Policies - Apple

- Apple policies
- Apps & profiles
- Kiosk and Home Screen
- Passcode
- Restrictions
- Networking
- Software updates
- macOS
- Location and geofences
- Accounts
- Compliance rules

Apple policies

Apple policies define management settings that Cerberus Enterprise applies to Apple devices via MDM. These settings are configured from the dashboard in the Apple policy editor.

Before you start

Apple device management requires Apple Management (APNs) to be configured. If needed, read the Apple Management setup (APNs) page.

Open the Apple Policy Editor

In the dashboard, open **Policies** and click **Create new Apple policy**. To edit an existing Apple policy, click its row in the policies table.

Editor layout

The Apple Policy Editor is organized as a set of expandable sections. At the top of the page you can always edit:

- **Name** (required)
- **Id** (read-only)
- **Description** (optional)

Policy sections

The sections below match the panels currently available in the Apple Policy Editor. When you open a saved policy, the sections with changed settings are already open and the others stay closed:

- **App management**: configure app-related restrictions, managed apps and the Cerberus Enterprise app settings.
- **Kiosk and Home Screen**: Single App Mode, allowed and blocked apps, Home Screen layout, web clips, notification settings and Lock Screen message.

- **Security**: control features such as Auto Unlock and biometric unlocking.
- **Passcode settings**: configure passcode requirements and related rules.
- **Multimedia**: allow/disallow camera and related features.
- **Cellular**: control cellular-related settings (app cellular data settings, eSIM, plan changes), and the cellular APN.
- **Networking**: Wi-Fi networks (also enterprise 802.1X), VPNs, certificates, global proxy, encrypted DNS and web content filter, and connectivity restrictions such as AirDrop and Bluetooth.
- **System and user interface**: restrict system features such as Handoff, the device name and the screen saver.
- **Software updates**: require an OS version by a deadline and control how updates are downloaded and installed.
- **macOS**: firewall, Gatekeeper, screen saver and lock, external and network storage, Screen Sharing, privacy permissions and login items of the Macs.
- **Apple apps and services**: restrict Apple apps and services such as iMessage, Game Center and Find My.
- **Siri and Apple Intelligence**: restrict Siri, dictation and the Apple Intelligence features.
- **iCloud**: allow or disallow specific iCloud services (backup, keychain sync, private relay, etc.).
- **Location and geofences**: areas where iPhones and iPads with the Cerberus Enterprise app report when they enter or leave, for event rules.
- **Accounts**: restrict account modification and configure Mail, Exchange, Google, calendar, contacts and directory accounts.
- **Compliance rules**: requirements (passcode, minimum OS version, supervision, FileVault) that make devices non-compliant, with an optional block.

> Many options in the Apple Policy Editor include a tooltip that documents requirements and supported OS versions.

Save, delete, and associated devices

Use **Save policy** to apply your changes. The button is disabled when there are no pending edits, or when the license is expired.

When editing an existing policy, the page also shows a **Delete policy** action. The editor can show an **Associated devices** list at the bottom, so you can see how many devices are currently using the policy.

Next pages

- Apps & profiles: configure installed apps and configuration profiles.
- Kiosk and Home Screen: Single App Mode, allowed apps, Home Screen layout and web clips.
- Passcode: configure passcode requirements and related security options.
- Restrictions: define allowed features and OS-level limitations.
- Networking: Wi-Fi networks, VPNs, certificates and connectivity restrictions.
- Software updates: required OS updates and software update settings.
- macOS: firewall, Gatekeeper, screen lock, storage, privacy permissions and login items of the Macs.
- Location and geofences: areas that iPhones and iPads report entering and leaving, and what they need to work.
- Accounts: mail, Exchange, Google, calendar, contacts and directory accounts.
- Compliance rules: requirements for the devices and blocking non-compliant devices.

Apps & profiles

This section documents how to configure managed applications and account payloads for Apple devices.

App management

The **App management** panel contains both general app-related restrictions and a list of managed apps.

General app restrictions

- **Allow app clips**
- **Allow app installation**
- **Allow app removal**
- **Allow automatic app downloads**
- **Allow apps to be hidden**
- **Allow apps to be locked**
- **Allow In-App Purchases**
- **Allow default browser modification**
- **Allow enterprise app trust**

Managed apps

Use **Add application** to add an app to the policy. Each managed app is displayed as a card. You can expand the card to edit its settings and remove the app using the delete action.

- **App Store ID**: the App Store identifier of the managed app.
- **Bundle ID**: the app bundle identifier.
- **Install behavior**: controls whether the app must remain installed or can be installed/removed by the user.
- **Assignment**: the license type, chosen by Cerberus for each device: a device license on company devices, a user license of the Managed Apple Account on personal iPhones and iPads (User Enrollment). The field can't be changed.
- **VPP license**: VPP license type used for installation through the App Store.

The app selection list shows the available and total licenses of each app. Free apps show **Unlimited**: Apple gives them unlimited licenses. An app can be added to a policy only once: an

app with the same App Store ID or bundle ID as an app already in the policy is refused, and so is a policy that lists the same app twice when you save it.

The app selection list shows the apps of your Apple Business library. To add an app, in Apple Business open **Apps & Services** → **Apps & Books** → **App Store**, search for the app and click **View**. For a free app, click **Add**, select **All current and future** (or the organizational unit of your Apps and Books token) in **Organizational Units**, click **Add**, then **Review** and confirm. For a paid app, click its price, choose **Managed Licenses** as **License Type** (redemption codes can't be used by Cerberus Enterprise), select the **Organizational Unit** of your Apps and Books token, enter the number of **Licenses** (up to 1,000 per order), then **Review** and buy. A payment method must be set up first in **Settings** → **Payments & Billing** → **Apps & Services** → **Payment Methods**. The apps then appear in the app selection list.

Apps on personal devices (User Enrollment)

Personal iPhones and iPads enrolled with User Enrollment have no serial number, so they get the App Store apps of the policy with a **user license** of the Managed Apple Account signed in on the device. Cerberus registers the account with Apps and Books and assigns the licenses by itself, free apps included. The Managed Apple Accounts and the Apps and Books token must belong to the same Apple Business organization. A paid app uses one license per account, shared by all the personal devices of that account.

- **Waiting for a user license**: the app is sent to the device only after Apple confirms its license, usually within a few minutes. The **Applications** panel of the device shows the reason.
- **User license failed**: Apple couldn't assign the license, for example because a paid app has no license left. Add licenses in Apple Business, then use **Request again**; Cerberus also tries again by itself every day.
- **Install prompt**: iOS asks the user before installing each app. If the user cancels, the app shows **Waiting for the user to approve the installation**; **Request again** shows the prompt again.
- **Release**: removing the app from the policy, or unenrolling the last personal device of the account, removes the app and releases its license, unless another enrolled device of the same account still needs it.

For the details, see **Apple account-driven enrollment**.

Managed configuration

A managed configuration gives an app its settings (for example a server address or an account name), the way the app's vendor documents them. Open the app in the policy and click **Add**

managed configuration. It's sent to iPhones and iPads with **iOS or iPadOS 18.4 or later** and to Macs with **macOS 27 or later**; it isn't sent to Apple Vision Pro yet. The app must support managed configuration: an app that doesn't read it ignores it. The app panel shows the number of keys next to the app name.

- **Keys**: each row has a key, a type and a value. The types are Text, Integer, Decimal number, Boolean, Date, Text list and Integer list. Keys are case-sensitive and must be unique. **Add key** adds a row; **Remove configuration** removes all the keys.
- **Variables**: Text and Text list values can contain $EMAIL$, $USERNAME$, $FULLNAME$, $SERIAL$ and $UDID$, replaced with the values of each device and of its user. A variable with no value (for example $EMAIL$ on a device without a user) is left empty, and the system logs say so. When a user's details change, the devices get the new values; for users synchronized from Google Directory the values are updated at the next change of the policy or of the device.
- **Import a configuration**: open or paste the file that the app's vendor publishes for MDM, then click **Read**. An **AppConfig specfile** (specfile.xml) lists every key the app supports with its description, type, default value and allowed values: select the keys to import (those with a default value are already selected) and fill in the others after the import. A warning appears when the specfile is for another app. A **property list** (a whole property list or only its <dict>) is imported as it is. Then choose whether to replace the current keys or add to them. Nested dictionaries, data values and lists with mixed or unknown types aren't supported and are listed as skipped. Apple doesn't publish the configuration keys of apps, so ask the vendor for the file when it isn't in their documentation. **Copy as plist** copies the configuration as the device receives it, before the variables are replaced.
- **Limits**: up to 200 keys per app; keys up to 255 characters; text values up to 4,096 characters; lists up to 200 items of up to 1,024 characters each; dates between 1970 and 9999; 64 KB per app in all.
- **Not for passwords**: the configuration is sent only to the device it's built for and the logs show only key names and types, but dashboard administrators can read the values, and the address the device downloads it from changes with the values, so a value that is easy to guess (a boolean or a small number) could be inferred from it. Don't put passwords or other secrets in a managed configuration.
- **Status**: when the device reports that the app didn't accept its configuration, the Applications panel of the device shows **Configuration not accepted** next to the app. It doesn't make the device non-compliant. A configuration that the device can't download does make it non-compliant.

App options

Each app of the policy (App Store and custom apps) has these options. **Device default** leaves the choice to the device. Devices that don't support an option don't get it; the app is still installed.

- **Automatic updates**: App Store setting, Always or Never. iOS, iPadOS, macOS and visionOS 26 or later. For custom apps, App Store setting behaves like Never.

- **Downloads over cellular**: App Store setting, Always or Never, for automatic installs and updates. Supervised iPhones and iPads with iOS or iPadOS 26 or later only.
- **Back up the app data**: whether the app data is included in the device backups. Not on Mac.
- **User can hide the app** and **User can lock the app**: iOS and iPadOS 18.1 or later, enforced on supervised devices. Not allowing locking also prevents hiding. The restrictions **Allow apps to be hidden** and **Allow apps to be locked** apply to all apps.
- **Lock the device after Tap to Pay transactions that need the customer's PIN**: iPhone only.
- **Associated domains**: the domains the app is associated with (for example for universal links or password autofill), up to 50. Enter plain host names such as www.example.com, without applinks:, wildcards or ?mode=. Not on Mac.
- **Verify the domains directly**: the device checks the domains' site association file directly instead of through Apple's servers. Only turn it on for internal domains that can't be reached from the internet.
- **Removal**: apps installed as **Required** can't be removed by the user.

Mac App Store apps

The app selection list shows the platforms of each app (iPhone/iPad, Mac, Apple TV, Vision); an app that exists only for Mac shows its Mac icon. An app that exists only for Mac applies to Macs when you add it. An App Store app without a Mac version isn't sent to Macs: the policy shows "This app has no Mac version: it isn't sent to Macs" under **Applies to**. Macs get App Store apps by their App Store ID, so apps with a different bundle ID on Mac are installed correctly.

Cerberus Enterprise app

Cerberus installs its **Cerberus Enterprise** app on every iPhone and iPad with **iOS or iPadOS 18.4 or later**, with or without a policy. The app shares the device location and receives notifications and live tracking requests. It's installed as **Required**, with a device license from Apps and Books (a user license on personal devices), so the user can't delete it.

- **One-time setup**: add the free app to your library once. In Apple Business, open **Apps & Services** → **Apps & Books** → **App Store**, search for "Cerberus Enterprise", click **View**, then **Add**, select **All current and future** in **Organizational Units**, click **Add**, then **Review** and confirm. If the app shows **Manage**, it's already in your library: open it and check that its organizational units are **All current and future units**. Until then the dashboard shows a banner and the devices show the app as not installed. See **Account settings**.
- **Not in policies**: the app can't be added to a policy. It is disabled in the app selection list and refused as a custom app (same bundle ID). A policy that already lists it gets the entry removed when you open it: save the policy to confirm.

- **Not installed on**: devices with iOS or iPadOS older than 18.4. Shared iPad is supported for the install, but the app features aren't verified there.
- **Personal devices**: on iPhones and iPads enrolled with User Enrollment, the app is installed too, with an install prompt and a user license of the Managed Apple Account. It receives messages from your organization, and it doesn't access the location unless the policy turns on **Location and geofences on personal devices** and the user allows it.
- **Kept installed**: if the Apps and Books token expires or the app leaves your library, the app stays on the devices and the device page says that Apps and Books needs attention. It's removed only if the app is still missing from your library 24 hours later.

Custom apps

Custom apps are your own iPhone and iPad apps (.ipa files), distributed without the App Store. In Xcode, export the app for **Enterprise** distribution (Apple Developer Enterprise Program: it installs on every device) or **Ad Hoc** distribution (it installs only on the devices registered in its provisioning profile). App Store builds can't be uploaded.

Click **Add custom app** to open the custom apps of your organization. **Upload app** adds an .ipa file (up to 2 GB). Select an app and click **Add to the policy**. The list shows the version, the type of signing, the number of devices in the profile of ad hoc apps and the expiry date of the provisioning profile.

- **New version**: upload an app with the same bundle ID. It replaces the previous version, and the devices of the policies that use it update the app.
- **Requirements**: iPhone and iPad with iOS 17.2 or later. Devices that can't install the app don't get it: an iPad-only app on an iPhone, an older iOS than the app needs, a device that isn't in the profile of an ad hoc app, or an expired provisioning profile.
- **Expiry**: when the provisioning profile expires, the app no longer opens. Upload a version signed with a new profile before the expiry date.
- **Reserved bundle ID**: a custom app with the bundle ID of the Cerberus Enterprise app (com.lsdroid.Cerberus.Enterprise) can't be added to a policy, because Cerberus installs that app by itself.
- **Delete**: remove the app from its policies first; the devices then remove it.

Mac packages

The custom apps also accept Mac installer packages (.pkg files). The package must be a distribution package built with **productbuild** (a component package made only with pkgbuild is refused) and signed with a **Developer ID Installer** certificate. Notarizing it and stapling the ticket (**notarytool** and **stapler**) is recommended. Unsigned packages, packages changed after signing and packages signed with other certificates are refused.

Upload the .pkg file with **Upload app** and add it to the policy: it appears under **Mac packages**. iPhones and iPads ignore it.

Macs with **macOS 26 or later** (supervised, which includes Macs enrolled manually, but not User Enrollment) get the package as a **declaration**: the Mac downloads, installs, retries and updates it by itself and reports its state. Older Macs, and packages installed as managed apps, get it with an **MDM command**, on any macOS version. A Mac that is upgraded to macOS 26 moves to the declaration by itself; it may install each package once more. The same can happen when a Mac moves to another policy that has the same package.

- **New version**: upload a package with the same package identifier. It replaces the previous version, and the Macs of the policies that use it install the new one.
- **Skipped Macs**: Macs with an older macOS than the package allows, and Intel Macs for packages built only for Apple silicon. The device page shows the package as Skipped, with the reason; the version installed before stays on the Mac.
- **Install**: **Required** (default) installs the package on the Mac; **Optional** makes the package available on the Mac without installing it. macOS 26 doesn't show optional packages to the user in System Settings or the App Store (tested on macOS 26.3), so an app or a script on the Mac has to install them; choose Optional only if you have one. It applies to Macs with macOS 26 or later; older Macs always install the package. The choice isn't available for packages installed as managed apps.
- **Remove the package files when it leaves the policy (macOS 27 or later)**: when the package is removed from the policy, Macs with macOS 27 or later delete the files it installed. It's off by default. On older Macs the files stay. Not available for packages installed as managed apps.
- **Install as a managed app**: turn it on for a package of the policy to install its app as a managed app (macOS 11 or later, not with user enrollment). macOS then reports when the installation fails, and MDM can remove the app. The app must be installed directly in the Applications folder, signed with a **Developer ID Application** certificate and notarized; on macOS 11 to 13 the package must contain only that app (no other files or scripts). The custom apps show which packages qualify. When the option can't be applied on a Mac, the package is installed there as usual and the device page says why.
- **Removal options** of a managed app: **Remove the app when the package is removed from the policy** (on by default) and **Remove the app when the Mac leaves management**. Removing an app also deletes its data. If the user had already installed the same app, that copy becomes managed and is removed too.
- **Removal**: the MDM command can't uninstall a package. For a package installed as a managed app, removing it from the policy removes the app (when the option is on); its other files and the package receipt stay. Packages with **Remove the package files** on are removed from Macs with macOS 27 or later. Other packages: removing them from the policy only stops new installs, the files stay on the Macs. A Mac that is skipped (for example after a new version that needs a newer macOS) keeps the app while the package is in the policy; removing the package from the policy then removes it too. Remove leftover files on the Mac (for example with the package's own uninstaller), and run `sudo pkgutil --forget <identifier>` to forget the package receipt.
- **State on the device**: the Applications panel of the Mac shows each package of the policy as Queued, Accepted (the Mac accepted the command), Downloaded, Installed (only for packages that contain apps, when the apps are found on the Mac) or Failed (with the

error). macOS doesn't report the installation of a package sent with the command, so Downloaded is the last state for packages without apps. A package installed as a managed app also becomes Failed when macOS reports that the app couldn't be installed or was removed from the Mac. **Install again** sends the package to the Mac again.
- **State with the declaration**: packages delivered as a declaration show the **Declaration** chip and the state the Mac reports: Queued, Available (an optional package the user hasn't installed), Waiting for the user, Downloading, Installing, Installed or Failed (with the reason, for example a failed download or installation, or a declaration the Mac rejected). The **Files removed with the policy** chip marks packages whose files a macOS 27 Mac removes when they leave the policy. **Install again** appears only after a failure and asks the Mac to try again; it isn't offered when the Mac rejected the declaration.

Accounts

The **Accounts** panel lets you configure accounts that are applied to managed devices. It also includes a restriction toggle for account modification.

Restriction

- **Allow account modification**: when disabled, users cannot modify accounts such as Apple Accounts and internet accounts.

Add accounts

Use **Add Google account** or **Add mail account** to add account payloads to the policy. Each account appears as a card with its configuration fields.

Account credentials from users

Both Google and Mail account cards provide a **Account credentials from users** toggle. When enabled, the system applies account credentials on a per-user basis. When disabled, you enter the account identity in the policy.

Google account fields

- **Visible name**: the name shown to the user for the account.
- **Google email address**: the user email address.
- **Full name**: the user’s full name.

Mail account fields

Mail accounts include identity fields plus incoming/outgoing server configuration. Host names are required.

- **Visible name**: the name shown to the user for the mail account.
- **Email address**: the user email address.
- **Full name**: the user's full name.

Incoming server

- **Server type**: mail protocol (for example IMAP or POP).
- **Authentication Method**: authentication method for the server.
- **IMAP path prefix**: shown only when Server type is IMAP.
- **Host name**: required.
- **Port**: server port (1–65535).

Outgoing server

- **Authentication Method**
- **Host name**: required.
- **Port**: server port (1–65535).

S/MIME options

For Mail accounts, you can also configure S/MIME encryption and signing behavior.

Encryption

- **S/MIME encryption**
- **Identity user-overrideable**
- **Per-message switch enabled**
- **User overrideable**

Signing

- **S/MIME Signing**
- **Identity user-overrideable**
- **User overrideable**

> Account and restriction options include tooltips in the dashboard that document prerequisites and supported OS versions.

Kiosk and Home Screen

The **Kiosk and Home Screen** section of the Apple Policy Editor locks devices into one app, chooses which apps can be opened, arranges the Home Screen and adds website icons. Each setting is installed on the devices as a configuration profile.

> Apple applies most of these settings only to supervised iPhone and iPad (devices enrolled with Automated Device Enrollment or prepared with Apple Configurator). Other devices don't get them. Web clips also work on unsupervised iPhone and iPad. Macs don't get any of these settings.

Choosing apps

Apps are identified by their bundle ID. Type the name of an app to pick it from the built-in Apple apps and the apps of the policy, or type any bundle ID, for example `com.example.app`.

Single App Mode

The device runs only the chosen app: users can't go back to the Home Screen or open other apps. Install the app with the policy first. The device leaves Single App Mode when you turn the setting off or remove the app.

- **Options**: disable the touch screen, rotation, the volume buttons, the Ring/Silent switch, the side button or Auto-Lock, and turn on accessibility features such as VoiceOver or Zoom.
- **Accessibility features the user can turn on**: features the user can turn on with a triple-click of the side button.

Allowed and blocked apps

- **Blocked apps**: the devices hide these apps and don't open them. `com.apple.webapp` blocks all the web clips. Settings and, on iPhone, Phone can't be blocked.
- **Allow only some apps**: the devices show only these apps. Leave the list empty to allow all apps. **Policy apps** adds the apps installed by the policy; `com.apple.webapp` allows all the web clips.

- **Apps that can start Single App Mode**: apps that can lock the device into themselves while they need it, for example during an exam.

> With **Allow only some apps**, every app that isn't in the list disappears from the devices, including the apps of the policy. Settings and, on iPhone, Phone always stay.

Home Screen layout

Arrange the Dock and the pages of the Home Screen. Add apps, folders of apps and the web clips of the policy, choosing where with **Add to**, and drag the icons to move them between the Dock and the pages. Click a folder to change its name and apps.

Users can't move the icons or remove apps from the Home Screen. Apps that aren't in the layout go to the pages after the ones you defined. An iPhone shows four icons in the Dock. Users can still delete apps that aren't Required in the storage screen of Settings. To stop that, turn off **Allow app removal** in App management.

Web clips

A web clip is an icon on the Home Screen that opens a website. Set the label and the URL, and optionally an image (PNG or JPEG, cropped to a square).

- **Users can remove it**: turn off to keep the icon on the devices.
- **Full screen**: the website opens as a web app, without the Safari address bar. **Stay full screen on other websites** also opens the links to other websites in the web app.
- **Open with this app**: an app that opens the URL instead of Safari.

Notification settings

Choose how the devices show the notifications of some apps: whether they are allowed, where they appear (Lock Screen, Notification Center, CarPlay), badges, sounds, critical alerts, banner style, grouping and previews. Users can't change these settings.

Lock Screen message

A message shown at the bottom of the Lock Screen, for example who to call if the device is found, and an asset tag. Both can contain the variables $EMAIL$, $USERNAME$, $FULLNAME$, $SERIAL$ and $UDID$: a variable without a value for the device is left empty.

Passcode

The **Passcode settings** section controls device passcode requirements and related security rules (for example, minimum length and complexity). On a Mac (macOS 13 or later, not with User Enrollment) the rules apply to the login password of the users.

Options

In the Apple Policy Editor, passcode options are configured using a mix of toggles and numeric fields. Many fields include tooltips that indicate supported OS versions and supervision requirements.

Passcode toggles

- **Change at next authentication**: force a passcode reset the next time the user authenticates.
- **Require alphanumeric passcode**: require at least one alphabetic character and one number.
- **Require complex passcode**: require a “complex” passcode (no repeating/sequential patterns and at least one non-alphanumeric character).
- **Require passcode**: require a passcode without additional length/quality requirements. Note: setting other passcode keys implicitly requires a passcode.

Numeric fields

- **Failed attempts reset (minutes)**: minutes before the failed-attempts counter resets (requires Maximum failed attempts).
- **Maximum failed attempts**: failed attempts allowed before an iPhone or iPad is erased, or a Mac is locked (range: 2–11).
- **Maximum grace period (minutes)**: how long the device can be unlocked without requiring the passcode (0 = none).
- **Maximum inactivity (minutes)**: idle time before the device locks (range: 0–15).
- **Maximum passcode age (days)**: max passcode age before a forced change (range: 0–730).
- **Minimum complex characters**: minimum number of “complex” characters (range: 0–4).
- **Minimum length**: minimum passcode length (range: 0–16).
- **Passcode reuse limit**: passcode history length to prevent reusing old passcodes (range: 1–50).

Restrictions

The Restrictions sections control which OS features are allowed on managed Apple devices. In the Apple Policy Editor, these options are exposed as grouped panels with multiple toggles.

The toggles of the **App management** and **Networking** panels are described in the Apps & profiles and Networking pages of Apple policies.

> Many restrictions are only supported on specific OS versions and may require supervised devices. Use the tooltips in the dashboard for authoritative requirements.

> On User Enrollment devices (personal devices enrolled by signing in with a work account), the restrictions that Apple doesn't allow there are not sent. If none of the policy's restrictions applies, these devices get no restrictions profile. The policy editor shows how many restrictions don't apply to User Enrollment devices.

Security

- **Allow auto unlock**
- **Allow Fingerprint for Unlock**
- **Allow Fingerprint Modification**
- **Allow erase content and settings**
- **Allow enabling restrictions**
- **Allow profile installation by users**
- **Allow host pairing**
- **Allow diagnostic submission**
- **Allow diagnostic submission modification**
- **Allow Apple personalized advertising**
- **Allow Remote Management modification**

Multimedia

- **Allow camera**
- **Allow File Sharing Modification**

- **Allow Files USB Drive Access**
- **Allow call recording**
- **Allow AirPrint**
- **Allow AirPrint credentials storage**
- **Allow AirPrint iBeacon discovery**

Cellular

- **Allow app cellular data modification**
- **Allow cellular plan modification**
- **Allow eSIM modification**
- **Allow eSIM outgoing transfers**
- **Keep the eSIM when the device is erased**
- **Allow background fetch when roaming**

Cellular APN

Cellular APN sets the access point name (APN) that your carrier gave you, for example for a private or business mobile network. Set the **access point name** and, if the carrier asks for them, the authentication (PAP or CHAP), the user name (variables such as $EMAIL$ are allowed), the password, the IP versions (also when roaming), a proxy server with its port and XLAT464. Turn on **Attach APN** only if the carrier gives you a separate APN for connecting to the network.

- Only iPhone and cellular iPad get it, also without supervision. Devices enrolled with User Enrollment (personal devices) don't get it.
- It applies to the SIM used for cellular data only. A device that already has an APN profile installed in another way doesn't install it.
- The password is protected like the other network passwords: the profile is sent only to the device it was made for and the password isn't written in the logs.

> A wrong APN leaves the devices without cellular data: try the settings on a test device with a SIM of the carrier first.

System and user interface

- **Allow Handoff**
- **Allow autocorrection**
- **Allow auto-dim**

- **Allow automatic screen saver**
- **Allow QuickPath keyboard**
- **Allow definition lookup**
- **Allow device name modification**
- **Allow device sleep**
- **Allow iPhone Mirroring**
- **Allow iPhone widgets on Mac**
- **Allow keyboard shortcuts**

Apple apps and services

- **Allow iMessage**
- **Allow Game Center**
- **Allow Adding Game Center Friends**
- **Allow Find My device**
- **Allow Find My friends**
- **Allow Find My friends modification**
- **Allow Book Store**
- **Allow Book Store erotica**
- **Allow enterprise book backup**
- **Allow enterprise book metadata sync**
- **Allow iTunes Store**
- **Allow explicit content**

Siri and Apple Intelligence

- **Allow Siri**
- **Allow Siri user-generated content**
- **Allow Siri while locked**
- **Allow dictation**
- **Allow external intelligence integrations**
- **Allow external intelligence sign-in**
- **Allow Genmoji**
- **Allow Image Playground**
- **Allow Image Wand**

iCloud

- **Allow iCloud address book**
- **Allow iCloud backup**
- **Allow iCloud bookmarks**
- **Allow iCloud calendar**
- **Allow iCloud desktop and documents**
- **Allow iCloud document sync**
- **Allow iCloud Freeform**
- **Allow iCloud keychain sync**
- **Allow iCloud Mail**
- **Allow iCloud Notes**
- **Allow iCloud Photo Library**
- **Allow iCloud Private Relay**
- **Allow iCloud Reminders**

Accounts (restriction)

The Accounts panel contains both a restriction and (optionally) account configuration. The restriction toggle controls whether the user can modify system accounts.

- **Allow account modification**

Networking

The **Networking** section of the Apple Policy Editor configures Wi-Fi networks and VPNs on the devices, installs certificates and restricts connectivity features such as AirDrop and Bluetooth. Each Wi-Fi network, VPN and certificate is installed as its own configuration profile, which you can see on the device in **Settings > General > VPN & Device Management** (Mac: **System Settings > General > Device Management**, or **Privacy & Security > Profiles** on macOS 13 and 14) and in the **Profiles, certificates and security** panel of the device page.

> Certificates are taken from the **Certificates** page: import them there first. The policy only refers to them, so a certificate is never copied in the policy.

Restrictions

- **Allow AirDrop**
- **Allow AirPlay incoming requests**
- **Allow Bluetooth modification**
- **Allow Bluetooth Sharing modification**
- **Allow Files Network Drive Access**
- **Allow Internet Sharing Modification**
- **Allow content caching**
- **Allow iTunes file sharing**

Wi-Fi networks

Click **Add Wi-Fi network** and fill in:

- **Network name (SSID)**: the name of the network, up to 32 characters.
- **Security**: none (open network), WEP, WPA or WPA2, WPA2 or WPA3, WPA3 only, or any.
- **Password**: the password of a personal network.
- **Join automatically** and **Hidden network** (for networks that don't broadcast their name).
- **Proxy**: none, manual (server and port) or automatic (a PAC file URL, or empty to find the settings on the network with WPAD).
- **Profile name** (optional) and **Applies to**, to limit the network to some device types.

Enterprise networks (802.1X)

Turn on **Enterprise (802.1X)** when users authenticate with their own credentials or a certificate instead of a shared password.

- **Authentication**: PEAP, TTLS (with its inner authentication, for example MSCHAPv2) or TLS (certificate only).
- **Credentials of the device user**: each device uses the EAP identity, anonymous identity and password of its user, set in the Users page. Devices without a user don't get the network.
- **Identity**, **Anonymous identity** and **Password**: the same for every device. The identity can contain the variables $EMAIL$, $USERNAME$, $FULLNAME$, $SERIAL$ and $UDID$. Leave the password empty to let the user type it on the device.
- **Identity certificate**: the client certificate the device presents (required for TLS). See the certificate options below.
- **Trusted certificates**: the CA certificates that issued the certificate of your RADIUS server. They are installed with the network.
- **Trusted server names**: the names of the RADIUS server certificates the devices accept, for example radius.example.com or *.example.com.

VPN

Click **Add VPN** to configure an IKEv2 or Cisco IPSec VPN, built into iOS, iPadOS and macOS (no VPN app needed).

- **Type** and **Server**. For IKEv2 also the **Remote identifier** (the server identity, empty means the server address) and the **Local identifier**, which can contain the variables $EMAIL$, $USERNAME$, $FULLNAME$, $SERIAL$ and $UDID$.
- **Authentication** of the device: a certificate (from the Certificates page, the certificate of the device user, or requested from a SCEP server), a shared secret (with the optional group name for IPSec), or none for IKEv2 servers that only check the user name and password.
- **Trusted certificates**: the CA certificates of your VPN server, installed with the VPN. For IKEv2 you can also set the expected server certificate issuer and name.
- **User name and password**: the user also authenticates with a user name (variables allowed) and a password (EAP for IKEv2, XAuth for IPSec). Leave the password empty to let the user type it on the device.
- **VPN name** (shown in Settings) and **Applies to**.

Connection

- **The user connects in Settings**: the VPN is installed and the user turns it on when needed.
- **On demand**: the device connects the VPN automatically. With **except on trusted Wi-Fi** the VPN is not used on the Wi-Fi networks you list, for example the office network.
- **Always On** (IKEv2, supervised iPhone and iPad only): all the traffic goes through the VPN on cellular and Wi-Fi and the user can't turn it off. Other devices don't get this VPN. If the VPN uses a user name, the password must be set in the policy.
- **Per-app VPN**: only some apps use the VPN. Choose the VPN for each app in the **App management** section (field **VPN** of the app, not available on Mac), and optionally list Safari domains that use it.

> Test an Always On or on-demand VPN on a test device first: if the server can't be reached, the device may have no connection.

Global proxy, encrypted DNS and web content filter

These settings apply to all the traffic of the device. Turn them on with their toggle.

- **Global HTTP proxy** (supervised iPhone and iPad, Mac): all the HTTP traffic goes through the proxy, on every network. Set the server and port (with an optional user name and password) or an automatic configuration (PAC file URL, or empty to find it on the network). **Allow Wi-Fi login pages** lets the user sign in to public Wi-Fi networks without the proxy.
- **Encrypted DNS**: the device resolves names with your DNS server over DNS over HTTPS (server URL) or DNS over TLS (server name), optionally only for some domains. On supervised devices you can prevent the user from turning it off.
- **Web content filter** (supervised iPhone and iPad): the filter built into iOS limits adult websites, with lists of websites always allowed or always blocked, or allows only a list of websites, shown as bookmarks in Safari.

> A wrong proxy or DNS server can leave the devices without a connection: try the settings on a test device first.

Certificates

Click **Add certificate** to install a certificate on its own: a certificate authority that Safari and the apps must trust, or an identity used by apps. The certificates of a Wi-Fi network or a VPN don't

need to be added here.

Certificate options

- **A certificate of the Certificates page**: the same certificate on every device. Click **Choose a certificate** to pick it.
- **The certificate of the device user**: each device gets the client certificate linked to its user. Devices whose user has no certificate don't get it. When you link another certificate to the user, the devices are updated.
- **Requested from a SCEP server**: each device requests its own certificate from your SCEP server (for example Microsoft NDES), so the private key never leaves the device. Set the server URL, the challenge if the server requires one, the subject (for example CN=$EMAIL$), the alternative names, the key size and the key usage.

> Client certificates (.p12) protected with the old RC2 encryption can't be installed on Apple devices. Export them again with 3DES or AES encryption before importing them.

Security

Profiles that contain passwords or private keys are sent only to the device they were made for: the device must sign its request with its management certificate. Passwords and certificates aren't written in the logs.

Software updates

The **Software updates** section of the Apple Policy Editor keeps the devices up to date: it can require a specific OS version by a deadline, and control how the devices download and install updates.

Required OS updates

Click **Add required update** to require an OS version. The device downloads the update, tells the user about the deadline and installs it at the deadline if the user hasn't done it before. Devices that already run that version or a later one ignore it. Requires iOS and iPadOS 17 or macOS 14.

- **Applies to** (required): iPhone and iPad, or Mac. Each OS has its own version numbers, so add one required update for each.
- **OS version**: the version to install, for example 18.5 or 15.4.1.
- **Build** (optional): a specific build, for example a beta or a Background Security Improvement.
- **Install by**: the deadline, in the local time of each device. A date in the past makes the devices install the update as soon as possible.
- **Details URL** (optional): a web page about the update, shown to the user.

> The device reports the progress of the update (download, installation, failures) in the **Device status** panel of the device page.

Software update settings

Turn on **Software update settings for iPhone and iPad** or **Software update settings for Mac** to control the updates on those devices. When a toggle is off, the devices keep their own settings. Requires iOS and iPadOS 18 or macOS 15: older devices ignore these settings.

- **Defer updates**: new updates are shown to the user only after the given number of days (1 to 90). On a Mac you can set different periods for major updates, minor updates and other system updates.
- **Updates shown to the user** (iPhone and iPad): when more than one update is available, show all of them, only the oldest minor update or only the newest one.

- **Standard users can install updates** (Mac): users without administrator rights can install macOS updates.
- **Notify the user about required updates**: show the notifications about the required updates and their deadline.
- **Automatic download**, **Automatic install** and **Automatic security updates**: let the user decide (Apple's default), or keep them always on or always off.
- **Background Security Improvements**: offer Apple's quick security fixes to the user, and choose whether the user can remove them.
- **Beta programs**: let the user join Apple beta programs, or remove the device from them and block them (on iPhone and iPad, supervised devices only).

Installing an update from the device page

The **Device status** panel of a device lists the updates available for it. Use **Check for updates** to ask the device for the list and **Install** to install one now or to download it and let the user install it. This works on supervised iPhones and iPads and on Macs. Apple removes these commands in iOS, iPadOS and macOS 27: on those versions use the required updates of the policy.

macOS

The **macOS** section of the Apple Policy Editor manages the security, privacy and login settings of the Macs: FileVault, firewall, Gatekeeper, screen saver and lock, external and network storage, Screen Sharing, privacy permissions, login items, and the system and kernel extensions allowed to load. It also installs configuration files of system services and background tasks. The other devices of the policy don't get these settings. Turn on a setting to manage it; while it's off, the users keep their own choices.

> Most settings are installed on the Mac as configuration profiles, visible in System Settings > General > Device Management. FileVault, privacy permissions, login items and the extension settings need user-approved MDM: on a Mac enrolled with the enrollment URL or QR code, the user approves the enrollment profile in System Settings. The device page shows it in the **User-approved MDM** field. When the user approves it later, the Mac gets these settings automatically.

FileVault

FileVault encrypts the disk of the Mac. With this setting on, users are asked to turn on FileVault with their password, and the Mac sends its personal recovery key to Cerberus Enterprise, encrypted with a key that belongs to your organization only. The key is shown and changed from the **FileVault** field of the device page (see Device commands). The setting needs macOS 13 or later and user-approved MDM; without user-approved MDM the Mac isn't asked to turn on FileVault, but if a user turns it on, its key is stored.

- **Required at login**: at the next login, after a number of skips (the user is asked again at each login and at logout), or only when the user logs out.
- **Don't ask at logout**: the user is asked only at login. It can't be combined with "Only asked at logout".
- **Show the recovery key to the user**: off by default, so only the administrators know the key, from the device page.
- **Prevent turning off FileVault**: on by default.
- **Remove the key from memory on standby**: the user types the password again when the Mac wakes from standby.
- **Where the key is kept**: text shown to the user at the login window, for example who to call to recover the Mac (default "Cerberus Enterprise").

>

> A Mac that was already encrypted when it got the setting keeps its current key, which Cerberus Enterprise doesn't receive: the key is stored only after the user turns FileVault off and on again (with "Prevent turning off FileVault" off). Turning the setting off, or removing the Mac from management, doesn't turn off FileVault, and the stored key keeps working. The setting can't be saved when the server isn't configured to store recovery keys.

Firewall

- **Turn on the firewall**: when it's off, the firewall stays off and users can't turn it on.
- **Block all incoming connections**: only the basic network services (DHCP, Bonjour, IPSec) can receive connections.
- **Stealth mode**: the Mac doesn't answer ping requests and connections to closed ports.
- **Allow built-in software** and **Allow downloaded signed apps** to receive connections without asking the user (macOS 12.3 or later).
- **Rules for specific apps**: allow or block the incoming connections of an app, by bundle ID.

Gatekeeper

Gatekeeper checks the apps before they open. Choose whether the Macs open only App Store apps, or also the apps of identified developers. With Gatekeeper off, any app can be opened, also unsigned or damaged ones. On macOS 15 or later you can also choose whether users are asked to send blocked malware to Apple.

Screen saver and lock

Start the screen saver after a time of inactivity, and ask for the password when the screen saver starts or the display sleeps, at once or after a delay. A setting left as **Not managed** stays the user's choice.

External and network storage

Allow, make read only or block the external storage (USB drives, SD cards, external disks) and the network storage (file shares) of the Macs. Needs macOS 15 or later.

Screen Sharing

Limits of the people who connect to these Macs with Screen Sharing: copying files from or to the Mac, high-performance connections and the number of virtual displays. Turning on Screen Sharing stays a choice of the user. Needs macOS 14 or later.

Privacy permissions

Give or deny apps the access that users would otherwise allow in System Settings > Privacy and Security, for example Full Disk Access for a backup app, or deny the camera to an app. For each permission choose:

- **Permission**: Full Disk Access, Accessibility, Automation, the folders, contacts, calendars, photos, camera, microphone, screen recording and the others of Privacy and Security.
- **Access**: allow or deny. The camera, the microphone, screen recording and input monitoring can't be allowed by a profile, only denied; for screen recording and input monitoring you can let users allow them also without admin rights.
- **App**: the bundle ID of an app, or the path of a program (for example a command line tool).
- **Code requirement**: identifies the app by its signature. Get it in Terminal with `codesign -dr - /Applications/Example.app` and copy the text after `designated =>`.
- **Automation** also needs the controlled app (for example the Finder) and its code requirement.

Login and background items

The login items and background items that match the rules are managed: they run without asking, and users can't turn them off in System Settings > General > Login Items. A rule matches the developer team ID of an app, its bundle ID or the label of a launch daemon or agent (also by their first part). Needs macOS 13 or later.

System extensions

System extensions are the modern way for apps to extend macOS: network filters and VPNs, endpoint security agents and drivers. With this setting on, the system extensions of the listed developer teams load without asking the user. Needs macOS 10.15 or later and user-approved MDM.

- **Team ID**: the 10-character team ID of the developer (for example in the "Signed by" details of the app, or with `codesign -dv`). Each team can appear only once.
- **Allow**: every extension of the team, or only the extensions with the listed bundle IDs.
- **Types**: limits the team to drivers, network or endpoint security extensions. With none chosen, all types are allowed.
- **Removal** (only for listed extensions): the app can remove its extension without an administrator (macOS 12 and later), the extension can't be turned off or removed while System Integrity Protection is on, or it can't be removed in System Settings and the Finder (macOS 15 and later). Macs with an older macOS still allow the extension, without the removal option.
- **Users can approve other system extensions**: on by default. When off, only the extensions allowed by a profile can load; if any profile on the Mac turns it off, it's off.

> On macOS 11.3 and later, removing this setting turns off the system extensions it allowed, for example a security agent or a network filter. This happens when the setting is turned off, when the policy is deleted, and when the Mac moves to a policy without it. The editor asks for confirmation before turning the setting off.

Kernel extensions (legacy)

Kernel extensions are the older, deprecated way to extend macOS: prefer system extensions when the vendor offers them. The kernel extensions of the listed teams (every extension of a team, or only the listed bundle IDs) load without asking the user, after a restart. Needs macOS 10.13.2 or later and user-approved MDM.

- **Users can approve other kernel extensions**: off by default, so only the listed teams can load kernel extensions.
- **Standard users can approve them too**: with user approvals on, users without administrator rights can approve them too (macOS 11 and later).

> On Macs with Apple silicon, a profile alone isn't enough: someone must start the Mac in recoveryOS, choose Reduced Security in Startup Security Utility and allow the user management of kernel extensions from identified developers. The Mac may also need a Bootstrap Token escrowed in Cerberus Enterprise to approve them (see the Bootstrap Token field of the device page).

Mac files

The service configuration files and the background tasks use files that you upload to the **Mac files** of your organization: zip archives and property lists (.plist) up to 20 MB. Upload, check where they are used or delete them with the **Mac files** button of the macOS section, or with **Choose file** in a setting. A file used by a policy can't be deleted. To change a file, upload the new version and choose it in the policy: the Macs download it again.

Service configuration files

Installs the configuration files of a system service from a zip archive. The built-in services are the SSH server (sshd), sudo, PAM, printing (CUPS), the Apache web server, bash, zsh, smart card login and authorization. You can also use your own service, with a reverse DNS identifier like `com.example.service`: it finds the files with the managed configuration files library of macOS. Each service can have one setting. Needs macOS 14 or later.

> A built-in service uses only the files of the archive and ignores its own files, for example /etc/ssh for sshd: put the whole configuration in the archive, in folders that start from the root of the disk (for example `etc/ssh/sshd_config`). The files are read-only on the Mac, and they are removed with the setting. Test the archive on one Mac first: a wrong sudo or PAM configuration can stop users from signing in or from using administrator rights.

Background tasks

Programs and scripts that run in the background, started by launchd. Users can't turn them off, and they are removed with the setting. Needs macOS 15 or later. For each task choose:

- **Task identifier**: a reverse DNS identifier, like `com.example.task`, different for each task.
- **Programs and scripts** (optional): a zip archive that the Mac unpacks in `/var/db/ManagedConfigurationFiles/BackgroundTaskServices/Services/` followed by the task identifier. Keep the execute permission of the files, and don't put property lists in it.
- **launchd property lists**: the jobs that start the programs, loaded as a daemon (for the whole system, as root) or as an agent (in each user session). Refer to the programs with their path in the folder above.

A task needs the programs or at least one property list. When the files change, the Macs restart the task.

Location and geofences

The **Location and geofences** section of the Apple Policy Editor defines areas, for example a site or a warehouse. iPhones and iPads with the Cerberus Enterprise app report when they enter or leave one of these areas, and event rules can react to it: send an email, lock or disable the device, or enable it again. Macs don't use geofences.

Geofence fields

Use **Add geofence** to create a new entry. A policy can contain up to **10 geofences**, and their names must be unique within the policy. Each geofence contains these fields:

- **Name**: required and unique.
- **Latitude** and **Longitude**: the center of the area. You can also click the map to set it.
- **Radius (m)**: required, from **200** to **10000** meters.
- **Description**: optional notes for administrators.
- **Report enter** and **Report exit**: choose which events are generated. At least one must stay on.
- **Active**: enables or disables the geofence without deleting it.

Each geofence has a map preview. The center and radius of a saved geofence are locked: use the lock button on the map to change them. The other buttons center the map on the area or on your current location.

When you save the policy, Cerberus Enterprise asks the iPhones and iPads of the policy to get the new geofences. This usually takes a few minutes, but iOS can delay it, for example in Low Power Mode or without a network connection. The device page shows when the app hasn't applied the latest geofences yet.

Requirements

Geofences work on iPhones and iPads with the Cerberus Enterprise app. Macs ignore them. On the device:

- the Cerberus Enterprise app is installed (Cerberus installs it) and the user has opened it at least once;
- location access for the app is set to **Always**;
- **Precise Location** is on for the app;

- **Background App Refresh** is on.

> **Location and geofences on personal devices**, at the top of the section, is off by default. While it's off, Cerberus doesn't collect the location of personal devices (User Enrollment), their geofence events aren't generated, and their device page doesn't offer Locate or live tracking. When you turn it on, personal devices of the policy share their location and geofence events like company devices, but only if the user allows location for the Cerberus Enterprise app; the app explains this to the user before iOS asks. Your organization is responsible for informing users and for following local rules on employee location. Lost Mode and the MDM location request are never available on personal devices (Apple doesn't allow them).

In **Low Power Mode** geofences keep working, but iOS reports leaving an area later and with a less precise location, and it can delay getting new geofences after a policy change by hours, until the app runs again (for example at the next crossing, or when the user opens it).

How events arrive

iOS watches the areas for the app and wakes it when the device crosses a border, also when the app isn't open. Events can arrive a few minutes after the crossing, and a small area or a short visit can be missed: iOS needs the device to move some distance past the border and to stay there for a while.

If the app can't run, for example after a restart before the device is unlocked, the server also detects the crossing from the locations the app sends, later. Each crossing triggers the event rules once, whether the app or the server detects it.

When the app hasn't contacted the server for several hours, the server asks iOS to wake it so that it can check its geofences again. iOS doesn't guarantee these requests.

Device page

On the device page, the **Geofence** tab of the **Location map** panel shows the transitions of the device. When the policy has active geofences, a status line tells whether geofencing works on the device, or why it can't: the app hasn't contacted the server or must be updated, location access isn't **Always**, **Precise Location** or **Background App Refresh** is off, **Low Power Mode** was on when the app last reported its settings, or the app hasn't applied the latest geofences yet.

>

> Geofence transitions and locations are kept for **90 days** and older data is deleted automatically every night. The latest location of each enrolled device is always kept, so it stays on the maps.

Limitations

- Copying a policy doesn't copy its geofences: add them again in the copy.
- Geofence events on iPhones and iPads are best effort: iOS decides when the app can run.

> To react to geofence events, create an event rule for **Device enters a geofence** or **Device exits a geofence**. See **Event rules**.

Accounts

The **Accounts** section of the Apple Policy Editor adds mail, Exchange, Google, calendar, contacts and directory accounts to iPhone, iPad and Mac. Click **Add account** and choose the type. The section also contains the **Allow account modification** restriction (supervised devices).

> On a Mac (macOS 13 or later) the accounts are installed in the session of the users enrolled in the management, usually the user who enrolled the Mac, when they're logged in. The device page lists these users under **Mac users**, with the state of their accounts. Accounts that take the email address from the device user use the user assigned to the Mac in Cerberus Enterprise.

Common settings

- **Name and email of the device user** (Mail, Exchange, Google): each device uses the name and email address of its user, set in the Users page (for Google, the user's Google account). Devices without a user, or whose user has no email address, don't get the account. Otherwise, set a fixed email address and full name.
- **Account name** (optional): the name of the account in Settings and in the apps.
- **User name** and **Password** (optional): the user name can contain the variables $EMAIL$, $USERNAME$, $FULLNAME$, $SERIAL$ and $UDID$. Leave the password empty to let the user type it on the device, or leave both empty to let the user type them.
- **Applies to**: limit the account to some device types.

Account types

- **Mail account**: IMAP or POP incoming server and outgoing server, with their ports and authentication. The outgoing server can use the same user name and password as the incoming one.
- **Exchange account** (Exchange ActiveSync): the server, or empty to find it from the email address. Choose the services (mail, contacts, calendars, reminders, notes) and whether the user can change them. With **Modern authentication (OAuth)** the user signs in with the page of your identity provider, for example Microsoft 365. An **Authentication certificate** (iOS 17 or later) authenticates the device with a certificate.
- **Google account**: the user signs in to the Google account with the given email address.

- **Calendar account (CalDAV)** and **Contacts account (CardDAV)**: the server, port and path.
- **Directory (LDAP)**: the server and port, and the searches (search base, scope and description) used to look up contacts.
- **Subscribed calendar**: the URL of the calendar (https, http or webcal).

S/MIME

Mail and Exchange accounts can sign and encrypt messages with S/MIME (iOS 17 or later). For each one, choose the certificate: one of the Certificates page, the certificate of the device user, or requested from a SCEP server. You can let the user turn it off, choose another certificate and, for encryption, show a switch for each message.

Security

Passwords and certificates aren't part of the settings sent to the devices: each device downloads its own credentials from Cerberus Enterprise, signing the request with its management certificate. They aren't written in the logs.

Compliance rules

The **Compliance rules** section of the Apple Policy Editor sets requirements for the devices of the policy. A device that breaks a rule is non-compliant at once: the device page shows why, in **Why the device is non-compliant**, and the devices list shows it as non-compliant. A device is also non-compliant when a setting of the policy can't be applied on it.

Rules

- **Passcode**: the device has a passcode, and it meets the passcode settings of the policy.
- **Minimum OS version**: for iPhone and iPad, or for Mac. To update the devices, add a required update in Software updates.
- **Supervised device**: devices enrolled with the enrollment profile or with User Enrollment aren't supervised.
- **FileVault on (Mac)**: the disk of the Mac is encrypted with FileVault. Other devices aren't affected.

A rule is checked when the device sends its status, when the policy is saved, and every hour. A device whose status is unknown (for example a Mac that didn't report FileVault yet) doesn't break the rule.

Blocking the device

Turn on **Block the device if it's still non-compliant** and set after how many days (0 blocks at once). Supervised iPhone and iPad go into lost mode with the message you set. The other devices are locked once; a Mac restarts and asks for a PIN, shown in the details of the command in the command history.

- **Unblock**: when the device is compliant again, lost mode is turned off automatically.
- **Turning lost mode off yourself**: the device isn't blocked again until it becomes compliant and breaks a rule again. Use it to let the user fix the device.

> To be notified, or to react in other ways, create an event rule for **Device becomes non-compliant**: it also gets the reasons.