



Ticket Audit events reference

ON THIS PAGE

- Create event
- Change event
- Comment event
- Comment redaction event
- Attachment redaction event
- Voice comment event
- Comment privacy change event
- Suspended ticket recovery event
- Notification event
- Notification with CCs event
- SMS notification event
- CC event
- Follower notification event
- Follower change event
- Email CC change event
- Macro reference event
- Satisfaction rating event
- Ticket sharing event
- Organization subscription notification event
- SLA target change event
- Error event
- Tweet event
- Facebook event
- Facebook comment event

External event
LogMeIn transcript event
Push event
Chat started event
Chat ended event
Skill assigned event
Survey offered event
Survey response submitted event
Offered To event
Routing channel event

The [Ticket Audits](#) resource in the Support REST API can use the following events to describe the ticket update:

- [Create event](#)
- [Change event](#)
- [Comment event](#)
- [Comment redaction event](#)
- [Attachment redaction event](#)
- [Voice comment event](#)
- [Comment privacy change event](#)
- [Suspended ticket recovery event](#)
- [Notification event](#)
- [Notification with CCs event](#)
- [SMS notification event](#)
- [CC event](#)
- [Follower notification event](#)
- [Follower change event](#)
- [Email CC change event](#)
- [Macro reference event](#)
- [Satisfaction rating event](#)
- [Ticket sharing event](#)
- [Organization subscription notification event](#)
- [SLA target change event](#)
- [Error event](#)
- [Tweet event](#)
- [Facebook event](#)
- [Facebook comment event](#)

- [External event](#)
- [LogMeIn transcript event](#)
- [Push event](#)
- [Chat ended event](#)
- [Skill assigned event](#)
- [Survey offered event](#)
- [Survey response submitted event](#)
- [Offered To event](#)
- [Routing channel event](#)

An event will have its own `via` object if it's different from the `via` object of the audit. See [Support API: Via object reference](#).

Note: Events may be added at any time by the Zendesk developer team. Please ignore undocumented event types in your integration.

Create event

A ticket property was set on a newly created ticket. A separate event is created for each property set.

Create events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>Create</code>
field_name	string	yes	The name of the field that was set
value	string, array, object	yes	The value of the field that was set

`value` is normally a string. However, the attribute is an array when the value of `field_name` is `"tags"`. It's an object when the value of `field_name` is a SLA event like `"first_reply_time"`. Example:

```
1  "value": {
2    "minutes": 1440,
3    "in_business_hours": false
4  }
```

Example

```
1  {
2    "id":      1274,
3    "type":    "Create"
4    "field_name": "status",
5    "value":    "new"
6  }
```

Change event

A ticket property was updated. The event describes the previous and newly updated value of each ticket property.

Change events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value Change
field_name	string	yes	The name of the field that was changed
value	string, array, object	yes	The value of the field that was changed
previous_value	string, array, object	yes	The previous value of the field that was changed

value and previous_value are normally strings. However, the attribute is an array when the value of field_name is tags. It's an object when the value of field_name is a SLA event like "first_reply_time".

Example

```
1  {
2    "id":      1274,
3    "type":    "Change"
4    "field_name": "subject",
5    "value":    "My printer is on fire!",
6    "previous_value": "I need help!"
7  }
```

Comment event

A comment was added to the ticket. The event's `type` is "Comment". The event's properties are identical to the ticket comment object. By default, inline images aren't included in the `attachments` array. See [Ticket comments](#).

Example

```
1  {
2    "id": 2127301148,
3    "type": "Comment",
4    "body": "This is a new private comment",
5    "html_body": "<p>This is a new private comment</p>",
6    "public": false,
7    "created_at": "2015-07-20T22:55:29Z",
8    "author_id": 123123,
9    "attachments": []
10 }
```

Comment redaction event

A word or string was redacted from a ticket comment using the API. See [Redact String in Comment](#). Comment redaction events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>CommentRedactionEvent</code>
comment_id	integer	yes	The comment with the redacted text

Example

```
1  {
2    "id": 59542664837,
3    "comment_id": "59733541888",
4    "type": "CommentRedactionEvent"
5  }
```

Attachment redaction event

An attachment was redacted, or permanently deleted, from a ticket comment using the REST API. See [Redacting comment attachments](#). These events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>AttachmentRedactionEvent</code>
attachment_id	integer	yes	The redacted attachment
comment_id	integer	yes	The comment with the redacted attachment

Example

```
1  {
2    "id": 59549109257,
3    "type": "AttachmentRedactionEvent",
4    "attachment_id": 1636097007,
5    "comment_id": 59738862068
6  }
```

Voice comment event

A voice comment was added to a ticket with Zendesk Talk.

Voice comment events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>VoiceComment</code>
data	object	yes	Properties describing the voice comment
formatted_from	string	yes	A formatted version of the phone number which dialed the call
formatted_to	string	yes	A formatted version of the phone number which answered the call

Name	Type	Read-only	Comment
body	string	yes	Comment added to the ticket
html_body	string	yes	The comment formatted to HTML
public	boolean	yes	If true, the ticket requester can see the comment. If false, only agents can see it
trusted	boolean	yes	If this comment is trusted or marked as being potentially fraudulent
author_id	integer	yes	The comment author, typically the agent assigned to the ticket
transcription_visible	boolean	yes	Whether the transcription is visible on the ticket
attachments	array	yes	The attachments on this comment as Attachment objects

Example

```

1  {
2    "id": 215546547,
3    "type": "VoiceComment",
4    "body": "'Inbound call from +1 (123) 654-7890\nCall Details:\n\nCall from...'",
5    "data": {
6      "answered_by_id": 63197591,
7      "answered_by_name": "Keith Hayward",
8      "author_id": 63197591,
9      "brand_id": 1156956,
10     "call_duration": 64,
11     "call_id": 129873628,
12     "from": "+11236547890",
13     "location": "Roselle, Illinois, United States",
14     "to": "+11233257890",
15     "public": false,
16     "recording_url":
17       "https://omniwear.zendesk.com/api/v2/channels/voice/calls/CAed671/twilio/recording",
18     "started_at": "2016-12-20T16:30:16Z",
19     "transcription_status": "completed",
20     "transcription_text": "Hello, I have a problem with...",
21     "via_id": 34
22   },
23   "formatted_from": "+1 (123) 654-7890",

```

```
23   "formatted_to":      "+1 (123) 325-7890",
24   "transcription_visible": false,
25   "public":            false,
26   "html_body":
    "<div class=\"zd-comment\">\n<p dir=\"auto\">Inbound call from +1 (123) 654-
    7890<br>\nCall Details..."
27   ,
28   "author_id":          63197591,
29   "trusted":            true,
30   "attachments":        []
}
```

Notes:

- `data.answered_by_id` is not present for voicemails
- `data.transcription_status` and `data.transcription_text` are only present for voicemails with transcription enabled

Comment privacy change event

A public comment was marked as private.

Ticket comment privacy change events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>CommentPrivacyChange</code>
comment_id	integer	yes	The id of the comment that changed privacy
public	boolean	yes	Tells if the comment was made public or private

Example

```
1  {
2    "id": 1274,
3    "type": "CommentPrivacyChange",
4    "comment_id": 453,
5    "public": false
6  }
```


Suspended ticket recovery event

A suspended ticket was recovered.

Suspended ticket recovery events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>SuspendedTicketRecovery</code>
via	object	yes	The suspended ticket that created the ticket. See Via object reference
cause	string	yes	The details of the suspension. See Cause of suspension reference
cause_name	string	yes	The key of the cause of suspension. See Cause of suspension reference
cause_id	integer	yes	The id of the cause of suspension. See Cause of suspension reference
recovery_details	string	yes	The details of the recovery, including the recovery type and the recovery author's name
recovery_type	string	yes	The type of recovery. May be <code>manual</code> or <code>automatic</code>
recovered_at	datetime	yes	The date of recovery
recovered_by	integer	yes	The user who performed the recovery

Example

```
1  {
2    "id": 10471060966419,
3    "type": "SuspendedTicketRecovery",
4    "via": {
5      "channel": "suspended_ticket",
6      "source": {
7        "to": {},
8        "from": {
9          "id": 10471036265747
10       },
11    },
12  }
```

```

11     "rel": "suspended_ticket"
12   }
13 },
14   "cause": "Automated response email, delivery failed",
15   "cause_name": "auto_delivery_failure",
16   "cause_id": 7,
17   "recovery_details": "Recovered automatically by Admin",
18   "recovery_type": "automatic",
19   "recovered_at": "2022-10-17T18:35:27Z",
20   "recovered_by": 2987483127
21 }

```

Notification event

A notification was sent by a business rule such as a trigger when the ticket was created or updated.

Notifications have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value Notification
subject	string	yes	The subject of the message sent to the recipients
body	string	yes	The message sent to the recipients
recipients	array	yes	An array of simple objects with the ids and names of the recipients of this notification
via	object	yes	The business rule that created the notification. See Via object reference

Example

```

1  {
2    "id": 1275,
3    "type": "Notification"
4    "subject": "Your ticket has been updated"
5    "body": "Ticket #235 has been updated"
6    "recipients": [847390, 93905],
7    "via": {
8      "channel": "system",

```

```
9      "source": {
10        "type": "rule",
11        "id": 61,
12        "title": "Notify assignee of comment update"
13      }
14    }
15  }
```

Notification with CCs event

A notification was sent to the requester and email CCs.

Notifications have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>NotificationWithCcs</code>
subject	string	yes	The subject of the message sent to the recipients
body	string	yes	The message sent to the recipients
recipients	array	yes	An array of simple objects with the ids and names of the recipients of this notification
via	object	yes	The business rule that created the notification. See Via object reference

Example

```
1  {
2    "id": 1275,
3    "type": "NotificationWithCcs"
4    "subject": "Your ticket has been updated"
5    "body": "Ticket #235 has been updated"
6    "recipients": [847390, 93905],
7    "via": {
8      "channel": "system",
9      "source": {
10        "type": "rule",
11        "id": 61,
```

```
12      "title": "Notify requester and email CCs of comment update"
13    }
14  }
15 }
```

SMS notification event

An SMS notification was sent by a business rule such as a trigger when the ticket was created or updated.

Notifications have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>SmsNotification</code>
body	string	yes	The message sent to the recipients
recipients	array	yes	An array of simple objects with the ids and names of the recipients of this notification

Example

```
1  {
2    "id":      1275,
3    "type":    "SmsNotification"
4    "body":    "Ticket #235 has been updated"
5    "recipients": [847390, 93905]
6  }
```

CC event

A cc (also known as a collaborator) was notified when the ticket was updated.

Ticket CC Events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value Cc
body	string	yes	The message sent to the recipients
recipients	array	yes	A array of simple objects with the ids and names of the recipients of this notification
via	object	yes	A reference to the business rule that created this notification. See Via object reference

Example

```

1  {
2    "id": 1275,
3    "type": "Cc",
4    "recipients": [93905],
5    "body":
      "You are registered as a CC on this request ({{ticket.id}}). Reply to this email
      to add a comment to the request.\n\n{{ticket.comments_formatted}}"
6    ,
7    "via": {
8      "channel": "system",
9      "source": {
10       "type": "rule",
11       "id": 62,
12       "title": "Notify collaborator of comment update"
13     }
14   }
15 }
```

Follower notification event

A follower was notified when the ticket was updated.

Ticket follower notification events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>FollowerNotificationEvent</code>
subject	string	yes	The subject of the message sent to the recipients
body	string	yes	The message sent to the recipients
recipients	array	yes	An array of simple objects with the ids and names of the recipients of this notification
via	object	yes	A reference to the business rule that created this notification. See Via object reference

Example

```

1  {
2    "id":      1275,
3    "type":    "FollowerNotification",
4    "subject": "Your ticket has been updated",
5    "body":    "You are a Follower on this request ({{ticket.id}}).
               {{ticket.follower_reply_type_message}}\n\n{{ticket.comments_formatted}}"
6    "recipients": [847390],
7    "via": {
8      "channel": "system",
9      "source": {
10       "type": "rule",
11       "id":   61,
12       "title": "Notify follower of comment update"
13     }
14   }
15 }
```

Follower change event

Followers have been added or removed from the ticket.

Follower change events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value "FollowersChange"
previous_followers	array	yes	The previous followers on the ticket
current_followers	array	yes	The current followers on the ticket

Example

```
1  {
2    "id": 1274,
3    "type": "FollowerChange"
4    "previous_followers": ["agent_uno@{subdomain}.com", "agent_dos@{subdomain}.com"]
5    "current_followers": ["agent_uno@{subdomain}.com"]
6  }
```

Email CC change event

Email CCs have been added or removed from the ticket.

Email CC change events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value EmailCcChange
previous_email_ccs	array	yes	The previous email CCs on the ticket
current_email_ccs	array	yes	The current email CCs on the ticket

Example

```
1  {
2    "id": 1274,
3    "type": "EmailCcChange"
4    "previous_email_ccs": ["agent_uno@{subdomain}.com", "end_user@example.com"]
5  }
```

```
5   "current_email_ccs": ["end_user@example.com"]
6 }
```

Macro reference event

Macro reference events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when creating events
type	string	yes	Has the value <code>MacroReference</code>
macro_id	integer	yes	The id of the macro this event refers to
macro_title	string	yes	The title of the macro this event refers to
macro_deleted	boolean	yes	Whether or not the macro this event refers to is deleted
via	object	yes	How the event was created. See Via object reference

Example

```
1  {
2    "id": 1274,
3    "type": "AgentMacroReference",
4    "via": {
5      "channel": "web",
6      "source": {
7        "from": {},
8        "to": {},
9        "rel": null
10   }
11 },
12 "macro_id": 45345,
13 "macro_title": "Downgrade ticket",
14 "macro_deleted": false
15 }
```

Satisfaction rating event

Satisfaction rating events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when creating events
type	string	yes	Has the value SatisfactionRating
score	string	yes	The rating state "offered", "unoffered", "good", "bad"
assignee_id	integer	yes	Who the ticket was assigned to upon rating time
body	string	yes	The users comment posted during rating

Example

```
1  {
2    "id": 1274,
3    "type": "SatisfactionRating",
4    "score": "good",
5    "assignee_id": 87374,
6    "body": "Thanks, you guys are great!"
7  }
```

Ticket sharing event

Ticket sharing events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when creating events
type	string	yes	Has the value TicketSharingEvent
agreement_id	integer	yes	The sharing agreement id
action	string	yes	Either shared or unshared

Example

```
1  {
2    "id": 1274,
3    "type": "TicketSharingEvent",
4    "agreement_id": 3454,
```

```
5   "action": "shared"
6 }
```

Organization subscription notification event

A notification was sent to the organization subscribers when somebody in the organization submitted a ticket.

This feature was available in the classic version of the Zendesk Support user interface. In the current version of Zendesk Support, you can use the [Organization Subscriptions API](#) to create the subscriptions.

Organization subscription notification events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>OrganizationActivity</code>
subject	string	yes	The subject of the message sent to the recipients
body	string	yes	The message sent to the recipients
recipients	array	yes	An array of simple objects with the ids and names of the recipients of the notification
via	object	yes	A reference to the trigger that created the notification. See Via object reference

Example

```
1  {
2    "id":      1275,
3    "type":    "OrganizationActivity"
4    "subject": "Your ticket has been updated"
5    "body":    "Ticket #235 has been updated"
6    "recipients": [847390, 93905],
7    "via": {
8      "channel": "system",
9      "source": {
10       "type": "rule",
11       "id":   61,
12       "title": "Notify requester of comment update"
13     }
14   }
```

```
14   }
15 }
```

SLA target change event

SLA target change events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when creating events
type	string	yes	Has the value <code>Change</code>
previous_value	object	yes	Previous value in minutes, and indicator of whether the value was in calendar hours or business hours. May be empty
value	object	yes	Current value in minutes, and indicator of whether the value was in calendar hours or business hours. May be empty
field_name	string	yes	The identifier of the SLA metric. May be <code>"first_reply_time"</code> or <code>"requester_wait_time"</code>
via	object	yes	Via information. See Via object reference

Example

```
1  {
2    "id": 4497,
3    "type": "Change",
4    "previous_value": { "minutes": 150, "business_hours": false },
5    "value": { "minutes": 45, "in_business_hours": false },
6    "field_name": "requester_wait_time",
7    "via": {
8      "source": {
9        "rel": "sla_target_change"
10     },
11    "current_sla_policy": "for demo"
12  }
13 }
```

Error event

An error occurred during the processing of the ticket.

Ticket errors have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is creating
type	string	yes	Has the value <code>Error</code>
message	string	yes	The error message

Example

```
1  {
2    "id": 1274,
3    "type": "Error",
4    "message": 453
5  }
```

Tweet event

A comment was added to the ticket from X (formerly Twitter).

Tweet events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>Tweet</code>
direct_message	boolean	yes	Whether this tweet was a direct message
body	string	yes	The body of the tweet
recipients	array	yes	The recipients of this tweet

Example

```
1  {
2    "id": 1274,
```

```
3  "type": "Tweet",
4  "direct_message": false,
5  "body": "Hi there",
6  "recipients": [847390, 93905]
7  }
```

Facebook event

A comment was posted on a Facebook Wall, or a private message was sent to a Facebook Page.

Facebook events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value FacebookEvent
page	hash	yes	The name and graph id of the Facebook Page associated with the event
communication	integer	yes	The Zendesk Support id of the associated communication (wall post or message)
ticket_via	string	yes	"post" or "message" depending on the association with a wall post or a private message
body	string	yes	The value of the message posted to Facebook

Example

```
1  {
2    "id": 1274,
3    "type": "FacebookEvent",
4    "page": {
5      "name": "Zendesk",
6      "graph_id": "61675732935"
7    },
8    "communication": 5,
9    "ticket_via": "post",
10   "body": "Thanks!"
11 }
```

Facebook comment event

A comment was added to a ticket from Facebook.

Facebook comments have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value FacebookComment
data	object	yes	Properties of the Facebook comment
body	string	yes	The actual comment made by the author
html_body	string	yes	The actual comment made by the author formatted as HTML
public	boolean	yes	If this is a public comment or an internal-agents-only note
trusted	boolean	yes	If this comment is trusted or marked as being potentially fraudulent
author_id	integer	yes	The id of the author of this comment
graph_object_id	string	yes	The graph object id of the associated Facebook Wall post or message

Example

```
1  {
2    "id": 1274,
3    "type": "FacebookComment",
4    "data": {
5      "type": "status"
6      "content": "asrk2d",
7      "attachments": [
8        {
9          "id": "70713f06c93b0cba705cc10239ea3e4c",
10         "mime_type": "image/png",
11         "name": "transpmentor.png",
12         "size": 26981
13       }
14     ],
```

```
15     "via_zendesk": false
16   },
17   "public": true,
18   "author_id": 1,
19   "body": "Thanks for your help!",
20   "html_body": "<p>Thanks for your help!</p>",
21   "trusted": true,
22   "graph_object_id": "152318411530606_1523184115123123",
23   "attachments": []
24 }
```

External event

External ticket events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value External
resource	string	yes	External target id
body	string	yes	Trigger message for this target event

Example

```
1  {
2    "id": 1274,
3    "type": "External",
4    "resource": 135476,
5    "body": "Target this ticket {{ticket.id}}"
6  }
```

LogMeIn transcript event

LogMeIn transcript events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when creating events

Name	Type	Read-only	Comment
type	string	yes	Has the value LogMeInTranscript
body	string	yes	An audit of the transcript

Example

```
1  {
2    "id": 1274,
3    "type": "LogMeInTranscript",
4    "body":
      "Session ID: 1234 Technician name: Johnny User Technician email:
      johnny@example.com ..."
5  }
```

Push event

Information was pushed to an external target. See [Notifying external targets](#) in Zendesk help.

Push events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value Push
value	string	yes	Data being pushed out of our system
value_reference	string	yes	A reference to the destination of the data

Example

```
1  {
2    "id": 1274,
3    "type": "Push",
4    "value":
      "<li><strong>Project</strong>: Internal</li><li><strong>Task</strong>: Admin...",
5    "value_reference": "Harvest Time Tracking"
```



```
6 }
```

Chat started event

A chat, Zendesk messaging, or social messaging conversation was started in Agent Workspace.

Chat started events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>ChatStartedEvent</code>
value	array	yes	Properties of the messaging conversation
history	array	yes	Data about chat events
webpath	array	yes	The webpath for the messaging conversation
attachments	array	yes	The attachments on this conversation as Attachment objects

Example

```
1  {
2    "id": 1717959154921,
3    "type": "ChatStartedEvent",
4    "value": {
5      "visitor_id": "10655845-15VkFHoZI7rnzr0",
6      "chat_id": "2108.10655845.SfiWb2lS6gX6c",
7      "tags": ["whatsapp_chat"]
8    }
9  }
```

Chat ended event

A chat, Zendesk messaging, or social messaging conversation was ended in Agent Workspace.

Chat ended events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value ChatEndedEvent
value	array	yes	Properties of the messaging conversation
attachments	array	yes	The attachments on this conversation as Attachment objects

Example

```
1  {
2    "id": 1920482841485,
3    "type": "ChatEndedEvent",
4    "value": {
5      "chat_id": "2108.10655845.SfiWb2lS6gX6c",
6      "visitor_id": "10655845-15VkJHoZl7rnzr0",
7      "is_served": false,
8      "tags": ["whatsapp_chat"]
9    },
10   "attachments": []
11 }
```

Skill assigned event

One or more [skills](#) were assigned to the ticket. In this event, skills are named attribute values. Skill types are named attributes.

Skill assigned events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value AssociateAttValsEvent
attribute_values	array	yes	Skills assigned to the ticket. attribute_id is the id for the skill type

Example

```
1  {
2    "id": 11654650767636,
3    "type": "AssociateAttValsEvent",
4    "attribute_values": [
5      {
6        "id": "2aa15dcf-7d4f-11ed-8b72-9b5bb2ceed2e",
7        "name": "Python",
8        "attribute_id": "2867e888-7d4f-11ed-8b72-85e72d02c5e0"
9      }
10   ]
11 }
```

Survey offered event

A survey was offered to a set of responders.

Survey offered events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>SurveyOffered</code>
assigned_user_id	integer	yes	The id of the user assigned to the ticket at the time the survey was offered. In case there is no user assigned at that time, the value of this field is <code>null</code>
assigned_group_id	integer	yes	The id of the group assigned to the ticket at the time the survey was offered. In case there is no group assigned at that time, the value of this field is <code>null</code>
survey_id	string	yes	A ULID representing the survey that was offered
survey_type	string	yes	The type of the survey that was offered

Example

```
1  {
2    "id": 11654650767636,
3    "type": "SurveyOffered",
4    "assigned_user_id": 123123,
5    "assigned_group_id": 30768,
```

```
6  "survey_id": "01J8M766YV6FTKCCBQ0Z85YND5",
7  "survey_type": "CustomerSatisfaction"
8  }
```

Survey response submitted event

A survey response was submitted by one of the responders.

Survey response submitted events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value SurveyResponseSubmitted
assigned_user_id	integer	yes	The id of the user assigned to the ticket at the time the survey response was submitted. In case there is no user assigned at that time, the value of this field is null
assigned_group_id	integer	yes	The id of the group assigned to the ticket at the time the survey response was submitted. In case there is no group assigned at that time, the value of this field is null
survey_response_id	string	yes	A ULID representing the survey response that was submitted
survey_type	string	yes	The type of the survey that was offered

Example

```
1  {
2    "id": 11654650767636,
3    "type": "SurveyResponseSubmitted",
4    "assigned_user_id": 123123,
5    "assigned_group_id": 30768,
6    "survey_response_id": "01J8M766YV6FTKCCBQ0Z85YND5",
7    "survey_type": "CustomerSatisfaction"
8  }
```

Offered To event

The ticket was offered to an agent by omnichannel routing. If using [skill-based routing](#), the audit event also contains the list of skills used to match the ticket to the agent.

Note: Offered To events are not logged for voice channel tickets.

Offered To events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value OfferedToEvent
assignee	integer	yes	The id of the agent the ticket was offered to
skills	array	yes	Skills used to route the ticket to an agent. The ticket might have other skills that timed out during the routing process. <code>attribute_id</code> is the id for the skill type

Example

```
1  {
2    "id": 8307193253373,
3    "type": "OfferedToEvent",
4    "assignee": 199404753,
5    "skills": [
6      {
7        "id": "189b8a13-2926-46b1-a6de-5b482183d4b2",
8        "name": "english",
9        "attribute_id": "ca712d80-232d-11ef-8f9d-55bad2e0a8c4"
10     },
11     {
12       "id": "7ab97a79-5f9b-425c-9e39-d4a9d3d143c6",
13       "name": "flemish",
14       "attribute_id": "ca712d80-232d-11ef-8f9d-55bad2e0a8c4"
15     }
16   ]
17 }
```

Routing channel event

The routing channel of the ticket was changed.

The routing channel is used by Omnichannel Routing to determine which channel the ticket is to be routed in. When a ticket is created, the routing channel of the ticket is determined by the [via type](#).

While the via type of a ticket does not change, the routing channel can change according to some action or setting. For example, in Omnichannel Routing settings, if the option to change agent-ended messaging sessions to email tickets is enabled, the routing channel can change from Messaging to Email.

Routing channel events have the following properties:

Name	Type	Read-only	Comment
id	integer	yes	Automatically assigned when the event is created
type	string	yes	Has the value <code>RoutingChannelEvent</code>
previous	string	yes	The previous routing channel
current	string	yes	The current routing channel
reason	string	yes	The reason for the routing channel change. Possible values: <code>SUPPORT_TRIGGER_ACTION_TRANSFORM_TO_EMAIL</code> , <code>AGENT_END_SESSION</code> , <code>UNKNOWN</code>

Example

```
1  {
2    "id": 1274,
3    "type": "RoutingChannelEvent",
4    "previous": "MESSAGING",
5    "current": "SUPPORT",
6    "reason": "AGENT_END_SESSION"
7  }
```

Join our developer community

 [Forum](#)  [Blog](#)  [Slack](#)

Zendesk 181 Fremont Street, 17th Floor, San Francisco, California 94105