

Document Code: **PM-LC-ITAG**

Document Version: **1.1**

Document Date: **7 August, 2023**

Linux Client: IT Admin Guide

Configure, deploy and manage your Linux workstations

 **Admin** By Request

Linux Product Version: **3.0**



Copyright © 2023 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

1390 Market Street, Suite 200
San Francisco, CA 94102

Phone and Email:
adminbyrequest.com/contact

www.adminbyrequest.com

Table of Contents

1

INTRODUCTION	5
In This Document	5
Audience	5
Product Release Notes	5

2

INSTALLING AND UNINSTALLING	6
Prerequisites	6
Installing Admin By Request	6
Upgrading Admin By Request	8
Uninstalling Admin By Request	8
User rights after installation	9
Tamper Prevention	9
Performance after Installation	9
File Locations	9

3

THE USER INTERFACE	10
About Admin By Request	10
Requesting Administrator Access	14
Using Run As Admin	17

4

PORTAL ADMINISTRATION FOR LINUX	18
Pre-Approval	18
Blocked Applications	18
Run As Admin	19
Machine Learning	20

Supplementary Technical Information	20
<i>Local Administrator Accounts</i>	20
<i>Sub-Settings</i>	20
<i>Sudo</i>	21
<i>Tampering</i>	21
Changing Admin Session Duration	21
5	
POLICIES FOR LINUX	22
About Policies	22
Overruling Portal Settings	22
Appendix	
TERMS AND DEFINITIONS	26
Privileged Access	26
Glossary	28
Document History	30
Index	31

1

Introduction

Admin By Request's Privileged Access Management (PAM) solution is designed to solve the security and productivity challenges relating to Local Administration rights usage within today's security conscious and highly distributed enterprises.

Employees achieve optimum productivity by using secure methods to safely elevate everyday trusted tasks. IT departments achieve significant time and resource savings as employee requests for elevation are offloaded and routed through streamlined, fully audited and automated workflows.

This guide describes key IT administrator concepts and tasks related to installing, configuring, deploying, and managing linux endpoints.

In This Document

The content of this guide describes:

- How to install the Admin By Request client on endpoints running Linux.
- How to uninstall Admin By Request.
- The user interface, including screen panels associated with menu selections.
- Key portal administration tasks, specific to Linux.
- Terms and definitions

Audience

The Linux Client: IT Admin Manual is intended for IT system administrators who install and manage user workstations running the Linux operating system and desktop software.

Product Release Notes

Release notes for all product versions are available on the Admin By Request website:

[Resources > Documentation > Release Notes \(Linux\)](#)

2

Installing and Uninstalling

Prerequisites

Admin By Request, version 3.0 supports the following Linux distributions:

- Ubuntu 20.04 LTS
- Ubuntu 22.04 LTS
- Fedora 36
- Red Hat Enterprise Linux (RHEL9)

You will need the following on every workstation that executes the installation client:

- Administrator privileges (e.g., the ability to run sudo).
- Python 3 installed - the installation client is a Python script. This is not required if Admin By Request is downloaded to the workstation as part of an image

NOTE: The installation script uses standard package management features and may install or update some dependencies if necessary. Once installed, future updates to Admin By Request are handled completely by package management.

You will also need valid credentials to access to your Admin By Request online portal at [Admin By Request Portal](#).

Installing Admin By Request

The following installation procedure is in two parts: the first outlines downloading and installing the Admin By Request package, and the second part describes how to test that installation was successful.

Installation steps are grouped into the following tasks:

A. Download and install the Admin By Request package.

1. Download the Linux client from <https://account.adminbyrequest.com/ABRDownload> and store the client file in a suitable temporary location.
2. If you haven't already, start a terminal session and make sure the file is executable:

```
chmod +x 'abr-installer'
```

3. Run the installation script:

```
sudo ./'abr-installer'
```

- When the installation completes, the Admin By Request icon appears in the top right corner of the screen. Click the icon to show details about the client or start an Admin Session.

Installation is now complete.

B. Test the installation.

- At the command line, enter a command that requires elevated privileges (e.g., **sudo apt update**)

The result should be a line explaining that sudo is not allowed by Admin By Request - an admin session is required.

- Log in to the [Admin By Request Portal](#).
- From the portal menu at the top, select **Settings > Linux Settings**.
- Under **AUTHORIZATION**, check the current settings and change any that you wish to test. For example, you might set the *Access time (minutes)* to **5**.
- Return to the Linux workstation and start an admin session:
 - Click the Admin By Request icon in the top right corner of the screen and select **Request administrator access**.
 - Confirm you want to start a session now (answering any questions that might pop up, such as **Reason**).
- Run the sudo command above again and confirm that it works this time.
- You can now finish the admin session or allow it to time out.

You might also want to check the audit log in the portal, to review the details that were logged as part of this admin session:

- From the portal menu at the top, select **Auditlog**.
- Under **ADMIN SESSIONS**, find the name of the logged-in user and expand the drop-down arrow:

The screenshot shows the 'Admin Sessions' interface. At the top, there are tabs for 'RUN AS ADMIN' and 'ADMIN SESSIONS'. The main heading is 'Admin Sessions' with a search box. Below this is a table with columns: User, Computer, Time, Duration, Activity, and Status. A session is listed for user 'Steve Dabson' on 'UBUNTU20' at '01-06-2023 08:16:37' with a duration of '00:02:04' and status 'Finished'. Below the table, there are two sections: 'Contact Information' and 'Execution'. 'Contact Information' shows 'Full name: Steve Dabson', 'User account: sdbson', and 'Approval: Not required'. 'Execution' shows 'Start time: 01-06-2023 08:16:37', 'End time: 01-06-2023 08:18:41', 'Duration: 00:02:04', 'Settings: Global Settings', and 'Trace no: 123257504'. At the bottom, there are two sections: 'Installed or uninstalled software' (No software was installed or uninstalled) and 'Programs executed using elevated privileges' (No files were executed using elevated privileges during the session).

- Note the activity - in the example shown, no software was installed or uninstalled, and no files were executed using elevated privileges during the session.

Upgrading Admin By Request

To upgrade Admin By Request on a Linux endpoint, simply run the standard `:system update / upgrade` commands at the command line:

NOTE: You can either start an Admin Session or execute each `sudo` command via Run As Admin.

1. Start a terminal session.
2. If you're starting an Admin Session and need Admin By Request approval to run `sudo` commands, request it.
3. Once approved, execute the `system update/upgrade` commands:

```
sudo apt update
sudo apt upgrade
```

Upgrading Admin By Request typically changes one or more of the following packages:

- `abr-gui`
- `abr-linux`
- `abr-pam-plugin`
- `abr-polkit-plugin`
- `abr-service`
- `abr-sudo-plugin`

Uninstalling Admin By Request

There are several ways to uninstall Admin By Request on a Linux endpoint, depending on the version currently installed:

A. From GRUB menu (all versions).

1. Shutdown and reboot the computer.
2. Try any of the following:
 - If your computer boots using BIOS, *press and hold down* the **Shift** key while GRUB is loading.
 - If your computer boots using UEFI, press the Escape key (**Esc**) while GRUB is loading.
 - As you're booting the computer, wait for the manufacturer logo to flash from the BIOS. If your computer boots too quickly, you're going to need to do this immediately after powering it on. Quickly press the **Escape** key.

The timing has to be near perfect on some computers, so you may have to press the key repeatedly. If you miss the window, reboot and try again.

3. At the GRUB boot menu, you'll see an entry for "Advanced Options ...". Select it and press **Enter**.
4. Choose the most recent *recovery mode* option and press **Enter**.
5. At the *Password:* prompt, enter the root password (or simply press **Enter** if you haven't yet given the root account any password).

6. Now you can uninstall Admin By Request for Linux by executing the following command:

```
apt -y purge abr-* && apt -y autoremove
```

B. Via root user (version 2.2.3 and earlier).

1. Start a terminal session, then start an Admin Session.
2. If you haven't already, set the root user password:

```
sudo passwd
```

3. Switch to the root user:

```
su
```

4. Execute the following command:

```
apt -y purge abr-* && apt -y autoremove
```

User rights after installation

When a user logs on, the account is downgraded from Admin to Standard User unless:

- You have turned off **Revoke Admins Rights** in the portal settings (**Settings > Lockdown > ADMIN RIGHTS**).
- Also under **Revoke Admins Rights**, the user is in the list of *Excluded accounts*.
- The computer is domain-joined and the user is a domain administrator.

Please refer to [Endpoint software > macOS Client](#) for more information (section *Technical Info*).

Tamper Prevention

When a user initiates an administrator session, the user's role is not actually changed from user to admin. The user is granted all administrator rights, *except* the right to add, modify or delete user accounts. Therefore, there is no case where the user can create a new account or change their own role and become a permanent administrator.

The user also cannot uninstall Admin By Request, as the only program, to keep the administrator session open forever. Furthermore, all settings, configuration and program files are monitored during administrator sessions. If the user tries to remove or change any of the Admin By Request files, these are restored straight away and the attempted activity is logged.

Performance after Installation

When users are not using Admin By Request, it does not consume resources, except for a brief daily inventory and settings check.

File Locations

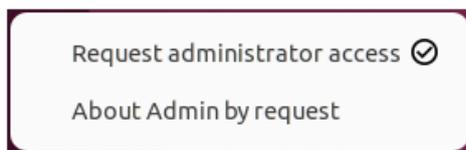
- Executable Files: `/usr/bin`
- Configuration Files: `/etc/abr` and `/usr/share/abr/configuration`
- Log Files: `/var/log/abr`

3

The User Interface

About Admin By Request

The user interface is graphical and is accessed via the icon menu in the top right corner of the screen. Click the icon to display the menu and select a menu option for further information or to carry out an admin task:



Selecting **About Admin By Request** shows the *About Admin By Request* panel:

- **About** – displays this panel, including current workstation edition, license details, website link, and copyright information:

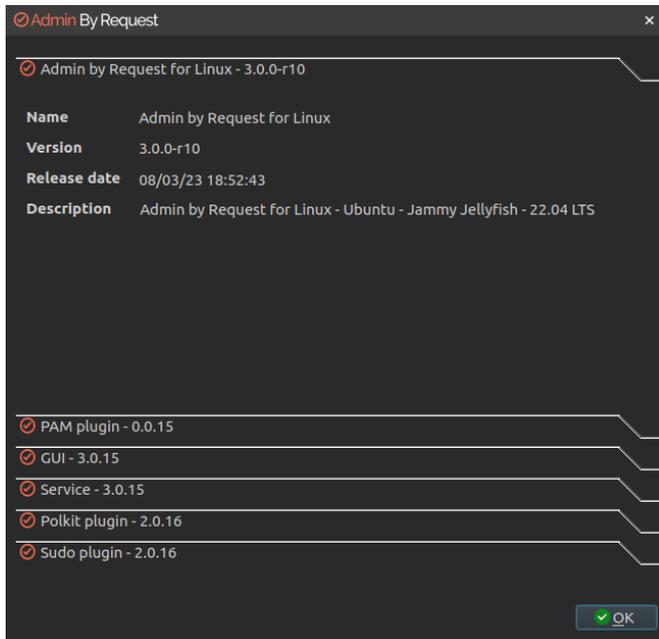


Clicking **Components** displays information about the individual modules that make up Admin By Request.

The modularized architecture means components can be updated as required via Linux package management with minimal impact on other parts of the system.

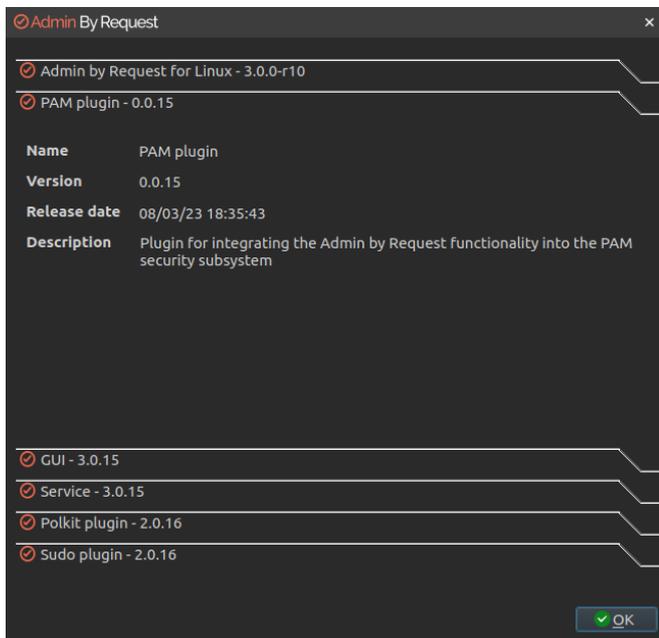
1. Admin By Request for Linux:

The main module for logic and functionality carried out by the application. This module also supplies the version number of the Linux client that is installed:



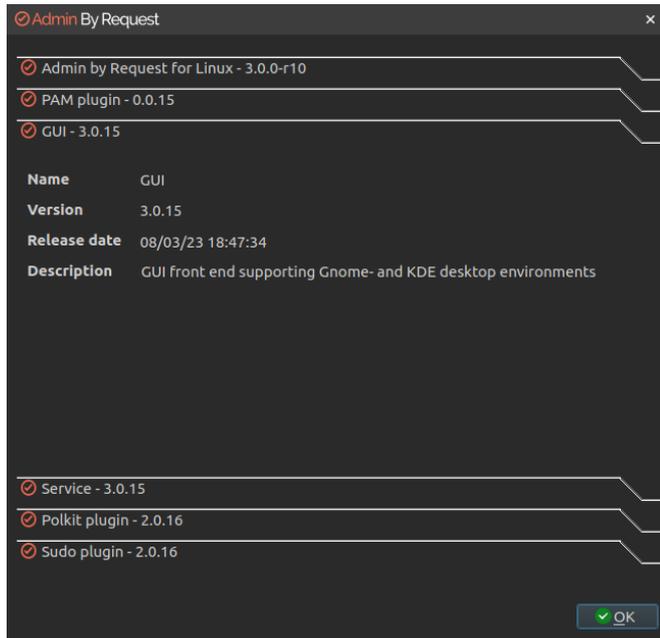
2. PAM plugin:

Privileged Access Management plugin, supporting the main module.



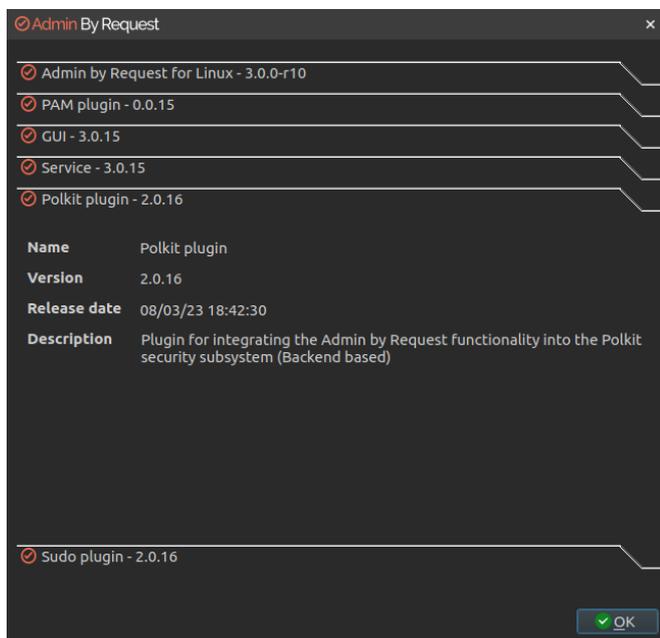
3. GUI:

User interface front-end, supporting both Gnome and KDE desktop environments.



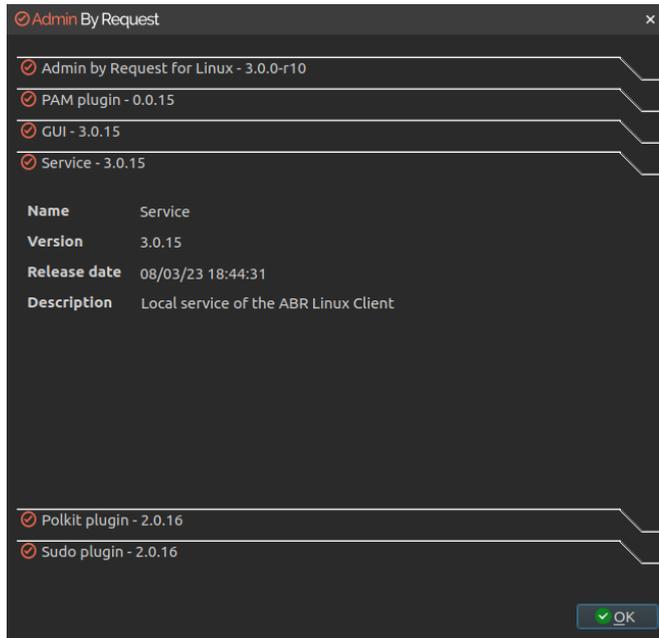
4. Polkit plugin:

A plugin for integrating application functionality into the Polkit security subsystem.



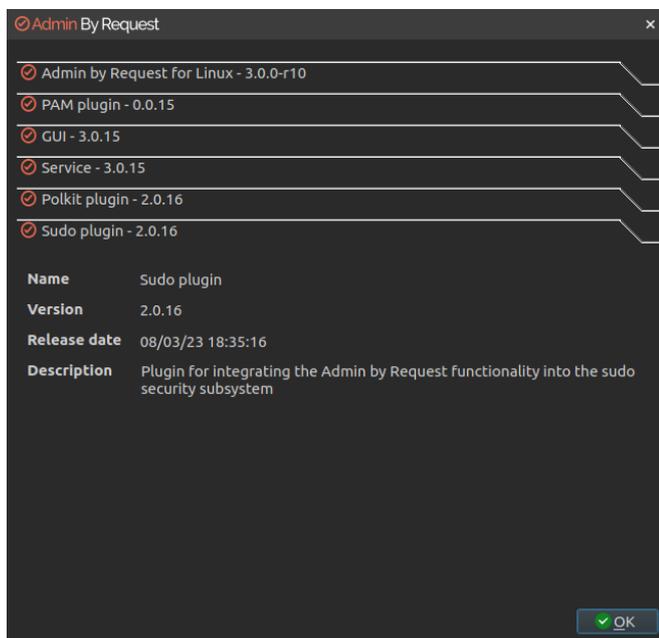
5. Service:

The local service for the Admin By Request Linux client.

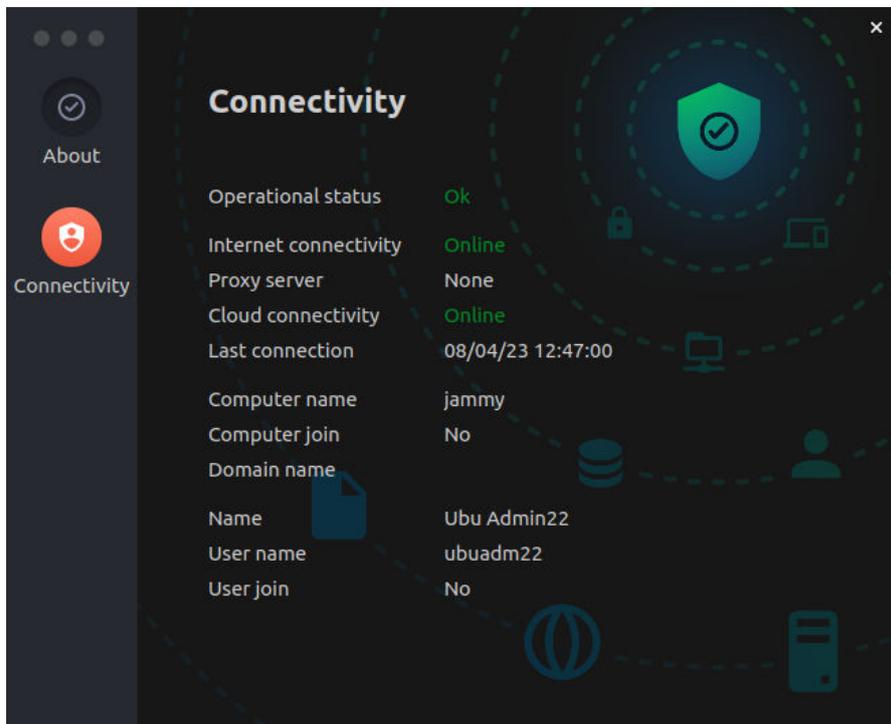


6. Sudo plugin:

A plugin for integrating application functionality into the sudo security subsystem.



- **Connectivity** – displays the current operational status of the Admin By Request system, including Internet and Cloud connectivity, and details about the current workstation and user:



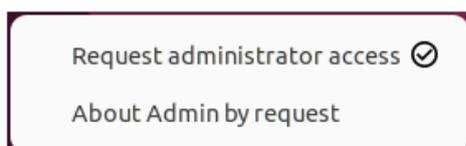
Requesting Administrator Access

Submitting a request for Administrator access is the primary mechanism for gaining elevated privileges.

NOTE: Timing can be important when an admin session is started for some GUI operations. If you start an admin session *after* you have started the GUI interface (for example, add a new user account in Settings), you will need to refresh the current GUI screen by selecting another option in Settings, then going back to User Accounts.

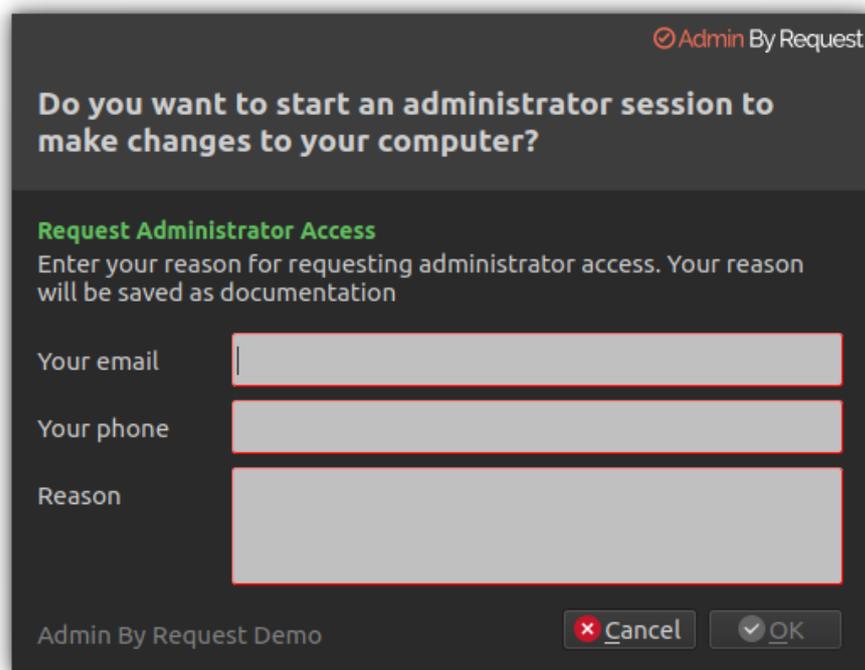
If you start the admin session *before* opening Settings, there is no need to refresh the user interface.

As with *About Admin By Request*, click the menu bar icon to display the menu and select **Request administrator access**:



A standard user making this selection initiates the following sequence of events:

1. An empty *Request Administrator Access* form appears:



Admin By Request

Do you want to start an administrator session to make changes to your computer?

Request Administrator Access
Enter your reason for requesting administrator access. Your reason will be saved as documentation

Your email

Your phone

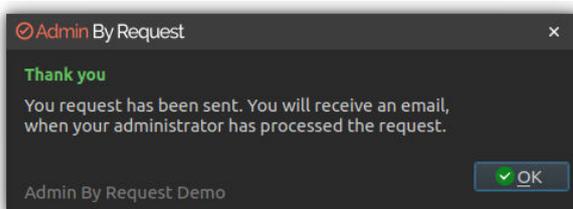
Reason

Admin By Request Demo

2. The user enters *email*, *phone* and *reason* information into the form and clicks **OK**.

NOTE: If approval is not required (**Portal > Settings > Linux Settings**), the approval steps are skipped.

3. The request is submitted to the IT administration team and the user is advised accordingly:

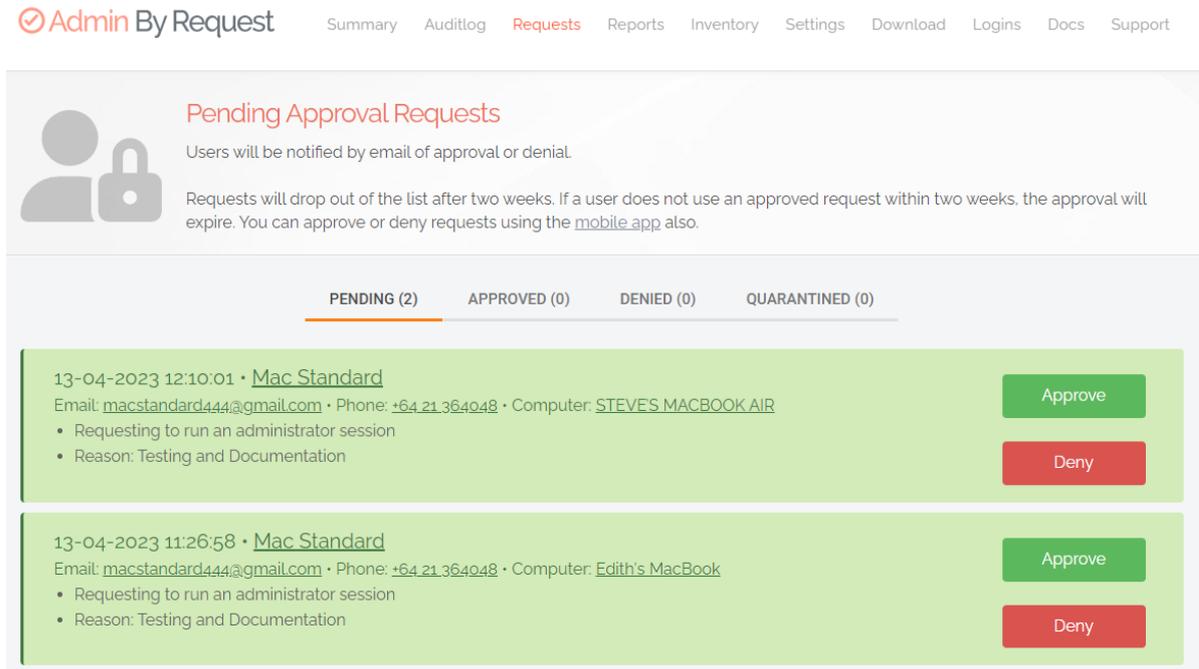


Admin By Request

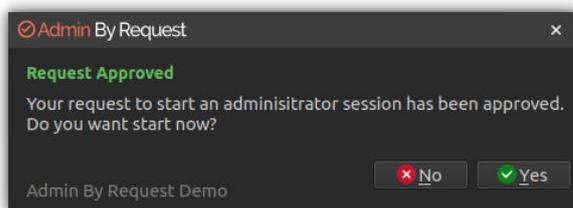
Thank you
Your request has been sent. You will receive an email, when your administrator has processed the request.

Admin By Request Demo

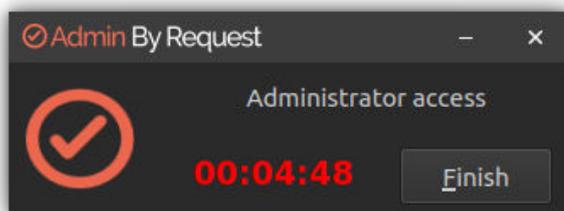
- The IT administration team is notified via the Admin By Request portal that a new request for administrator access has arrived. The following example shows how two new requests might appear in the portal:



- One of the team either approves or denies the request. If approved, the user is advised accordingly:



- The user clicks **Yes**, which starts the session and displays a countdown timer:



- The duration of an admin session is set via the portal (5 minutes in this example) and the countdown timer ticks down to zero, at which time the session ends. The user can optionally end the session at any time once it has started by clicking **Finish**.

See "[Changing Admin Session Duration](#)" on page 21 for more information on changing the duration of the countdown timer.

During an admin session, users can install programs requiring admin rights, install drivers and change system settings other than user administration.

IMPORTANT: During an admin session, users cannot run sudo or add, remove or modify user accounts.

Using Run As Admin

The Admin By Request *Run as Administrator* feature allows for the elevation of a single application. This capability negates the need for users to initiate an Administrator Access session (i.e., an extended period of time during which the user has elevated privileges on the device) to simply install one program.

Elevating privileges for execution of a single file is the much safer option compared to elevating the user's privileges across the endpoint.

In Linux, a single line sudo command implements *Run As Admin*.

For example:

1. Run a sudo command.
2. If approval is required, a pop-up will appear asking for information. Sometimes approval is not required, but a reason must still be given for logging purposes.
3. When the sudo command is complete, check the portal under **Auditlog > RUN AS ADMIN** rather than **Auditlog > ADMIN SESSIONS**. The sudo command is logged under RUN AS ADMIN.

Pre-approved applications run without prompting for a reason and the activity is logged under RUN AS ADMIN. (e.g. the `sleep` command).

The elevated privileges last only for the duration of the install and apply only to the particular application or package authorized.

4

Portal Administration for Linux

This topic describes several key areas of the Admin Portal that can be used to manage *Linux Settings* and *Linux Sub Settings*, specifically Pre-Approval, Machine Learning, Azure AD Support and Admin Session Duration.

Pre-Approval

Pre-Approval (known sometimes as Whitelisting) refers to the method of working out which applications are trusted and frequently used, and adding them to a list that automatically allows users to elevate those applications when they need to. This is essentially the opposite of Block-listing/Blacklisting – creating a list of applications that cannot be elevated.

This method of “allow most, deny some” has proven to be extremely resource-efficient for large enterprises compared to the method of denying all applications and only allowing elevations on a case-by-case basis.

Admin By Request v3.0 for Linux allows for pre-approval of trusted applications. Once an application has been installed with Admin By Request:

1. Log in to the portal and navigate to the application's corresponding entry in the portal **Audit-log**.
2. Expand on the application entry, and select **Pre-approve this file** under Actions:
3. Click **Save**.

The list of pre-approved Linux applications can be found under **Settings > Linux Settings > App Control > PRE-APPROVE**:

Pre-Approval is based on the application vendor or checksum.

Blocked Applications

You can specify programs and applications that you wish to prevent users from executing with administrator privileges. You can block applications based on one or more of the conditions: file name, checksum, vendor or file location.

NOTE: You should never block solely based on the file name, as this will open up the endpoint to simple file renaming to bypass the blocking.

PIN code exceptions: The option is available to use a PIN code in case you allow the execution as an exception - simply retrieve the PIN code from the computer's inventory. If you do not wish to offer a PIN option, you can disable this under the Run As Admin tab.

Defining a blocked application:

Block application

This page allows you to define an application that will be blocked from executing.

Application	About Applications
<p>Type <input style="width: 100%;" type="text" value="Block file from running as administrator"/></p> <p>Condition <input style="width: 100%;" type="text" value="No condition (block always)"/></p> <p>Application name <input style="width: 100%;" type="text"/></p> <p>File name <input style="width: 100%;" type="text"/></p> <p>Blocking message (optional) <input style="width: 100%; height: 30px;" type="text"/></p> <p>Internal comments (optional) <input style="width: 100%; height: 30px;" type="text"/></p>	<p>Block Application allows you to point to a file name that will be blocked from executing. You can specify wildcards as file name, such as *.msi.</p> <p>Application name is only used for convenience in the overview list.</p> <p>Condition is when a file is only blocked, if the condition is met:</p> <p>Directory File must be located in this directory or a sub-folder.</p> <p>Certificate The vendor digital certificate of a file.</p> <p>Checksum A checksum of a specific file version. If the file is updated, the checksum no longer matches and a new must be collected.</p> <p>Blocking Message will appear as a rejection to the user, when the application is attempted to be executed. Requires endpoint software version 8.0 or later.</p> <p>Please contact us using the "Contact" menu, if you have questions about blocking.</p>

Type:

- Block file from running as administrator
- Block vendor files from running as admin (digital certificate)
- Block location from running as admin (all files in folder tree)
- Block always

Condition:

- No condition (block always)
- Block if located in directory
- Block if matching digital certificate
- Block if matching checksum

Application name is a label only - used for convenience in the overview list.

File name allows you to point to a file name that will be blocked from executing. You can specify wildcards in the file name, such as *.sh.

Blocking message will appear as a denial message to the user when execution of the application is attempted.

Run As Admin

The core Admin By Request Run as Administrator feature, which allows for the elevation of a single application, is new and improved in version 3.0. This feature negates the need for users to initiate an Admin Session (i.e., an extended period of time during which the user has elevated privileges on the device) to simply install on program. Elevating a single file is the much safer option compared to elevating the user's privileges across the endpoint.

Machine Learning

The idea behind Machine Learning Auto-Approval is to kill two birds with one stone by allowing customers to build a Pre-Approved list as their employees use the software. This removes the need for enterprises to spend considerable amounts of time and effort figuring out and manually configuring which applications should be pre-approved ahead of time.

The way it works is, it allows you to create a simple rule that says:

"If approval for elevation of an application is granted X times, that application is now automatically approved for incoming requests from then on."

This allows the system to handle creating the list of applications that are safe for approval as applications are used.

For more information, including step-by-step procedures, refer to [Features > Machine Learning](#).

Supplementary Technical Information

Local Administrator Accounts

By default, users logging into a Linux workstation are not downgraded from administrator to user unless the setting 'Revoke admin rights' is enabled in the portal and the user is not in the excluded accounts list. The reason all users are not downgraded immediately is because you may have service accounts that you have forgotten to list in the excluded accounts list.

Also, if someone cleared the excluded accounts list and clicked **Save** by mistake, the result would be unusable endpoints; no users would be able to gain elevated privileges and would instead have very limited ability on their devices.

The following graphic shows Revoke admin rights **ON**, *except* for user account helpdesk:

The screenshot displays the 'Linux Global Settings' interface. On the left is a navigation menu with options: Authorization, Endpoint, Lockdown (highlighted), App Control, Data, and Emails. The main content area has three tabs: ADMIN RIGHTS (selected), SUDO, and ROOT. Under the ADMIN RIGHTS tab, there are two panels: 'Revoke Admin Rights' and 'About Admin Rights Revoke'. The 'Revoke Admin Rights' panel shows a toggle switch for 'Revoke admin rights' set to 'ON' and a text input field for 'Excluded accounts' containing 'helpdesk'. A 'Save' button is at the bottom. The 'About Admin Rights Revoke' panel contains explanatory text: 'Revoke admins rights removes the user from the local administrator's group when logging on to an endpoint. Domain groups, such as Domain Admins, are never removed from the local administrator's group.' and 'Excluded accounts are not removed from the local administrator's group at logon. Multiple accounts must be specified on separate lines. Domain accounts must be prefixed with a domain and a backslash. For Azure joined devices, use either the email address (the Azure "email" field) or AzureAD as domain.'

Sub-Settings

The portal has two levels of settings. *Linux Settings* apply to all users by default, unless overridden under *Linux Sub Settings*. With sub settings, you can define special settings based on Active Directory computer or user groups and/or Organizational Unit(s).

Sub settings will *override* the default settings for the users or computers to which they apply. If a user or computer hits multiple sub settings, the first in listed order that includes the setting concerned wins.

This can be used, for example, to allow sudo access for *developers* or automatically approve requests from *users in the IT department*.

Sudo

For security reasons, sudo access is disabled during administrator sessions by default. This can be enabled in the settings. We do not recommend enabling sudo access unless absolutely necessary.

Admin By Request has checks in place to prevent system tampering using sudo, but due to the root-level access, it is impossible to fully protect against tampering using sudo.

If only certain commands need to be run with sudo, consider using the built-in `/etc/sudoers` file. The Admin By Request sudo settings will not override normal `/etc/sudoers` settings.

Tampering

To prevent tampering with Admin By Request, the software monitors all important files during an administrator session. During a session, access to the Users & Groups preference panel is disabled to prevent users from adding new administrators. Further, by default, sudo access is disabled to prevent calling system-critical tools and user management from the terminal.

The service also monitors users and groups during the session to prevent tampering if sudo access is enabled. If Admin By Request detects that the clock has been changed, the administrator session will end instantly to prevent users from extending their session.

Changing Admin Session Duration

Admin session duration (access time) is the maximum amount of time in minutes an Admin Session may last. This time must be sufficient for the user to install software or perform any other necessary tasks.

To change the time allocated for an administrator session:

1. Log in to the Portal and select menu **Settings > Linux Settings**.
2. From the *Authorization* left menu, make sure the **AUTHORIZATION** tab is displayed (it is the default) and update the **Access time (minutes)** field in the Admin Session panel:
3. Click **Save** when done.

5

Policies for Linux

About Policies

Settings in the Admin By Request client application are controlled under "Linux Settings" in the *Settings* menu, when logged in to the portal. If, for whatever reason, you want to overrule these settings on specific clients, you can set overruling policies in a policy file.

IMPORTANT: Please note we do not recommend that you use a policy file to control client behavior. Instead, we recommend that you use portal settings and sub settings for better transparency and for real-time control of computers not connected to your LAN.

If you have any questions about portal settings or would like a demo of these, please feel free to contact us.

Overruling Portal Settings

To overrule portal settings with a policy file, edit this file:

```
/etc/abr/policies.d/adminbyrequest.policy.template
```

Note that this file is protected during administrator sessions and therefore cannot be hacked by end-users. The file is in json format and has an example non-used setting by default, as shown below. Simply add more settings from the following table to overrule web settings.

```
{
  "ExampleSetting": "ExampleValue"
}
```

Also note that any change to the policy file will take effect after the next reboot. Alternatively, if a policy change must take effect immediately without a reboot, an admin user or MDM can restart the service using `sudo killall adminbyrequest`.

Key	Type	Default	Description
AdminMinutes	Integer	15	Number of minutes the user is administrator. This can also be set in your portal settings.
AllowSudo	Boolean	0	Allow users to run sudo commands. Should not be enabled unless there is a good reason to, because it allows the user to

Key	Type	Default	Description
			tamper the endpoint software.
CompanyName	String		Overrules the company name that appears on user interfaces, which is by default the licensed company name.
ComputerGroups	Array of Strings		Computer groups to match machine to sub settings when not using Active Directory.
ExcludedAccounts	Array of Strings		List of accounts that will not be downgraded to user role, such as service accounts.
EnableSessions	Boolean	1	User can request an admin session.
EnableAppElevations	Boolean	1	User can authenticate apps without session.
Instructions	String		Body text on Code of Conduct ("Instructions") screen.
InstructionsHeader	String		Header text on Code of Conduct ("Instructions") screen.
LogoUrl	String		URL from which to download logo. If not specified, default icons will be used.
RemoveRights	Boolean	1	Downgrade users from Admin to User, unless the account is in excluded accounts or is a domain administrator in on a domain-joined device.
RequireApproval	Boolean	0	Elevate without requiring someone to approve requests.
RequireReason	Boolean	1	Require reason to elevate.
RequireAppApproval	Boolean	0	Elevate Run As Admin without requiring someone to approve requests.

Key	Type	Default	Description
RequireAppReason	Boolean	1	Require reason to Run As Admin.
ShowInstructions	Boolean	0	Show Code of Conduct screen.
UploadInventory	Boolean	1	Upload inventory data to the portal.
UserGroups	Dictionary with Array of Strings		User groups to match machine to sub settings when not using Active Directory.

Appendix

The appendix includes supplementary documentation such as terms and definitions, specifications, legacy product information, and references to in-depth material online.

In this appendix:

Terms and Definitions	26
Privileged Access	26
Glossary	28

Terms and Definitions

Privileged Access

Privileged access refers to abilities and permissions that go above and beyond what is considered "standard", allowing users (with privileged access) more control and reach in the system and network.

The following table describes several common privileged access terms.

Term	Definition
Blocklist	<p>The opposite of a pre-approved list. A list of blocked programs or applications that are denied access in an IT environment (i.e., they are denied the ability to run) when everything is allowed by default. All items are checked against the list and granted access unless they appear on the list. Might also be known as a "blacklist" – a term no longer used.</p> <p>See also "Pre-Approved List" on the next page.</p>
Elevated Application	<p>An application that has been given greater privileges than what is considered standard, which enables the application user to have more control over its operation, and the app itself to have more abilities and access within the computer.</p>
Elevated Privileges	<p>Also known as "privileged access". Elevated privileges provide the ability to do more than what is considered standard; for example, install and uninstall software, add and edit users, manage Group Policy, and modify permissions. Elevated privileges are sought after by attackers, who can use them to propagate through a network, remain undetected, and gain a strong foothold from which to launch further attacks.</p>
Endpoint	<p>A physical device that is capable of connecting to and exchanging information with a computer network. Endpoints include mobile devices, desktop computers, virtual machines, embedded devices, servers, and Internet-of-Things (IoT) devices.</p>
Horizontal Privilege Escalation	<p>Also known as "account takeover". Occurs when access to an account of a certain level (e.g., Standard User) is obtained from an account at that same level. Usually occurs when a malicious actor compromises a lower-level account and propagates through the network by compromising other lower-level accounts.</p> <p>See also "Vertical Privilege Escalation" on the next page.</p>

Term	Definition
Just-In-Time Access (JIT)	A way of enforcing the Principle of Least Privilege (POLP) by allowing access to privileged accounts and resources only when it is needed, rather than allowing “always on” access (also known as “standing access”). This reduces an organization’s attack surface by minimizing the amount of time an internal or external threat has access to privileged data and capability.
Lateral Movement	A common technique used by malicious actors, in which they spread from the initial entry point further into the network, while evading detection, retaining access, and gaining elevated privileges using a combination of tactics. The purpose is generally to compromise as many accounts as possible, access high-value assets, and/or locate a specific target or payload.
Phishing	A type of social engineering attack in which the victim is tricked into clicking a malicious link that can lead to malware installation or further duping of the victim into providing sensitive information such as credentials or credit card details.
Pre-Approved List	<p>The opposite of a blocklist. A list of approved programs or applications that are trusted (considered safe) when everything is denied by default. Items are checked against the already approved list and are only able to run if they are included in that list. Might also be known as a “whitelister” – a term no longer used.</p> <p>See also "Blocklist" on the previous page.</p>
Privileged Account	An account that has been granted access and privileges beyond those granted to non-privileged accounts. More sought after by attackers because, if compromised, they provide a better vantage point from which to launch an attack.
Privileged User	A trusted user who is authorized to leverage privileged access, such as through a privileged account, to perform high-value functions for which standard users are not authorized.
Standard User Account	A basic account for undertaking day-to-day tasks, for users who is not authorized or required to perform activities that require elevated privileges. These accounts are typically safer than those with higher access and permissions, as they do not provide the capability to perform administrative tasks, such as change system settings, install new software, manage the domain, and change local user credentials.
Vertical Priv-	Occurs when a lower-privileged account gains privileged access bey-

Term	Definition
ilege Escalation	<p>ond what it is intended to have. Usually occurs when a malicious actor compromises an account (e.g., a “Standard User” account) and then exploits system flaws or overrides privilege controls to escalate that account to one with higher privileges (e.g., a “Local Administrator” account).</p> <p>See also "Horizontal Privilege Escalation" on page 26.</p>

Glossary

Term	Short for	Definition
FIDO	Fast Identity Online	<p>With FIDO Authentication, users sign in with phishing-resistant credentials, called "Passkey" on the next page. Passkeys can be synced across devices or bound to a platform or security key and enable password-only logins to be replaced with secure and fast login experiences across websites and apps.</p> <p>Passkeys are more secure than passwords and SMS OTPs, simpler for consumers to use, and easier for service providers to deploy and manage.</p>
Intune	Microsoft Intune	Microsoft Intune is a cloud-based UEM solution. It manages user access and simplifies device and application management for multiple platforms, including mobile devices, desktop computers, and virtual endpoints.
MAM	Mobile Application Management	Software and processes that secure and enable IT control over enterprise applications on end users' corporate and personal devices.
MDM	Mobile Device Management	A methodology and toolset used to provide a workforce with mobile productivity tools and applications, while keeping corporate data secure.
PAM	Privileged Access Management	A set of cybersecurity technologies and strategies that allow organizations to secure their infrastructure and applications by managing privileged access and permissions for all users across the IT environment.

Term	Short for	Definition
Passkey	Passkey	Passkeys are a replacement for passwords that provide faster, easier, and more secure sign-ins to websites and apps across a user's devices. Unlike passwords, passkeys are always strong and phishing-resistant.
POLP	Principle of Least Privilege	The idea that users, applications, programs, and processes should be allowed only the bare minimum privileges necessary to perform their respective functions.
UEM	Unified Endpoint Management	A way to securely manage all the endpoints in an enterprise or an organization from a central location.

Document History

Document	Product	Changes
1.0 - 31 May 2023	2.2 - 19 September 2022	<ul style="list-style-type: none">• Initial document release
1.1 - 7 August 2023	3.0 - 7 August 2023	<ul style="list-style-type: none">• Include 3.0 features• Apply new document template and formatting

Index

A

- About ABR 10
 - About 10
 - Connectivity 14
- Administrator Access 14
- Audience 5

B

- BIOS 8
- Blocked Applications 18

C

- chmod 6
- Component
 - GUI 12
 - Main module 11
 - PAM plugin 11
 - Polkit plugin 12
 - Service 13
 - sudo plugin 13
- Condition (blocking) 19

D

- Download 6

- Duration 17

F

- Fedora 36 6
- File Locations 9

G

- GRUB menu 8

I

- Installing 6
- Introduction 5

L

- Local Administrator Accounts 20

M

- Machine Learning 20

O

- Overruling Portal Settings 22

P

Packages	8
Performance	9
Policies	
macOS	22
Portal	6
Pre-Approval	18
Prerequisites	6

R

Red Hat EL 9	6
Release Notes	5
root user	9
Run As Admin	17, 19

S

Session Duration	21
Sub-Settings	20
sudo	6, 8, 17, 21

T

Tamper prevention	9
Tampering	21
Test	7

Type (blocking)	19
-----------------------	----

U

Ubuntu 20.04	6
Ubuntu 22.04	6
UEFI	8
Uninstalling	6
Upgrading	8
User accounts	17
User rights	9