

Integration Manual

Slack V2

Document Information

Code: **IM-Slack**

Version: **2.0**

Date: **8 May 2026**

Copyright © 2026 Admin By Request

All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement (NDA). The software may be used or copied only in accordance with the terms of those agreements.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the customer's stated use without the written permission of Admin By Request.

Contact Admin By Request



+64 21 023 57020



marketing@adminbyrequest.com



adminbyrequest.com



Unit C, 21-23 Elliot St, Papakura, NZ

Table of Contents

Overview (V2)	1
Introduction	1
Assumptions / Limitations	1
Prerequisites	2
Something Missing?	2
Related Documents	3
Integration Tasks (V2)	4
Introduction	4
A. Create a Channel	4
Planning your channel layout	4
B. Install the Integration	5
Installing to additional channels	7
Updating settings	7
C. Manage Requests	8
Receiving requests	8
Approving or denying a request	8
Multi-channel synchronization	9
Requests handled outside of Slack	9
User notification	9
Document History	10

Overview (V2)

Introduction

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

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 guide provides a step-by-step guide on how to configure Version 2 of the application and integrate it into your Slack workspace.

Version 2 of the Slack integration introduces several significant improvements over V1:

- **Sub Settings Name scoping** - channels can be configured to receive only requests that match a specific Sub Settings Name, enabling targeted routing across multiple channels within the same workspace.
- **Multi-channel synchronization** - when a request is handled in any channel (or in the Admin By Request portal), all other channels that received the same request automatically update their messages to reflect the outcome and where it was handled.
- **Portal-handled notification** - if a request is approved or denied in the Admin By Request portal or through any other integration rather than through Slack, all Slack channels are updated to indicate it was handled outside of Slack.
- **Run As Admin program details** - Run As Admin request messages now include the program version, program path, and scan result alongside the standard request information.
- **Deny reason** - when denying a request from Slack, a deny reason can be entered; it is recorded in the audit log.
- **Approver identity** - the Slack message now shows who approved or denied the request. If it was handled in Slack, the approver's Slack display name is shown. If it was handled in the portal, the portal admin's username is shown.
- **Auditlog attribution** - if the Slack user's email matches a portal admin's email, the portal admin's name is written to the auditlog. If no match is found, the "Approved by" field is omitted from the auditlog entry.
- **User notification** - optionally, the requesting user receives a direct message in Slack when their request is handled, provided their endpoint email address matches their Slack account email.

Assumptions / Limitations

The tasks described in this guide 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 **overrides User Portal settings**.

Once the Slack integration described in this guide is configured, all users with access to the Slack channel(s) 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.

Prerequisites

Before setting up the integration, the following need to be in place:

1. A **Slack workspace** that the organization uses for internal communication. The integration is installed into a specific workspace, and the channel(s) created for requests must belong to that workspace.
2. An **ABR API key**. API keys are generated in the ABR portal under **Settings > Tenant Settings > Data > API KEYS**. Create a new key, give it a descriptive name ("Slack" is fine), and click the **Save** button to activate it.

Each integration should have its own dedicated API key - this makes it easier to revoke a specific connection without affecting others.

Existing API Keys for your tenant can be found in the portal under **Settings > Tenant Settings > Data > API KEYS**. Your API prefix (data center) is shown at the bottom of that page under *About API Keys*.

3. **Access to the ABR portal** with sufficient permissions to configure integrations (the *Settings* permission for a user in **Logins > User Logins > [User]**).
4. A **Channel plan** for your environment. Decide whether to start with a single global channel (which will receive all requests from all endpoints) or multiple channels with sub-setting scoping (V2 only). This decision affects how you structure the setup.

IMPORTANT: Upgrading from V1

V1 and V2 of the Slack integration are independent and cannot be updated in place. To use V2, you must remove the existing V1 integration and complete the V2 installation described in this guide.

V1 will continue to function until it is removed. You can run both integrations in parallel during a transition period if needed.

Something Missing?

If you've identified a bug or have a suggestion for this integration, or another SIEM integration you'd like us to add, contact us [here](#) and we'll see what we can do.

NOTE

The task descriptions in this guide (and screenshots in particular) cover the state of Slack V2 at the time of writing. While every effort is made to ensure currency, the screens you see during setup may look a little different, especially color schemes and the placement of buttons and links.

Related Documents

This guide may refer to, and should be read in conjunction with, the following:

- Commitments and responsibilities in ABR's [Data Processing Agreement](#)
- Support provisions in ABR's [Terms and Conditions](#) and [Customer Support Services](#)
- Collection, use and disclosure of personal data in ABR's [Privacy Policy](#) and [Data Privacy Settings](#)

Refer also to ABR's [Trust Center](#) documents.

This guide is available online:



[Slack Integration Manual](#)

Integration Tasks (V2)

Introduction

The following tasks are covered in this section:

"A. Create a Channel" below

"B. Install the Integration" on the next page

"C. Manage Requests" on page 8

A. Create a Channel

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

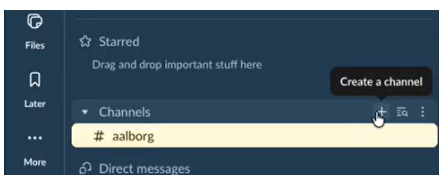
Planning your channel layout

- A channel with no Sub Settings Name set receives all requests — from both global settings and all sub-settings. This is the simplest configuration and suitable for most deployments.
- A channel with a specific Sub Settings Name set will only receive requests whose settings name matches. Requests from global settings are not sent to a scoped channel.
- You can have multiple channels with no scope (catch-all), multiple scoped channels, or a combination of both. When a request matches a channel's scope, it is sent there — and all matched channels remain synchronized. See Task C for details.

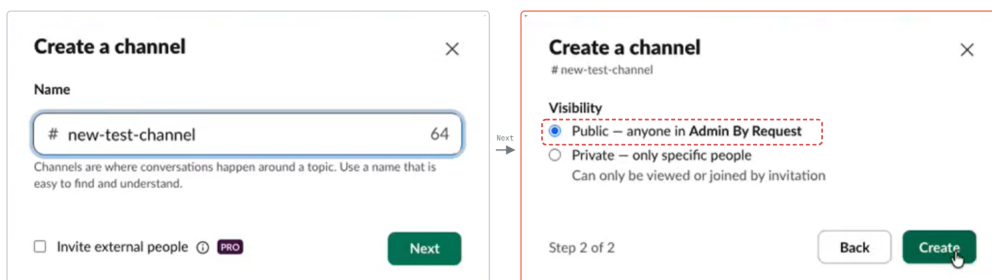
IMPORTANT: Channel must be public during installation

In V2, the integration cannot be installed directly to a private channel. You must install to a public channel first, and then change the channel to private in Slack after the installation is complete.

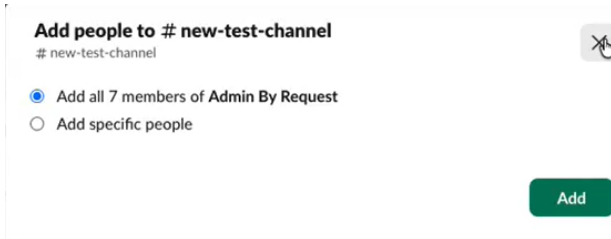
1. In the left-hand menu in Slack, under **Channels**, click **+** to **Create a channel**:



2. In the *Create a channel* pop-up, give the channel your desired name and click **Next**. At Step 2, leave the **Public** toggle selected for now, and click **Create**:



- Skip adding members for now. You will add members after the installation is complete:



- If you plan to route requests to multiple channels (for example, by Sub Settings Name), repeat steps 1–3 to create each additional channel. You will configure the scope of each channel in Task B.

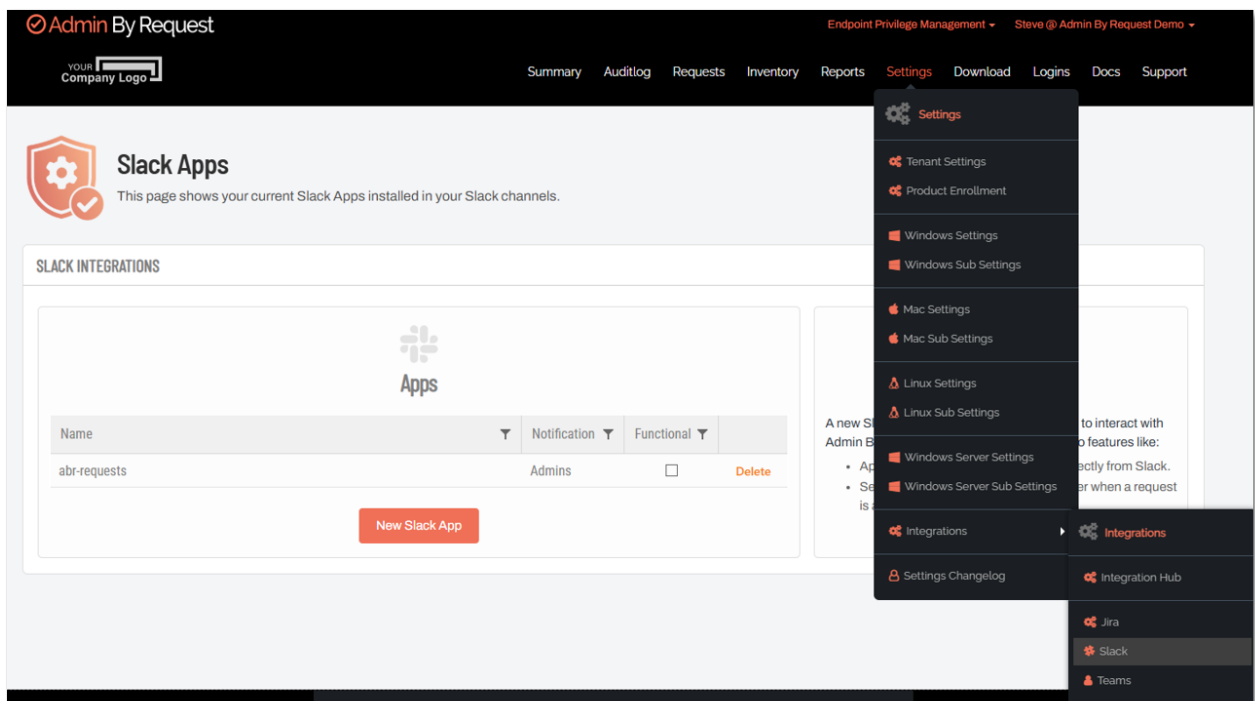
IMPORTANT

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

B. Install the Integration

The V2 integration is installed via a web-based setup flow, accessed from your Admin By Request User Portal. Each channel requires its own installation run. The same API key can be reused across all channels.

- In your Admin By Request User Portal, navigate to **Settings > Integrations > Slack** and click **New Slack App**:



- On the *Install Admin By Request app for Slack* page, click button **Add to Slack**.
- You will need to sign-in to Slack at this point (if you are not already).

4. Enter your Admin By Request **API Key** (which you should already have from "[Prerequisites](#)" on [page 2](#)) in the field provided:

SLACK INTEGRATION - V2

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

If defined, the channel will only receive requests with a matching Sub Settings Name.
If left blank, all requests will be forwarded to the channel.

(Optional) Sub Settings Name

Attempt to send chat message when a requests has been handled

[Continue installation](#) [Cancel](#)

The same API key can be reused when installing the integration to additional channels.

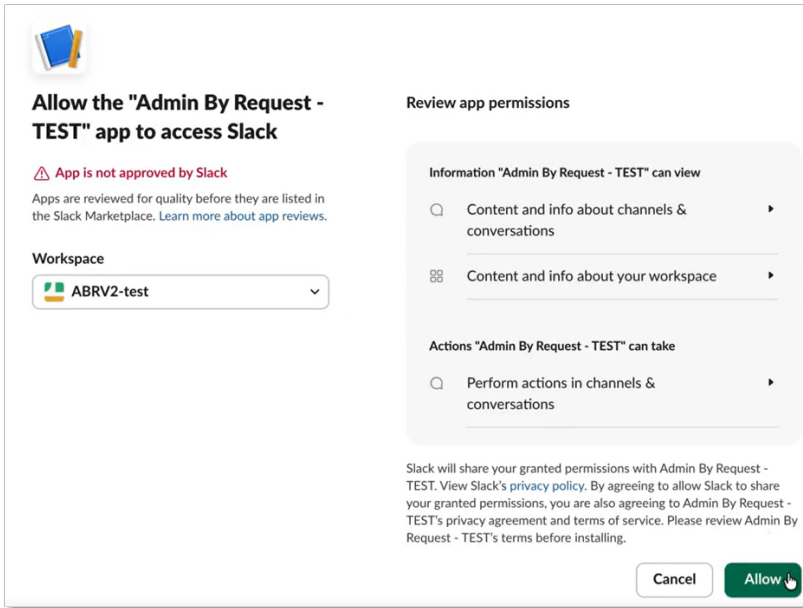
5. (Optional) Enter a **Sub Settings Name** for this channel.

Sub Settings Name

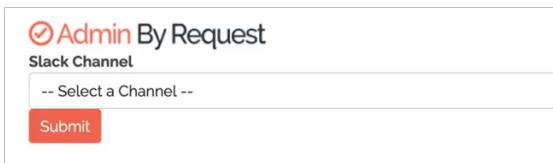
The Sub Settings Name controls which requests are routed to this channel. It must match the `settingsName` value returned in the Admin By Request Audit Log API for a request to be delivered here.

- Leave this field blank (or enter *Global*) to receive all requests, including those from sub-settings and global settings.
- Enter a specific sub-settings name (e.g., *Finance*) to receive only requests originating from that sub-setting. Requests from global settings will not be sent to this channel.
- To confirm the correct Sub Settings Name to use, check your audit log entries - the `settingsName` field in the JSON payload is the value to enter here.

6. Ensure the **Send new requests to Slack channel** toggle is set to **on**.
When this toggle is off, incoming requests are logged in the integration's internal database but are not posted to the Slack channel.
7. (Optional) Set the **Attempt to send chat message when a request has been handled** toggle to **on**.
When enabled, the requesting user receives a direct message in Slack notifying them of the outcome of their request. The message is only sent if the user's endpoint email address matches their Slack account email address.
8. Click **Continue Installation**.
9. When prompted, sign in to Slack and choose the **workspace** in which you created the channel in Task A, then review the permissions requested by the Admin By Request app and click **Allow**:



- From the channel list, select the channel created in Task A and click **Submit**:



NOTE

Only public channels are shown in this list. If the channel you created does not appear, confirm it is currently set to public in Slack.

- The installation is now complete for this channel. To make the channel private, navigate to the channel settings in Slack and enable the **Make private** option.

IMPORTANT

Once a Slack channel is changed from public to private, this action cannot be undone. Ensure you have completed the installation and verified it is working before converting the channel to private.

Installing to additional channels

To install the integration to additional channels, repeat steps 1–11 for each channel, selecting a different channel in step 10. Use the same API key for all channels.

Updating settings

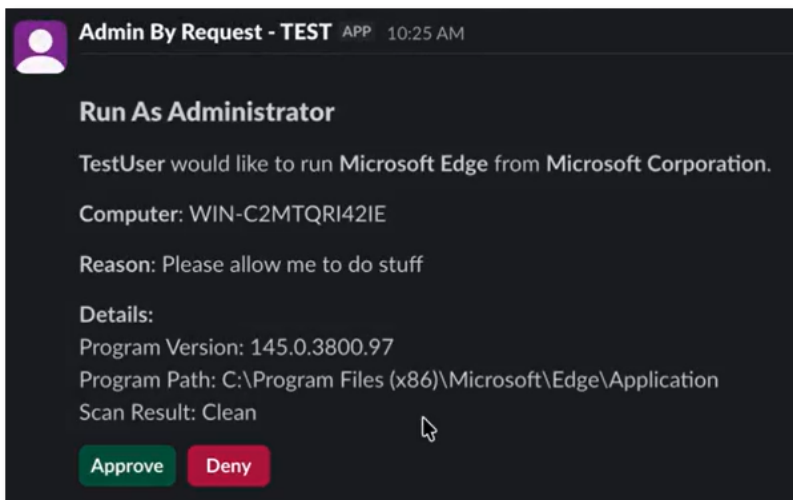
To update the settings (for example, to change the Sub Settings Name or toggle states) for an existing channel, repeat the installation steps using the same API key and select the same channel in step 10. The existing configuration will be updated.

C. Manage Requests

This task demonstrates how to use the Slack channel(s) to handle and keep track of requests made by your users.

Receiving requests

1. New and pending requests appear in the appropriate Slack channel(s) as a message, displaying the request type, the time the request was made, the name of the user, the computer name, and the reason provided (if the Reason feature is enabled in your ABR settings):



A request is delivered to every channel whose Sub Settings Name matches the request's `settingsName`. Channels with no scope set (or with "Global" as their value) receive all requests. Channels scoped to a specific sub-setting receive only requests from that sub-setting.

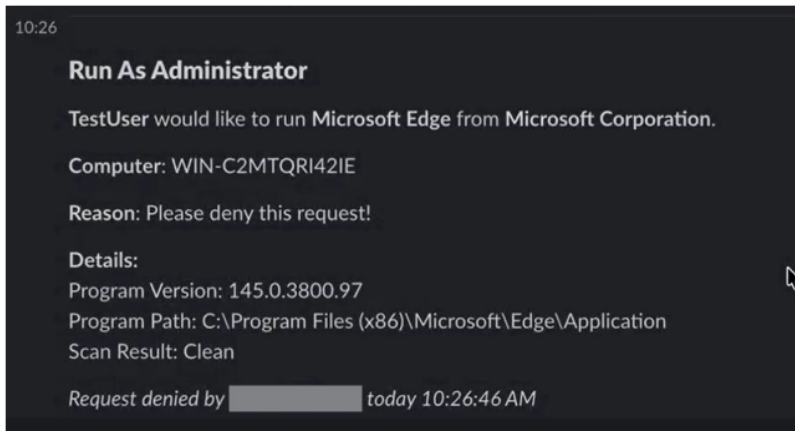
NOTE

For *Run As Admin* requests, the Slack message also displays the **Program Version**, **Program Path**, and **Scan Result** for the application being elevated. *Admin Session* requests display the standard request information only.

Approving or denying a request

1. Click **Approve** or **Deny** on the request message - the same way you would in the Requests page of the Admin By Request portal.
2. If denying, you may optionally enter a **Deny Reason**. The deny reason is recorded in the Admin By Request audit log for the request.

The action taken is reflected in the channel message, along with the name of the person who handled the request and the time it was handled:



Multi-channel synchronization

If the same request was delivered to more than one Slack channel, approving or denying it in any channel automatically updates the messages in all other channels. Each updated message reflects the outcome and identifies who handled the request and where.

Handling attribution

The message displays the Slack display name of the user who handled the request.

In the Admin By Request audit log, if the Slack user's email address matches a portal admin's email address, the portal admin's name is recorded as the approver. If no match is found, the "Approved by" field is omitted from the audit log entry.

Requests handled outside of Slack

If a request is approved or denied in the Admin By Request portal or through any other integration (rather than through Slack), all Slack channels that received the request message are updated to indicate it was handled outside of Slack, along with the portal admin's username and the time of the action.

User notification

If the **Attempt to send chat message when a request has been handled** option was enabled during installation (Task B, step 4), the requesting user receives a direct message in Slack when their request is approved or denied. This message is only sent if the user's endpoint email address matches their Slack account email address.

Document History

Version	Author	Changes
June 2022 1.0	Sophie Dodson	Initial document release
8 May 2026 2.0	Steve Dodson	Updates for V2 integration: <ul style="list-style-type: none">• New V2 features: Sub Settings Name channel scoping, multi-channel synchronization, portal-handled notification, Run As Admin program details (version, path, scan result), deny reason, approver identity display, auditlog attribution, and user DM notification on outcome.• Revised installation flow: web-based OAuth setup (accessed via portal), channel must be public during installation and can be converted to private afterward, multiple channels supported by repeating the installation process.