check box next to it.

h. **Port Number** auto-populates.

i. Click **Save**.

Step 7 At the General tab, follow these steps:

a. Check the Admin Status **Enabled** check box to enable event definition (disabled by default).

b. Set the event definition 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.

c. Choose the frequency of notifications:
   1. Check the **All the Time** check box to continuously report events. Proceed to step g.
   2. Uncheck the **All the Time** check box to select the day and time of the week that you want event notifications sent. Days of the week and time fields appear for selection. Proceed to step d.

d. Check the check box next to each day that you want the event notification sent.

e. Choose a start time for the event notification by selecting the hour, minute, and AM or PM using the Apply From drop-down menus.

f. Choose an end time for the event notification by selecting the hour, minute, and AM or PM from the Apply Until drop-down menus.

g. Click **Save**.

Step 8 Verify that the new event definition is listed for the event group (Services > Context Aware Notifications > Notification Settings > *Group Name)*.

Deleting an Event Definition

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

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2 Choose **Notification Settings** (left panel).

Step 3 Click the name of the group from which you want to delete an event definition.

Step 4 Select the event definition that you want to delete by checking its corresponding check box.

Step 5 From the Select a command drop-down menu, choose **Delete Event Definition(s)**. Click **Go**.

Step 6 Click **OK** to confirm that you want to delete the selected event definition.

Testing Event Definitions

You can use Cisco WCS to verify that the mobility service engine is sending an event notification over the transport protocol you have specified in an event definition. The mobility service engine sends three fictitious event notifications (absence, containment, and distance) to the destination you have specified in the event definition. The messages contain dummy MAC addresses.

Note Emergency and chokepoint event notifications are not tested.

To test one or more event notifications of an event definition, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Step 2 Choose **Notification Settings** (left panel).

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 Select a command drop-down menu, choose **Test-Fire Event Definition(s)**. Click **Go**.

Step 6 Click **OK** to confirm that you want to test the event notifications.

Step 7 Ensure that notifications were sent to the designated recipient.

Viewing Event Notification Summary

The mobility services engine sends event notifications and does not store them. However, if WCS is a destination of notification events, it stores the notifications it receives and groups them into the following seven categories:

- Absence (Missing)—The mobility services engine generates an absence event when an asset goes missing. In other words, the mobility services engine cannot detect the asset in the WLAN for the specified time.
- In/Out Area (Containment)—The mobility services engine generates a containment event when an asset moves in or out of 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 mobility services engine generates a movement event when an asset is moved beyond a specified distance from a designated marker you define on a map.
- Location Changes—The mobility services engine generates location change events when a client station, asset tag, rogue client and rogue access point changes its location.
- Battery Level—The mobility services engine generates battery level events for all tracked asset tags.
- Emergency—The mobility services engine generates an emergency event for a Cisco CX v.1 compliant asset tag when the tag's panic button is triggered or the tag becomes detached, is tampered with, becomes inactive, or reports an unknown state. This information is reported and displayed only for Cisco CX v.1 compliant tags.
- Chokepoint Notifications—The mobility services engine generates an event when a tag is stimulated by a chokepoint. This information is reported and displayed only for Cisco CX v.1 compliant tags.

Note All element events are summarized hourly and daily.

To view event notifications, follow these steps:

Step 1 In Cisco WCS, choose **Services > Context Aware Notifications**.

Cisco WCS displays a summary of event notifications for each of the seven event notification categories.

Note Emergency and chokepoint notifications are reported and displayed only for Cisco CX v.1 compliant tags.

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.

Clearing Notifications

A mobility services engine sends event notifications when it clears an event condition in one of the following scenarios:

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

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

Notification Message Formats

This section describes the notification message formats for XML and text.

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 Mobility Services Engine 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 08 Jun 2009"
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 quality 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="nycTestArea,5th Floor,Bldg-A,Rochester_Group,Rochester,"
```

Note The `containerID` string represents a coverage area called `nycTestArea`, located in the 5th floor of Bldg-A of the campus *Rochester.*

```
entityType="Tag" entityID="00:0c:cc:5b:fa:44"/>

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

Distance Condition

Message format for elements on 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 on 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 08 Jun 2009" 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"/>
```

Chokepoint Condition

An example:

```
<ChokepointTrackEvent
lastSeen="11:10:08 PST 08 Jun 2009"
chokepointMac="00:0c:cc:60:13:a3"
chokepointName= "chokeA3"
trackDefn="choke"
entityType="Tag"
entityID="00:12:b8:00:20:4f"/>
```

An example for the clear state:

```
<ChokepointTrackEvent
state="clear"
entityType="Tag"
trackDefn="choke"
entityID="00:12:b8:00:20:4f"/>
```

Emergency Condition

An example for element location:

```
<ChokepointTrackEvent
lastSeen="11:36:46 PST June 08 2009"
emergencyReason= "detached"
trackDefn="emer"
entityType="Tag"
entityID="00:12:b8:00:20:50"/>
```

Note Emergency events are never cleared.

Notification Formats in Text

When you specify that notification be sent in text format, the mobility services engine 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.

Note XML is the recommended format for systems that need to parse or analyze notification contents.

Cisco WCS as a Notification Listener

Cisco WCS acts as a notification listener. Cisco WCS receives the notifications from the mobility services engine in the form of the trap `locationNotifyTrap` as part of the MIB file bsnwras.my. The mobility services engine stores the content of the notification message in XML format in the variable `locationNotifyContent` (see the "Notification Formats in XML" section on page 6-8).

```
locationNotifyTrap  NOTIFICATION-TYPE
    OBJECTS { locationNotifyContent}
    STATUS current
    DESCRIPTION
        "This trap will be generated by the mobility services engine
        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 }
```

Cisco 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 08 June 2009.
  ```

- In/Out (Containment)

  ```
  Tag with MAC 00:0c:cc:5b:fa:44 is In the Area 'Rochester > Rochester > 5th Floor >
  nycTestArea'
  ```

- 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 08 June 2009
  ```

- Location Change

  ```
  Mobile Station 00:01:02:03:04:05 has moved
  158.11388300841898ft, where the trigger distance was 5.0
  ```