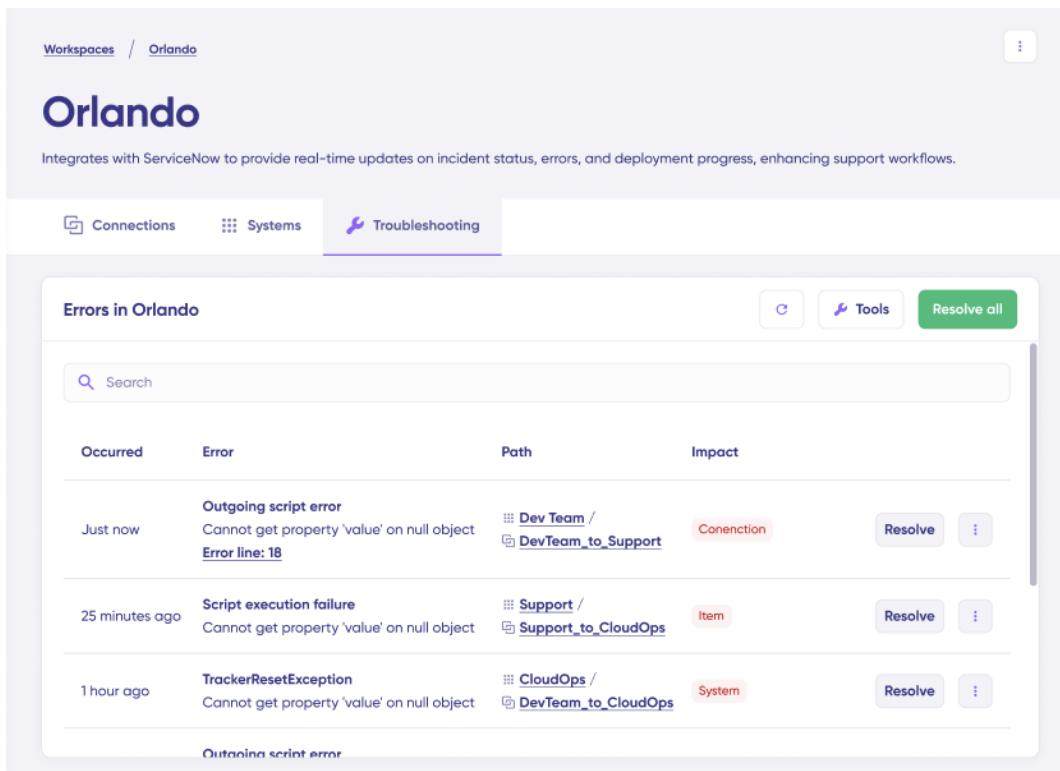


Why is an item not syncing?

Last Modified on 02/17/2026 10:10 am EST

Exalate Troubleshooting Section

New Exalate includes a dedicated [Troubleshooting section](#) that displays errors preventing synchronization. The unified console shows recent errors with timestamps, error types, affected connections, and specific failure messages like "Outgoing script error: Cannot get property value on null object, Error line 18". This centralized error view makes it significantly easier to identify sync problems compared to checking separate consoles in Exalate Classic.



The screenshot shows the Exalate interface for the 'Orlando' workspace. The left sidebar includes 'Workspaces', 'Users', 'Settings', and a 'Resources' section with a dropdown. The main area shows the 'Orlando' workspace with a brief description: 'Integrates with ServiceNow to provide real-time updates on incident status, errors, and deployment progress, enhancing support workflows.' Below this are tabs for 'Connections', 'Systems', and 'Troubleshooting', with 'Troubleshooting' being the active tab. A sub-section titled 'Errors in Orlando' lists three errors:

| Occurred | Error | Path | Impact | Actions |
|----------------|---|-----------------------------------|------------|--|
| Just now | Outgoing script error Cannot get property 'value' on null object Error line: 18 | Dev Team / DevTeam_to_Support | Connection | Resolve More |
| 25 minutes ago | Script execution failure Cannot get property 'value' on null object | Support / Support_to_CloudOps | Item | Resolve More |
| 1 hour ago | TrackerResetException Cannot get property 'value' on null object | CloudOps / DevTeam_to_CloudOps | System | Resolve More |

At the bottom of the error list, there is a note: 'Outgoing script error'.

Aida AI Diagnosis for Errors

Aida AI provides intelligent diagnosis when errors occur, explaining issues in easy-to-digest language with context-aware suggestions for resolution. When you encounter a script-related error, Aida analyzes it and explains it. You can view the full analysis and receive guidance on fixing the problem directly within the troubleshooting interface.

Orlando

Integrates with ServiceNow to provide real-time updates on incident status, errors, and deployment progress, enhancing support workflows.

Connections Systems Troubleshooting

Errors in Orlando

Search

| Occurred | Error | Details | Actions |
|-----------------------|--|---|-------------------------|
| Just now | Outgoing script error Cannot get property 'value' on null object Error line: 18 | <p>Dev Team / DevTeam_to_Support</p> <p>Support / Support_to_CloudOps</p> <p>CloudOps / DevTeam_to_CloudOps</p> | Resolve |
| 25 minutes ago | Script execution failure Cannot get property 'value' on null object | <p>Support / Support_to_CloudOps</p> | Resolve |
| 1 hour ago | TrackerResetException Cannot get property 'value' on null object | <p>CloudOps / DevTeam_to_CloudOps</p> | Resolve |
| Outgoing script error | | | |

John Doe
System Admin

Logout

Aida Diagnosis

The error message you're seeing indicates a script-related issue: Cannot get property 'value' on null object. This means there's likely an attempt in your script to access a property on an object that doesn't exist or hasn't been initialized....

[View Full Analysis >](#)

Resolve all

Outgoing script error

Just now

Resolve X

Aida Diagnosis

The error "Cannot get property 'value' on null object" in your outgoing sync script means that the script is trying to access a property (like .value) on an object that doesn't exist (it's null). This typically happens when a custom field or property you're referencing isn't present on the issue being synced.

Here's how to fix it:

1. Check your outgoing sync script, especially around line 18 (as mentioned in the stack trace: Script13.groovy:18).
2. Look for any line trying to access .value on a field, such as issue.customFields."SomeField".value.
3. The field "SomeField" may not exist on the issue, meaning issue.customFields."SomeField" is null.

To prevent this error, use Groovy's null-safe operator (?) like this:

```
def fieldValue = issue.customFields."SomeField"? .value
```

This ensures that if the field doesn't exist, it won't throw an error.

After updating your script, retry the synchronization. If you want to debug further, you can add a line like:

```
debug.error("Field value is: " + issue.customFields."SomeField")
```

This will help you see what value (if any) is being passed.

Let me know if you need help finding the exact line or want to share your script for a quick review!

Copy

Common Reasons for Sync Failures

Common reasons for sync failures include:

- script execution errors
- trigger configuration issues
- or missing field mappings

The [Product](#) [error list](#) categorizes problems by type such as "Outgoing script error," "Script execution failure," "Network," and "TrackerResetException," helping you quickly identify whether the issue stems from your [configuration](#) or system connectivity. Each error entry shows which connection and [specific](#) nodes are affected, making it easy to pinpoint where to focus your troubleshooting efforts! [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](#)