

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

Document Version: **1.4**

Document Date: **24 May 2024**

# Linux Client: IT Admin Guide

Configure, deploy and manage your Linux workstations

 **Admin** By Request

Linux Product Version: **3.1.9**



Copyright © 2024 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](https://adminbyrequest.com/contact)

[www.adminbyrequest.com](https://www.adminbyrequest.com)  
[linktr.ee/adminbyrequest](https://linktr.ee/adminbyrequest)

# Table of Contents

<b>Linux Client - Overview</b>	<b>5</b>
Introduction	5
In this document	5
Audience	5
Product Release Notes	5
<b>Linux Client - Install / Uninstall</b>	<b>6</b>
Prerequisites	6
Installing Admin By Request	6
Upgrading Admin By Request	8
Deploying new releases	8
Uninstalling Admin By Request	8
User rights after installation	9
Tamper Prevention	9
Performance after Installation	10
File Locations	10
<b>The Linux GUI Client User Interface</b>	<b>11</b>
About Admin By Request	11
In this topic	11
About Admin By Request	12
Connecting via a Proxy Server	17
Ports and IP addresses	17
Using Run As Admin	18
Requesting Administrator Access	19
<b>The Linux Command Line Interface</b>	<b>22</b>
Introduction	22
Prerequisites	22
Commands	23
abr finish	23
abr settings	24
abr start	25
abr status	26
abr version	26
abr --help	27
abr --master-config-file	27

abr --system-config-file .....	28
abr --log-level .....	28
Auditlog .....	28
<b>Portal Administration for Linux .....</b>	<b>30</b>
Introduction .....	30
In this topic .....	30
Run As Admin Settings .....	31
Admin Session Settings .....	32
Changing Admin Session Duration .....	32
Lockdown Settings .....	33
Admin Rights .....	33
Sudo .....	33
Root .....	34
Pre-Approval Settings .....	35
File Blocking Settings .....	37
Privacy Settings .....	39
Entra ID Support .....	40
Preventing Abuse .....	42
Policies for Linux .....	43
Supplementary Technical Information .....	43
Local Administrator Accounts .....	43
Sub-Settings .....	44
Sudo .....	44
Tampering .....	44
<b>Terms and Definitions .....</b>	<b>45</b>
Privileged Access .....	45
Glossary .....	47
<b>Document History .....</b>	<b>49</b>
<b>Index .....</b>	<b>50</b>

# Linux Client - Overview

## 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.
- Selected Settings tables, describing how to use each setting.
- 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.

### NOTE:

Although the guide is written from the point of view of an IT Administrator, the procedure steps and screenshots are described from an end user's perspective. This has two benefits:

1. You can clearly see how something works from an end user's point of view.
2. If required, you can create your own customized end user documentation by simply copying and pasting the procedures with minimal rework.

## Product Release Notes

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

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

# Linux Client - Install / Uninstall

## Prerequisites

Admin By Request, version 3.1.9 supports the following Linux distributions:

- Ubuntu 20.04 LTS
- Ubuntu 22.04 LTS
- 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'
```

4. 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.

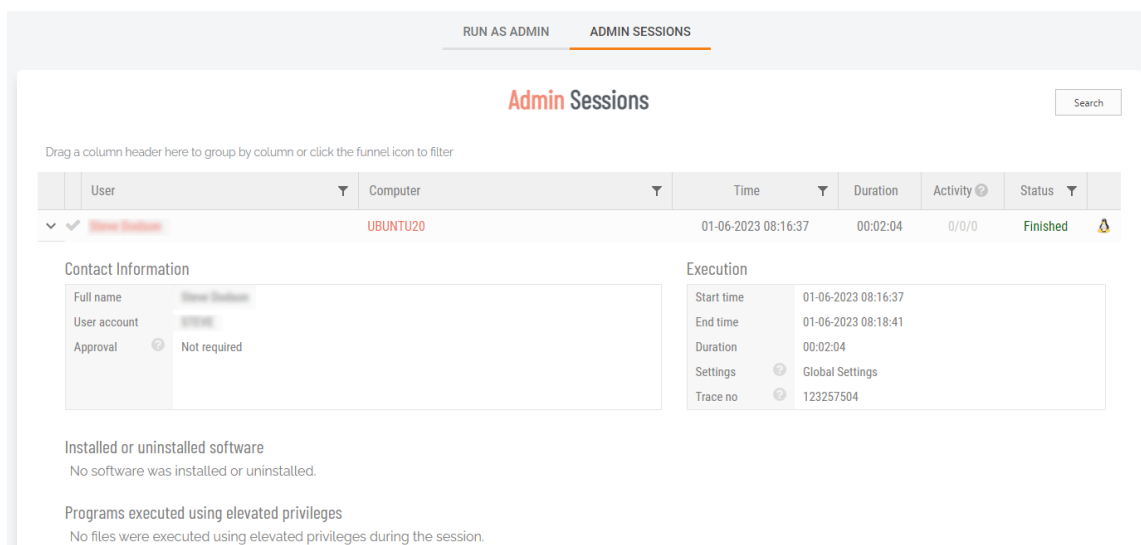
Installation is now complete.

## B. Test the installation.

1. 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.
2. Log in to the [Admin By Request Portal](#).
3. From the portal menu at the top, select **Settings > Workstation Settings > Linux Settings**
4. 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**.
5. Return to the Linux workstation and start an admin session:
  1. Click the Admin By Request icon in the top right corner of the screen and select **Request administrator access**.
  2. Confirm you want to start a session now (answering any questions that might pop up, such as **Reason**).
6. Run the sudo command above again and confirm that it works this time.
7. 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:

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



The screenshot shows the 'Admin Sessions' portal with tabs for 'RUN AS ADMIN' and 'ADMIN SESSIONS'. The 'ADMIN SESSIONS' tab is active, displaying a table of sessions. The first session is for user 'Steve Davidson' on computer 'UBUNTU20', starting at 01-06-2023 08:16:37 and ending at 01-06-2023 08:18:41, with a duration of 00:02:04. The status is 'Finished'. Below the table, there are sections for 'Contact Information', 'Execution', 'Installed or uninstalled software', and 'Programs executed using elevated privileges'.

User	Computer	Time	Duration	Activity	Status
Steve Davidson	UBUNTU20	01-06-2023 08:16:37	00:02:04	0/0/0	Finished

**Contact Information**

Full name	Steve Davidson
User account	STEVE
Approval	Not required

**Execution**

Start time	01-06-2023 08:16:37
End time	01-06-2023 08:18:41
Duration	00:02:04
Settings	Global Settings
Trace no	123257504

**Installed or uninstalled software**

No software was installed or uninstalled.

**Programs executed using elevated privileges**

No files were executed using elevated privileges during the session.

3. 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 immediately 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

## Deploying new releases

Admin By Request software updates are deployed by our Auto-Update process. However, when we release a new version we do not deploy it right away to all customers via auto-update. This is simply to mitigate any issues that arise after beta testing.

Our rule-of-thumb is to activate auto-update of new releases within 4 - 8 weeks of release, but this is subject to change, depending on feedback and any potential issues that might arise.

## 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**.



- If a menu similar to that shown below appears, choose the option that gets you to a shell prompt:

```
Recovery Menu (filesystem state: read-only)

resume           Resume normal boot
clean            Try to make free space
dpkg            Repair broken packages
fsck            Check all file systems
grub            Update grub bootloader
network        Enable networking
root            Drop to root shell prompt
system-summary  System summary

<Ok>
```

- At the *Password:* prompt, enter the root password (or simply press **Enter** if you haven't yet given the root account any password).
- 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).

- Start a terminal session, then start an Admin Session.
- If you haven't already, set the root user password:  
`sudo passwd`
- Switch to the root user:  
`su`
- 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 Workstation Settings > Linux 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 ["Supplementary Technical Information" on page 43](#) for more information.

## 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

Admin By Request maintains files and logs in the following locations:

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

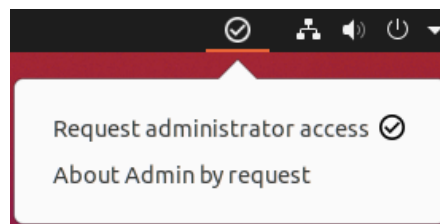
# The Linux GUI Client User Interface

## About Admin By Request

The user interface is graphical and is accessed via the icon menu in the menu bar (top right) of the screen:



Click the icon to display the menu and select *About Admin By Request* for further information:



### In this topic

- "About Admin By Request" on the next page
- "Connecting via a Proxy Server" on page 17
- "Using Run As Admin" on page 18
- "Requesting Administrator Access" on page 19

## About Admin By Request

Once installed, Admin By Request is running in the background for as long as the endpoint is powered-on. Selecting the app from the menu bar launches the *user interface*, which comprises a simple window with two buttons down the left-hand side:



The default panel is *About Admin By Request*, which is accessed via the top button. It shows the current workstation edition, license details, website link, and copyright information.

Click the **About** button to get back to this panel if viewing one of the other panels.

**Other Panels** (accessed via their respective buttons).

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

### Components

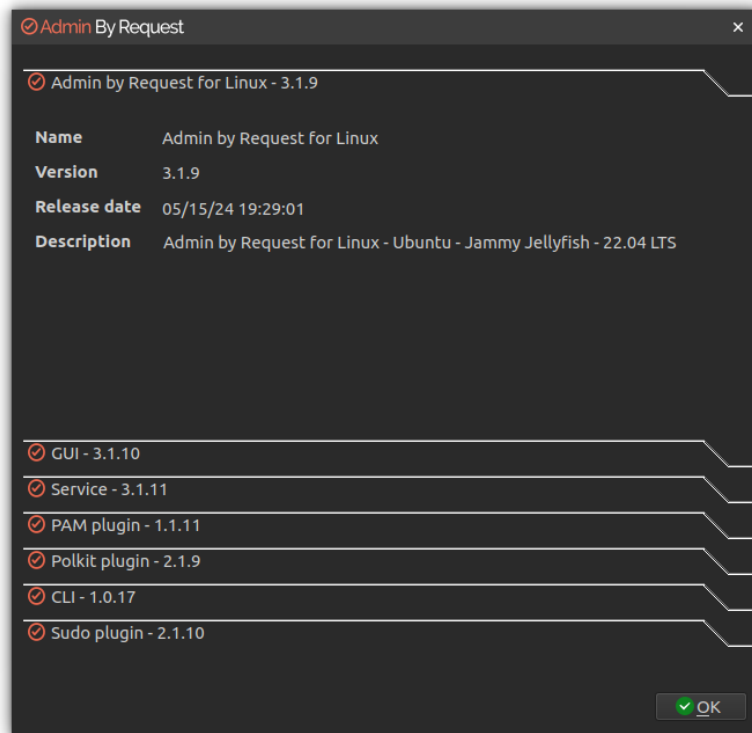
Clicking *Components* displays information about the individual modules that make up Admin By Request.

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

The following screens show current component versions. These can be useful should the need arise during troubleshooting.

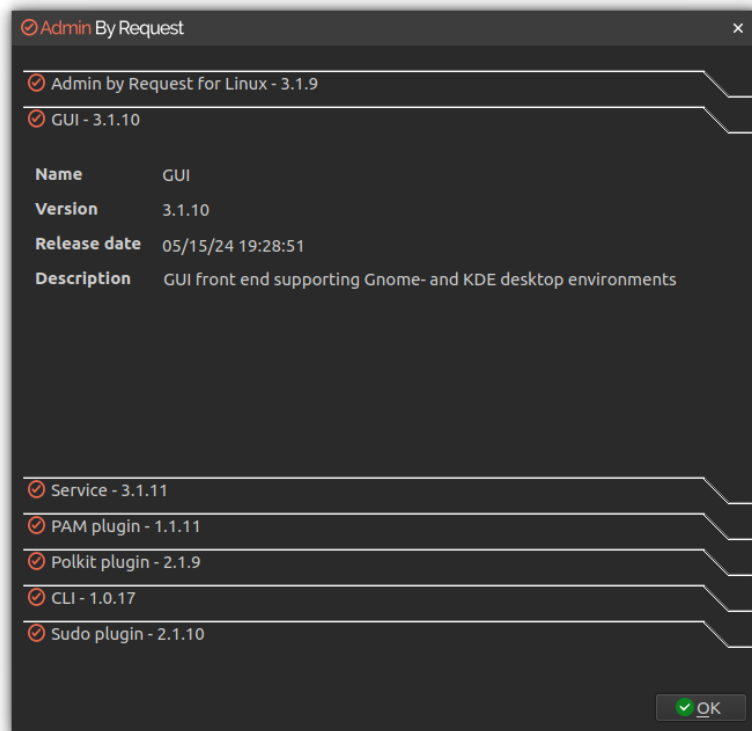
# 1. Admin By Request for Linux:

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.



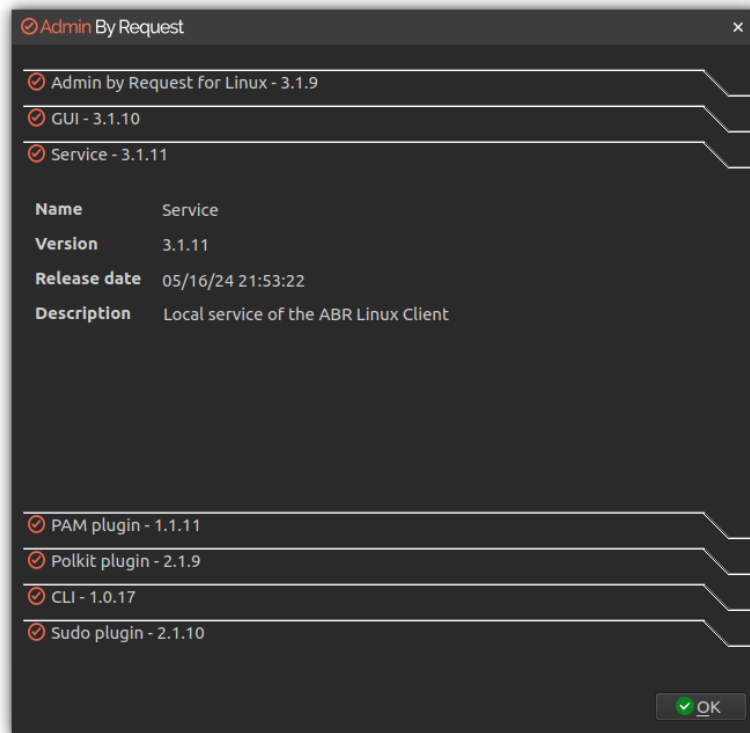
# 2. GUI:

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



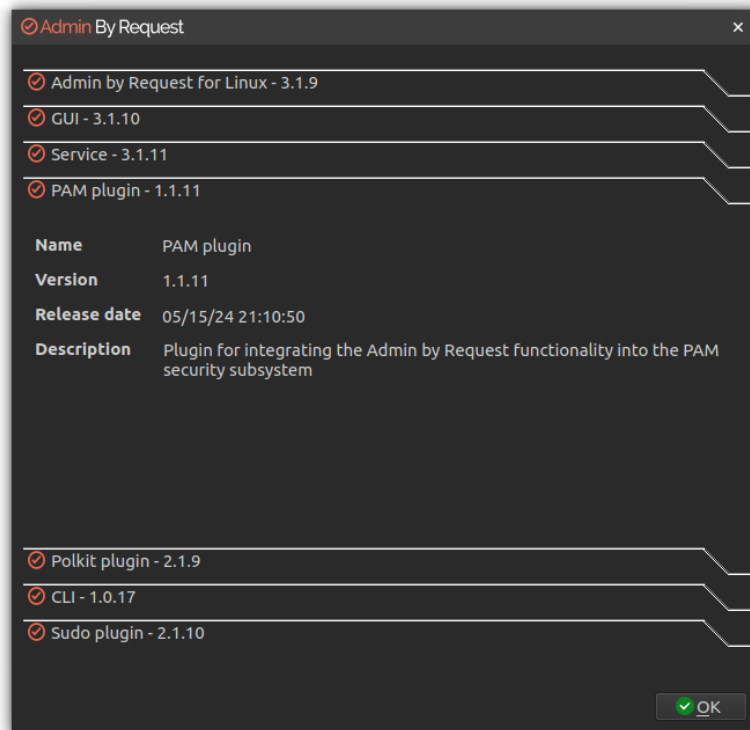
### 3. Service:

The local service for the Admin By Request Linux client.



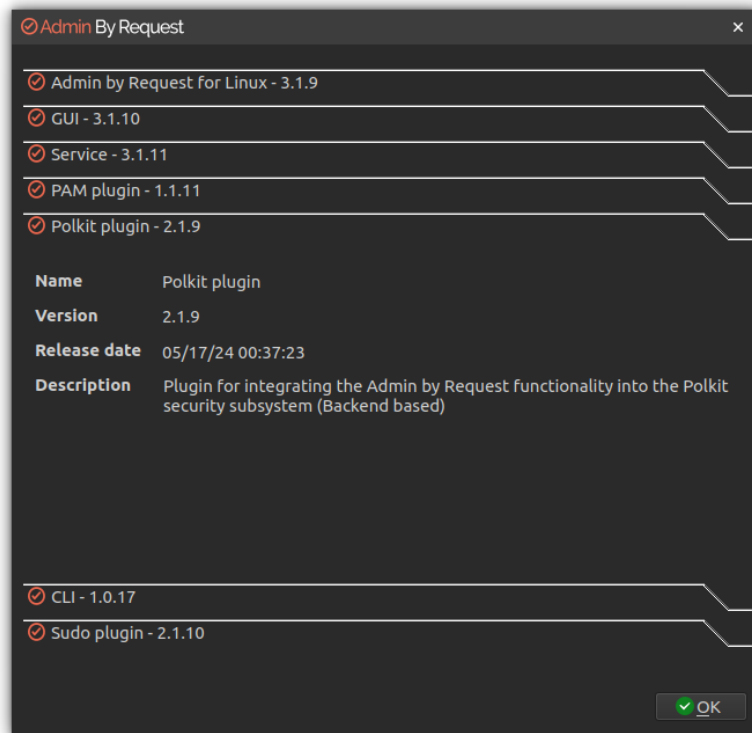
### 4. PAM plugin:

Privileged Access Management plugin, supporting the main module.



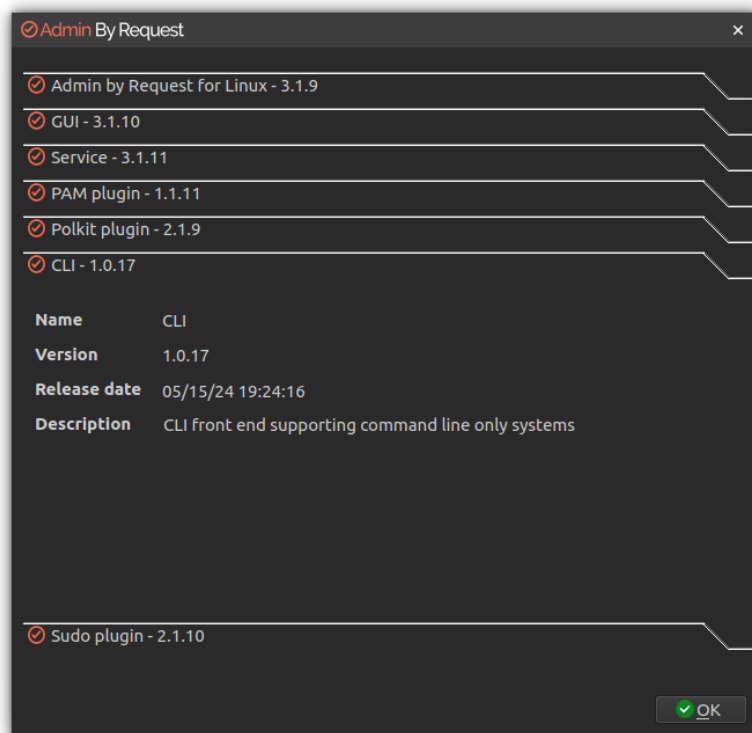
## 5. Polkit plugin:

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



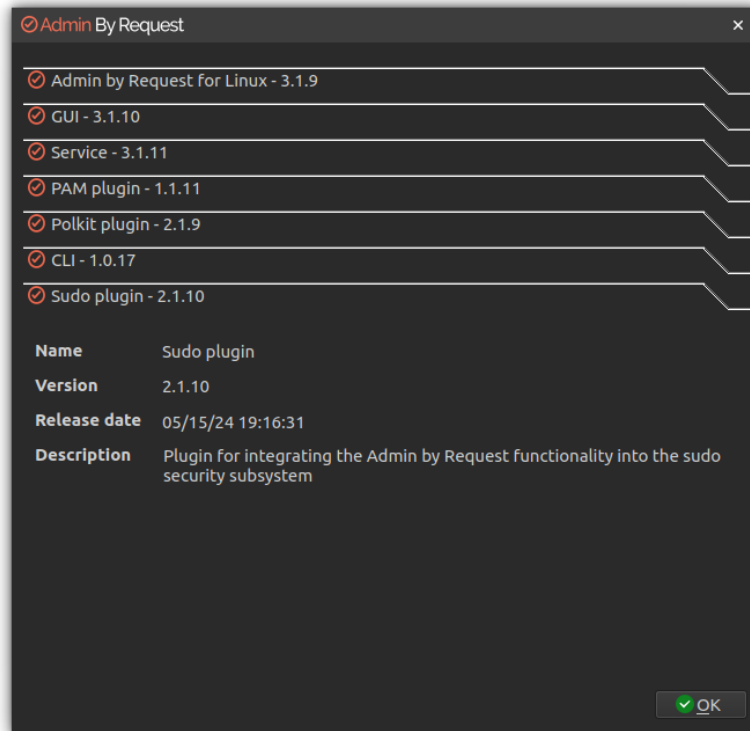
## 6. CLI plugin:

The Command Line Interface plugin for enabling user input and app responses via the command line.



## 7. 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 (see next section for screenshot).

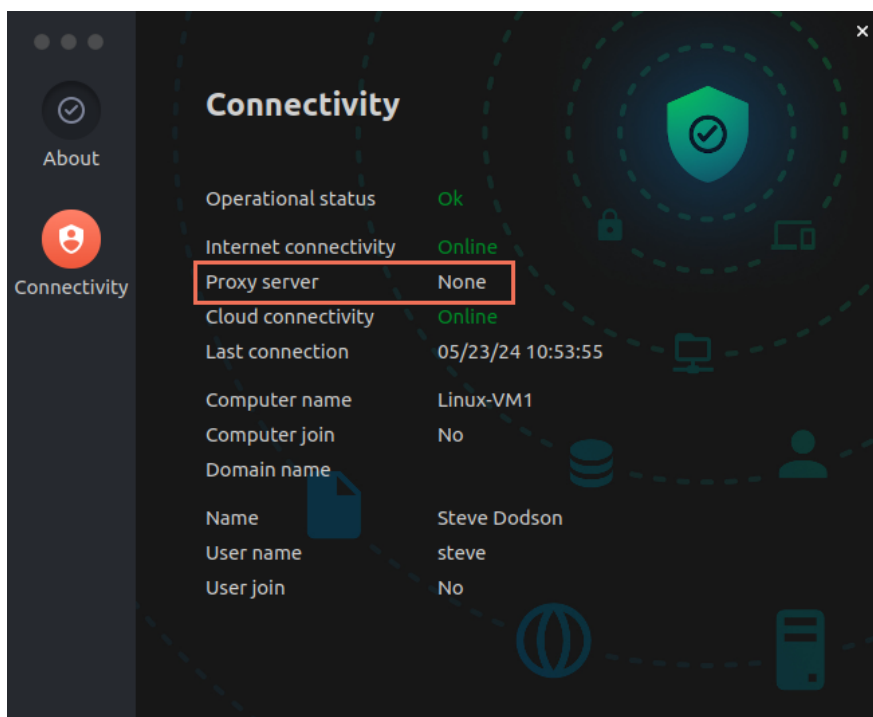


## Connecting via a Proxy Server

Endpoints can be configured to route privilege requests through a proxy server, which works transparently with Admin By Request.

If the user does have a proxy server enabled, its configuration is passed to the underlying service that will in turn use this proxy for cloud service communications. The proxy traffic uses NO-AUTH (no credentials) and will be seen as the computer account generating the traffic.

The *Connectivity* panel indicates whether or not a proxy server is used for an endpoint:



## Ports and IP addresses

Admin By Request uses port **443** and the IP addresses and URLs that need access through firewalls are as follows.

If your data is located in Europe:

- IP: **104.45.17.196**
- DNS: **linuxapi1.adminbyrequest.com**

If your data is located in the USA:

- IP: **137.117.73.20**
- DNS: **linuxapi2.adminbyrequest.com**

When the endpoint starts up, Admin By Request checks to see if it can connect directly to its host cloud server. If it can, then no proxy server is required and the value of *Proxy server* will be **None**.

If it cannot connect directly, it checks the following configuration file and works through the listed servers one by one until a connection is possible:

```
/etc/abr/configurations.d/proxy.conf.template
```

The default entries in this file are listed below. If you need to configure a proxy server, replace the information in this file with your proxy server information.

```
{
  "proxy":
  [
    {
      "type": "HTTPS",
      "hostname": "my-proxy-01.anyone.com",
      "port": 8080
    },
    {
      "type": "HTTPS",
      "hostname": "my-proxy-02.anyone.com",
      "port": 8080
    }
  ],
}
```

If the endpoint connects via a server configured in this file, **None** is replaced by the *hostname* of the proxy server and all privilege requests are routed through it.

Refer to [How We Handle Your Data](#) for more information.

## Using Run As Admin

*Run As Admin* (also known as *App Elevation*) allows for the elevation of a single application.

This capability negates the need for users to initiate an *Admin Session*. 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, including reason. If approval is not required, 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).

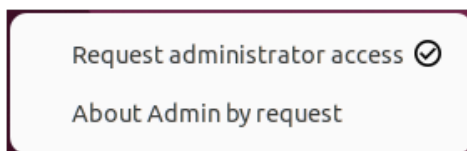
The elevated privileges last only for the duration of the install and apply only to the particular application or package authorized.

Check the audit log in the portal for details on the user, the endpoint, the application run and execution history.

## Requesting Administrator Access

Requesting administrator access is also known as requesting an *Admin Session*, which is a time-bound period during which a standard user has elevated privileges and can carry out administrator-level tasks..

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



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

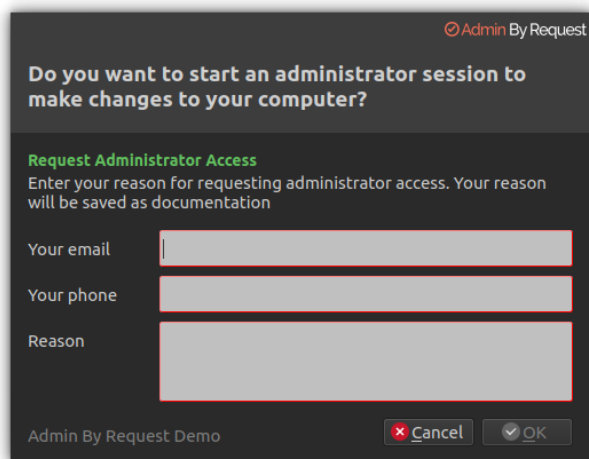
### NOTE:

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 might 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.

A standard user making this selection *where approval is required* initiates the following sequence of events.

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



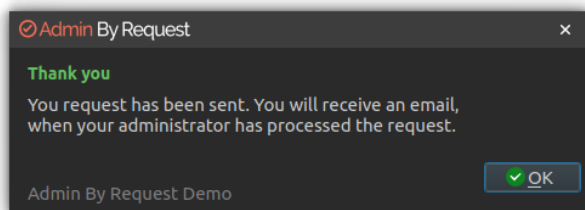
2. The user enters *email*, *phone* and *reason* information into the form and clicks **OK**.

### NOTE:

Settings in the portal control the full extent of what is displayed to the user:

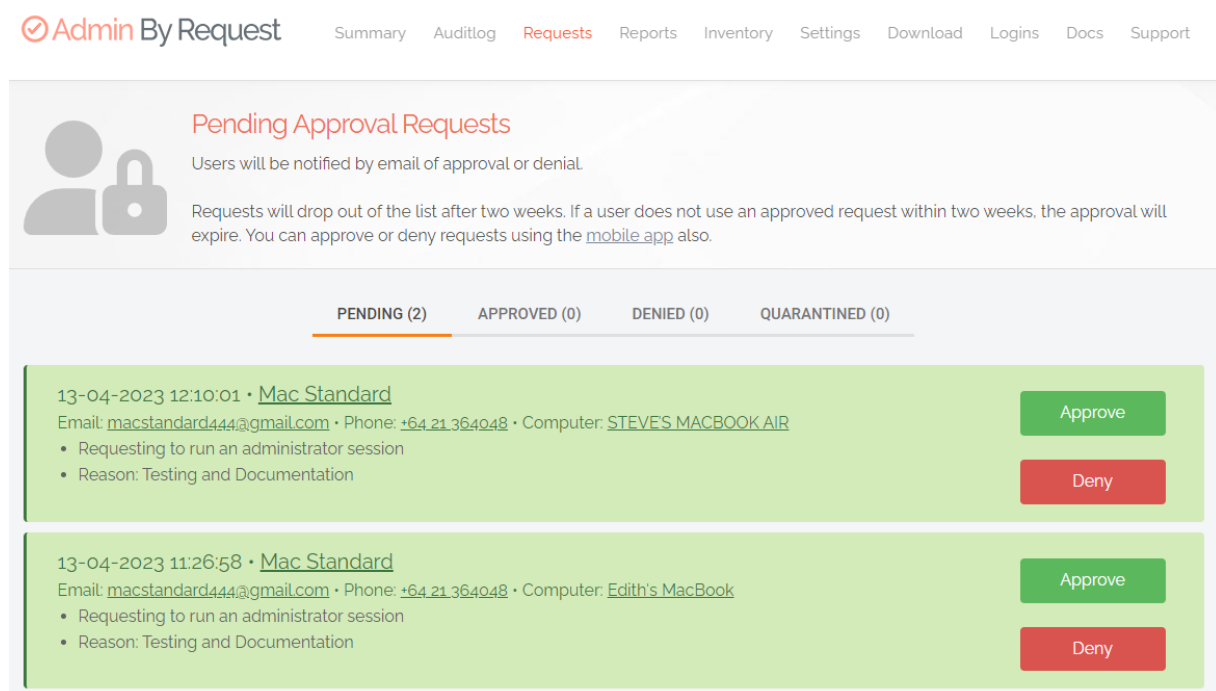
- If *Code of Conduct* is enabled, the user must acknowledge a Code of Conduct pop-up to continue (**Portal > Settings > Workstation Settings > Linux Settings > Endpoint > INSTRUCTIONS**).
- If *Require approval* is OFF, the approval steps are skipped (**Portal > Settings > Workstation Settings > Linux Settings > Authorization > AUTHORIZATION > Admin Session**).

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

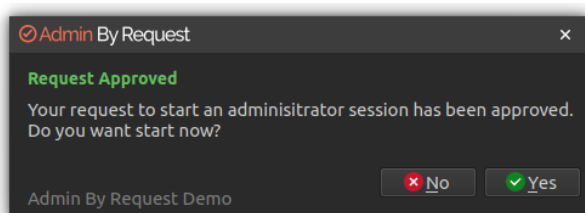


- The IT administration team is notified via the Admin By Request portal that a new request for administrator access has arrived.

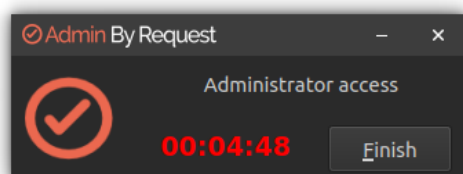
The following example shows how two new requests might appear in the portal:



- 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 (15 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 32 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. All activity during the elevated session is audited, so you can see in the audit log the reason why the person needs the elevation; anything installed, uninstalled, or executed.

**IMPORTANT:**

During an *Admin Session*, users **cannot** uninstall Admin By Request, or add, remove or modify user accounts.

# The Linux Command Line Interface

## Introduction

This topic describes the Linux command line interface (CLI).

## Prerequisites

- 1. The CLI commands are designed for the command line. If run inside a graphical interface's terminal window, certain commands will defer to the GUI version.

### Example - abr start

Running **abr start** in a GUI terminal window shows the following message:

```
steve@Linux-VM1:~$
steve@Linux-VM1:~$
steve@Linux-VM1:~$ abr start
Please use the ABR for Linux GUI instead to start an administration session.
steve@Linux-VM1:~$
```

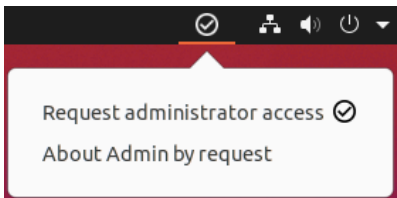
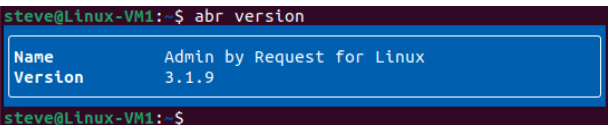
**NOTE:**  
The screenshots in this section were taken on a Linux Ubuntu 22.04 desktop, switching between GUI and command line interfaces:

- From GUI to command line:  
`systemctl set-default multi-user.target`  
`reboot`
- From command line to GUI:  
`systemctl set-default graphical.target`  
`reboot`

- 2. Using the CLI requires Admin By Request for Linux **version 3.1.9** or greater.

### Check version at endpoint

You can check which version of Admin By Request you have installed on a Linux endpoint as follows:

GUI	Command line
<p>From the icon in the menu bar, select <b>About Admin By Request:</b></p> 	<p>At the command prompt, enter <b>abr version:</b></p> 

## Check version in portal

You can also check the version using the Admin Portal. Log in to the portal and click **Inventory**. Find the device concerned and note the version listed in column **SW**:

[Search](#)

Drag a column header here to group by column or click the funnel icon to filter. You can select more columns by right-clicking the header.

Computer	User	Operating system	Model	SW	Remote	PIN	Details
EDITH	Steve	Windows 10 Pro	Precision M6700	8.3.0		PIN	<a href="#">Details</a>
HOME-SRV-1	Administrator	Windows Server 2019 Datacenter	VMware20,1	8.3.0	Remote	PIN	<a href="#">Details</a>
HOME-VM1-L22	Olivia Lim	Ubuntu 22.04.4 LTS	VMware Virtual Platform	3.1.8		PIN	<a href="#">Details</a>
HOME-VM2-W11	Eric Hastie	Windows 11 Pro	VMware20,1	8.3.0		PIN	<a href="#">Details</a>
LINUX-VM1	Steve Dodson	Ubuntu 22.04.4 LTS	VMware Virtual Platform	3.1.9		PIN	<a href="#">Details</a>
STEVE'S MACBOOK PRO	Steve Dodson	macOS 12 Monterey	MacBookPro 14,3	4.2.2		PIN	<a href="#">Details</a>
WINDOWS-VM3-11	Steve	Windows 11 Pro	VMware20,1	8.3.0		PIN	<a href="#">Details</a>

## Commands

Version 3.1.9 supports the following commands:

- `abr finish`
- `abr settings`
- `abr start`
- `abr status`
- `abr version`

There are also four global options:

- `abr --help`
- `abr --master-config-file <arg>`
- `abr --system-config-file <arg>`
- `abr --log-level <arg>`

Run **abr --help** to see a full list of commands and options.

### abr finish

Command	Output
<b>abr finish</b>  Ends an admin session.	<div> <pre>Status Finished admin session Connecting ...</pre> </div> <p>When the pop-up message above disappears, the countdown timer in the bottom right of the status bar also disappears.</p>

## h2&gt;abr settings

Command	Output
<p><b>abr settings</b></p> <p>Lists the settings from the portal that currently apply to this endpoint.</p> <p>To output settings to a text file:</p> <pre>abr settings &gt; settings.txt</pre>	<pre> steve@Linux-VM1:~\$ abr settings ===== Settings ===== UpdateTime = "2024-05-23T11:53:55.906" ComputerDistinguishedName = "" ComputerGroups = [] ComputerOrganizationalUnit = [] SIDC = "8fb8c607-5263-41ba-a35c-7f37ac5fd8b8" SID = "9697897e-ba55-4bb8-9819-2f43d764af78" UploadInventory = "true" DebugMode = "true" SupportAssistEnabled = "false" EnableAppElevations = "true" BlockAppElevations = "false" RequireAppApproval = "false" RequireAppReason = "false" EnableSessions = "true" RequireApproval = "false" RequireReason = "false" AdminMinutes = "180" AuthenticateMode = "Authenticate" SSOEmailMatch = "true" SSORestrictiveSessions = "false" SSORestrictivePreApprovals = "false" AllowAppStore = "true" AllowSudo = "true" AllowSudoForEveryone = "true" AllowSudoInteractive = "true" AllowRootLogin = "false" AllowChangeRootPassword = "false" ShowInstructions = "false" ShowAppInstructions = "false" InternetUpdate = "false" AzureADConnector = "false" PreferAzureAD = "true" UseLogo = "false" DockIcon = "true" OwnerLock = "false" IntuneComplianceLock = "false" RemoveRights = "false" LastAdminCheck = "false" EnableBlockedApps = "false" LockedPreferences = ["com.apple.LoginItems-Settings.extension", "com.apple.Users-Groups-Settings.extension"] EmailFieldBehavior = "Mandatory" PhoneNoFieldBehavior = "Mandatory" ComputerName = "Linux-VM1" ConnectionStatus = "true" ===== Policies ===== SessionEndRemindTime = "30" RemoveGroups = ["root", "adm", "sudo"] AddGroups = []  steve@Linux-VM1:~\$ </pre>



## h2&gt;abr start

Command	Output
<b>abr start</b>  Starts an admin session. The sequence of messages that follow depends on settings in the portal; i.e. whether or not the user must have requests approved before running elevated tasks.	<div> <div>1.</div> <div> <div>User access control</div> <div>Do you want to request administrative access now?</div> <div>ABR NZ Demo [Yes] No</div> </div> </div> <div> <div>2.</div> <div> <div>Request administrator access</div> <div>Your email stevehd@slingshot.co.nz</div> <div>Your phone 555 123456</div> <div>Reason Operating system updates</div> <div>ABR NZ Demo Cancel Ok</div> </div> </div> <div> <div>3.</div> <div> <div>User access control</div> <div>You request has been sent. You will receive an email, when your administrator has processed the request.</div> <div>ABR NZ Demo [Ok]</div> </div> </div> <div> <div>IMPORTANT:</div> <div>Check email for approval.</div> <div>Once approval is granted, start the admin session <i>before</i> entering any commands that require elevation, including sudo. If an admin session is not running, sudo is treated as "Run As Admin" and will prompt again for a reason even if approval has already been given for the admin session.</div> <div>To start the session, enter <b>abr start</b> a second time.</div> </div> <div> <div>4.</div> <div> <div>Request approved</div> <div>Your request to start an administrator session has been approved. Do you want start now?</div> <div>ABR NZ Demo [Yes] No</div> </div> </div> <div> <div>5.</div> <div> <div>Admin By Request 02:59:56</div> </div> </div> <p>The final screenshot shows a countdown timer in the status bar. When the timer reaches zero, the session will terminate.</p>

## abr status

Command	Output
<b>abr status</b>  Equivalent to selecting <b>About Admin By Request &gt; Connectivity</b> in the GUI app (see <a href="#">"Connecting via a Proxy Server" on page 17</a> ).	<pre> <b>steve@Linux-VM1:~\$ abr status</b>  Operational status      Ok Internet connectivity   Online Proxy server            None Cloud connectivity      Online Last connection         05/23/24 10:19:55  Computer name           Linux-VM1 Computer join           No Domain name  Name                    Steve Dodson User name               steve User join               No  <b>steve@Linux-VM1:~\$</b> </pre> <p>In this example, both <i>Internet</i> and <i>Cloud</i> connectivity are <b>Online</b>, and neither the computer nor the user is joined to a domain.</p>

## abr version

Command	Output
<b>abr version</b>	<pre> <b>steve@Linux-VM1:~\$ abr version</b>  Name      Admin by Request for Linux Version   3.1.9  <b>steve@Linux-VM1:~\$</b> </pre>

## abr --help

Entering **abr --help** shows all available commands and options:

```
steve@Linux-VM1:~$ abr --help
Description:
  Command line client for ABR for Linux

  It is possible to start and stop Admin sessions through the various commands below.
  For help with any of those, simply call them with --help.

Usage:
  abr [command]

Available commands:
  finish
  settings
  start
  status
  version

Global options:
  --help                produce help message
  --master-config-file arg (=usr/share/abr/configuration/cli.conf)
                        File path to the master configuration
                        file, which will be loaded first.
  --system-config-file arg (=etc/abr/cli.conf)
                        File path to the system configuration
                        file
  --log-level arg       log level for the application
```

## abr --master-config-file

Shows the master configuration file.

### NOTE:

This is for the use of Admin By Request and not something customers should be changing. It is provided here for information only and may be hidden in future releases.

```
steve@Linux-VM1:~$ cat /usr/share/abr/configuration/cli.conf
{
  "log_level": "INFO",
  "terminal_raw_input": "/var/log/abr/${username}/raw-input.txt",
  "terminal_unhandled_input": "/var/log/abr/${username}/unhandled-input.txt",
  "terminal_raw_output": "/var/log/abr/${username}/raw-output.txt",
  "terminal_unhandled_output": "/var/log/abr/${username}/unhandled-output.txt",
  "log_destinations":
  [
    {
      "type": "journal",
      "enable": false
    },
    {
      "type": "console",
      "enable": false
    },
    {
      "type": "file",
      "enable": true,
      "file_path": "/var/log/abr/${username}/cli.log",
      "rotation_size": 10485760
    }
  ]
}
```

## abr --system-config-file

Shows the system configuration file.

### NOTE:

This is for the use of Admin By Request and not something customers should be changing. It is provided here for information only and may be hidden in future releases.

```
steve@Linux-VM1:~$ cat /etc/abr/cli.conf
{
    "log_level": "INFO",
    "terminal_raw_input": "/var/log/abr/${username}/raw-input.txt",
    "terminal_unhandled_input": "/var/log/abr/${username}/unhandled-input.txt",
    "terminal_raw_output": "/var/log/abr/${username}/raw-output.txt",
    "terminal_unhandled_output": "/var/log/abr/${username}/unhandled-output.txt",
    "log_destinations":
    [
        {
            "type": "journal",
            "enable": false
        },
        {
            "type": "console",
            "enable": false
        },
        {
            "type": "file",
            "enable": true,
            "file_path": "/var/log/abr/${username}/cli.log",
            "rotation_size": 10485760
        }
    ]
}
```

## abr --log-level

Sets the level for logging (e.g. info, debug etc.). Use this option together with a command as indicated in the following examples:

```
abr -log-level info start
```

```
abr -log-level info status
```

```
abr -log-level debug start
```

```
abr -log-level debug status
```

The default level is **info** and the current level can be seen in either the master config file or the system config file.

## Auditlog

In the same way as the graphical interface, all activity is logged, both for *Run As Admin* and *Admin Session*.

The following examples show typical activity recorded in the auditlog.

RUN AS ADMINADMIN SESSIONSSERVERS

Run As Admin

Search

Drag a column header here to group by column or click the funnel icon to filter

	Application	User	Computer	Time	Duration	Activity	Status	
▼	✔ apt	Steve Dodson	LINUX-VM1	24-05-2024 13:50:58	00:00:16	1/1/1	Finished	🔔

Contact Information

Full name

Steve Dodson

User account

STEVE

Email

stevehd@slingshot.co.nz

Phone

555 123456

Approved by

Steve Dodson

Response In

00:00:20

Reason

Need to run as root

Execution

Start time

24-05-2024 13:50:58

End time

24-05-2024 13:51:14

Duration

00:00:16

Settings

Global Settings

Trace no

71504904

Application

Name

apt

File name

apt

Path

/usr/bin

Actions

Malware scan

Not available

Virustotal

[Check status](#)

Installed or uninstalled software

Action	Application	Version	Publisher
Install	rr	5.5.0-1ubuntu0.1	Ubuntu
Uninstall	rr	5.5.0-1build1	Ubuntu

Programs executed using elevated privileges

Program	File	Parameters		
apt	/usr/bin/apt	update	<a href="#">Check</a>	<a href="#">Path</a>

RUN AS ADMINADMIN SESSIONSSERVERS

Admin Sessions

Search

Drag a column header here to group by column or click the funnel icon to filter

	User	Computer	Time	Duration	Activity	Status	
▼	✔ Steve Dodson	LINUX-VM1	24-05-2024 21:04:52	00:01:56	0/0/2	Finished	🔔

Contact Information

Full name

Steve Dodson

User account

STEVE

Email

stevehd@slingshot.co.nz

Phone

555 123456

Approved by

Steve Dodson

Response In

00:00:40

Reason

Operating system updates

Execution

Start time

24-05-2024 21:04:52

End time

24-05-2024 21:06:48

Duration

00:01:56

Settings

Global Settings

Trace no

61513892

Installed or uninstalled software

No software was installed or uninstalled.

Programs executed using elevated privileges

Program	File	Parameters		
apt	/usr/bin/apt	upgrade	<a href="#">Check</a>	<a href="#">Path</a>
apt	/usr/bin/apt	update	<a href="#">Check</a>	<a href="#">Path</a>

# Portal Administration for Linux

## Introduction

This topic describes several key areas of the Admin Portal that can be used to manage *Linux Settings* and *Linux Sub Settings*. Fields that can be set and/or configured in the portal are presented in tables, with each table showing:

- **Setting** - the name of the field that controls the setting
- **Type** - the type of value that can be entered or selected and its default value
- **Description** - how the setting is used and notes about any implications it may have on other settings

To change any of the settings in the portal, [log in to the portal](#) and select the setting from the menu.

## In this topic

["Run As Admin Settings" on the next page](#)

["Admin Session Settings" on page 32](#)

["Lockdown Settings" on page 33](#)

["Pre-Approval Settings" on page 35](#)

["File Blocking Settings" on page 37](#)

["Privacy Settings" on page 39](#)

["Entra ID Support" on page 40](#)

["Preventing Abuse" on page 42](#)

["Policies for Linux" on page 43](#)

["Supplementary Technical Information" on page 43](#)

# Run As Admin Settings

Portal menu: **Settings > Workstation Settings > Mac Settings > Authorization > AUTHORIZATION**

## Settings Table - Run As Admin

*Run As Admin* (also known as Application Elevation) elevates privileges for only the file or application selected.

It is invoked when a user runs a sudo command or double-clicks an executable file.

Setting	Type	Description
Allow Run As Admin	Toggle On   Off Default: <b>On</b>	<p><b>On</b> - Allows users to elevate privileges for a selected file. Enables <i>Require approval</i> and <i>Require reason</i>. Disables <i>Block Run As Admin</i>.</p> <p><b>Off</b> - Denies users the ability to elevate privileges for a selected file. Enables <i>Block Run As Admin</i>, which is how users with admin credentials can still elevate privileges.</p>
Block Run As Admin (enabled only if <i>Allow Run As Admin</i> is Off)	Toggle On   Off Default: <b>Off</b>	<p><b>On</b> - Denies users the ability to execute <i>Run As Admin</i> even if administrator credentials are available (i.e. no authentication window is presented).</p> <p><b>Off</b> - Allows users with administrator credentials to execute <i>Run As Admin</i> (i.e. authentication window pops-up asking for admin credentials).</p>
Require approval	Toggle On   Off Default: <b>Off</b>	<p><b>On</b> - Sends a request to the IT team, which must be approved before elevation is granted. Makes <i>Require reason</i> mandatory (i.e. must be On).</p> <p><b>Off</b> - Allows the user to elevate file privileges (and thus perform the action) as soon as the action is selected. For example, selecting "Run as administrator" to execute a program occurs immediately, without requiring approval. Makes <i>Require reason</i> optional (i.e. can be either On or Off).</p>
Require reason	Toggle On   Off Default: <b>Off</b>	<p><b>On</b> - Extends the authentication window and asks the user to enter email address, phone number and reason. Reason must comprise at least <i>two words</i>. This information is stored in the Auditlog.</p> <p><b>Off</b> - No reason is required by the user, but details of the actions performed are stored in the Auditlog.</p>
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

# Admin Session Settings

Portal menu: **Settings > Workstation Settings > Mac Settings > Authorization > AUTHORIZATION**

## Settings Table - Admin Session

*Admin Session* (also known as User Elevation) elevates the current user's privileges across the endpoint for the duration of the session.

Invoked when the user clicks the menu bar icon to request a protected administrator session, which makes the user a temporary member of the local administrator's group for a limited period of time under full audit.

Setting	Type	Description
Allow Admin Sessions	Toggle On   Off Default: <b>On</b>	<b>On</b> - Allows users to effectively become a local administrator for the number of minutes specified in <i>Access time (minutes)</i> . Enables <i>Require approval</i> , <i>Require reason</i> and <i>Access time (minutes)</i> .  <b>Off</b> - Denies users the ability to become a local administrator. Hides all other options under Admin Session.
Require approval	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Sends a request to the IT team, which must be approved before the request is granted. Makes <i>Require reason</i> mandatory (i.e. must be On).  <b>Off</b> - Allows the user to become a local administrator as soon as the request is made. Makes <i>Require reason</i> optional (i.e. can be either On or Off).
Require reason	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Extends the authentication window and asks the user to enter email address, phone number and reason. This information is stored in the Auditlog.  <b>Off</b> - No further information is required by the user, but user and computer details are stored in the Auditlog.
Access time (minutes)	Integer Default: <b>15</b> (minutes)	The maximum duration in minutes an Admin Session may last. This time must be sufficient for the user to install software or perform any other tasks that require elevation.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

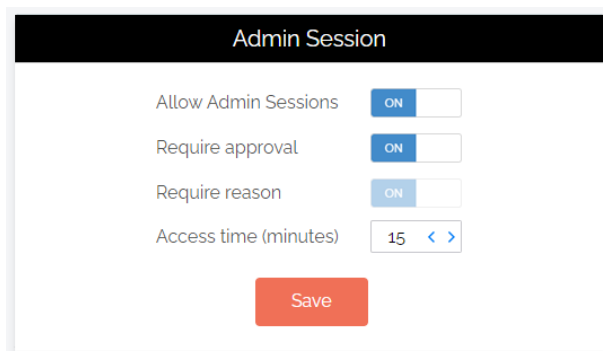
## Changing Admin Session Duration

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.

## Lockdown Settings

The Lockdown menu allows control over the following settings:

- Admin Rights
- Sudo
- Root

### Admin Rights

Portal menu: **Settings > Workstation Settings > Linux Settings > Lockdown > ADMIN RIGHTS**

#### Settings Table - Admin Rights

*Revoke admin rights* at logon means that all user accounts will be downgraded from an Admin role to a User role, unless the account appears in the *Excluded accounts* list.

Excluded accounts are not removed at logon.

Setting	Type	Description
Revoke admin rights	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Admin privileges are removed for all users except those appearing in the <i>Excluded accounts</i> list..  <b>Off</b> - Admin privileges are not removed for users configured locally as administrators.
Excluded accounts	Text	The account name(s) to retain local admin privileges. Multiple accounts must be specified on separate lines. Domain accounts must be prefixed with domain and backslash.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

### Sudo

Portal menu: **Settings > Workstation Settings > Linux Settings > Lockdown > SUDO**

## Settings Table - Sudo

Allowing sudo is strongly discouraged, because it gives the user full control over the endpoint and therefore allows the user to tamper with or completely remove any endpoint software.

Excluded accounts are not removed at login.

Setting	Type	Description
Allow sudo terminal commands	Toggle On   Off Default: <b>Off</b>	<b>On</b> - The logged-in user is in the sudoers file and can run sudo commands. <b>Off</b> - The user cannot run sudo commands, even though they are in the sudoers file.
Allow sudo for non-sudoers	Toggle On   Off Default: <b>Off</b>	<b>On</b> - The logged-in user is not in the sudoers file, but can run sudo commands. <b>Off</b> - The user cannot run sudo commands.
Allow sudo interactive sessions	Toggle On   Off Default: <b>Off</b>	<b>On</b> - The logged-in user can start a sudo interactive session. <b>Off</b> - The user cannot start a sudo interactive session.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

## Root

Portal menu: **Settings > Workstation Settings > Linux Settings > Lockdown > ROOT**

## Settings Table - Root

*Allow root login* controls whether or not it is possible to log in as root on Linux devices.

*Allow root password change* controls whether or not it is possible to change the password of the root account.

Setting	Type	Description
Allow root login	Toggle On   Off Default: <b>Off</b>	<b>On</b> - This endpoint allows users to login as root, or the logged-in user can su (switch user) to root. <b>Off</b> - The endpoint does not allow root logins.
Allow root password change	Toggle On   Off Default: <b>Off</b>	<b>On</b> - This endpoint allows users to login as root, and also allows the logged-in user to change the root password. <b>Off</b> - The endpoint allows root logins, but the root password cannot be changed.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

## Pre-Approval Settings

Portal menu: **Settings > Workstation Settings > Linux Settings > App Control > PRE-APPROVE**

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 Blocklisting/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 allows for quick pre-approval of trusted applications from the Auditlog. Pre-Approval is based on the application vendor or checksum, visible when the *Application Control* screen is displayed (step 3 below).

Once an application has been installed on an endpoint with Admin By Request:

1. Log in to the portal and navigate to the application's corresponding entry in the portal **Auditlog**.
2. Expand on the application entry, and select **Pre-approve this file** under Actions:
3. On the *Application Control* screen, modify any settings as required. For more information on pre-approval settings, refer to the Settings Table below.
4. Click **Save** verify that the app has been added to the list of pre-approved applications.

For example, the following applications are pre-approved:

### Settings Table - Pre-Approve

Pre-approved applications are SUDO commands that are pre-approved to run *Run As Admin*, when approval would normally be required. The intention is to remove trivial approval flows and avoid flooding the audit log with trivial data for applications known to be good, such as Visual Studio or Adobe Reader installs.

When an application is on the pre-approval list, the difference is:

- The application is auto-approved, so the approval flow is bypassed
- A reason is not required, as the application is known to be good
- You have the option to not log to the Auditlog (e.g. for trivial data)
- If *Run As Admin* is disabled, a pre-approved application will still run

### New entry

Click button **New entry** to create a new pre-approved application.

Setting	Type	Description
Log to auditlog (hidden if <i>User confirmation</i> is Off)	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Relevant details about the application are logged. <b>Off</b> - No logging is performed for this application.
User confirmation	Toggle On   Off Default: <b>On</b>	<b>On</b> - The user must confirm elevation on the endpoint before the application can be run. This is the typical authentication window. <b>Off</b> - The user does not need to confirm elevation on the endpoint before execution. Hides the <i>Log to auditlog</i> field.

Setting	Type	Description
Type	Selection Default: <b>Run As Admin application pre-approval</b>	<p><b>Run As Admin application pre-approval</b> - Pre-approve this application for Run As Admin.</p> <p><b>Run As Admin location pre-approval (all files in folder tree)</b> - Pre-approve all applications in the specified folder, including any sub-folders. Selecting this option enables the <i>Directory</i> field and hides all other fields.</p>
Protection	Selection Default: <b>File must be located in read-only directory</b>	<p>Prevent users from bypassing pre-approval by file renaming.</p> <p><b>File must be located in read-only directory</b> - The recommended method. File must be in a read-only location. You only need to know the name and location and you are not bound to a specific file version.</p> <p><b>File must match checksum</b> - A checksum of a specific file version. If the file is updated, the checksum no longer matches and a new one must be collected.</p> <p><b>No protection (not recommended)</b> - Not recommended for anything except testing. The file can be located anywhere and is a file renaming vulnerability, in case a user is aware of (or can guess) the file name.</p>
Directory (enabled when other selections are in effect):	Text	<p>A read-only location where the application to be added is stored.</p> <p>If the directory entered is not on the local machine, a UNC path can be used. The endpoint software will automatically translate drive letters to UNC path.</p>
Application name	Text	The name of the application. Mandatory, although used for convenience only to help identify applications in the list.
File name	Text	Enter file name. Adding the app via the Auditlog will auto-populate this field.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.
Cancel	Button	Cancels all work done in this setting and returns to the Linux Workstation Global Settings page.

## Enabled toggle

A global setting that indicates whether pre-approved applications are allowed at all (**On**) or not (**Off**).

## File Blocking Settings

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:

The screenshot shows the 'Application Control' interface. On the left, the 'Blocked Application' form has the following fields:

- Type:** A dropdown menu with 'Block file from running as administrator' selected.
- Condition:** A dropdown menu with 'No condition (block always)' selected.
- Application name:** A text input field.
- File name:** A text input field.
- Blocking message (optional):** A large text area.
- Internal comments (optional):** A large text area.

On the right, the 'About Blocked Application' sidebar contains the following information:

- Block Application:** allows you to point to a file name that will be blocked from executing.
- Application name:** is only used for convenience in the list.
- Condition:** is when a file is only blocked, if the condition is met:
  - Directory:** File must be located in this directory or a sub-folder.
  - Checksum:** A checksum of a specific file version. If the file is updated, the checksum no longer matches and a new must be collected.
- Blocking Message:** will appear as a rejection to the user, when the application is attempted to be executed.
- Please contact us using the 'Contact' menu, if you have questions about blocking.

*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.

## Settings Table - Block

Blocked applications are effectively the opposite of pre-approved applications - the feature allows you to point to a file name that will be *blocked* from executing rather than pre-approved for execution.

As with other activity, attempts to run blocked applications are recorded in the auditlog.

### NOTE:

Please contact us using the [Contact menu](#), if you have questions about blocking.

## New entry

Click button **New entry** to create a new pre-approved application.

Setting	Type	Description
Type	Selection Default: <b>Block file from running as administrator</b>	<b>Block file from running as administrator</b> - Block this application for Run As Admin.  <b>Block location from running as admin (all files in folder tree)</b> - Block all applications in the specified folder, including any sub-folders. Selecting this option enables the <i>Directory</i> field and hides all fields except the optional fields.
Condition	Selection Default: <b>No condition (block always)</b>	Condition applies only when a file is blocked.  <b>No condition (block always)</b> - The default (and recommended) method.  <b>Block if located in directory</b> - File must be located in this folder or directory to be blocked.  <b>Block if matching checksum</b> - A checksum of a specific file version. If the file is updated, the checksum no longer matches and a new one must be collected.
Directory (enabled when other selections are in effect): <ul style="list-style-type: none"><li><i>Block location from running as admin (all files in folder tree)</i></li><li><i>Block if located in directory</i></li></ul>	Text	File must be located in this directory or a sub-folder.
Application name	Text	The name of the application. Mandatory, although used for convenience only to help identify applications in the list.
File name	Text	The file name of the app to be blocked.
SHA256 checksum (enabled when other selections are in effect): <ul style="list-style-type: none"><li><i>Block if matching checksum</i></li></ul>	Text	A checksum of a specific file version. If the file is updated, the checksum no longer matches and a new one must be collected.

Setting	Type	Description
Blocking message (optional)	Text (multiline)	A message that appears as a rejection to the user, when the application is attempted to be executed.
Internal comments (optional)	Text (multiline)	Optional comments that IT admins might wish to add about the blocked application.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.
Cancel	Button	Cancels all work done in this setting and returns to the Linux Workstation Global Settings page.

### Enabled toggle

A global setting that indicates whether blocked applications are allowed at all (**On**) or not (**Off**).

## Privacy Settings

Portal menu: **Settings > Workstation Settings > Mac Settings > Data > PRIVACY**

### Settings Table - Privacy

The PRIVACY tab provides a way to anonymize data collection, so that data is still logged and available for analysis, but identification of individual users is not possible.

Key points:

- Obfuscation creates an alias for each user. You can track activity, but you cannot decode the true identity of any user.
- Collection of data should be left on unless you have a reason not to do this. If disabled, you will have to find contact information elsewhere.
- Inventory is a hardware and software inventory. If disabled, only the computer name is collected and shown in the "Inventory" menu.
- Geo-tracking maps the endpoint IP address to location using a public IP-to-location database to show in inventory and reports.

#### NOTE:

Changes apply *only to new data*. This is by design to avoid accidentally deleting existing data.

Setting	Type	Description
Obfuscate user accounts	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Create an alias for each user. <b>Off</b> - Do not create aliases for users.
Collect user names	Toggle On   Off Default: <b>On</b>	<b>On</b> - Record the name of each user associated with an ABR event. <b>Off</b> - Do not record user names.
Collect user email addresses	Toggle On   Off Default: <b>On</b>	<b>On</b> - Record email addresses associated with a user. <b>Off</b> - Do not record email addresses.
Collect user phone numbers	Toggle On   Off	<b>On</b> - Record phone numbers associated with a user. <b>Off</b> - Do not record phone numbers.

Setting	Type	Description
	Default: <b>On</b>	
Collect inventory	Toggle On   Off Default: <b>On</b>	<b>On</b> - Record hardware and software inventory data. <b>Off</b> - Do not record inventory data.
Allow geo-tracking	Toggle On   Off Default: <b>On</b>	<b>On</b> - Record the location of the public IP address associated with the user's endpoint. <b>Off</b> - Do not record IP addresses.
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

## Entra ID Support

### NOTE:

Azure AD has been renamed by Microsoft to Entra ID. This version of the document uses both terms interchangeably, but future versions will refer to Entra ID only.

A huge selling point for the Admin By Request PAM solution is its flexibility and tools for granular access control; organizations can configure every setting to their specific needs and the needs of all, some, or even individual users.

Settings act as rules, such as whether the *Run as Admin* or *Admin Session* features are enabled, and whether or not users need approval to use them. You likely wouldn't want the rules applied for an IT Administrator to be the same as those applied for a Customer Relations employee, so settings can be differentiated based on Sub-Settings, which allow different rules to be applied to different users and/or groups.

For all the clients, we've built in support for Entra ID groups, meaning you can now apply Sub-Settings to existing Entra ID / Azure AD user and device groups.

**Tenant Settings**  
Settings here are global tenant settings on top of all other settings. If you have any questions, feel free to contact us [here](#).

**Groups**

- Retention
- API Keys
- Webhooks
- Email Domain
- Policies

**ENTRA ID / AZURE AD**

### Identity Groups

#### Entra ID Connector

Enable Connector ☐ OFF

Tenant

Application ID

Secret Key

Hybrid Preference

National Cloud ☐ OFF

**Save**

#### About Entra ID Connector

Entra ID Connector allows endpoints to retrieve Entra ID (previously Azure Active Directory) groups for subsettings.

If you are using **on-premises Active Directory**, you do not need to configure anything. Collection of groups for Active Directory is configuration-less.

The Entra ID Connector is **NOT** used for single sign-on to the portal; it is solely used for subsetting groups. Example values:

Tenant **acme.onmicrosoft.com**  
Application ID **xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx**  
Secret Key **azVqedkQIVXgbHLBZjGCOZ6+Zih4goI7u53jgWIZN8-**

Hybrid Preferences is when a computer is AD joined and the user made an Azure Workjoin.

[Please refer to this page for Entra ID Connector documentation](#)

For more information on the Entra ID / Azure AD feature, refer to [Features > Azure AD Connector](#).



## Settings Table - Entra ID

The *Entra ID Connector* allows endpoints to retrieve Entra ID (previously Azure AD) groups for sub-settings.

### NOTE:

If you are using on-premise Active Directory, you do not need to configure anything - collection of groups for Active Directory is "configuration-less".

The Entra ID Connector is NOT used for single sign-on to the portal; it is used solely for sub-setting groups. Example values:

- Tenant **acme.onmicrosoft.com**
- Application ID **xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx**
- Secret Key **azVqedkQLVX9bHLBZjGCQZ6+iZlh4gol7u53igWlZN8=**

Refer to <https://learn.microsoft.com/en-us/entra/identity-platform/quickstart-register-app> for more information on registering apps with the Microsoft identity platform.

### NOTE:

The *National Cloud* regions of Azure are designed to make sure that data residency, sovereignty, and compliance requirements are honored within geographical boundaries.

Setting	Type	Description
Enable Connector	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Turns on the Entra ID Connector and allows endpoints to retrieve Entra ID groups for sub-settings.  <b>Off</b> - The Entra ID Connector is disabled and endpoints will use sub-settings as described under "Sub-Settings", rather than using Entra ID rules.
Tenant	Text	Standard email address format. Use a new line for each address.
Application ID	Text	The value assigned to an application when it is registered with the Microsoft identity platform.
Secret Key	Text	The application certificate or client secret generated when the app is registered.
Hybrid Preference	Selection Default: <b>Prefer Active Directory</b>	An option available for selection when a computer is both AD-joined and the user makes an Entra ID Workjoin: <ul style="list-style-type: none"> <li>• <b>Prefer Active Directory</b> - User is AD-joined only.</li> <li>• <b>Prefer Entra ID / Azure AD</b> - User is AD-joined and makes an Entra ID Workjoin.</li> </ul>
National Cloud	Toggle On   Off Default: <b>Off</b>	<b>On</b> - Enables selection of a physically isolated instance of Azure. Unhides <i>National Service</i> , which is where the actual geographic instance is selected.  <b>Off</b> - Disables selection of a physically isolated instance of Azure.
National Service (hidden if <i>National Cloud</i> is Off)	Selection	The geographic instance selected: <ul style="list-style-type: none"> <li>• <b>US Government L4 / GCC High</b> - Azure portal (global service)</li> </ul>

Setting	Type	Description
	Default: <b>US Government L4 / GCC High</b>	<ul style="list-style-type: none"> <li><b>US Government L5 / DoD</b> - Azure portal for US Government</li> <li><b>China (21Vianet)</b> - Azure portal China operated by 21Vianet</li> </ul>
<b>Save</b>	Button	Saves customization and changes to any fields. Note that reloading any defaults does not take effect until <b>Save</b> is clicked.

## Preventing Abuse

So what prevents the user from abusing an Admin Session? The fact that the user has to ask IT for access will in itself prevent the most obvious abuse. But as part of your settings, you can also configure a *Code of Conduct* page. Here you customize wording that suits your company policy. For example, what the penalty is for using the administrator session for personal objectives. You can also choose to explain the things you can monitor from the portal.

When you enable the *Code of Conduct* ("instructions") screen in the settings, this screen appears right before the administrative session starts. You can also customize company name and logo for all screens, so there is no doubt this message is authentic and indeed from the user's own company. This is the configuration part of the portal, where you set authorization, company logo, policies, email communications, etc:

### Linux Global Settings

Settings here are the global settings for all endpoints. You can overrule settings for certain domain users or computers under the sub settings menu. If you have any questions, feel free to contact us [here](#).

Authorization

**Endpoint**

Lockdown

App Control

Data

Emails

LOOK & FEEL

**INSTRUCTIONS**

### Linux Endpoint

#### Run As Administrator Instructions

Show instructions before start
☐ OFF

Code of Conduct

Your request to start the application as administrator has been approved. Please note that while the application is running, IT staff will be able to see the activity. If you have concerns about this, please close right away and contact the IT staff.

Show every time

Save

#### Admin Session Instructions

Show instructions before start
☐ OFF

Code of Conduct

Your request for temporary administrator access has been approved. When you close this screen, you will temporarily become administrator on your computer and a small countdown window will appear on the lower right side of your screen. To run a program with administrative rights, you must right-click a program and select 'Run as administrator'.

Please note that while the countdown window is visible, IT staff

Show every time

Save

## Policies for Linux

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](#).

## Supplementary Technical Information

This section provides more information on the following:

- Local Administrator Accounts
- Sub-Settings
- Sudo
- Tampering

### Local Administrator Accounts

By default, users logging on to 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 accounts Steve, Jo and Mary:

The screenshot displays the 'Linux Global Settings' interface. On the left is a sidebar with navigation links: Authorization, Endpoint, Lockdown (selected), App Control, Data, and Emails. The main content area is titled 'Linux Lockdown' and has three tabs: ADMIN RIGHTS (selected), SUDO, and ROOT. Under the 'ADMIN RIGHTS' tab, there are two sections. The 'Admin Rights' section shows a toggle for 'Revoke admin rights' set to 'ON' and a list of 'Excluded accounts' containing 'Steve', 'Jo', and 'Mary'. A 'Save' button is at the bottom. The 'About Admin Rights' section provides 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:

1. *Linux Settings* (also known as Global Settings) apply to all users by default, **except** those settings overridden under Sub Settings.
2. *Linux Sub Settings*, where you can define special settings based on Active Directory computer or user groups and/or Organizational Unit(s).

Settings here are the global settings for all endpoints. You can overrule settings for certain domain users or computers under the sub-settings menu.

Sub settings will *override* the global settings for the users or computers to which they apply. Both users and computers can be in Active Directory groups or organizational units.

If a user or computer hits multiple sub settings, the first in listed order *that includes the setting concerned* wins.

### Example sub-settings

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.

# 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
<b>Blocklist</b>	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. See also <a href="#">"Pre-Approved List" on the next page</a> .
<b>Elevated Application</b>	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.
<b>Elevated Privileges</b>	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.
<b>Endpoint</b>	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.
<b>Endpoint Security</b>	An holistic approach to securing a network that goes beyond traditional anti-malware and aims to protect every endpoint from potential threats. See also <a href="#">"EDR" on page 47</a> in the glossary.
<b>Horizontal Privilege Escalation</b>	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. See also <a href="#">"Vertical Privilege Escalation" on the next page</a> .

Term	Definition
<b>Just-In-Time Access (JIT)</b>	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.
<b>Lateral Movement</b>	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.
<b>Phishing</b>	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.
<b>Pre-Approved List</b>	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 "whitelist" – a term no longer used. See also <a href="#">"Blocklist" on the previous page</a> .
<b>Privileged Account</b>	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.
<b>Privileged User</b>	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.
<b>Standard User Account</b>	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.
<b>Vertical Privilege Escalation</b>	Occurs when a lower-privileged account gains privileged access beyond 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). See also <a href="#">"Horizontal Privilege Escalation" on the previous page</a> .

## Glossary

The following table lists the meanings of many acronyms used when discussing privileged access and endpoint protection.

Term	Short for	Definition
<b>Azure AD</b>	Azure Active Directory	Azure Active Directory is part of Microsoft Entra, which is an enterprise identity service that provides single sign on, multi-factor authentication, and conditional access to guard against security threats.
<b>Entra ID</b>	Microsoft Entra	Microsoft Entra is a family of multi-cloud identity and access solutions that includes Azure AD. The term "Entra ID" replaces the term "Azure AD".
<b>EDR</b>	Endpoint Detection and Response	A method of securing endpoints that focuses on detecting and responding to threats that are present. Works in conjunction with EPP.
<b>EPP</b>	Endpoint Protection Platform	A method of securing endpoints that focuses on preventing threats from arriving. Combines analysis, monitoring & management, anti-malware software, EDR capabilities and other security features into a comprehensive endpoint security platform.
<b>FIDO</b>	Fast Identity Online	<p>With FIDO Authentication, users sign in with phishing-resistant credentials, called "<a href="#">Passkey</a>" <a href="#">on the next page</a>. 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>
<b>Intune</b>	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.
<b>MAM</b>	Mobile Application Management	Software and processes that secure and enable IT control over enterprise applications on end users' corporate and personal devices.
<b>MDM</b>	Mobile Device Management	A methodology and toolset used to provide a workforce with mobile productivity tools and applications, while keeping corporate data secure.

Term	Short for	Definition
<b>PAM</b>	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.
<b>Passkey</b>	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.
<b>POLP</b>	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.
<b>UEM</b>	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	Initial document release
1.1 7 August 2023	3.0 7 August 2023	<p>Included 3.0 features</p> <ul style="list-style-type: none"> <li>Improved Run As Admin sudo sessions</li> <li>Support for pre-approved and blocked applications in sudo sessions</li> <li>Revoke admin rights now automatically removes local admin groups from user accounts</li> <li>Audit logging on programs executed in sudo sessions</li> <li>Option for disabling interactive sudo sessions</li> <li>Option for disabling the root account</li> <li>Option for allowing or disallowing changing of the root password</li> <li>Integration of identity management such as Active Directory, FreeIPA and LDAP domain using SSSD client software</li> </ul> <p>Applied new document template and formatting</p>
1.2 16 February 2024	3.0 7 August 2023	<p>Added proxy server configuration section.</p> <p>Added multiple Settings Tables to chapter Portal Administration.</p> <p>Restructured User Interface chapter.</p> <p>Fixed pagination.</p>
1.3 28 March 2024	3.0 7 August 2023	<p>Added Settings Table for Linux Settings &gt; Data &gt; PRIVACY.</p> <p>Added Settings Tables for tabs under Linux Settings &gt; Lockdown.</p> <p>[Online only] Added FAQ explaining distorted fonts under UI scaling.</p> <p>Updated portal menu selection paths.</p>
1.4 24 May 2024	3.1 22 May 2024	<p>Added Command Line Interface chapter.</p> <p>Updated screenshots for v3.1.9</p> <p>Added remaining settings tables for <i>Endpoint</i> and <i>App Control</i> menus.</p>

# Index

## A

About ABR	
About .....	12
Connectivity .....	16
About Admin By Request .....	12
Admin Rights	
Tab .....	33
Admin Session	
Settings .....	32
Administrator Access	
User Interface .....	19
App .....	18
Audience .....	5
Auditlog .....	28
Azure AD .....	40

## B

BIOS .....	8
------------	---

## C

Check version .....	22
chmod .....	6
Command	
abr finish .....	23
abr settings .....	24
abr start .....	25
abr status .....	26
abr version .....	26
Command Line Interface .....	22
Command option	
abr --help .....	27
abr --log-level .....	28
abr --master-config-file .....	27

abr --system-config-file .....	28
Component	
CLI plugin .....	15
GUI .....	13
Main module .....	13
PAM plugin .....	14
Polkit plugin .....	15
Service .....	14
sudo plugin .....	16
Condition (blocking) .....	37

## D

Download .....	6
Duration .....	18

## E

Entra ID .....	40
Execution history .....	18

## F

File Blocking	
Settings .....	37
File Locations .....	10

## G

GRUB menu .....	8
GUI User Interface .....	11

## I

Install .....	6
IP addresses .....	17

## L

Local Administrator Accounts .....	43
Lockdown	
Setting .....	33

## O

Overview .....	5
----------------	---

## P

Packages .....	8
Performance .....	10
Policies	
Linux .....	43
Policy file .....	43
Portal .....	6
Portal Administration	
Linux .....	30
Ports .....	17
Pre-Approval	
Settings .....	35
Prerequisites .....	6
Preventing Abuse .....	42
Privacy	
Settings .....	39
Proxy Server .....	17

## R

Red Hat EL 9 .....	6
Release Notes .....	5
Root	
Tab .....	34

root user .....	9
Run As Admin	
Settings .....	31
User Interface .....	18

## S

Single app (execution of) .....	18
Sub-Settings .....	44
sudo .....	6, 8, 18
Sudo .....	44
Tab .....	33
Supplementary Technical Information ...	43

## T

Tamper prevention .....	9
Tampering .....	44
Test .....	7
Type (blocking) .....	37

## U

Ubuntu 20.04 .....	6
Ubuntu 22.04 .....	6
UEFI .....	8
Uninstall .....	6
Upgrading .....	8
User accounts .....	21
User rights .....	9