



P R O C E S S M A N U A L

Document Code: PM-SLI

Slack

Integration

Handle Requests for administrator access via Slack.

 **FastTrack** Software

 **Admin** By Request

Table of Contents

Introduction	3
Assumptions	3
Prerequisites	3
Breakdown of Tasks.....	4
Integration Tasks.....	5
Task A: Create a Channel.....	5
Task B: Add and Connect Slack App.....	7
Task C: Manage Requests.....	10

Introduction

Slack is a versatile solution for workforce communication, bringing together instant messaging, file-sharing, and third-party applications.

We've created a custom Admin By Request app for Slack which enables users to handle (i.e., approve, deny, and view) Requests for administrative access from within a dedicated Slack channel. This manual provides a step-by-step guide on how to configure the application and integrate a core Admin By Request feature into your Slack workspace.

Assumptions / Limitations

The tasks described in this manual assume that the user has access to Slack, the Admin By Request User Portal, and some familiarity with both environments.

IMPORTANT: The access provided to users through our integrations **override User Portal settings**. Once the Slack integration described in this manual is configured, all users with access to the Slack channel created in Task A will have the ability to approve or deny requests via Slack, regardless of whether they have been granted these abilities in your Admin By Request User Portal Sub-Settings. Ensure you only add the integration to internal channels and reserve access for authorized users.

Prerequisites


To enable this integration, you must first obtain your Admin By Request API Key. This key can be self-generated through your Admin By Request User Portal via **Settings > [OS] Settings > Data > API**:

The screenshot shows the Admin By Request User Portal interface. The top navigation bar includes links for Summary, Auditlog, Requests, Reports, Inventory, Settings (highlighted with a red arrow), Download, Logins, Docs, and Contact. The main content area is titled 'Windows Workstation Global Settings'. On the left, a sidebar lists various settings categories, with 'Data' selected and highlighted by a red arrow. The 'Data' section contains a sub-menu with 'API' selected, also highlighted by a red arrow. The 'API' settings page shows a toggle for 'API access' set to 'ON' (circled in red) and an 'API Key' field containing a long alphanumeric string (circled in red). A red arrow points to the API Key field. Below the key field is a 'Save' button. To the right, the 'About API Access' section provides additional context and a link to documentation.



IMPORTANT: Click the **Save** button after Regenerating an API Key, to ensure this is the key used to establish the connection to Slack. A green tick icon will appear next to the **Save** button when the action is complete:



 **NOTE:** The API Key has been blurred out in the above example.

Breakdown of Tasks

Seven tasks are covered in this manual:

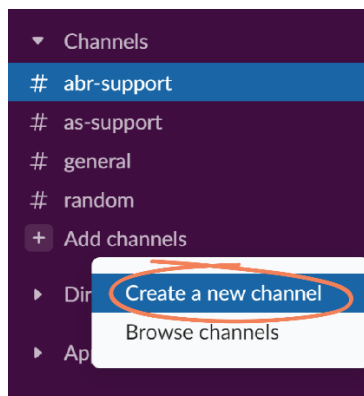
1. Task A: Create a Channel
2. Task B: Add and Connect Slack App
3. Task C: Manage Requests

Integration Tasks

Task A: Create a Channel

This integration requires a dedicated Admin By Request Slack channel for Requests to be sent to, and managed from.

1. In left-hand menu in Slack, under *Channels*, select **+ Add channels > Create a new channel**:



2. In the *Create a channel* pop-up, give the channel your desired name (required) and a description (optional). Switch the *Make private* toggle **on**, and click **Create**:

A screenshot of the 'Create a private channel' dialog box in Slack. The title is 'Create a private channel' with a close button (X) in the top right. Below the title is a brief explanation: 'Channels are where your team communicates. They're best when organized around a topic — #marketing, for example.' The form has two input fields: 'Name' with the value 'abr-requests' and a lock icon, and 'Description (optional)' which is empty. Below the description field is the text 'What's this channel about?'. There is a 'Make private' section with a toggle switch that is turned on (green) and the text 'This can't be undone. A private channel cannot be made public later on.' At the bottom, there is a checkbox for 'Share outside FastTrack Softwa...' (partially visible), a 'PREMIUM' badge, and a green 'Create' button which is circled in red.

IMPORTANT: Ensure you set the *Make private* toggle to **on** for this channel, otherwise all members of the organization will have the ability to approve or deny Requests.

NOTE: In this example, our channel is named *abr-requests*.

3. Clicking Create displays an *Add people to* pop-up. In this example we **Skip for now**:



Add people to  **abr-requests** ×

ex. Nathalie, or james@acme.com

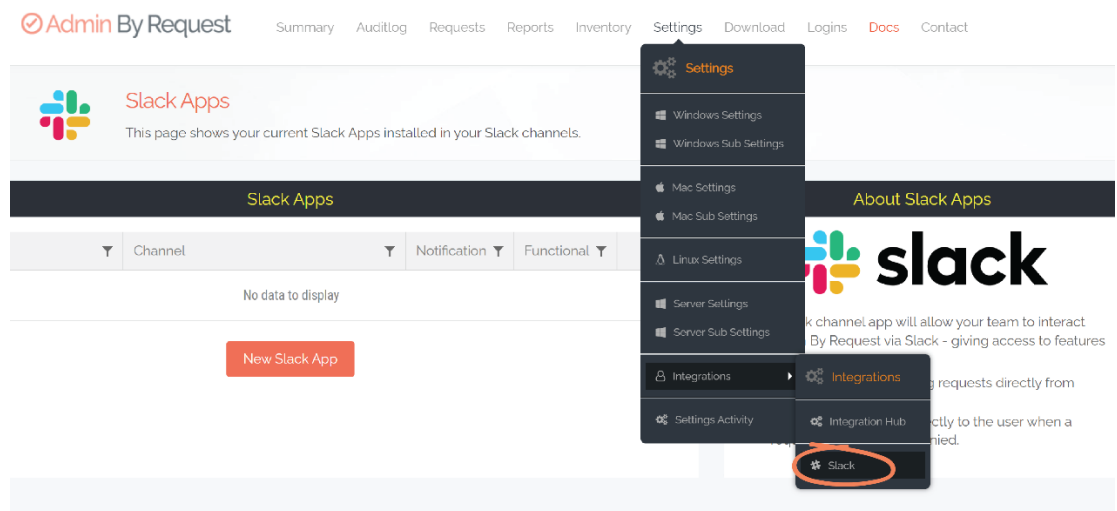
Skip for now

! **IMPORTANT:** Remember that this integration **overrides User Portal settings**, so all users with access to the Slack channel created above will have the ability to approve or deny requests via Slack, regardless of whether they have been granted these abilities in your Admin By Request User Portal Sub-Settings.

Task B: Add and Connect Slack App

The app used in this integration comes preconfigured and ready to 'plug-in-and-go'. This Task involves adding the app from you Admin by Request User Portal and allowing it to form a connection to the Slack channel.

1. In your Admin By Request User Portal, navigate to **Settings > Integrations > Slack**:



2. Select **New Slack App**:



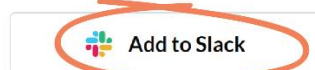
3. Click the **Add to Slack** button:


Install Admin By Request Slack app

Click the button below to install the Admin By Request app into one of your Slack channels.

The app will allow your team to interact with Admin By Request via Slack - giving access to features like:

- Approve or deny incoming requests directly from Slack.
- Send a chat message directly to the user when a request is approved or denied.





 **NOTE:** The page that opens describes the nature of the Admin By Request / Slack integration – what the app is able to view and do. Click the drop-down arrows for more details:



ABR Requests is requesting permission to access the FastTrack Software APS Slack workspace

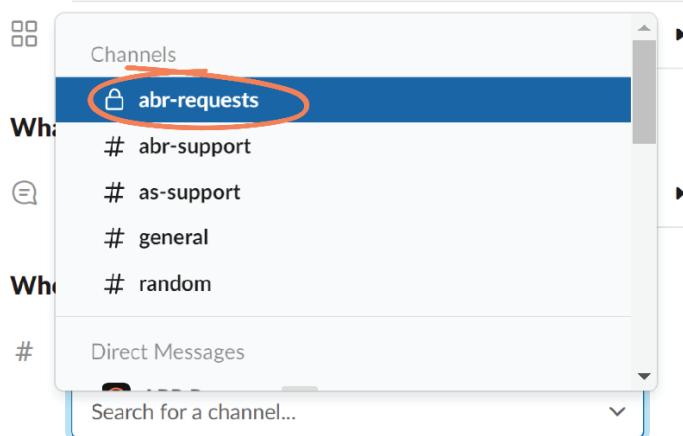
What will ABR Requests be able to view?

-  Content and info about channels & conversations ▶
-  Content and info about your workspace ▶

What will ABR Requests be able to do?

-  Perform actions in channels & conversations ▶

4. Under *Where should ABR Requests post?*, click the drop-down arrow and select the Slack channel created in step 2 – in this case, *abr-requests*:



5. Click **Allow**:

Where should ABR Requests post?

ABR Requests requires a channel to post to as an app

 abr-requests ▼

Cancel 

6. In the next window, paste your Admin By Request API Key (see the [Prerequisite](#) section of this document) into the *API Key* field and ensure the *Send new requests to Slack channel* toggle is set to **on**. The second toggle (*Attempt to send chat message when a request has been handled*) is optional, but in this demonstration, we will switch it **on**:

Enter your Admin by Request API key

In order to verify your Admin by Request account, please enter your API key below.

You can find your key in the [API Access section](#).

API Key

- ☒ Send new requests to Slack channel
- ☒ Attempt to send chat message when a requests has been handled



NOTE: Switching the second toggle on sends a Slack message of the outcome of the Request to the user who made it.

7. Click **Continue installation**, and wait for the connection to be established:

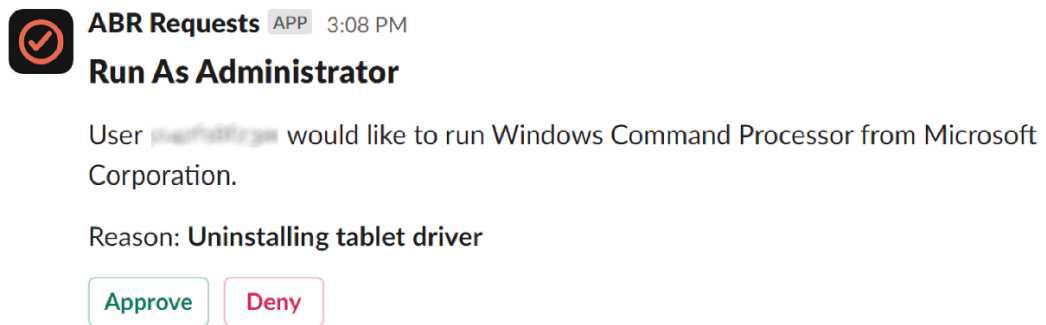
Continue installation

Cancel

Task C: Manage Requests


This Task demonstrates how to use the new channel and app to handle and keep track of Requests in Slack.

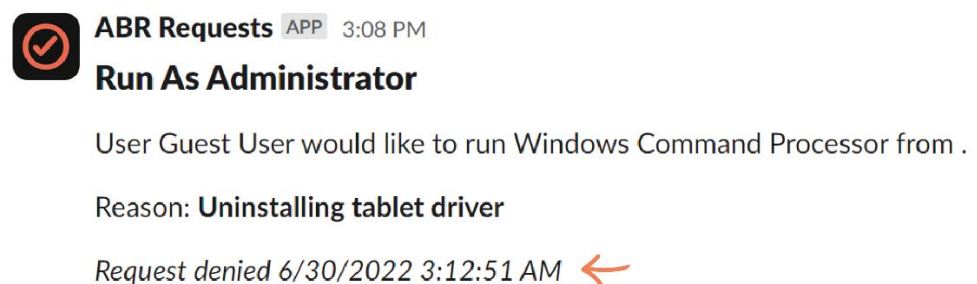
1. New / pending Requests now appear in Slack, in our newly created channel (i.e., *abr-requests*), displaying the time the request was made, the name of the user, and the reason provided for making the Request (if the Reason feature is enabled in your ABR settings):



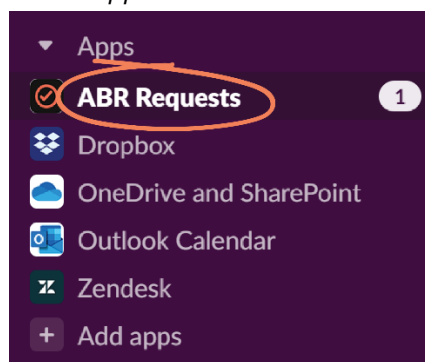
2. Click **Approve** or **Deny** – the same way you would in the Requests page of your Admin By Request User Portal. In this example we **Deny** the request:



 **NOTE:** The chosen action is reflected in the channel, and, if configured in Task B, step 6, a Slack notification of the outcome is displayed to the user:




3. Under *Apps* in the left-hand menu, select the **ABR Request** app:



4. All previous Requests and the action taken on them by the admin are logged here, listing the date, time, and outcome for each one:

Today ▾

New

 **ABR Requests** APP 3:12 PM
Your request to run Windows Command Processor has been denied
