

# How to Impersonate a Comment in Jira On-premise

Last Modified on 02/11/2026 9:29 am EST

By default, Exalate adds a comment, received from the other side on behalf of [the proxy user](#). The comment will look like the following:

---

▼  [Exalate](#) added a comment - 1 hour ago

Administrator (blue) commented:  
Have a look at following solution

- Install exalate
- Configure relation
- Done

---

You can change this behavior. For example, create the received comment with the original comment author. The [comment object](#) has a field **Executor** which can be set to any user. This is how you can impersonate synced comments.

## Source Side

### Outgoing Sync

To send comments to the destination instance use the following script:

```
replica.comments = issue.comments
```

## Destination Side

You can apply the received comment in different ways:

1. Assign the original comment author if exists in the local system and create a new user if not found.

**Note:** This method works only for Jira Server.

### Incoming sync

- Check if the author of the comment exists in the local system by doing a user lookup
- If the author does not exist in the local JIRA, create a new user
- Assign this user as a comment executor

```

issue.comments = commentHelper.mergeComments(issue, replica, {
  comment ->

    def authorUser = userHelper.getByEmail(comment.author.email)

    // set the executor, optionally create the user if the local user is not found
    comment.executor = authorUser ?: userHelper.createUser(comment.author.username,
      "changeme", //password !
      comment.author.email,
      comment.author.displayName
    )

  comment
})

```

2. Assign the original comment author if exists in the local system, use the proxy user if not found.

**Note:** This method works for Jira Server and Jira Cloud.

## Incoming Sync

- Change the comment executor from the proxy user to the original comment author
- If the author does not exist in the local JIRA, use the proxy user as an author

```

replica.addedComments.each { it.executor = nodeHelper.getUserByEmail(it.author?.email) }
replica.changedComments.each { it.executor = nodeHelper.getUserByEmail(it.updateAuthor?.email) }
issue.comments = commentHelper.mergeComments(issue, replica, { it })

```

## Result

The first comment was created by the proxy user, while the second comment was impersonated and the local administrator is the comment executor.

1  Exalate added a comment - 13 minutes ago - [edited](#)  
This comment has been created by the administrator on the blue tracker, but is not impersonated

2  administrator added a comment - Now - [edited](#)  
This comment has been created by the administrator on the blue issue tracker.

### ON THIS PAGE

[About Us](#) 

[Release History](#) 

[Glossary](#)  [Destination Side](#)  
[API Reference](#) 

[Security](#) 

[Pricing and Licensing](#) 

### Resources

[Subscribe for a weekly Exalate hack](#) 

[Academy](#) 

[Blog](#) 

[YouTube Channel](#) 

[Ebooks](#) 

### Still need help?

[Join our Community](#) 

[Visit our Service Desk](#) 

[Find a Partner !\[\]\(cead67df4d82d6c83effe4f8699a7d8f\_img.jpg\)](#)