Users, Roles and RBAC

Users, Roles and RBAC

PatchMon uses role-based access control (RBAC) to decide who can see and do what inside the application. Every user has exactly one role, and every role is a collection of permissions. This page covers the built-in roles, the full permission list, and how to manage users and roles from the Settings UI.

> **Related pages:**
>
> - Setting Up OIDC / Single Sign-On: authenticate users against an external IdP
> - Setting Up Microsoft Azure Entra ID (SSO) with PatchMon: Entra-specific walkthrough
> - Two-Factor Authentication: per-user TOTP and trusted devices

---

The Built-In Roles

PatchMon ships with five roles. You see these in **Settings → Users** (in the **Role** dropdown) and in **Settings → Roles** (as the matrix columns).

Role	Default Permissions	Typical Use
Super Admin (`superadmin`)	Everything, including managing other superadmins	The very first user, or dedicated platform owners
Admin (`admin`)	Everything except managing other superadmins	Day-to-day platform administrators
Host Manager (`host_manager`)	Monitoring + host/infrastructure management + operations (patching, compliance, alerts, automation, remote access)	NOC / Ops engineers
User (`user`)	Monitoring + data export	Engineers who need to look but not break

Role	Default Permissions	Typical Use
Readonly (`readonly`)	Monitoring only	Auditors, read-only dashboards, management

Two important rules about built-ins:

- **Cannot be deleted.** `superadmin`, `admin`, `host_manager`, `user` and `readonly` are always present. The **Delete** button does not appear for them.
- **The core three cannot have their permissions edited.** `superadmin`, `admin` and `user` are *locked*: their permission matrix is hardcoded and the **Edit** button is disabled. `host_manager` and `readonly` can still be edited if you want to tune them.

> **First user is always Super Admin.** When PatchMon is first installed and has no users, the setup wizard creates the initial account as `superadmin`, regardless of what role you type. If OIDC is configured for auto-create before first boot, the very first OIDC login is also promoted to `superadmin` automatically so you cannot lock yourself out.

The Full Permission List

Permissions are grouped into four risk tiers. The colour you see in the **Roles** matrix corresponds to this risk level.

Monitoring & Visibility (Low risk)

Read-only access to dashboards, hosts, packages, reports, and logs.

Permission key	Label	What it lets the user do
`can_view_dashboard`	View Dashboard	View the main dashboard and its stat panels
`can_view_hosts`	View Hosts	See the host list, host detail pages, and connection status
`can_view_packages`	View Packages	See the package inventory across all hosts
`can_view_reports`	View Reports	See compliance scan results and alert reports
`can_view_notification_logs`	View Notification Logs	See notification delivery history and status

Host & Infrastructure (Medium risk)

Create, modify and delete hosts, packages, and containers.

Permission key	Label	What it lets the user do
`can_manage_hosts`	Manage Hosts	Create / edit / delete hosts, host groups, repositories and integrations
`can_manage_packages`	Manage Packages	Edit package inventory and metadata
`can_manage_docker`	Manage Docker	Delete Docker containers, images, volumes and networks

Operations (Medium-High risk)

Day-to-day NOC tasks.

Permission key	Label	What it lets the user do
`can_manage_patching`	Manage Patching	Trigger patches, approve patch runs, manage policies
`can_manage_compliance`	Manage Compliance	Trigger compliance scans, remediate findings, install scanners
`can_manage_alerts`	Manage Alerts	Assign, delete and bulk-action alerts
`can_manage_automation`	Manage Automation	Trigger and manage automation jobs
`can_use_remote_access`	Remote Access	Open SSH and RDP terminals against managed hosts

Administration (High risk)

Organisation-wide control.

Permission key	Label	What it lets the user do
`can_view_users`	View Users	See the user list and account details
`can_manage_users`	Manage Users	Create, edit and delete user accounts
`can_manage_superusers`	Manage Superusers	Manage `superadmin` accounts and elevated privileges
`can_manage_settings`	Manage Settings	System configuration, OIDC / SSO, AI, alert config, enrollment tokens
`can_manage_notifications`	Manage Notifications	Configure notification destinations and routing rules

Permission key	Label	What it lets the user do
`can_export_data`	Export Data	Download and export data and reports

> **Billing:** On PatchMon Cloud there is also a `can_manage_billing` permission that governs access to the Billing page. On self-hosted instances this permission exists in the schema but the Billing page is not enabled by default.

Viewing the Role Matrix

1. Sign in as a user with `can_manage_settings`.
2. Go to **Settings → Roles**.
3. You'll see a matrix: rows are permissions (grouped by tier), columns are roles. A green tick means the role has that permission.

Each column header also shows an `n/N` counter showing the number of permissions that role currently holds out of the total 20.

Creating a Custom Role

Custom roles let you tailor the permission set beyond the built-in five.

> **Availability:** The **Add Role** button is only shown when the `rbac_custom` module is enabled on your PatchMon deployment. On self-hosted installs this module is typically enabled by default; on PatchMon Cloud it depends on your plan. If you don't see **Add Role** and the URL `https://patchmon.example.com/settings/roles` shows a "Not Available" screen, the module isn't enabled on your plan.

To create one:

1. Go to **Settings → Roles**.
2. Click **Add Role** in the top-right.
3. Fill in the modal:
   - **Role Name:** lowercase, underscores instead of spaces. Examples: `host_manager`, `compliance_auditor`, `noc_operator`. This is the internal key; it cannot be renamed later.

- **Preset** (optional): four quick-start presets are available:
  - **Read Only:** just the Monitoring & Visibility group
  - **Operator:** everything except the Administration group
  - **Admin:** every permission
  - **Clear All:** start from zero
- **Permissions:** tick / untick individual permissions, or use the **Select all / Deselect all** shortcut on each group header.

4. Watch the counter at the bottom (`n/20 permissions selected`) as a sanity check.
5. Click **Create Role**.

The new role appears as a new column in the matrix and is selectable when creating or editing users.

Editing a Custom Role

1. In the matrix, click the pencil icon in the column header of the role you want to edit.
2. An editor panel opens below the matrix with all permissions listed.
3. Tick / untick as needed, then click **Save**.

Changes take effect immediately. Any session held by a user with that role has its in-memory permissions refreshed on their next request.

Deleting a Custom Role

You can only delete a role that is **not assigned to any user**. If any user holds that role, the delete endpoint rejects the request with "Cannot delete role: users are assigned to it". Reassign those users to a different role first (see Editing a Role for an Existing User).

To delete:

1. Click the pencil in the role's column header to open the editor panel.
2. Click **Delete** (appears only for non-built-in roles).
3. Confirm.

Creating Users

Go to **Settings → Users** and click **Add User** in the top-right.

Field	Notes
Username	Minimum 3 characters. Lowercase recommended

Field	Notes
Email	Must be a valid email. Used for OIDC account linking and email alerts
First Name / Last Name	Optional
Password	Must satisfy the active password policy (configured under **Settings → Server Config → Security**)
Role	Choose from built-in or custom roles

Click **Add User**. The account is created immediately and can sign in straight away.

> **Role escalation protection:** You cannot create a user with a role that's more privileged than your own. Only `superadmin` users can create new `admin` or `superadmin` accounts. Non-superadmin accounts that hold the `can_manage_superusers` permission can also create and manage `superadmin` accounts.

Self-Service Sign-Up

PatchMon can also let users register themselves rather than having an admin invite them.

1. Go to **Settings → Users**.
2. Scroll to **User Registration Settings**.
3. Tick **Enable User Self-Registration**.
4. Pick a **Default Role for New Users**: the role that self-registered accounts are assigned.
5. Click **Save Settings**.

A sign-up link now appears on the login page. Anyone who can reach the login page can create an account.

> **Security warning:** Only enable self-registration on internal or private-network deployments. If your PatchMon is internet-facing, leave it off and invite users manually, or front it with OIDC SSO (which lets your IdP decide who can log in).

Editing a Role for an Existing User

1. Go to **Settings → Users**.
2. Find the user in the table and click the **Edit** (pencil) icon.

3. Change **Role** in the dropdown and click **Save**.

Important side effects:

- **Sessions are revoked.** When a user's role changes, all of their existing JWT sessions are invalidated on the server. They must sign in again. This ensures the old role's privileges cannot be replayed from an existing browser tab.
- **You cannot change your own role.** The API rejects a self-role change with "Cannot change your own role". This is a deliberate safety net: two admins must cooperate to demote each other.
- **You cannot promote a user above yourself.** An `admin` cannot promote a user to `superadmin`. Only a `superadmin` can create or promote to `superadmin`, and likewise only `superadmin` can assign the `admin` role.

Resetting a User's Password

1. In the users table, click the **Reset** (key) icon on that user's row.
2. Enter a new password.
3. Click **Reset Password**.

After a reset, all of that user's sessions and trusted-device records are revoked. This is the standard post-compromise response. The user must sign in with the new password on every device.

> You cannot reset the password of an inactive user. Reactivate them first.

Disabling (Deactivating) a User

Disabling is the safer alternative to deletion. The user record, their history, and their audit trail are preserved, but they cannot log in.

1. Go to **Settings → Users**.
2. Click the **Edit** icon on the user you want to disable.
3. Untick the **Active** checkbox.
4. Click **Save**.

Effects:

- All their sessions are revoked immediately.
- All their trusted devices are revoked (so re-activating them later cannot reuse a "remember this device" cookie that predates the deactivation window).

- The user's row is shown with a red **Inactive** badge in the users table.

To re-enable: edit and tick **Active** again.

Deleting a User

Deletion is permanent and removes the user record and their associated dashboard preferences, sessions, trusted devices and notification preferences.

1. Click the **Delete** (trash) icon on the user's row.
2. Confirm.

Restrictions:

- You cannot delete your own account.
- You cannot delete the last `superadmin` (the API refuses).
- You cannot delete the last `admin` if there are no `superadmin` users (ensures at least one admin always exists).
- You cannot delete a user who holds a role that's more privileged than yours.

How Permissions Are Evaluated

- **Admin and Super Admin** always have every permission, even if the `role_permissions` table says otherwise. The middleware short-circuits their permission checks. This is a safety net: if someone mis-edits the `admin` row (which shouldn't be possible via the UI, but could happen via direct database access), admins don't get locked out.
- **Every other role** (built-in or custom) has its permissions read from the database at each request. Changes made in **Settings → Roles** take effect on the user's next API call; no restart required.
- **Role hierarchy for user management** is enforced separately from the permissions above:
  - `superadmin` → rank 100
  - `admin` → rank 90
  - `host_manager` → rank 50
  - custom roles → rank 30 (mid-tier)
  - `user` → rank 20
  - `readonly` → rank 10

You can only modify, delete, or reset the password of users whose role rank is less than or equal to your own. This is distinct from the permission checks. Even if a custom role were granted `can_manage_users`, its holder still could not touch `admin` or `superadmin` accounts unless they additionally had `can_manage_superusers`.

When OIDC Role Sync Is Enabled

If **Settings → OIDC / SSO → Sync roles from IdP** is on, PatchMon stops letting admins manage users and roles from the UI. Instead:

- The **Add User** and **Add Role** buttons disappear.
- The Users tab shows a read-only list.
- The Roles tab shows a banner reminding you that group membership in your IdP drives role assignment via environment variables: `OIDC_SUPERADMIN_GROUP`, `OIDC_ADMIN_GROUP`, `OIDC_HOST_MANAGER_GROUP`, `OIDC_USER_GROUP`, `OIDC_READONLY_GROUP`.
- Users' roles are re-evaluated on every login based on their current IdP group membership.

If you want to use OIDC for authentication but still manage roles locally in PatchMon, leave **Sync roles from IdP** off. See Setting Up OIDC / Single Sign-On for the full toggle reference.

Troubleshooting

"You do not have permission to assign the role: admin"

Only a `superadmin` can create or promote users to `admin` or `superadmin`. If you're an `admin` and try to promote someone to `admin`, the API refuses. Ask a superadmin to do it.

"Cannot modify built-in role permissions"

The `superadmin`, `admin` and `user` rows are locked against permission edits. If you need a role with tweaked permissions, create a custom role based on a preset and assign users to that instead.

"Cannot delete role: users are assigned to it"

Before a role can be deleted, reassign every user who holds it. Use **Settings → Users → Edit** to change each user's role, then try the delete again.

"Cannot delete the last superadmin user" / "Cannot delete the last admin user"

At least one `superadmin` must always exist. If there are no superadmins at all, at least one `admin` must exist. Create a replacement first (and sign in as them to confirm the login works) before deleting the final one.

User's old role is still in effect after I changed it

Changing a role revokes all existing sessions, but the user's browser may still hold an old JWT cookie that hasn't been rejected yet. Ask them to refresh the page or sign out and back in; the server will reject the stale token and redirect them to login.

"Add User" / "Add Role" button is missing

Three possible causes:

1. **Your role doesn't have `can_manage_settings` or `can_view_users`.** Check `/settings/users`: if the page is empty or you get a Forbidden, your role lacks the view permission.
2. **OIDC role sync is on.** See When OIDC Role Sync Is Enabled.
3. **The `rbac_custom` module is not enabled.** This only affects the **Add Role** button on the Roles tab. Custom role creation is a gated feature. The **Add User** button on the Users tab is always available when the other two conditions are met.

Revision #1
Created 2026-04-25 10:16:05 UTC by Iby
Updated 2026-04-25 10:16:27 UTC by Iby