Settings reference

- General
  - Send usage statistics
  - Automatically check for updates
  - Automatically update components
  - Enable Gordon
  - Block docker load
  - Hide onboarding survey
  - Enable Docker terminal
  - Expose Docker API on TCP 2375 {{< badge color=blue text="Windows only" >}}
- Extensions
  - Enable Docker extensions
  - Allow only extensions distributed through the Docker Marketplace
  - Enable a private marketplace
- AI
  - Enable Docker Model Runner
    - Enable host-side TCP support
      - Port
      - CORS Allowed Origins
  - Enable GPU-backed inference {{< badge color=blue text="Windows only" >}}
- File sharing and emulation
  - File sharing directories
  - VirtioFS {{< badge color=blue text="Mac only" >}}
  - gRPC FUSE {{< badge color=blue text="Mac only" >}}
  - Rosetta {{< badge color=blue text="Mac only" >}}
- Scout
  - Enable Scout image analysis
  - Enable background Scout SBOM indexing
- Proxy

    - Embedded PAC script
    - PAC file URL
    - Override Windows "dockerd" port {{< badge color=blue text="Windows only" >}}
    - Enable Kerberos and NTLM authentication
    - Proxy bypass
- Containers proxy
    - Air-gapped container proxy
- LinuxVM
    - Enable WSL engine {{< badge color=blue text="Windows only" >}}
    - Docker daemon options
    - VPNKit CIDR {{< badge color=blue text="Mac only" >}}
- Windows containers
    - Docker daemon options
- Kubernetes
    - Enable Kubernetes
    - Show system containers
    - Kubernetes image repository
    - Cluster provisioning method
    - Node version
    - Nodes count
- Features in development
    - Access beta features
    - Enable Docker MCP Toolkit (Beta)
- Enhance container isolation
    - Enable Enhanced Container Isolation
    - Docker socket access control (ECI exceptions)
- Network
    - Networking mode
    - Inhibit DNS resolution for IPv4/IPv6
    - Port binding behavior
- Other
    - Enable Docker Offload

This reference documents Docker Desktop settings that administrators can configure using Settings Management. Use this page to understand which settings are available, their accepted values, platform compatibility, and which configuration methods apply.

> [!NOTE]
>
> This page only covers configurable settings for administrators who are deploying Docker Desktop to their organization. For the full list of Docker Desktop user-facing settings, see Change settings.

General

Send usage statistics

Controls whether Docker Desktop collects and sends local usage statistics and crash reports to Docker. Does not affect server-side telemetry collected via Docker Hub or other backend services such as sign in timestamps, pulls, or builds.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`analyticsEnabled`
Docker Home	**Send usage statistics**

> [!NOTE]
>
> Organizations using the Insights Dashboard may need this setting enabled to ensure that developer activity is fully visible. If users opt out and the setting is not locked, their activity may be excluded from analytics views.

Automatically check for updates

Controls whether Docker Desktop checks for and notifies users about available updates. When set to `true`, update checks and notifications are disabled.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`disableUpdate`
Docker Home	**Disable update**

> [!NOTE]
>
> In hardened environments, enable this setting and lock it. This guarantees that only internally vetted versions are installed.

Automatically update components

Allows Docker Desktop to automatically update components that do not require a restart, such as Docker Compose, Docker Scout, and the Docker CLI.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`silentModulesUpdate`
Docker Home	**Automatically update components**

Enable Gordon

Property	Value
Default	`false`
Accepted values (individuals)	`true`, `false`
Accepted values (Business tier)	`"Disabled"`, `"Enabled"`, `"Always Enabled"`
JSON key	`enableDockerAI`

Property	Value
Docker Home	**Enable Gordon**

> [!IMPORTANT]
>
> Docker Business customers must set this to `"Enabled"` or `"Always Enabled"` in Docker Home. Setting to `"User Defined"` alone will not activate Gordon.

Block `docker load`

Prevents users from loading local Docker images using the `docker load` command, enforcing image provenance by requiring all images to come from registries.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`blockDockerLoad`
Docker Home	**Block Docker Load**

> [!NOTE]
>
> In hardened environments, enable and lock this setting. This forces all images to come from your secure, scanned registry.

Hide onboarding survey

Prevents the onboarding survey from being shown to new users.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`displayedOnboarding`

Property	Value
Docker Home	**Hide onboarding survey**

Enable Docker terminal

Allows or restricts access to the built-in terminal for host system interaction. When set to `false`, users cannot use the Docker terminal to interact with the host machine or execute commands directly from Docker Desktop.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
Docker Desktop GUI	**General** tab
JSON key	`desktopTerminalEnabled`
Docker Home	Not available

Expose Docker API on TCP 2375 {{< badge color=blue text="Windows only" >}}

Exposes the Docker API over an unauthenticated TCP socket on port 2375. Only recommended for isolated and protected environments. Supports legacy integrations that require TCP API access.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`exposeDockerAPIOnTCP2375`
Docker Home	**Expose Docker API**

> [!NOTE]

> In hardened environments, disable and lock this setting. This ensures the Docker API is only reachable via the secure internal socket.

Extensions

Enable Docker extensions

Controls whether users can install and run Docker Extensions.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`extensionsEnabled`
Docker Home	**Allow Extensions**

> [!NOTE]
>
> In hardened environments, disable and lock this setting. This prevents third-party or unverified plugins from being installed.

Allow only extensions distributed through the Docker Marketplace

Prevents installation of third-party or locally developed extensions.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`onlyMarketplaceExtensions`

Property	Value
Docker Home	**Only marketplace extensions**

Enable a private marketplace

Ensures Docker Desktop connects to content defined and controlled by the administrator instead of the public Docker Marketplace.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`extensionsPrivateMarketplace`
Docker Home	**Extensions private marketplace**

AI

Enable Docker Model Runner

Enables Docker Model Runner functionality for running AI models in containers.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`enableInference`
Docker Home	**Enable Docker Model Runner**

Enable host-side TCP support

Enables TCP connectivity for Docker Model Runner services, allowing external applications to connect to Model Runner via TCP.

Property	Value
Default	`false`

Property	Value
Accepted values	`true`, `false`
Format	Boolean
JSON key	`enableInferenceTCP`
Docker Home	**Host-side TCP support**
Requires	Docker Model Runner enabled

Port

Specifies the port used for Model Runner TCP connections.

Property	Value
Default	`12434`
Accepted values	Integer
Format	Integer
JSON key	`enableInferenceTCPPort`
Docker Home	**Host-side TCP port**
Requires	Docker Model Runner and host-side TCP support enabled

CORS Allowed Origins

Controls cross-origin resource sharing for Model Runner web integration.

Property	Value
Default	Empty string
Accepted values	Empty string (deny all), `*` (accept all), or comma-separated list of origins
Format	String
JSON key	`enableInferenceCORS`
Docker Home	**CORS Allowed Origins**
Requires	Docker Model Runner and host-side TCP support enabled

Enable GPU-backed inference {{< badge color=blue text="Windows only" >}}

Enables GPU-backed inference. Additional components will be downloaded to `~/.docker/bin/inference`.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`enableInferenceGPUVariant`
Docker Home	**Enable GPU-backed inference**

File sharing and emulation

File sharing directories

Defines which host directories containers can access for development workflows.

Property	Value
Default	Varies by OS
Accepted values	List of file paths
Format	Array of strings
JSON key	`filesharingAllowedDirectories`
Docker Home	Yes — **Allowed file sharing directories**

VirtioFS {{< badge color=blue text="Mac only" >}}

Uses VirtioFS for fast, native file sharing between host and containers. If both VirtioFS and gRPC FUSE are set to `true`, VirtioFS takes precedence.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean

Property	Value
JSON key	`useVirtualizationFrameworkVirtioFS`
Docker Home	**Use VirtioFS for file sharing** tab

gRPC FUSE {{< badge color=blue text="Mac only" >}}

Enables gRPC FUSE for macOS file sharing.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`useGrpcfuse`
Docker Home	**Use gRPC FUSE for file sharing**

Rosetta {{< badge color=blue text="Mac only" >}}

Uses Rosetta for x86_64/amd64 emulation on Apple Silicon.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`useVirtualizationFrameworkRosetta`
Docker Home	**Use Rosetta for x86_64/amd64 emulation on Apple Silicon**

Scout

Enable Scout image analysis

Turns on vulnerability scanning and software bill of materials (SBOM) analysis for container images.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`sbomIndexing`
Docker Home	**SBOM indexing**

Enable background Scout SBOM indexing

Keeps image metadata current by indexing during idle time or after image operations.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`useBackgroundIndexing`
Docker Home	**Background indexing**

Proxy

> [!NOTE]
>
> Proxy configuration is a special case because it must be configured in two places:
>
> 1. In the Docker Home for your organization.
> 2. On the user's system where Docker Desktop is installed.
>
> On the user's machine, configure the proxy either through the `admin-settings.json` file or by using installer flags during Docker Desktop installation.

> For detailed instructions, refer to the installation guide.
>
> This additional configuration is required because Docker Desktop must know which proxy server to use before it can complete user sign-in and retrieve organization settings from the Docker Home.

Embedded PAC script

Specifies an embedded Proxy Auto-Config (PAC) script. For example: `"embeddedPac": "function FindProxyForURL(url, host) { return \"DIRECT\"; }"`.

Property	Value
Default	`""`
Accepted values	Embedded PAC script content
Format	String
JSON key	`embeddedPac`
Docker Home	Yes **Embedded PAC script**

PAC file URL

Specifies a PAC file URL for Docker Desktop to use when routing network traffic. For example: `"pac": "http://proxy/proxy.pac"`.

Property	Value
Default	`""`
Accepted values	PAC file URL
Format	String
JSON key	`pac`
Docker Home	**PAC file**

Override Windows "dockerd" port {{< badge color=blue text="Windows only" >}}

Exposes Docker Desktop's internal proxy locally on this port for the Windows Docker daemon to connect to. If it is set to 0, a random free port is chosen. If the value is greater than 0, use that exact value for the port.

Property	Value
Default	`-1`
Accepted values	`-1` `0`
Format	String
JSON key	`windowsDockerdPort`
Docker Home	**Override Windows “dockerd” port**

Enable Kerberos and NTLM authentication

Enables enterprise proxy authentication support for Kerberos and NTLM protocols.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`proxy.enableKerberosNtlm`
Docker Home	**Kerberos NTLM**

Proxy bypass

Defines network addresses that containers should bypass when using proxy settings.

Property	Value
Default	`""`
Accepted values	List of addresses
Format	String
Docker Desktop GUI	**Proxies** tab
JSON key	`proxy` (with `manual` and `exclude` modes)
Docker Home	Yes — **Proxy** section

Containers proxy

Air-gapped container proxy

Configures an HTTP/HTTPS proxy that governs two distinct traffic paths:

- Daemon image pulls (always enforced): Docker Desktop always injects `http.docker.internal:3128` as the daemon's proxy in `daemon.json` at VM startup. All `docker pull` and Compose pull operations are routed through `containersProxy`, including any PAC file rules. This applies regardless of whether `transparentPorts` is configured.
- Running container outbound traffic (opt-in): Container TCP traffic is only subject to `containersProxy` rules when `transparentPorts` is configured. Without it, running containers connect directly and PAC file rules do not apply to their outbound traffic.

> [!IMPORTANT]
>
> If you configure a PAC file under `containersProxy`, the PAC file must return an appropriate proxy server to connect to the registries where your images are hosted.

The `proxy` setting governs Docker Desktop host-level traffic: the Desktop application, Docker CLI, and extensions. It serves as a fallback for the daemon only when `containersProxy` is not explicitly configured. Once `containersProxy` is set, `proxy` plays no role in daemon or container traffic.

Property	Value
Default	See example below
Accepted values	JSON object
Format	JSON object
JSON key	`containersProxy`
Docker Home	**Containers proxy** section

```
"containersProxy": {
  "locked": true,
  "mode": "manual",
  "http": "",
  "https": "",
  "exclude": [],
```

```
  "pac": "",
  "transparentPorts": ""
}
```

For more information, see Air-gapped containers.

LinuxVM

Enable WSL engine {{< badge color=blue text="Windows only" >}}

When set to `true`, Docker Desktop uses the WSL 2 based engine. Overrides any backend flag set at installation using `--backend=<backend name>`.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`wslEngineEnabled`
Docker Home	**Windows Subsystem for Linux (WSL) Engine**

Docker daemon options

Overrides the Docker daemon configuration used in containers, without modifying local configuration files.

Property	Value
Default	`{}`
Accepted values	JSON object
Format	Stringified JSON
JSON key	`linuxVM.dockerDaemonOptions`
Docker Home	**Docker Daemon options** in the LinuxVM dropdown

VPNKit CIDR {{< badge color=blue text="Mac only" >}}

Sets the network subnet used for Docker Desktop's internal VPNKit DHCP/DNS services. Prevents IP address conflicts in environments with overlapping network subnets.

Property	Value
Default	`192.168.65.0/24`
Accepted values	CIDR notation
Format	String
JSON key	`vpnkitCIDR`
Docker Home	**VPNKit CIDR**

Windows containers

Docker daemon options

Overrides the Docker daemon configuration used in Windows containers, without modifying local configuration files.

Property	Value
Default	`{}`
Accepted values	JSON object
Format	Stringified JSON
JSON key	`windowsContainers.dockerDaemonOptions`
Docker Home	**Docker Daemon options** in the **Windows containers dropdown**

Kubernetes

Enable Kubernetes

Enables the local Kubernetes cluster integration with Docker Desktop.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`kubernetes`
Docker Home	**Enable Kubernetes**

Show system containers

Controls visibility of Kubernetes system containers in the Docker Desktop Dashboard.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
Docker Home	**Show system containers**

Kubernetes image repository

Specifies a registry used for Kubernetes control plane images instead of Docker Hub. Overrides the `[registry[:port]/][namespace]` portion of image names. Images must be mirrored from Docker Hub with matching tags.

Property	Value
Default	`""`
Accepted values	Registry URL
Format	String
JSON key	`KubernetesImagesRepository`
Docker Home	**Kubernetes Images Repository**

> [!NOTE]

> Images must be mirrored from Docker Hub with matching tags. Required images depend on the cluster provisioning method.

> [!IMPORTANT]
>
> When using custom image repositories with Enhanced Container Isolation, add these images to the ECI allowlist: `[imagesRepository]/desktop-cloud-provider-kind:*` and `[imagesRepository]/desktop-containerd-registry-mirror:*`.

Cluster provisioning method

Controls Kubernetes cluster topology and node configuration.

Property	Value
Default	`kubeadm`
Accepted values	`kubeadm`, `kind`
Format	String
Docker Home	**Kubernetes mode**

Node version

Pins the Kubernetes version used for cluster nodes.

Property	Value
Default	`1.31.1`
Accepted values	Semantic version (e.g. `1.29.1`)
Format	String
Docker Home	**Node version** tab

Nodes count

Sets the number of nodes in multi-node Kubernetes clusters.

Property	Value

Default	`1`
Accepted values	Integer
Format	Integer
Docker Home	**Nodes count**

Features in development

Access beta features

Controls whether users can access all Docker Desktop features that are in public beta.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`allowBetaFeatures`
Docker Home	**Access beta features**

Enable Docker MCP Toolkit (Beta)

Enables Docker MCP Toolkit in Docker Desktop for AI model development workflows.

Property	Value
Default	`true`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`enableDockerMCPToolkit`
Docker Home	Not available

Enhance container isolation

Enable Enhanced Container Isolation

Prevents containers from modifying Docker Desktop VM configuration or accessing sensitive host areas.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`enhancedContainerIsolation`
Docker Home	**Enable enhanced container isolation**

Docker socket access control (ECI exceptions)

Defines specific images and commands allowed to use the Docker socket when Enhanced Container Isolation is active. Supports tools like Testcontainers, LocalStack, or CI systems that need Docker socket access while maintaining security.

Property	Value
Accepted values	JSON object
Format	JSON object
JSON key	``dockerSocketMount`
Docker Home	**Image list**, **Command list**

```
"enhancedContainerIsolation": {
  "locked": true,
  "value": true,
  "dockerSocketMount": {
    "imageList": {
      "images": [
        "docker.io/localstack/localstack:*",
        "docker.io/testcontainers/ryuk:*"
      ]
    },
```

```
    "commandList": {
      "type": "deny",
      "commands": ["push"]
    }
  }
}
```

Network

Networking mode

Sets the default IP protocol used when Docker creates new networks.

Property	Value
Default	`dual-stack`
Accepted values	`ipv4only`, `ipv6only`
Format	String
JSON key	`defaultNetworkingMode`
Docker Home	**Default network IP mode**

For more information, see Networking.

Inhibit DNS resolution for IPv4/IPv6

Filters unsupported DNS record types to improve reliability in environments where only IPv4 or IPv6 is supported. Requires Docker Desktop 4.43 and later.

Property	Value
Default	`auto`
Accepted values	`ipv4`, `ipv6`, `none`
Format	String
JSON key	`dnsInhibition`
Docker Home	**DNS filtering behavior**

For more information, see Networking.

Port binding behavior

Specify how port bindings are handled for new containers.

Property	Value
Default	`default-port-binding`
Accepted values	`default-local-port-binding`, `local-only-port-binding`, `default-port-binding`
Format	String
JSON key	`portBindingBehavior`
Docker Home	**Port binding behavior**

Other

Enable Docker Offload

Controls Docker Offload availability. When enabled, users see the Docker Offload toggle in the Docker Desktop header.

Property	Value
Default	`false`
Accepted values	`true`, `false`
Format	Boolean
JSON key	`enableCloud`
Docker Home	**Enable Docker Offload**

> [!NOTE]
>
> This setting is only available when Docker Offload capability is enabled for the organization.

□□□ #2
□ □□□ □□□ 2026-10-31 19:10:42 UTC
□ □□□ □□□ 2026-10-31 19:10:45 UTC