



Remediate

Contents

Welcome to Puppet Remediate™	4
Remediate 2.x release notes	4
Remediate 1.x release notes.....	6
Installing Remediate	9
Architecture.....	10
System requirements.....	13
Install Remediate on online nodes.....	15
Install Remediate on offline nodes.....	17
Install Remediate from a custom container registry.....	20
Install Remediate in online multi-network deployments.....	21
Install Remediate in offline multi-network deployments.....	22
Verify Docker Compose file for online installs.....	22
Using TLS certificates with Remediate.....	23
Analytics data collection.....	24
Uninstall Remediate.....	24
Upgrading Remediate	25
Upgrade Remediate on online nodes.....	25
Upgrade Remediate on offline nodes.....	25
Upgrade Remediate from a custom registry.....	26
Managing user access	27
Add new users.....	29
Update a user.....	30
Delete a user.....	30
Force log out.....	31
LDAP configuration.....	31
Active Directory configuration.....	33
LDAP mappers.....	35
Work with user groups in Remediate.....	36
Remediate Vulnerability Dashboard	40
Dashboard Metrics bar.....	40
Dashboard Vulnerability Overview tab.....	41
Dashboard Node Overview tab.....	42
Configuring Remediate	43
Add sources	51
Vulnerability scanners.....	52

Infrastructure sources.....	55
Puppet Enterprise sources.....	57
Adding node credentials.....	57
Secure Shell (SSH).....	58
Windows Remote Management (WinRM).....	59
Tagging nodes.....	60
Assign a tag to a node.....	61
Assign a tag to multiple nodes.....	61
Add multiple tag values.....	62
Edit tag values.....	62
Remove tags from nodes.....	63
Filter nodes by tag.....	63
Prioritizing vulnerabilities.....	64
Viewing vulnerability details.....	66
Latest events for vulnerabilities.....	67
Latest event vulnerability task details.....	67
Accepting vulnerability risk.....	67
Filtering and exporting data.....	68
Remediating vulnerabilities.....	69
Upload scripts.....	69
Upload modules.....	70
Run tasks.....	70
Delete a task.....	74
Remediate your top vulnerabilities.....	74
Remediate high risk vulnerabilities.....	75
Discovering and managing resources.....	76
Discovering resources.....	77
Viewing resource details.....	79
Review Latest events.....	87
Discovery events.....	88
Tasks events.....	89
Integration status.....	90
Remediate audit log.....	90
Troubleshooting.....	91

Welcome to Puppet Remediate™

Remediate helps you minimize the risk of external attacks and data breaches by providing you visibility into the vulnerabilities across your infrastructure, and the tools to prioritize and take action. With Remediate, you can eliminate the repetitive and error-prone steps of manual data handovers between teams.

Remediate has the following capabilities built-in:

- Shared vulnerability data: Integrating with Tenable, Qualys and Rapid7.
- Risk-based prioritization: A dashboard that displays your most critical vulnerabilities, prioritized based on infrastructure context.
- Task-based remediation: Allowing you to fix vulnerabilities at scale. You can upload your own scripts for Remediate to convert to a task or use task-based modules from Puppet Forge.



Helpful Puppet Remediate docs links	Other useful places
<p>Before you upgrade or install:</p> <ul style="list-style-type: none"> System requirements Architecture <p>Getting started:</p> <ul style="list-style-type: none"> Install Remediate Manage user access Add sources Add node credentials <p>Prioritize and fix vulnerabilities:</p> <ul style="list-style-type: none"> Prioritize vulnerabilities Remediate vulnerabilities <p>Discover and manage resources:</p> <ul style="list-style-type: none"> Discover resources Manage resources 	<p>Docs for related Puppet products:</p> <ul style="list-style-type: none"> Puppet Enterprise Bolt Open source Puppet <p>Why and how people are using Remediate:</p> <ul style="list-style-type: none"> Read recent blog posts about Remediate and other Puppet products <p>Get support:</p> <ul style="list-style-type: none"> Search the Support portal and knowledge base <p>Share and contribute:</p> <ul style="list-style-type: none"> Engage with the Puppet community Puppet Forge Open source projects from Puppet on GitHub

Remediate 2.x release notes

New features, enhancements, resolved issues, and known issues for Puppet Remediate 2.x release series.

Version 2.0.1

Released 8 July 2021.

Resolved issues in this release:

- **Tenable.sc integration** - Fixed an issue where Tenable.sc was not verifying Remediate SSL certificates. This fixes CVE-2021-27018.
- **Tenable.io integration** - Fixed a bug where integration fails occasionally after Tenable.io is added as a source.
- **Remediate UI** - Corrected an issue with the **Cancel activity** button where it was not disabled on first click.
- **High severity vulnerabilities** - Remediate image updated to resolve the following high severity vulnerabilities:
 - CVE-2021-20305
 - CVE-2019-25013
 - CVE-2019-5827
 - CVE-2021-27219
 - CVE-2021-3517
 - CVE-2021-3560
- **Accept risk** - Corrected an issue where the **Accept risk** review timed out if a large number of nodes were selected.
- **Accept risk table** - Fixed an issue where the **Accept risk** table did not display properly.
- **Tagging** - Removed a redundant **Assign tags** button from the **Node overview** page.
- **Medium severity vulnerabilities** - Several medium severity vulnerabilities were also resolved in the Remediate image update.
- **Tenable.io integration** - Remediate now detects if there is a corrupted export in Tenable.io and informs the user in the **integration status** page.

Version 2.0.0

Released 14 April 2021.

New in this release:

- **Node tagging** - This release introduces the ability to assign tags to nodes to group them according to any criteria you choose. You can now For full details of this feature, see [Tagging nodes](#) on page 60.
- **Import scanner tags** - You can elect to import tags that are assigned to nodes by the Qualys and Tenable.io vulnerability scanners by selecting the **Import Tags** option when adding them as a source.
- **Cancel data discovery runs** - Data discovery runs can now be cancelled before completion with the click of a button.
- **Vulnerability scanner SSL certificates** - The process for adding certificates signed by an internal certificate authority for vulnerability scanners has been simplified. Certificates signed by an internal certificate authority are now validated by Remediate and customers are strongly recommended to upgrade to version 2.0.0. This fixes CVE-2021-27018. For more information, see [SSL certificate verification for scanners](#) on page 55.
- **Admin user group** - A new 'admin' user group was added to allow individual users to have admin superuser privileges. For more information on this feature, see [Managing user access](#) on page 27.
- **AWS provider improvements** - Improved the way in which the AWS provider discovers regions.
- **Improved provider logging** - Messages previously written to the messages channel are now outputted to edge docker logs.

Resolved issues in this release:

- **Source management** - Simplified the process of deleting a source by removing an unnecessary pop-up drop-down menu.
- **Puppet Tasks updated** - The default Puppet Tasks shipped with Remediate were updated to the latest versions.
- **Remediate tasks** - Resolved an issue where task output was not captured for all tasks.
- **Working with multiple sources** - Added the **Canonical ID** field to **Node Attributes** table to aid troubleshooting of deduplication issues when using multiple discovery sources.
- **Mayday logs** - Information on accessible nodes is now included in the mayday logs.
- **Data discovery UI** - Fixed a bug where the status icon did not update when the **Discover all** button was clicked on the **Manage Sources** pages

- **Puppet Enterprise integration** - Fixed an issue where adding a trailing slash to the Puppet Enterprise source URL resulted in no tasks being discovered.
 - **Running tasks** - Corrected an issue where multiple tasks jobs could be run if the **Run Task** button was clicked multiple times before the initial API request was sent.
 - **Vulnerable node filtering** - Fixed an issue where the filter on the **Nodes** table on the **Vulnerability Details** page did not handle multiple filters.
 - **Accept risk functionality** - Resolved a bug where accepting risks on all nodes for a vulnerability caused an error.
 - **Node filtering** - Corrected an issue where filter requests were aborted or sent twice on the **Nodes by operating system** page.
 - **Run task wizard** - Updated the Run tasks wizard to fix a bug that caused the loss of user-set credentials if the user went back to the **Add Credentials** step from the **Node Overview** step.
 - **Run task functionality** - Fixed an issue where it was possible to run tasks without adding mandatory parameters.
 - **Node filtering options** - Corrected a bug where any attempt to remove a source filter results in an error if the only active filter applies on sources on the **Nodes** page.
 - **Integration status icon** - Updated Remediate to correct an issue where On the first run of a fresh install with a security source, the integration status icon remained 'unset'.
 - **Run task workflow** - Resolved a bug that caused the run task workflow to work incorrectly if only a single node was selected.
 - **unset-override command** - Corrected an issue that prevented the `unset-override` from working for global keys.
 - **Initializing Remediate** - Updated Remediate so that IPv6 no longer needs to be configured on the host for Remediate to initialise.
 - **Node filtering** - Fixed an issue where if all nodes from a filter were selected and the filter removed, select all was still applied in the **Nodes** page.
 - **Adding sources** - Improved the look and feel of the page loading message in the **Add source** page.
 - **Data discovery UI** - The number of nodes that a task can be run against has been capped at 10,000 nodes to prevent slow loading of the UI. A warning message is presented if the number of selected nodes exceeds 10,000.
 - [Remediate 1.x release notes](#) on page 6
- New features, enhancements, resolved issues, and known issues for Puppet Remediate 1.x release series.

Remediate 1.x release notes

New features, enhancements, resolved issues, and known issues for Puppet Remediate 1.x release series.

Version 1.4.2

Released 15 December 2020.

Resolved issues in this release:

- **Node filtering** - A content completion drop-down field was added to help correct an issue where node filter results on Nodes pages were not returned if the node source name was not entered in full.
- **Task parameters** - Remediate was updated to fix an issue where it was possible to run a task without mandatory parameters.
- **Node identification** - An issue was corrected where canonical ID matching between hosts found from VR sources and a PDB source failed.
- **Running tasks on nodes** - Fixed an issue where the selection of individual nodes on Node pages when running a task did not pass the provider information accurately.
- **Install image** - The `docker-compose.yml` file has been updated to fix an issue where Docker pull limits caused the install process to fail.
- **Puppet Enterprise integration** - This release fixes an issue where incompatibilities between the names of identical assets retrieved from Puppet Enterprise and other sources negatively affected PE integration with Remediate.

Version 1.4.1

Released 10 November 2020

Resolved issues in this release:

- **Security hardening** - A number of Remediate open source components have been updated to improve software security.
- **Password management** - Remediate has been updated to prevent browsers from saving login credentials.
- **Installation improvements** - Previously, Remediate could be installed without a license and display very limited functionality. This is no longer possible in this release.
- **UI selection updates** - Custom column selection in the **Vulnerability details** page now persists when the user navigates away from the page.
- **Source/Credential removal improved** - Removal of a source or credential set in Remediate now requires additional confirmation in the UI.
- **Source loading information** - Users are now given information on the progress of the loading of sources.

Known issues in this release:

- IPv6 must be enabled on the host for Remediate to start properly.

Version 1.4.0

Released 30 September 2020

New in this release:

- **Puppet Enterprise tasks** - Remediate has been updated to allow you to add a single Puppet Enterprise source. Use this source to apply PE tasks on your network.
- **Risk management** - In this release, you can use Remediate to accept risk for, and pause reporting of, individual vulnerabilities on selected nodes.
- **TLS certificate management** - Additional CLI commands have been added to help users add and manage third-party signed TLS certificates.

Resolved issues in this release:

- **Package management** - Remediate has been updated to show statuses for Windows packages.
- **User management password hardening** - Password hardening has been implemented for the Remediate user management interface. Users must now employ strong passwords of at least 6 characters in length.
- **Qualys vulnerability scanner integration** - You can now set limits on the date range within which you want vulnerability results to be returned. Previously, Remediate returned all available results.

Version 1.3.0

Released 3 June 2020

New in this release:

- **Record of Remediation** - Puppet Remediate has been updated to provide persistent records of the events initiated to combat vulnerabilities. You can now see data on the latest events and remediation tasks that have been carried out, who initiated them, and whether they were successful or not, for any given vulnerability.
- **Audit log** - Remediate now contains a fuller audit log that provides data on user, source, credential, and task management events. This is available via command line and can be stored or examined as an audit trail.
- **Activity feed** - Data on latest remediation events are now persisted even after restart.
- **Improved data polling configuration** - The **Update Interval** option has been added to the security source configuration. This allows users to set how often Remediate checks for new data. This defaults to 30 minutes. Users who use Remediate in large environments can adjust this value accordingly.
- **Severity threshold configuration** - The **Severity threshold** option has been added to the security source configuration. When configured Remediate does not import any vulnerabilities with a severity less than the configured value. This defaults to importing all vulnerabilities, regardless of severity.

- **Tenable.io integration** - The Tenable.io integration now uses the `last_found` field instead of the `since` field.

Resolved issues in this release:

- **Tenable.sc integration** - Improved error handling when requesting authentication tokens from the Tenable.sc server.

Version 1.2.1

Released 23 March 2020

Resolved issues in this release:

- **Service logs timestamps** - Remediate has been updated to fix an issue where timestamps were missing from service logs.
- **Custom registry installs** - A fix was added for an issue that stopped the install image from loading when using a custom container registry.
- **Vault container shutdown issue** - A fix was added to Remediate to resolve an issue that caused the vault container to be inadvertently shut down.
- **Non-UTF-8 encoded status message issue** - Remediate was updated to fix an issue where the controller panicked if it encountered a non-UTF-8 encoded status message.
- **Non-admin login issue** - This release fixes an issue where non-admin accounts were permitted to log in before the initial configuration was complete.

Version 1.2.0

Released 26 February 2020

New in this release:

- **Vulnerabilities dashboard redesign** - The Remediate Vulnerabilities dashboard has been redesigned with a new Metrics bar, data visualizations, and reorganization of vulnerability and affected node information.
- **Performance improvements** - The database behind Remediate has been completely redesigned for this release to provide significant performance and scale improvements.
- **Puppet Risk Score** - To enhance risk-based prioritisation, the addition of the Puppet Risk Score (the risk score assigned by your vulnerability scanner multiplied by the number of nodes affected) allows you to reduce the risk in your environment even faster.
- **Offline install updated to use custom container registries** - The offline install process has been expanded to allow users to install from their own custom container registries.
- **Integration status redesign** - The **Integration status** indicator has been moved to navigation sidebar to give you immediate information on the health of your latest scans.
- **Source credential testing** - You can now test access credentials when setting up a scanner on the **Add sources** page before you run a scan.

Resolved issues in this release:

- **Duplication of nodes** - The Remediate database redesign incorporates a fix to prevent possible duplication of nodes information in the Remediate UI.
- **Tenable.sc integration** - Remediate has been updated to correct an issue where some vulnerabilities reported in Tenable.sc were not reflected in the Remediate UI.

Version 1.1.1

Released 9 January 2020

New in this release:

- **Scan refresh functionality** - You can now manually trigger a rescan of all or selected resources from the **Manage Sources** page.
- **SUSE Linux 11 integration** - Remediate now fully supports SUSE Linux 11.

- **Activity feed updates** - The **Recent Events** table has been updated to include the username of the event initiator.

Resolved issues in this release:

- **Offline install** - Remediate now uses a dedicated Docker image bundle and `docker-compose.yml` file for offline installs.
- **SSH Updates** - Remediate has been updated to enable SSH access to hosts that use CBC ciphers.
- **Container DNS issue** - Remediate has now been updated to fix a DNS issue where the container added `ndots` configuration to the `/etc/resolv.conf` file.
- **Qualys API integration** - Remediate is now able to parse human-readable durations used by the Qualys API that may be reported during daylight savings.

Version 1.1.0

Released 31 October 2019

New in this release:

- **Tenable.sc support** - Remediate now supports the Tenable.sc (Security Center) vulnerability scanner.
- **Multi-user support** - User management in Remediate has been completely updated. Administrators can now:
 - Create and manage multiple users accounts.
 - Assign different group privileges to user accounts.
 - Configure Remediate to work with LDAP or Active Directory servers.
- **RBAC Permissions** - Assign user privileges to:
 - Add, remove, or run tasks
 - Add or remove credentials
 - Add or remove sources
- **Remediation workflow improvements** - The vulnerability remediation workflow has been improved and additional information on the vulnerability and the steps needed to remediate it are provided.

Resolved issues in this release:

- **Offline install not working** - The `-o` flag has been introduced to the `remediate start` command to ensure you can start remediate when there is no internet access.
- **OpenSSH private key support** - Remediate now supports the latest version of OpenSSH private keys.

Version 1.0.1

Released 1 August 2019.

This is the initial release of Remediate.

Known issues:

- **Unable to install Remediate on Debian8 with the default kernel module.** Upgrade to Kernel 4.9 and install Remediate again.
- **Network discovered nodes being shown as cloud instances.** Hosts discovered via their IP address are counted as a cloud instance and visible in the top cloud instance by region card.
- **Due to inconsistent DNS lookups, tasks fail to run on discovered hosts.** When discovered hosts are running on the same domain, an inconsistent DNS lookup between discovering hosts and running tasks on discovered hosts results in tasks failing.
- **In a multi-network environment, the first discovery run might not identify the IP or hostname.** Wait for the second discovery run, which happens automatically after four hours.

Installing Remediate

- [Architecture](#) on page 10

Puppet Remediate consists of a number of components and services, each one running as an individual Docker container.

- [System requirements](#) on page 13

Before installing Puppet Remediate, check to ensure your system meets these requirements.

- [Install Remediate on online nodes](#) on page 15

Install Puppet Remediate on a Linux or Windows machine that is connected to the internet.

- [Install Remediate on offline nodes](#) on page 17

If any of your swarm nodes are offline (do not have external connectivity), you must manually import the Puppet Remediate images in order to install the product.

- [Install Remediate from a custom container registry](#) on page 20

You can also use a custom Docker registry to install Puppet Remediate.

- [Install Remediate in online multi-network deployments](#) on page 21

Puppet Remediate connects to security providers to discover hosts with vulnerabilities. To take action and fix the vulnerabilities, the system needs to connect directly to the hosts. If the host you want to fix is deployed in different network segments that are not directly accessible from where you installed Remediate, you can setup a multi-network deployment.

- [Install Remediate in offline multi-network deployments](#) on page 22

Puppet Remediate connects to security providers to discover hosts with vulnerabilities. To take action and fix the vulnerabilities, the system needs to connect directly to the hosts. If the host you want to fix is deployed in different network segments that are not directly accessible from where you installed Remediate, you can setup a multi-network deployment.

- [Verify Docker Compose file for online installs](#) on page 22

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

- [Using TLS certificates with Remediate](#) on page 23

By default Puppet Remediate uses self-signed TLS certificates. You can also optionally set and configure your own certificates for Remediate if you prefer.

- [Analytics data collection](#) on page 24

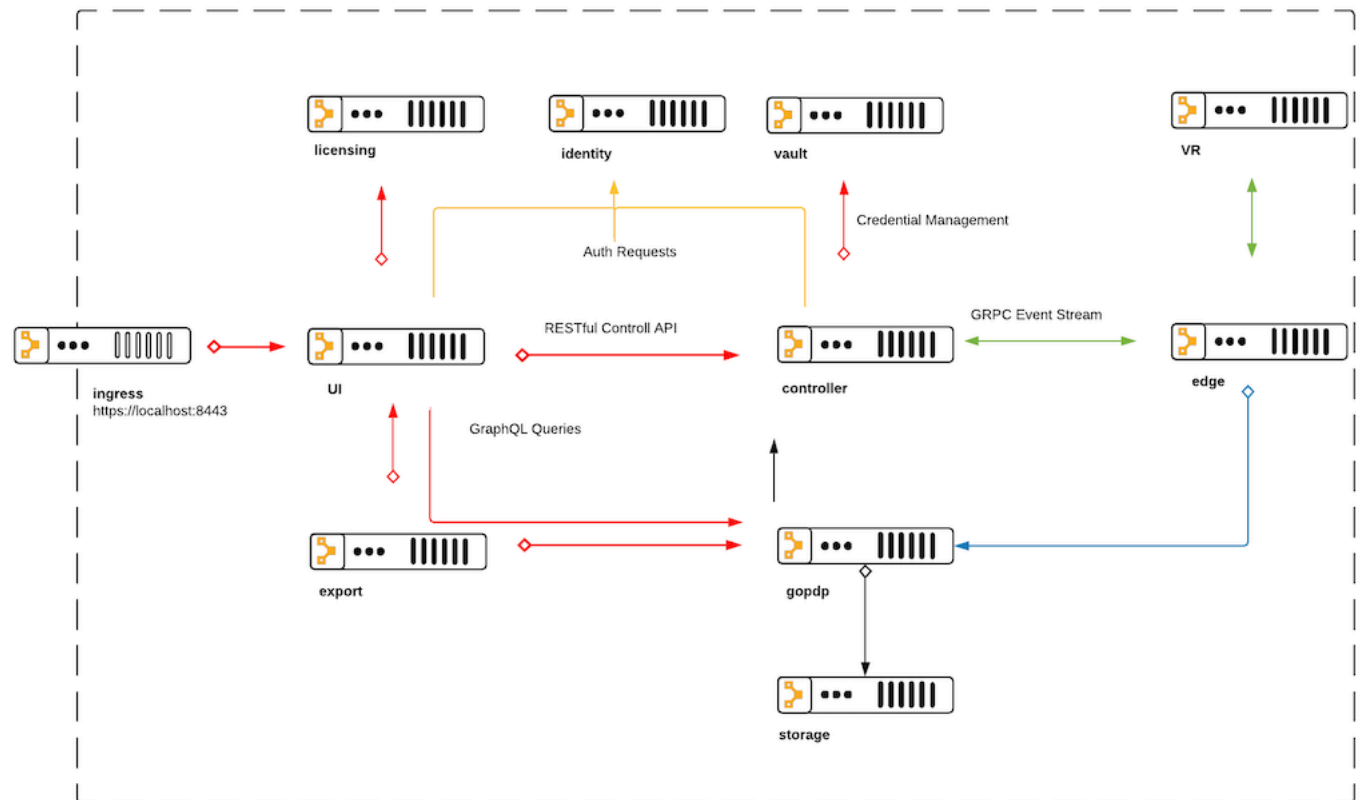
Puppet Remediate automatically collects data about how you use the product. If you want to opt out of providing this data, you can do so, either during or after installing.

- [Uninstall Remediate](#) on page 24

You can uninstall Puppet Remediate easily with a couple of commands.

Architecture

Puppet Remediate consists of a number of components and services, each one running as an individual Docker container.



Service	Container	Description
Licensing	remediate_licensing	Stores the user licensing information and is queried by the UI using the licensing API.
Ingress	remediate_frontdoor	The nginx front end listening on port 8443 (HTTPS).
VR	remediate_vr	Connects to and retrieves host and vulnerability data from Tenable, Qualys and Rapid7.
Storage	remediate_storage	The storage layer for discovered data which communicates with the remediate_gopdp container on port 5432.
Identity	remediate_identity	Generates the bearer token for the default user roles: admin and viewer.

Service	Container	Description
Export	remediate_export	The export service consumes API requests from the UI on port 9200 (HTTPS) and queries the data platform on port 8082 (GRPC).
gopdp	remediate_gopdp	The data platform is an ingest service implementation that processes discovered data from the edge on port 8082 (GRPC), and exposes the query API to the UI on port 8084.
Controller	remediate_controller	<p>The controller manages the discovery process by:</p> <ul style="list-style-type: none"> • Authenticating each API request by validating the bearer token with remediate_identity on port 5556 (HTTPS). • Retrieving source and host credentials from the vault on port 8200 (HTTPS). • Consuming the discovery and task API requests from the UI on port 9999 (HTTPS). • Dispatching discovery and task commands to the edge on port 8081 (GRPC).
Edge	remediate_edge	The edge consumes the discovery API requests from the controller and invokes each source the user adds. It discovers vulnerabilities, resources, executes ad hoc tasks on target hosts, and submits data to the data platform. The edge services consist of a set of pluggable providers that are determined by which sources are added.
Vault	remediate_vault	The secure store for source and node credentials.

Service	Container	Description
UI	remediate_ui	The UI enables you to add sources, credentials, and run tasks on target hosts by initiating discovery API requests to the controller on port 9999 (HTTPS). To populate the dashboards and provide a high-level summary view of your infrastructure, the UI queries the data platform on port 8084 (HTTPS) for vulnerabilities and discovered resources.

System requirements

Before installing Puppet Remediate, check to ensure your system meets these requirements.

Supported operating systems

You can install Remediate on the following operating systems.

Operating System	Versions	System	Prerequisite
CentOS	7 or higher	<ul style="list-style-type: none"> • Architecture: x64 • Minimum Memory: 8GB • Minimum Storage: 20GB • Minimum CPUs: 2 	Docker CE 17.04.0-ce or higher, or Docker EE 17.06.1-ee or higher for Linux. Docker for Windows. When you install Remediate on a virtual machine, you must enable nested virtualization. For more information, see the documentation for the hypervisor you are using.
Red Hat Enterprise Linux	7		
Debian	8 or higher		
Ubuntu	14.04 or higher		
Windows	10		

Docker requirements

Both Docker CE and Docker EE editions include the option to run Kubernetes as a single-node cluster on a local machine using [port 8080](#). Remediate requires that no other application use ports 8080 and 8443. Note that Docker swarm mode requires [additional ports](#).

If using Docker for Windows, the virtual machine must be configured with 8 GB of memory.

To prevent running out of storage, configure log rotation by editing the `log-driver` and `log-opts` parameters within the daemon configuration file that is located here:

- Linux: `/etc/docker/daemon.json`
- Windows: `%programdata%\docker\config\daemon.json`

To deploy the Docker Compose file, you need to install [Docker Compose](#) version 1.24.1.

Discoverable operating systems

Discover resources that run on these operating systems.

Operating system	Versions	Prerequisite
Enterprise Linux: <ul style="list-style-type: none"> CentOS Red Hat Enterprise Linux 	5 or higher	For the account Remediate authenticates with, configure bash as the login shell.
Debian	7 or higher	
SUSE Linux Enterprise Server	12	
Ubuntu	14.04 or higher	
Windows Server	2012 or higher	

System configuration

Before installing Remediate, make sure that your network is properly configured, and that the time is correctly set and managed on each server. These are the port requirements for a Remediate installation.

Port	Description
<ul style="list-style-type: none"> 443 (HTTPS) 	Required to install or update to the latest version of Remediate from: <ul style="list-style-type: none"> storage.googleapis.com gcr.io
<ul style="list-style-type: none"> 8443 (HTTPS) 	Required to view the Remediate dashboards.
<ul style="list-style-type: none"> 22 	Required for SSH authentication on discovered Linux hosts.
<ul style="list-style-type: none"> 5986 (HTTPS) 5985 (HTTP) 	Required for WinRM authentication on discovered Windows hosts.

Important: Remediate requires all interfaces to allow traffic to the loopback network (127.0.0.0/8).

Hardware requirements

Remediate requires:

Storage	20.0 GB
Memory	8.0 GB
CPUs	2

Supported browsers

The Remediate user interface is supported on the latest versions of the following browsers:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

Install Remediate on online nodes

Install Puppet Remediate on a Linux or Windows machine that is connected to the internet.

Before you begin

Make sure you meet the system requirements for installing and running Remediate, and that you have received your [license](#).

1. Install and run Docker on the node where you want to install Remediate.

a) Download Docker for your operating system:

- Linux: [Docker CE](#) 17.04.0-ce or higher, or [Docker EE](#) 17.06.1 or higher.
- Windows: [Docker for Windows](#).

b) Download [Docker Compose](#).

Note: If installing Docker Compose on Windows, create a new environment variable called `COMPOSE_CONVERT_WINDOWS_PATHS` and set it to 1. By doing this, you enable path conversion from Windows-style to Unix-style in volume definitions. For more information, see the [Docker Compose documentation](#).

c) Initialize a swarm by running the following command:

```
docker swarm init
```

2. Download the Remediate [Docker Compose file](#) to the same directory as your license.

Note: If you want to manually validate the signature, see [Verify Docker Compose file for online installs](#) on page 22 for more information.

3. Run the following command (replacing `your-license.json` with your own license):

```
docker-compose run remediate start --license-file your-license.json
```

The container images are pulled from the Google Cloud Platform.

- To check that all the images have downloaded and that the containers have started running, run the following command:

```
docker ps
```

The list of running containers:

CONTAINER ID	IMAGE	CREATED	STATUS
36139cda23ca	gcr.io/puppet-discovery/pdp-go:<version>	3 minutes ago	Up 3 minutes (healthy)
app/entrypoint.sh"	remediate_godp.1.w035aad0ifvu0ynaimxy64lcq		8082/
43709401f394	gcr.io/puppet-discovery/pd-storage:<version>	3 minutes ago	Up 3 minutes (healthy)
"storageEntryPoint.s..."	remediate_storage.1.omvlrrd3rwhnwo3ngurgtcnzk		8087/tcp
5432/tcp	gcr.io/puppet-discovery/licensing-api:<version>	3 minutes ago	Up 3 minutes (healthy)
551b061acc98	remediate_licensing.1.kdoj7s492el77rdpc2rza3fx7		"/
entrypoint.sh"	vault:1.1.2	3 minutes ago	Up 3 minutes (healthy)
43a3568e6b5a	"docker-entrypoint.s..."		8200/tcp
"docker-entrypoint.s..."	remediate_vault.1.k0jgyxxi451dn4pqadhgvjfo	3 minutes ago	Up 3 minutes (healthy)
8200/tcp	gcr.io/puppet-discovery/identity:<version>	3 minutes ago	Up 3 minutes (healthy)
11b90d8564ef	remediate_identity.1.blwkt4kx4ps107949k14ctsqx		8080/
opt/jboss/tools/do..."	gcr.io/puppet-discovery/frontdoor:<version>	3 minutes ago	Up 3 minutes (healthy)
tcp, 8443/tcp	remediate_frontdoor.1.oib8jsr7u4z6wqxvjz02mxcxz		80/
1235fd27cbdc	gcr.io/puppet-discovery/pdp-proxy:<version>	3 minutes ago	Up 3 minutes (healthy)
"nginx -g 'daemon of..."	remediate_export.1.uin509pnc2zb4pf3rdjrlso0x		8080/
tcp	gcr.io/puppet-discovery/node-ui:<version>	3 minutes ago	Up 3 minutes (healthy)
a3a641af5269	remediate_ui.1.3u0tewgou7t4hz2c46nn4mydo		"/
pdp-proxy-svc"	gcr.io/puppet-discovery/edge:<version>	3 minutes ago	Up 3 minutes
tcp	remediate_edge.1.koacwnjoce2tabwcbi73619fu		9997/
c60250b8a2eb	gcr.io/puppet-discovery/controller:<version>	4 minutes ago	Up 4 minutes (healthy)
usr/src/app/entryp..."	remediate_controller.1.mww2fm9up4lmeisjorul89hr4		9999/
f9af99dc9ca7	gcr.io/puppet-discovery/vr:<version>	4 minutes ago	Up 4 minutes (healthy)
edge-svc"	remediate_vr.1.yjlliup91g4mac1bklvww2nqq		"/
tcp			9997/
149485b54fec			"/
controller-svc"			9999/
tcp			"/
f9f1ab4a029d			"/
vr-svc"			"/

- To access Remediate on a local workstation, the URL is `https://localhost:8443`, or port 8443 on the host where you installed Remediate.

Note: When you first open the site, a warning message is displayed that indicates the site certificate is untrusted. This is because Remediate uses a self-signed certificate and is expected behavior. Accept the certificate to continue.

- Read and accept the software license agreement.
- Sign in to Remediate.

For default usernames and passwords, see [Managing user access](#) on page 27.

Related information

[System requirements](#) on page 13

Before installing Puppet Remediate, check to ensure your system meets these requirements.

[Install Remediate on offline nodes](#) on page 17

If any of your swarm nodes are offline (do not have external connectivity), you must manually import the Puppet Remediate images in order to install the product.

[Install Remediate from a custom container registry](#) on page 20

You can also use a custom Docker registry to install Puppet Remediate.

[Verify Docker Compose file for online installs](#) on page 22

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

[Managing user access](#) on page 27

As a Remediate administrator, you can create new user accounts and assign group-based access privileges to them.

Install Remediate on offline nodes

If any of your swarm nodes are offline (do not have external connectivity), you must manually import the Puppet Remediate images in order to install the product.

Before you begin

Prior to installing Remediate on any offline nodes, check that you have carried out the following prerequisite tasks:

- Make sure you meet the system requirements for installing and running Remediate, and that you have received your license.
- Ensure that Docker and Docker Compose are installed on the nodes where you want to install the Remediate images. If you are installing Docker Compose on Windows, ensure that you create a new Windows environment variable called `COMPOSE_CONVERT_WINDOWS_PATHS` and set it to 1. This enables path conversion from Windows-style to Unix-style in volume definitions.
- Ensure that your license is added to the nodes where you want to install the Remediate images.
- Remediate may fail to start in offline (`-o`, `--offline`) mode if the Docker host is unable to reach its configured DNS server. Please ensure that you have DNS configured correctly.

To install Remediate on nodes that do not have network connectivity:

1. On a node with internet connectivity:
 - a) Download the offline Remediate image bundle (<https://storage.googleapis.com/remediate/stable/latest/offline/images.tar.gz>)
 - b) Download the offline `docker-compose.yml` file (<https://storage.googleapis.com/remediate/stable/latest/offline/docker-compose.yml>)

Note: Skip this step if you are using your own custom Docker registry.

2. Optionally, you can verify the image bundle and offline `docker-compose.yml` files signatures:

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

- a) Download the [offline docker-compose.yml file signature](#) and the [image bundle signature](#), along with the [public key](#) to the same directory as your license file.
- b) Run the following commands:

```
openssl dgst -sha256 -verify puppet-remediate-signing-key.pub -signature
docker_compose_signature docker-compose.yml
```

And:

```
openssl dgst -sha256 -verify puppet-remediate-signing-key.pub -signature
images_signature images.tar.gz
```

If the signature is valid, you get the following response for each command:

```
Verified Ok
```

3. Copy the Remediate image bundle and offline `docker-compose.yml` file to the offline node where you want to install Remediate.
4. On the node where you want to install Remediate, initialize a swarm by running the following command:

```
docker swarm init
```

5. Run the Docker load command:

```
docker load -i images.tar.gz
```

6. Use the following command to start Remediate (replacing `your-license.json` with your own license):

```
docker-compose run remediate start -o --license-file ./your-license.json
```

7. To check that the containers have started running, run this command:

```
docker ps
```

The list of running containers:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
36139cda23ca	gcr.io/puppet-discovery/pdp-go:<version>	app/entrypoint.sh"	3 minutes ago	Up 3 minutes (healthy)	8082/tcp, 8087/tcp	remediate_godp.1.w035aad0ifvu0ynaimxy64lcq
43709401f394	gcr.io/puppet-discovery/pd-storage:<version>	"storageEntryPoint.s..."	3 minutes ago	Up 3 minutes (healthy)	5432/tcp	remediate_storage.1.omvlrrd3rwhnwo3ngurgtcnzk
551b061acc98	gcr.io/puppet-discovery/licensing-api:<version>	entrypoint.sh"	3 minutes ago	Up 3 minutes (healthy)		remediate_licensing.1.kdoj7s492el77rdpc2rza3fx7
43a3568e6b5a	vault:1.1.2	"docker-entrypoint.s..."	3 minutes ago	Up 3 minutes (healthy)	8200/tcp	remediate_vault.1.k0jgyxxi451dn4pqadhgvjfo
11b90d8564ef	gcr.io/puppet-discovery/identity:<version>	opt/jboss/tools/do..."	3 minutes ago	Up 3 minutes (healthy)	8080/tcp, 8443/tcp	remediate_identity.1.blwkt4kx4ps107949k14ctsqx
1235fd27cbdc	gcr.io/puppet-discovery/frontdoor:<version>	"nginx -g 'daemon of..."	3 minutes ago	Up 3 minutes (healthy)	80/tcp	remediate_frontdoor.1.oib8jsr7u4z6wqxvz02mxcxz
a3a641af5269	gcr.io/puppet-discovery/pdp-proxy:<version>	pdp-proxy-svc"	3 minutes ago	Up 3 minutes (healthy)	9200/tcp	remediate_export.1.uin509pnc2zb4pf3rdjrlso0x
c60250b8a2eb	gcr.io/puppet-discovery/node-ui:<version>	usr/src/app/entryp..."	3 minutes ago	Up 3 minutes (healthy)		remediate_ui.1.3u0tewgou7t4hz2c46nn4mydo
f9af99dc9ca7	gcr.io/puppet-discovery/edge:<version>	edge-svc"	3 minutes ago	Up 3 minutes	9997/tcp	remediate_edge.1.koacwnjoce2tabwcbi73619fu
149485b54fec	gcr.io/puppet-discovery/controller:<version>	controller-svc"	4 minutes ago	Up 4 minutes (healthy)	9999/tcp	remediate_controller.1.mww2fm9up4lmeisjorul89hr4
f9f1ab4a029d	gcr.io/puppet-discovery/vr:<version>	vr-svc"	4 minutes ago	Up 4 minutes (healthy)		remediate_vr.1.yjlliup91g4maclbklvww2nqq

8. To access Remediate on a local workstation, the URL is `https://localhost:8443`, or port 8443 on the host where you installed Remediate.

Note: When you first open the site, a warning message is displayed that indicates the site certificate is untrusted. This is because Remediate uses a self-signed certificate and is expected behavior. Accept the certificate to continue.

9. Read and accept the software license agreement.
10. Sign in to Remediate.

For default usernames and passwords, see [Managing user access](#) on page 27.

Related information

[System requirements](#) on page 13

Before installing Puppet Remediate, check to ensure your system meets these requirements.

[Install Remediate on online nodes](#) on page 15

Install Puppet Remediate on a Linux or Windows machine that is connected to the internet.

[Install Remediate from a custom container registry](#) on page 20

You can also use a custom Docker registry to install Puppet Remediate.

[Managing user access](#) on page 27

As a Remediate administrator, you can create new user accounts and assign group-based access privileges to them.

Install Remediate from a custom container registry

You can also use a custom Docker registry to install Puppet Remediate.

Before you begin

Prior to installing Remediate using a custom Docker registry, check that you have carried out the following prerequisite tasks:

- Make sure you meet the system requirements for installing and running Remediate, and that you have received your license.
- Ensure that Docker and Docker Compose are installed on the nodes where you want to install the Remediate images. If you are installing Docker Compose on Windows, ensure that you create a new Windows environment variable called `COMPOSE_CONVERT_WINDOWS_PATHS` and set it to 1. This enables path conversion from Windows-style to Unix-style in volume definitions.
- Ensure that your license is added to the nodes where you want to install the Remediate images.
- If you use a custom Docker registry to store the Remediate Docker images, you must set the `REGISTRY` environment variable to point to the URL of your custom docker registry, e.g.:

```
export REGISTRY=my-custom-docker-registry.local.net
```

In this example, Remediate contacts `my-custom-docker-registry.local.net` to download the docker images it needs.

1. On the node where you want to install Remediate, initialize a swarm by running the following command:

```
docker swarm init
```

2. Download the Remediate [Docker Compose file](#) to the same directory as your license.

Note: If you want to manually validate the signature, see [Verify Docker Compose file for online installs](#) on page 22 for more information.

3. Run the following command (replacing `your-license.json` with your own license):

```
docker-compose run remediate start --license-file your-license.json
```

The container images are pulled from the custom registry location specified by your `REGISTRY` environment variable.

4. To check that all the images have downloaded and that the containers have started running, run the following command:

```
docker ps
```

5. To access Remediate on a local workstation, the URL is `https://localhost:8443`, or port 8443 on the host where you installed Remediate.

Note: When you first open the site, a warning message is displayed that indicates the site certificate is untrusted. This is because Remediate uses a self-signed certificate and is expected behavior. Accept the certificate to continue.

6. Read and accept the software license agreement.

7. Sign in to Remediate.

For default usernames and passwords, see [Managing user access](#) on page 27.

Related information

[System requirements](#) on page 13

Before installing Puppet Remediate, check to ensure your system meets these requirements.

[Install Remediate on online nodes](#) on page 15

Install Puppet Remediate on a Linux or Windows machine that is connected to the internet.

[Install Remediate on offline nodes](#) on page 17

If any of your swarm nodes are offline (do not have external connectivity), you must manually import the Puppet Remediate images in order to install the product.

[Verify Docker Compose file for online installs](#) on page 22

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

[Managing user access](#) on page 27

As a Remediate administrator, you can create new user accounts and assign group-based access privileges to them.

Install Remediate in online multi-network deployments

Puppet Remediate connects to security providers to discover hosts with vulnerabilities. To take action and fix the vulnerabilities, the system needs to connect directly to the hosts. If the host you want to fix is deployed in different network segments that are not directly accessible from where you installed Remediate, you can setup a multi-network deployment.

In a multi-network deployment, an edge service is deployed inside each network segment. Remediate instructs the edge to connect to the hosts when executing tasks, and then leverages Docker swarm to deploy an edge service on each swarm worker. Workers must have direct connectivity to the manager — the main node where you installed Remediate. For more information on workers and managers, see the [Docker documentation](#).

For more information on installing Remediate on nodes that are not connected to the internet, see instructions on how to [Install Remediate on offline nodes](#) on page 17.

Note: The following procedure assumes that you have already run `docker swarm init` on the main Remediate node as part of the installation process to designate it as swarm manager.

1. To add a worker to the swarm, run the following command on the manager node:

```
docker swarm join-token worker
```

2. Using the key output from the above command as the <TOKEN>, run the following command on each worker node:

```
docker swarm join --token <TOKEN>
```

This command adds a worker to each network segment.

3. On the manager node, follow the normal installation instructions for Remediate. The swarm automatically deploys the edge on the workers.
4. After the installation is complete, verify the expected number of edges is running with the following command:

```
docker service ps remediate_remote-edge
```

Related information

[Install Remediate on offline nodes](#) on page 17

If any of your swarm nodes are offline (do not have external connectivity), you must manually import the Puppet Remediate images in order to install the product.

Install Remediate in offline multi-network deployments

Puppet Remediate connects to security providers to discover hosts with vulnerabilities. To take action and fix the vulnerabilities, the system needs to connect directly to the hosts. If the host you want to fix is deployed in different network segments that are not directly accessible from where you installed Remediate, you can setup a multi-network deployment.

In a multi-network deployment, an edge service is deployed inside each network segment. Remediate instructs the edge to connect to the hosts when executing tasks, and then leverages Docker swarm to deploy an edge service on each swarm worker. Workers must have direct connectivity to the manager — the main node where you installed Remediate. For more information on workers and managers, see the [Docker documentation](#).

Note: The following procedure assumes that you have already run `docker swarm init` on the main Remediate node as part of the installation process to designate it as swarm manager.

1. On a node with internet connectivity:
 - a) Download the offline Remediate image bundle (<https://storage.googleapis.com/remediate/stable/latest/offline/images.tar.gz>).

Note: Skip this step if you are using your own custom Docker registry.

2. Optionally, you can verify the image bundle file signature.

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

- a) Download the [image bundle signature](#), along with the [public key](#) to the same directory as your license file.
- b) Run the following command:

```
openssl dgst -sha256 -verify puppet-remediate-signing-key.pub -signature
images_signature images.tar.gz
```

If the signature is valid, you get the following response:

```
Verified Ok
```

3. Copy the Remediate image bundle to the offline node where you want to install Remediate.
4. Using the key output from the above command as the `<TOKEN>`, run the following command on each worker node:

```
docker swarm join --token <TOKEN>
```

This command adds a worker to each network segment.

5. On the manager node, follow the normal installation instructions for Remediate. The swarm automatically deploys the edge on the workers.
6. After the installation is complete, verify the expected number of edges is running with the following command:

```
docker service ps remediate_remote-edge
```

Verify Docker Compose file for online installs

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

1. Download the [signature](#) file and the [public key](#) to the same directory as your `docker-compose.yml` and license file.

Note: For instructions on downloading the `docker-compose.yml` and license files, see the instructions on how to **Install Remediate on online nodes**.

2. Run the following command:

```
openssl dgst -sha256 -verify puppet-remediate-signing-key.pub -signature
signature docker-compose.yml
```

If the signature is valid, you get the following response:

```
Verified Ok
```

Related information

[Install Remediate on online nodes](#) on page 15

Install Puppet Remediate on a Linux or Windows machine that is connected to the internet.

[Install Remediate on offline nodes](#) on page 17

If any of your swarm nodes are offline (do not have external connectivity), you must manually import the Puppet Remediate images in order to install the product.

Using TLS certificates with Remediate

By default Puppet Remediate uses self-signed TLS certificates. You can also optionally set and configure your own certificates for Remediate if you prefer.

When you log into Remediate, it uses a self-signed site certificate. If you don't want to use this automatically-generated self-signed cert, you can employ your own certificate signed by a third-party certificate authority.

The Remediate CLI tool contains commands you use to set, unset, and refresh certificates.

- `start --certfile [string] --keyfile [string]` - Starts Remediate using the specified certificate and key instead of the default self-signed certificate.
- `set-frontdoor-cert --certfile [string] --keyfile [string]` - Restarts Remediate using the specified certificate and key.
- `unset-frontdoor-cert` - Unsets any certificates set using either of the above two commands and tells Remediate to use the default sign-signed certificates.
- `refresh-internal-certs` - Remediate uses a set of internal Docker certificates that are valid for one year only. On offline instances, expired certificates are not updated automatically on upgrade. Use this command refresh any expired certificates. This command does not refresh any certificates you have set using the above two commands.

For more detailed information on these Remediate CLI commands, see [Configuring Remediate](#) on page 43.

Related information

[Configuring Remediate](#) on page 43

After installing Puppet Remediate, configure and administer the application by using the command line interface (CLI).

[SSL certificate verification for scanners](#) on page 55

You can verify SSL certificates signed by an internal certificate authority for Rapid7 and Tenable.sc.

Analytics data collection

Puppet Remediate automatically collects data about how you use the product. If you want to opt out of providing this data, you can do so, either during or after installing.

What data does Remediate collect?

When Puppet Remediate starts, and restarts, it collects the following information:

- Browser name and version.
- Device type.
- Form submission events, but not form input data.
- Anonymized IP address
- Geographic data inferred by the anonymized IP address.
- JavaScript exceptions.
- Link, button, and form element clicks.
- Operating system and version.
- Page views.
- Session durations and session details including:
 - number and title of screens viewed
 - screens arrived at
 - exit screens

How does Puppet use the data?

The data we collect is one of many methods we use for learning about our customers. For example, understanding how you navigate through the web interface can help us optimize the navigation so that you can complete your work faster. And, learning what kinds of data sources are used, helps us to prioritize new functionality like increasing how many resources we can discover for those sources.

Opt out before installing

Remediate does not collect any analytics data during the installation process. However, if you want to disable data collection before you start Remediate for the first time, run the following Docker Compose command from the folder where your `docker-compose.yml` file is located:

```
docker-compose run remediate config set-override analytics false
```

Opt out after installing

If you have already installed Remediate, and want to disable data collection, run the following Docker Compose command from the folder where your `docker-compose.yml` file is located:

```
docker-compose run remediate config set-override analytics false
```

Restart Remediate to complete the update:

```
docker-compose run remediate restart
```

Uninstall Remediate

You can uninstall Puppet Remediate easily with a couple of commands.

To uninstall Remediate:

1. Stop and remove the running Remediate containers and any associated volumes by issuing the following command from the folder where your `docker-compose.yml` file is located:

```
docker-compose run remediate reset
```

2. To delete the Remediate containers on your system, use the following Docker command:

```
docker system prune --filter label=com.docker.compose.service=remediate
```

Upgrading Remediate

Upgrading Puppet Remediate is easy can be done for both online and offline nodes.

- [Upgrade Remediate on online nodes](#) on page 25

Upgrading to a new version of Puppet Remediate on a node with internet access can be done with a few simple commands.

- [Upgrade Remediate on offline nodes](#) on page 25

Upgrading to new version of Puppet Remediate on offline machines can be done with a few simple commands.

- [Upgrade Remediate from a custom registry](#) on page 26

You can also upgrade to a new version of Puppet Remediate if you use a custom registry with a few simple commands.

Upgrade Remediate on online nodes

Upgrading to a new version of Puppet Remediate on a node with internet access can be done with a few simple commands.

To upgrade to the latest version of Remediate:

Run the Remediate update command:

```
docker-compose run remediate update
```

Important: If you are upgrading from Remediate 1.1.x, you must stop Remediate and remove the `oauth_client.json` file *before* running the update command:

```
docker-compose run remediate stop
```

```
docker secret rm oauth_client.json
```

There is no need to do this if you are upgrading from Remediate 1.2.x or later.

Upgrade Remediate on offline nodes

Upgrading to new version of Puppet Remediate on offline machines can be done with a few simple commands.

To upgrade to the latest version of Remediate on an offline node:

1. On a node with internet connectivity:
 - a) Download the offline Remediate image bundle (<https://storage.googleapis.com/remediate/stable/latest/offline/images.tar.gz>)
 - b) Download the offline `docker-compose.yml` file (<https://storage.googleapis.com/remediate/stable/latest/offline/docker-compose.yml>)

Note: Skip this step if you are using your own custom Docker registry.

2. Optionally, you can verify the image bundle and offline `docker-compose.yml` files signatures:

With each Puppet Remediate release, a digital signature is created using the private key portion of an asymmetric key. You can manually validate the signature using the public key portion of the same asymmetric key.

 - a) Download the [offline docker-compose.yml file signature](#) and the [image bundle signature](#), along with the [public key](#) to the same directory as your license file.
 - b) Run the following commands:

```
openssl dgst -sha256 -verify puppet-remediate-signing-key.pub -signature
docker_compose_signature docker-compose.yml
```

And:

```
openssl dgst -sha256 -verify puppet-remediate-signing-key.pub -signature
images_signature images.tar.gz
```

If the signature is valid, you get the following response for each command:

```
Verified Ok
```

3. Copy the Remediate image bundle and offline `docker-compose.yml` file to the offline node where you want to install Remediate.
4. Run the Docker load command:

```
docker load -i images.tar.gz
```

5. Stop Remediate:

```
docker-compose run remediate stop
```

6. Use the following command to start Remediate (replacing `your-license.json` with your own license):

```
docker-compose run remediate start -o --license-file ./your-license.json
```

Upgrade Remediate from a custom registry

You can also upgrade to a new version of Puppet Remediate if you use a custom registry with a few simple commands.

Before you begin

Prior to upgrading Remediate using a custom Docker registry, check that you have carried out the following prerequisite tasks:

- Make sure you meet the system requirements for installing and running the new version of Remediate, and that you have received your license.
- Ensure that your license is added to the nodes where you want to install the Remediate images.

- If you use a custom Docker registry to store the Remediate Docker images, you must set the *REGISTRY* environment variable to point to the URL of your custom docker registry, e.g.:

```
export REGISTRY=my-custom-docker-registry.local.net
```

In this example, Remediate contacts `my-custom-docker-registry.local.net` to download the docker images it needs.

1. Download the Remediate [Docker Compose file](#) to the same directory as your license.

Note: If you want to manually validate the signature, see [Verify Docker Compose file for online installs](#) on page 22 for more information.

2. Stop Remediate:

```
docker-compose run remediate stop
```

3. Run the Remediate `start` command (replacing `your-license.json` with your own license):

```
docker-compose run remediate start --license-file your-license.json
```

The container images are pulled from the custom registry location specified by your *REGISTRY* environment variable.

Related information

[Install Remediate from a custom container registry](#) on page 20

You can also use a custom Docker registry to install Puppet Remediate.

[Managing user access](#) on page 27

As a Remediate administrator, you can create new user accounts and assign group-based access privileges to them.

Managing user access

As a Remediate administrator, you can create new user accounts and assign group-based access privileges to them.

When you first install Remediate, you have one default superuser account: `admin`. Admin users have full access to all Remediate user interface functionality and can add and manage other users.

Role	Username	Default password	Permissions
Administrator	admin	@Admin	<ul style="list-style-type: none"> • Add and remove sources and credentials. • Upload scripts and modules. • Run tasks. • View dashboards. • View all node, package, container listing and details pages. • Add, update, delete, log out other users. • Accept risk for vulnerabilities.

Manage user accounts

As an admin user, you can add, view, update or delete additional user accounts and assign them to groups.

The following groups are available to add to accounts:

Permission Group	Description
accept-risk	Accept risk for a given vulnerability on selected nodes.
add-credential	Add access credentials for a node.
add-source	Add a vulnerability scanner or infrastructure source.
add-task	Add a new remediation task.
admin	Complete first-run configuration. Members of this group also have all the permissions of the admin superuser role except the ability to login into the user administration UI and manage users.
remove-credential	Remove access credentials for a node.
remove-source	Remove a vulnerability scanner or infrastructure source.
remove-task	Remove a new remediation task.
run-task	Run a remediation task.

Each group represents a user privilege that be granted to a user account. If you do not add any groups to a user account, that user only has read-only access to the Remediate UI.

In addition to creating user accounts manually, you can also configure remediate to pull user information from your LDAP or Active Directory server.

- [Add new users](#) on page 29

As a user administration UI admin user, you can add new user accounts and assign them group privileges.

- [Update a user](#) on page 30

From time to time you might need to change a user's password, or update the groups assigned to their account.

- [Delete a user](#) on page 30

Remediate administrators can delete other user accounts, including admin accounts.

- [Force log out](#) on page 31

Holders of the admin account can force other users to log out.

- [LDAP configuration](#) on page 31

You can set up Remediate to use LDAP content to authenticate users.

- [Active Directory configuration](#) on page 33

You can set up Remediate to use Active Directory to authenticate users.

- [LDAP mappers](#) on page 35

LDAP mappers are listeners, which are triggered by the LDAP Provider at various points, and provide another extension point to LDAP integration.

- [Work with user groups in Remediate](#) on page 36

Puppet Remediate provides a limited number of roles that allow you to control what users can and can't do.

Related information

[Add new users](#) on page 29

As a user administration UI admin user, you can add new user accounts and assign them group privileges.

[Update a user](#) on page 30

From time to time you might need to change a user's password, or update the groups assigned to their account.

[Add sources](#) on page 51

Add your vulnerability scanner to detect and fix vulnerabilities across your infrastructure. To discover nodes, packages, and containers running on your entire infrastructure, add multiple infrastructure sources.

[Force log out](#) on page 31

Holders of the admin account can force other users to log out.

[LDAP configuration](#) on page 31

You can set up Remediate to use LDAP content to authenticate users.

[Active Directory configuration](#) on page 33

You can set up Remediate to use Active Directory to authenticate users.

Add new users

As a user administration UI admin user, you can add new user accounts and assign them group privileges.

Important: Until the first run wizard is completed, you must only log in as the default admin user.

To add a new user:

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. Select **Users** in the sidebar, and click **Add user**.

4. On the **Add user** page:
 - a) Enter a **Username**, and click **Save**.

Important: Do not enter anything in the **Required User Actions** or **Email Verified** fields as it may prevent the user from logging in.

- b) On the **User** page for the new account, click the **Credentials** tab.
- c) Enter and confirm a password in the relevant fields, and click **Reset password** when you are done.

Note: The password that you enter here cannot be the same as the username, must be at least 6 characters in length, and contain:

- At least 1 upper case character.
- At least 1 lower case character.
- At least 1 special character.

Note also that when changing the password, the new password must be different from previous 4 passwords employed by the user.

- d) On the **Groups** tab, select the groups you want to add to the new account, and click **Join**.

Update a user

From time to time you might need to change a user's password, or update the groups assigned to their account.

Important: Until the first run wizard is completed, you must only log in as the default admin user.

To update a user account:

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. Select **Users** in the sidebar, and click **Edit** on the row assigned to the user account you want to edit.
4. Make changes on the relevant tabs in the **User** page for the account.

Important: Do not enter anything in the **Required User Actions** or **Email Verified** fields as it may prevent the user from logging in.

Note: The password that you enter here cannot be the same as the username, must be at least 6 characters in length, and contain:

- At least 1 upper case character.
- At least 1 lower case character.
- At least 1 special character.

Note also that when changing the password, the new password must be different from previous 4 passwords employed by the user.

Delete a user

Remediate administrators can delete other user accounts, including admin accounts.

To delete a user:

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. Select **Users** in the sidebar, and click **Delete** on the row assigned to the user account you want to remove.
4. Click **Delete** on the pop-up dialog that appears to confirm.
The user account is no longer displayed on the **Users** page.

Force log out

Holders of the admin account can force other users to log out.

To log a user out:

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. Select **Users** in the sidebar, and click **Edit** on the row assigned to the user account you want to log out.
4. On the **Sessions** tab, click **Logout**.

LDAP configuration

You can set up Remediate to use LDAP content to authenticate users.


To configure Remediate to work with your LDAP server:

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. On the sidebar, click **User Federation** and select *ldap* from the drop-down list.

4. In the **Required Settings** area:

- a) Ensure **Import Users** is set to *ON*. Failure to do so can cause issues when you try to sync users.
- b) Select the LDAP server type from the **vendor** drop-down list.

Note: Some of the required parameters are automatically filled out depending on the vendor option you selected. These parameters can be changed if they are unsuitable.

Hover over the  icon beside each field for more information about the required content for that field.

- c) Enter the **Username LDAP Attribute**.
For many LDAP vendors this is `uid`, but may be different in your case.
- d) Enter the **RDN LDAP Attribute**.
This is the attribute that defines the relative distinguished name of the typical user distinguished name. Usually, it is the same as the username attribute.
- e) Enter the **UUID LDAP attribute**.
For many LDAP vendors this is `entryUUID`, but may be different in your case.
- f) Enter values for all LDAP **User Object Classes**. Values must be comma-separated, for example:

```
InetOrgPerson,organizationPerson
```

- g) Enter the **Connection URL** of your LDAP server.
You can use the **Test connection** button to ensure that Remediate can connect to the server.
- h) Enter the **User DN**.
This is the full distinguished name of the LDAP tree where your users are located. For example:

```
ou=myusers,dc=myhost,dc=com
```

- i) Select the **Authentication Type**. There are two options:
 - *None* - anonymous LDAP authentication.
 - *Simple* - Bind credential + Bind password authentication.
- j) Enter the **Bind DN** field. This is the distinguished name of the LDAP admin account used to access the LDAP server. For example:

```
uid=admin,dc=myhost,dc=com
```

- k) Enter the LDAP admin password in the **Bind Credential** field.
You can use the **Test authentication** button to see if Remediate can access the LDAP server.
- #### 5. To configure the synchronization schedule between your LDAP or Active Directory server and Remediate:
- a) Enter the maximum number of users to be imported in any transaction in the **Batch Size** field. The default is *1000*.
 - b) If you want to set up a full synchronization, set **Periodic Full Sync** to *ON*, and enter a value in seconds in the **Full Sync Period** field. The default is *604800* seconds (every 7 days).
 - c) If you want to synchronize only new and changed users, set **Periodic Changed Users Sync** to *ON* and enter a value in seconds in the **Changed Users Sync Period** field. The default is *86400* seconds (every 24 hours).

6. When you are done, click **Save**.

Related information

[LDAP mappers](#) on page 35

LDAP mappers are listeners, which are triggered by the LDAP Provider at various points, and provide another extension point to LDAP integration.

Active Directory configuration

You can set up Remediate to use Active Directory to authenticate users.


To configure Remediate to use your Active Directory server:

1. In the main Remediate UI, click **User admin** on the navigation sidebar. The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. On the sidebar, click **User Federation** and select *ldap* from the drop-down list.

4. In the **Required Settings** area:

- a) Select *Active Directory* from the **vendor** drop-down list.

Note: Some of the required parameters are automatically filled. These parameters can be changed if they are unsuitable.

Hover over the  icon beside each field for more information about the required content for that field.

- b) Enter the **Username LDAP Attribute**.

- c) Enter the **RDN LDAP Attribute**.

This is the attribute that defines the relative distinguished name of the typical user distinguished name.

Usually, it is the same as the username attribute. However, you might also use `cn` for this attribute if you used `sAMAccountName` for the username attribute.

- d) Enter the **UUID LDAP attribute**.

For Active Directory, this is usually `objectGUID`.

- e) Enter values for all LDAP **User Object Classes**. Values must be comma-separated, for example:

```
InetOrgPerson,organizationPerson
```

- f) Enter the **Connection URL** of your LDAP server.

For example:

```
ldap://ds.example.com:389
```

Or:

```
ldaps://ds.example.com:636
```

You can use the **Test connection** button to ensure that Remediate can connect to the server.

- g) Enter the **User DN**.

This is the full distinguished name of the LDAP tree where your users are located. For example:

```
ou=Users,dc=remediate,dc=io
```

- h) Select the **Authentication Type**. There are two options:

- *None* - anonymous LDAP authentication.
- *Simple* - Bind credential + Bind password authentication.

- i) Enter the **Bind DN** field. This is the distinguished name of the LDAP admin account used to access the LDAP server. For example:

```
cn=admin,dc=remediate,dc=io
```

- j) Enter the LDAP admin password in the **Bind Credential** field.

You can use the **Test authentication** button to see if Remediate can access the LDAP server.

5. To configure the synchronization schedule between your LDAP or Active Directory server and Remediate:

- a) Enter the maximum number of users to be imported in any transaction in the **Batch Size** field. The default is *1000*.

- b) If you want to set up a full synchronization, set **Periodic Full Sync** to *ON*, and enter a value in seconds in the **Full Sync Period** field. The default is *604800* seconds (every 7 days).

- c) If you want to synchronize only new and changed users, set **Periodic Changed Users Sync** to *ON* and enter a value in seconds in the **Changed Users Sync Period** field. The default is *86400* seconds (every 24 hours).

6. When you are done, click **Save**.

Related information

[LDAP mappers](#) on page 35

LDAP mappers are listeners, which are triggered by the LDAP Provider at various points, and provide another extension point to LDAP integration.

LDAP mappers

LDAP mappers are listeners, which are triggered by the LDAP Provider at various points, and provide another extension point to LDAP integration.

LDAP mappers are triggered when a user logs in via LDAP and needs to be imported, or when a user is queried from the user admin interface. When you create an LDAP provider, a default set of built-in mappers for this provider is made available. You are free to change this set and create a new mapper, or update/delete existing ones.

Mapper Type	Description
User Attribute Mapper	This mapper allows you to specify which LDAP attribute is mapped to which user attribute. So, for example, you can configure that LDAP attribute <code>mail</code> to the attribute <code>email</code> in the Remediate user database. For this mapper implementation, there is always a one-to-one mapping (one LDAP attribute is mapped to one Remediate user attribute)
FullName Mapper	This mapper allows you to specify that the full name of the user, which is saved in some LDAP attribute (usually <code>cn</code>), is to be mapped to <code>firstName</code> and <code>lastName</code> attributes in the Remediate user database. Using <code>cn</code> to contain the full name of a user is a common case for some LDAP deployments.
Role Mapper	This mapper is not applicable for the current release.
Hardcoded Role Mapper	This mapper grants a specified role to each user linked with LDAP.
Group Mapper	This mapper allows you to configure group mappings from LDAP into Remediate user group mappings. A group mapper can be used to map LDAP groups from a particular branch of an LDAP tree into groups in Remediate. It also propagates user-group mappings from LDAP into user-group mappings in Remediate.
MSAD User Account Mapper	This mapper is specific to Microsoft Active Directory (MSAD). It can tightly integrate the MSAD user account state into the Remediate user account state (account enabled, password is expired etc) using the <code>userAccountControl</code> and <code>pwdLastSet</code> LDAP attributes (both of which are specific to MSAD and are not LDAP standard).
Certificate Mapper	This mapper is used specifically for mapping X.509 certificates. It is generally used in conjunction with X.509 authentication and <i>Full certificate in PEM</i> format as an identity source. It behaves the same way as the User Attribute Mapper, but allows you to filter for an LDAP attribute which stores a certificate in either PEM or DER format. It is generally advised to enable Always Read Value From LDAP with this mapper.

Adding LDAP mappers

By default, there are User Attribute mappers that map basic Remediate user attributes like `username`, `firstname`, `lastname`, and `email` to corresponding LDAP attributes. You are free to extend these and provide additional attribute mappings.

To add an LDAP mapper:

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.

2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. On the sidebar, click **User Federation**, and then click the row assigned to the LDAP/AD connection you want to update.
4. Click the **Mappers** tab, and then **Create**.
5. Give the mapper a **Name**, and select the **Mapper Type** from the drop-down list.
6. Enter the required information in the relevant fields for your mapper type.

Tip: Hover over the  icon beside each field for more information about the required content for that field.

7. Click **Save** when you are done.

Tip: To edit your mapper settings, click on the row assigned to it on the **Mappers** tab.

Work with user groups in Remediate

Puppet Remediate provides a limited number of roles that allow you to control what users can and can't do.

The following permission groups are available to add to accounts:

Permission Group	Description
accept-risk	Accept risk for a given vulnerability on selected nodes.
add-credential	Add access credentials for a node.
add-source	Add a vulnerability scanner or infrastructure source.
add-task	Add a new remediation task.
admin	Complete first-run configuration. Members of this group also have all the permissions of the admin superuser role except the ability to login into the user administration UI and manage users.
remove-credential	Remove access credentials for a node.
remove-source	Remove a vulnerability scanner or infrastructure source.
remove-task	Remove a new remediation task.
run-task	Run a remediation task.

When using LDAP, you may want to grant a certain user group full permissions but restrict other user groups to more limited permissions.

This following example describes a sample scenario. It shows how to grant everyone in the LDAP group called *secops* full permissions, while letting all other users just run tasks.

To achieve this, we'll configure 2 LDAP providers with default mappers, and assign a different priority to each provider.

1. In the main Remediate UI, click **User admin** on the navigation sidebar.
The user admin login page is displayed.
2. Log in with the user administration UI administrator username and password. For more information on setting the user administration UI admin user password, see [Configuring Remediate](#) on page 43.
3. On the sidebar, click **User Federation** and select *ldap* from the drop-down list.

4. Create an LDAP Provider for the *secops* user group:
 - a) Set the **Priority** to 0. This is the highest priority.
 - b) Specify a **Custom User LDAP Filter** that only includes members of *secops*. As in the following example:

Provider ID	<input type="text" value="e91fdd55-6365-4868-bc0a-10727af274fe"/>
Enabled ?	<input checked="" type="checkbox"/> ON
Console Display Name ?	<input type="text" value="ldap"/>
Priority ?	<input type="text" value="0"/>
Import Users ?	<input checked="" type="checkbox"/> ON
Edit Mode ?	<input type="text"/>
Sync Registrations ?	<input type="checkbox"/> OFF
* Vendor ?	<input type="text" value="Other"/>
* Username LDAP attribute ?	<input type="text" value="uid"/>
* RDN LDAP attribute ?	<input type="text" value="uid"/>
* UUID LDAP attribute ?	<input type="text" value="entryUUID"/>
* User Object Classes ?	<input type="text" value="inetOrgPerson, organizationalPerson"/>
* Connection URL ?	<input type="text" value="ldaps://dap.puppetlabs.com:636"/>
* Users DN ?	<input type="text" value="ou=users,dc=puppetlabs,dc=com"/>
* Bind Type ?	<input type="text" value="simple"/>
* Bind DN ?	<input type="text" value="cn=foobar,ou=service,ou=users,dc=puppetlabs,dc=com"/>
* Bind Credential ?	<input type="text" value="*****"/>
Custom User LDAP Filter ?	<input type="text" value="(memberOf=cn=secops,ou=groups,dc=puppetlabs,dc=com)"/>
Search Scope ?	<input type="text" value="One Level"/>

For a full step-by-step guide to creating an LDAP provider, see [LDAP configuration](#) on page 31.

5. Click **Save** when you are done.

- Next, create default mappers for each default permissions group you want to assign to the *secops* user group. This example shows a mapper for the *add-source* permission: group:

Add user federation mapper

Name *	<input type="text" value="add-source"/>
Mapper Type	<input type="text" value="hardcoded-ldap-group-mapper"/>
Group	<input type="text" value="add-source"/>

For a full step-by-step guide to creating a default mapper, see [LDAP mappers](#) on page 35.

7. Create an LDAP Provider for all other users:

- a) Set the **Priority** to 10. This number must be lower than 0 so that users are evaluated initially against the first LDAP provider, and then only this provider if they are not members of the *secops* LDAP group.

Required Settings

Provider ID	057db858-6188-4c8e-9d33-828394366a23
Enabled	<input checked="" type="checkbox"/> ON
Console Display Name	ldap
Priority	10
Import Users	<input checked="" type="checkbox"/> ON
Edit Mode	
Sync Registrations	<input type="checkbox"/> OFF
* Vendor	Other
* Username LDAP attribute	uid
* RDN LDAP attribute	uid
* UUID LDAP attribute	entryUUID
* User Object Classes	inetOrgPerson, organizationalPerson
* Connection URL	ldaps://ldap.puppetlabs.com:636
* Users DN	ou=users,dc=puppetlabs,dc=com
* Bind Type	simple
* Bind DN	cn=foobar ou=service,ou=users,dc=puppetlabs,dc=com
* Bind Credential	*****
Custom User LDAP Filter	LDAP Filter
Search Scope	One Level

For a full step-by-step guide to creating an LDAP provider, see [LDAP configuration](#) on page 31.

- b) Add default mapper for this second group to let all users run tasks by default. As in the following example:

Run-task

ID	0ad5dd55-c6d5-422c-8e9e-ddca2551a218
Name *	run-task
Mapper Type	hardcoded-ldap-group-mapper
Group	run-task

For a full step-by-step guide to creating a default mapper, see [LDAP mappers](#) on page 35.

Related information

[LDAP configuration](#) on page 31

You can set up Remediate to use LDAP content to authenticate users.

[LDAP mappers](#) on page 35

LDAP mappers are listeners, which are triggered by the LDAP Provider at various points, and provide another extension point to LDAP integration.

Remediate Vulnerability Dashboard

The Remediate **Vulnerability Dashboard** provides you with a wealth of information on the health of your system.

The Remediate dashboard contains a wide variety of indicators about what is happening on your system. It is composed of three main components:

- The **Metrics bar**
- **Vulnerabilities overview** tab
- **Nodes overview** tab
- [Dashboard Metrics bar](#) on page 40

The **Metrics bar** at the top of the **Vulnerabilities** dashboard provides a breakdown of key metrics regarding affected nodes on your network.

- [Dashboard Vulnerability Overview tab](#) on page 41

The **Vulnerability Overview** tab on the Puppet Remediate **Vulnerabilities dashboard** displays information on the number, criticality and extent to which the nodes on your network are vulnerable.

- [Dashboard Node Overview tab](#) on page 42

The **Nodes overview** tab on the **Vulnerabilities Dashboard** displays information on the nodes affected by vulnerabilities on your network.

Dashboard Metrics bar

The **Metrics bar** at the top of the **Vulnerabilities** dashboard provides a breakdown of key metrics regarding affected nodes on your network.

The **Metrics bar** at the top of the **Vulnerabilities** dashboard provides a breakdown of key metrics regarding affected nodes, including:

- **Nodes**: - The total number of nodes currently discovered by Puppet Remediate. Click on this area to go to the **Nodes** page.

Note: The result shown in this area may differ from that displayed in the **Vulnerable Nodes** area if your scanner duplicates nodes owing to IP address reuse. The **Nodes** area ignores duplicates in this instance.

- **Vulnerable Nodes**: The total number of discovered nodes currently assessed as vulnerable. Clicking on this area takes you to the **Most vulnerable nodes** page.
- The total number of vulnerabilities across all discovered nodes. Click this area to visit the **Total Vulnerabilities** page.

Note: The result shown in this area may differ from that displayed in the **Vulnerable Nodes** area if your scanner duplicates nodes owing to IP address reuse. The **Nodes** area ignores duplicates in this instance.

- **Highest Risk Score**: The risk score of the most severe vulnerabilities according to your vulnerability scanner. Click this area to go to the **Vulnerability detail** page for this vulnerability, where you'll see what nodes are affected, and from where you can run remediation tasks.

- **Most Vulnerable Node**The number of vulnerabilities on the node with the most vulnerabilities. Clicking this area takes you to the **Node detail** page for your most vulnerable node. Again, from this page you can initiate remediate tasks to clean vulnerabilities from the node.
- **Accepted Risks:** The total number of vulnerabilities that have been designated as an accepted risk on selected nodes. Click on this panel to visit the **Accepted Risks** page to view a list of the vulnerabilities where reporting has been paused.
- The **Metrics bar** also contains a search box you can use to find out more detail on a vulnerability if you have its CVE number. Searching here take you to the **Vulnerability detail** page where you can see nodes affected by the selected vulnerability, and also run tasks to remediate it.

A full overview of the information provided by the **Vulnerabilities** dashboard **Metrics bar** is contained in the following video.

Related information

[Accepting vulnerability risk](#) on page 67

You may not want or need to deal with every vulnerability on every node in your network. You can also tell Puppet Remediate not to report on selected vulnerabilities on individual hosts.

Dashboard Vulnerability Overview tab

The **Vulnerability Overview** tab on the Puppet Remediate **Vulnerabilities dashboard** displays information on the number, criticality and extent to which the nodes on your network are vulnerable.

The **Key Statistics** area gives you an idea of the overall health of your network. Here you'll see the total number of vulnerabilities affecting nodes within your network, as well as the total number of vulnerabilities designated as medium risk, high risk or critical.

Note: The result shown in this area may differ from that displayed in the **Vulnerable Nodes** area if your scanner duplicates nodes owing to IP address reuse. The **Nodes** area ignores duplicates in this instance.

The **Criticality Breakdown** chart displays the percentage of vulnerabilities in each criticality division. Mouse over each section in the chart to see more information on the number of vulnerabilities in that category affecting nodes in your network. Click on a section to view all vulnerabilities in this criticality division in the **Vulnerabilities** table.

The **Top 5 common vulnerabilities** chart displays the number of nodes compromised by the most widespread vulnerabilities on your network. Mouse over each bar for the name of the vulnerability in question. Click a bar to see more information on the selected vulnerability in the **Vulnerabilities table**.

The **Vulnerabilities table** itself provides top-level information on individual vulnerabilities:

- Click the export icon to export the **Vulnerabilities table** data in CSV format.
- Filter the content by vulnerability severity by using filter option menu and create filter button at the top of the table. If you need to filter by vulnerability name or analysis content, click **Create Filter** and select the appropriate column, choose the required operator, and add the value you want to search for.
- Sort by clicking on the appropriate column header. You can also choose which columns are displayed and which are hidden from the **Columns** drop-down menu.

Each row of the **Vulnerabilities table** provides:

- The name of the vulnerability.
- The risk score assigned by the vulnerability scanner. This is a score out of 1000.

As each vulnerability scanner provider uses a different scoring system, Remediate regularizes the score using the formulae outlined in the table below:

Scanner provider	Risk score regularization formula
Qualys	Severity (1-5) * CVSS Modifier field (1-10) * 20
Rapid7	The Rapid7 is not regularized and is presented as is.

Scanner provider	Risk score regularization formula
Tenable.io	Severity(1-5) * CVSS3 base score (if present, else CVSS base score) (1-10) * 20
Tenable.sc	Severity(1-5) * CVSS3 base score (1-10) * 20

- An analysis of the vulnerability provided by your vulnerability scanner. This is a description of the vulnerability threat, and the possible consequences that can occur if the vulnerability is successfully exploited.
- The number of nodes affected by the particular vulnerability.
- This table also shows the Puppet risk score. This is the vulnerability scanner risk score multiplied by the number of nodes affected by the vulnerability. The Puppet risk score lets you see straight away which vulnerabilities need to be addressed first.

Click the name of the vulnerability to go to the **Vulnerability detail** page. Here, you'll find fuller information on the vulnerability, the nodes it affects, and advice about how to the threat. You can also launch remediation tasks from this page.

Dashboard Node Overview tab

The **Nodes overview** tab on the **Vulnerabilities Dashboard** displays information on the nodes affected by vulnerabilities on your network.

The **Key Statistics** area on this tab, provides information on:

- The number of accessible nodes with and without vulnerabilities.
- The total number of inaccessible nodes. In other words, the total number of nodes that Puppet Remediate does not have access credentials for.
- The number of nodes that have vulnerabilities that are deemed to be critical.

The **Vulnerable Nodes by OS** chart shows the nodes on your network affected by vulnerabilities listed by the top 5 operating systems affected. Click on a bar in this diagram to list all nodes using the selected operating system in the **Nodes table** below.

The **Nodes with vulnerabilities** diagram breaks down the percentage of nodes on your network that are affected by vulnerabilities by accessibility. Mouse over each section in the ring to see more information on the number of nodes with vulnerabilities in each accessibility type. Click on a section to list nodes by accessibility type in the **Nodes table** below.

Each row of the **Nodes table** provides:

- The resource name of the node on the network.
- The number of vulnerabilities affecting the node.
- IP address.
- Operating system name and version.
- The version of Puppet running on the node.
- The infrastructure source type.
- The node's uptime.
- The date and time a scan was last initiated on the node.

Click the export icon to export the **Nodes table** data in CSV format. Filter by node accessibility by using the filter option menu. If you need to filter by column, click **Create Filter** and select the appropriate column, choose the required operator, and add the value you want to search for.

Sort by clicking on the appropriate column header. You can also choose which columns are displayed and which are hidden from the **Columns** drop-down menu.

Clicking the **resource name** entry for a node takes you to the **Attribute** tab on that node's **Node details** page with more information on the node. Clicking the number of vulnerabilities for the node, brings you to the **Vulnerabilities**

tab on the **Node details** page. From here, you can run tasks to remediate the vulnerabilities affecting the selected node.

Configuring Remediate

After installing Puppet Remediate, configure and administer the application by using the command line interface (CLI).

Command line options

The CLI provides additional administrative and troubleshooting options beyond the user interface.

```
docker-compose run remediate [command]
```

Command	Description
config	<p>View and modify the configuration.</p> <p>Usage:</p> <pre>docker-compose run remediate config [command]</pre> <p>The following commands are used with the <code>config</code> command:</p> <ul style="list-style-type: none"> • <code>get-compose-file</code> • <code>set-override</code> • <code>unset-override</code> • <code>view</code>
config get-compose-file	<p>Gets the docker compose file.</p> <p>Usage:</p> <pre>docker-compose run remediate config get-compose-file [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>config get-compose-file</code> command:</p> <ul style="list-style-type: none"> • <code>--filename <i>string</i></code> - The filename to output the docker compose file to. (default: <code>remediate-docker-compose.yml</code>).

Command	Description
config set-override	<p>Set configuration override for Remediate (requires restart).</p> <p>Usage:</p> <pre>docker-compose run remediate config set-override [key] [value] [flags]</pre> <p>See Configuration overrides below for more information about how to use this command.</p>
config unset-override	<p>Unset configuration override for Puppet Remediate (requires restart).</p> <p>Usage:</p> <pre>docker-compose run remediate config unset-override [key]</pre> <p>See Configuration overrides below for more information about how to use this command.</p>
config view	<p>View configuration for Remediate.</p> <p>Usage:</p> <pre>docker-compose run remediate config view [flags]</pre>
export	<p>Export the Remediate images to a <i>tar</i> file.</p> <p>Usage:</p> <pre>docker-compose run remediate export [flags]</pre>
generate-docs	<p>Generates the CLI documentation.</p> <p>Markdown files are created at the specified path detailing the CLI commands.</p> <p>Usage:</p> <pre>docker-compose run remediate generate-docs [path] [flags]</pre>
help	<p>Displays the help commands.</p> <p>Usage:</p> <pre>docker-compose run remediate help [command] [flags]</pre>

Command	Description
import	<p>Import the Remediate images from a <i>tar</i> file.</p> <p>Usage:</p> <pre data-bbox="634 306 1455 363">docker-compose run remediate import [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>import</code> command:</p> <ul data-bbox="634 457 1455 514" style="list-style-type: none"> • <code>--config <i>string</i></code> The config file (default is <code>\$HOME/.installer.yaml</code>) (default: <code>.installer.yaml</code>)
logs	<p>Displays the application service logs.</p> <p>Usage:</p> <pre data-bbox="634 646 1455 703">docker-compose run remediate logs [flags]</pre> <p>In addition to the available global flags, you can use the following flags with the <code>logs</code> command:</p> <ul data-bbox="634 798 1455 896" style="list-style-type: none"> • <code>--service <i>string</i></code> - Display logs for only this service. • <code>--since <i>string</i></code> - Show logs since timestamp (e.g. <code>2013-01-02T13:23:37</code>) or relative (e.g. <code>42m</code> for 42 minutes) (default <code>"4h"</code>). <p>Note: By default the <code>logs</code> command returns logs for the last 8 hours only. Use the <code>--since</code> flag to return logs for a time period different from the default.</p>
mayday	<p>Creates a mayday tarball containing debug information (including the audit log) for troubleshooting with the Puppet support team.</p> <p>Usage:</p> <pre data-bbox="634 1192 1455 1249">docker-compose run remediate mayday [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>mayday</code> command:</p> <ul data-bbox="634 1344 1455 1442" style="list-style-type: none"> • <code>--since <i>string</i></code> - Show logs since timestamp (e.g. <code>2013-01-02T13:23:37</code>). You can also use an integer with <code>m</code> or <code>h</code>. For example. <code>42m</code> for 42 minutes, or <code>1h</code> for 1 hour. The default is <code>"4h"</code>.

Command	Description
reset	<p>Deletes sources, credentials, user passwords, and resets the application to an uninstalled state.</p> <p>Usage:</p> <pre data-bbox="634 338 1458 394">docker-compose run remediate reset [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>reset</code> command:</p> <ul data-bbox="634 491 1036 520" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen.
restart	<p>Restarts the application.</p> <p>Usage:</p> <pre data-bbox="634 653 1458 709">docker-compose run remediate restart [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>restart</code> command:</p> <ul data-bbox="634 806 1036 835" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen.
refresh-internal-certs	<p>Refreshes internal docker certificates. Remediate uses a set of internal Docker certificates that are valid for one year only. On offline instances, expired certificates are not updated automatically on upgrade. Use this command refresh any expired certificates. This command does not refresh any certificates you have set using the <code>set-frontdoor-cert</code> or <code>start</code> commands.</p> <p>Usage:</p> <pre data-bbox="634 1115 1458 1192">docker-compose run remediate refresh-internal-certs [flags]</pre> <p>In addition to the available global flags, you can use the following flags with the <code>refresh-internal-certs</code> command:</p> <ul data-bbox="634 1289 1110 1358" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen. • <code>-o, --offline</code> - start without internet access

Command	Description
set-frontdoor-cert	<p>Restarts Remediate using the specified certificate and key.</p> <p>Usage:</p> <pre data-bbox="633 304 1453 388">set-frontdoor-cert --certfile [string] --keyfile [string] [flags]</pre> <p>In addition to the available global flags, you can use the following flags with the <code>set-frontdoor-cert</code> command:</p> <ul data-bbox="633 483 1453 556" style="list-style-type: none"> • <code>--cert-file</code> - TLS certificate to be used by the front door. • <code>--key-file</code> - TLS certificate private key to be used by the front door. <p>Note: The <code>--cert-file</code> and <code>--key-file</code> flags must be used together.</p> <ul data-bbox="633 661 1453 735" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen. • <code>-o, --offline</code> - start without internet access
set-license	<p>Sets a new license.</p> <p>Usage:</p> <pre data-bbox="633 861 1453 934">docker-compose run remediate set-license [license-file] [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>set-license</code> command:</p> <ul data-bbox="633 1029 1453 1060" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen.
set-password	<p>Sets a new user administration login password that allows you to log in to the user administration UI. This operation may take several minutes.</p> <p>Important: This command sets the password for the user administration UI admin user only. The main Remediate UI admin user password is set via the user administration UI. For more information on setting the Remediate UI admin user password, see Add new users on page 29.</p>

Command	Description
start	<p>Initializes and starts the application.</p> <p>Usage:</p> <pre data-bbox="634 306 1463 363">docker-compose run remediate start [flags]</pre> <p>In addition to the available global flags, you can use the following flags with the <code>start</code> command:</p> <ul data-bbox="634 459 1463 527" style="list-style-type: none"> • <code>--cert-file</code> - TLS certificate to be used by the Remediate UI. • <code>--key-file</code> - TLS certificate private key to be used by the Remediate UI. <p>Note: The <code>--cert-file</code> and <code>--key-file</code> flags must be used together.</p> <ul data-bbox="634 606 1463 709" style="list-style-type: none"> • <code>--license-file <i>string</i></code> - License file (default is <code>\$PWD/license.json</code>) • <code>--quietly</code> - Hides the splash screen. • <code>-o, --offline</code> - start without internet access
stop	<p>Stops the application.</p> <p>Usage:</p> <pre data-bbox="634 831 1463 888">docker-compose run remediate stop [flags]</pre> <p>In addition to the available global flags, you can use the following flag with the <code>stop</code> command:</p> <ul data-bbox="634 984 1463 1026" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen.

Command	Description
unset-frontdoor-cert	<p>Unsets any TLS certificate set by the <code>set-frontdoor-cert</code> or <code>start</code> commands and reverts to internally generated, self signed remediate certificate.</p> <p>Usage</p> <pre data-bbox="634 369 1463 457">docker-compose run remediate unset-frontdoor-cert [flags]</pre> <p>In addition to the available global flags, you can use the following flags with the <code>unset-frontdoor-cert</code> command:</p> <ul data-bbox="634 552 1112 617" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen. • <code>-o, --offline</code> - start without internet access
update	<p>Updates to the latest version.</p> <p>Usage:</p> <pre data-bbox="634 743 1463 793">docker-compose run remediate update [flags]</pre> <p>In addition to the available global flags, you can use the following flags with the <code>update</code> command:</p> <ul data-bbox="634 894 1263 959" style="list-style-type: none"> • <code>--quietly</code> - Hides the splash screen. • <code>--version <i>string</i></code> - Version to update to (default "latest"). <p>The <code>update</code> command can also be used with the <code>list</code> sub-command or with its own local flags.</p>
update list	<p>Shows a list of available versions.</p> <p>Usage:</p> <pre data-bbox="634 1163 1463 1213">docker-compose run remediate update list [flags]</pre>

Command	Description
vr-cert	<p>Manage SSL certificates for VR containers:</p> <ul style="list-style-type: none"> • add - Adds a CA certificate to VR service. The path must be relative and must be a sub-folder of the current folder. • list - Lists CA certificates for VR service. • remove - Removes a CA certificate from VR service. <p>Usage:</p> <pre>docker-compose run remediate vr-cert add <CA cert path></pre> <pre>docker-compose run remediate vr-cert list</pre> <pre>docker-compose run remediate vr-cert remove <CA cert></pre>

Global flags

In addition to the flags that can be used with individual commands, the following flags can be used with all commands:

Flag	Description
--config <i>string</i>	Config file (default is \$HOME/.installer.yaml) (default ".installer.yml")
-h, --help	<p>Help for remediate</p> <p>Use <code>remediate [command] --help</code> to get more information about a command.</p>

Configuration overrides

After installing Remediate, optimize the application for your environment by configuring and tuning settings as needed.

To view the current configuration:

```
docker-compose run remediate config view
```

To override a configuration key:

```
docker-compose run remediate config set-override [key] [value]
```

To reset a configuration key:

```
docker-compose run remediate config unset-override [key]
```

Key	Description	Default value
controller.discoveryinterval	The interval between each discovery run.	every 4h
controller.loglevel	The logging levels for the controller container.	INFO (default) DEBUG WARN ERROR FATAL PANIC
edge.discoworkers	The maximum number of discovery processes that run in parallel.	512
edge.taskworkers	The maximum number of tasks that run in parallel.	32
edge.workertimeout	The time out value, in minutes, for each job.	25
edge.loglevel	The logging levels for the edge container.	WARN (default) INFO DEBUG ERROR FATAL PANIC

Add sources

Add your vulnerability scanner to detect and fix vulnerabilities across your infrastructure. To discover nodes, packages, and containers running on your entire infrastructure, add multiple infrastructure sources.

1. On the sidebar, click **Manage sources**.

2. Click **Add sources**, and then select a source:

- Vulnerability scanner:
 - Qualys
 - Rapid7
 - Tenable.io
 - Tenable.sc
- Infrastructure source:
 - Amazon Web Services
 - Google Cloud Platform
 - Microsoft Azure
 - OpenStack
 - VMware vSphere
 - IP addresses
- Puppet Enterprise (PE) source

3. Enter relevant configuration parameter values for the source (see below for details).

Tip: If you are adding a vulnerability scanner, click **Test Credentials** to check that Remediate has the correct access credentials for the scanner before you kick off a discovery.

4. Click **Discover**.

Note: The quantity of data that is contained in your source affects the amount of time it takes for information to be displayed in Remediate. Depending on the scanner you use, it might take as much as 24 hours or more for content to be displayed if your source uses a very large data set.

- [Vulnerability scanners](#) on page 52

Puppet Remediate integrates with Tenable, Qualys and Rapid7.

- [Infrastructure sources](#) on page 55

Discover node instances on the following infrastructure sources.

- [Puppet Enterprise sources](#) on page 57

You can install a single Puppet Enterprise source to gather information on and run vulnerability remediation tasks against.

Related information

[Run tasks](#) on page 70

Run ad hoc tasks on target nodes to upgrade packages, restart services, execute shell commands, or perform any other type of single-action executions on your nodes.

[Discovering and managing resources](#) on page 76

The **Resources** dashboard provides a high-level summary view of your infrastructure, and consists of a number of dashboard cards to drill down from for detailed information about nodes, packages, and containers.

Vulnerability scanners

Puppet Remediate integrates with Tenable, Qualys and Rapid7.

Note: Ask your security team for the permissions to import vulnerability scan data.

Qualys

Add the details for your Qualys Vulnerability Manager account.

Parameter	Description
Name	A unique and descriptive name to identify this vulnerability scanner.
API server URL	The HTTPS URL and port number to the platform where your Qualys account is located. Note: Qualys CE is not API compatible and therefore is not supported by Remediate. For more information, see the Qualys CE user guide .
Username	Your Qualys username to authenticate with.
Password	Your Qualys password to authenticate with.
Update interval	The time interval before Remediate polls the vulnerability scanner for new data. This parameter is optional. If not specified, the update interval defaults to 30 minutes.
Severity threshold	The severity level on or above which vulnerability data is passed to Remediate. This parameter is optional. If not specified, severity level 1 is used by default.
Date range	Use the options in this drop-down menu to limit the time period for which results are returned.
Import tags	Use this option to import tags from Qualys. By default this option is OFF.

Rapid7

Add the details for your Rapid7 Nexpose (on-prem) or InsightVM (cloud) account.

Parameter	Description
Name	A unique and descriptive name to identify this vulnerability scanner.
InsightVM URL	The HTTPS URL and port number to your InsightVM or Nexpose instance.
Username	Your Rapid7 username to authenticate with.
Password	Your Rapid7 password to authenticate with.
Enable SSL certification verification	To verify the signature on the SSL certificate returned by Rapid7 using your CA cert, select this option. Remember that you must add your own CA certificate. For more information, see SSL certificate verification for scanners on page 55.
Update interval	The time interval before Remediate polls the vulnerability scanner for new data. This parameter is optional. If not specified, the update interval defaults to 30 minutes.
Severity threshold	The severity level on or above which vulnerability data is passed to Remediate. This parameter is optional. If not specified, severity level 1 is used by default.

Tenable.io

Add the details for your Tenable.io (cloud) account.

Parameter	Description
Name	A unique and descriptive name to identify this vulnerability scanner.
Access key	Your Tenable.io access key to authenticate with the Tenable.io API. For more information about generating an access key, see the Tenable.io documentation .
Secret key	Your Tenable.io secret key to authenticate with the Tenable.io API. For more information about generating a secret key, see the Tenable.io documentation .
Update interval	The time interval before Remediate polls the vulnerability scanner for new data. This parameter is optional. If not specified, the update interval defaults to 30 minutes.
Severity threshold	The severity level on or above which vulnerability data is passed to Remediate. This parameter is optional. If not specified, severity level 1 is used by default.
Import tags	Use this option to import tags from Tenable.io. By default this option is OFF.

Note: You must use the *Administrator* role in Tenable.io to export data using the Tenable.io API.

Tenable.sc

Add the details for your Tenable.sc account.

Parameter	Description
Name	A unique and descriptive name to identify this vulnerability scanner.
URL	The URL of your Tenable.sc instance.
Username	Your Tenable.sc account username. For more information, see the Tenable.sc documentation .
Password	Your Tenable.sc account password. For more information, see the Tenable.sc documentation .
Enable SSL certificate verification	Select this checkbox if you want to verify the SSL certificate returned by Tenable.sc. Remember that you must add your own CA certificate. For more information, see SSL certificate verification for scanners on page 55.
Refresh interval	The time interval before Remediate polls the vulnerability scanner for new data. This parameter is optional. If not specified, the update interval defaults to 30 minutes.
Severity threshold	The severity level on or above which vulnerability data is passed to Remediate. This parameter is optional. If not specified, severity level 1 is used by default.

Tip: The Tenable.sc *Auditor* role is the role with the least permissions that you can use to connect from Remediate.

SSL certificate verification for scanners

You can verify SSL certificates signed by an internal certificate authority for Rapid7 and Tenable.sc.

If you decide to enable verification for certificates signed by an internal certificate authority when configuring Rapid7 or Tenable.sc to work with Remediate, use the following procedure to add a self-signed certificate:

1. Copy your certificate to a sub-folder of your current working directory.
2. Issue the following command to tell Remediate to use the new certificate:

```
docker-compose run remediate vr-cert add <CA cert path>
```

Infrastructure sources

Discover node instances on the following infrastructure sources.

Amazon Web Services (AWS)

Add the AWS authentication credentials to discover the EC2 instances running on your AWS account.

Parameter	Description
Name	A unique and descriptive name to identify this source.
Access key	The access key ID that you generated in the AWS Management Console.
Secret key	The secret access key that corresponds to your access key ID.

Google Cloud Platform (GCP)

Add the GCP authentication credentials to discover the node instances running on each of your accounts. The client email, the private key, the private key ID, and the project ID values are located in the service account key file (.json) you saved after generating your GCP credentials in the GCP console.

Parameter	Description
Name	A unique and descriptive name to identify this source.
Client email	The service account email associated with your GCP account.
Private key	The private key you generated in the GCP console.
Private key ID	The private ID that corresponds to your private key.
Project ID	The GCP project that corresponds to your service account.

Microsoft Azure

Add the Microsoft Azure authentication credentials to discover the node instances on each of your Microsoft Azure accounts.

Make sure to log into the Azure portal and register your application. Name it `PuppetRemediate` and select the **Web app / API** application type. You must also assign the **Reader** permission to the application.

Parameter	Description
Name	A unique and descriptive name to identify this source.
Subscription ID	The subscription ID that identifies your Azure services subscription.
Tenant ID	The AAD tenant ID (also known as the directory ID).
Application ID	The Azure application ID (also known as the client ID).
Client ID	The client key (also known as the authentication key) generated for your application within the AAD.

OpenStack

Add the OpenStack authentication credentials to discover the node instances running on each of your OpenStack accounts.

Parameter	Description
Name	A unique and descriptive name to identify this source.
Endpoint	The authentication URL for the identity (Keystone) service.
Username	Your OpenStack username to authenticate with.
Password	Your OpenStack password to authenticate with.
Domain name	The authentication domain name used to connect to OpenStack.
Tenant ID	The tenant ID, also known as the project ID, used for OpenStack.

VMware vSphere

Add the VMware vSphere authentication credentials to discover the node instances running on each of your VMware vSphere accounts.

Parameter	Description
Name	A unique and descriptive name to identify this source.
vCenter server	The FQDN of the vCenter server.
vCenter username	The VMware vSphere username used to authenticate to the vCenter server.
vCenter password	The VMware vSphere password required to authenticate to the vCenter server.

Network nodes

Discover nodes by specifying an IP address range, a CIDR block, or by uploading a comma-separated IP address list.

Parameter	Description
Name	A unique and descriptive name to identify this source.

Parameter	Description
Type	<p>The IP address input type:</p> <ul style="list-style-type: none"> • IP range: Using the From and To fields, enter the IP address range. • CIDR: Enter the base IP address and the subnet mask to determine the network portion of the address. • CSV: Upload a comma-separated list of: <ul style="list-style-type: none"> • IP addresses. • Hyphen-separated IP ranges. • CIDR notations.

Puppet Enterprise sources

You can install a single Puppet Enterprise source to gather information on and run vulnerability remediation tasks against.

Add Puppet Enterprise (PE) credentials to authenticate your Puppet account and discover nodes running PE. Once added, your Puppet Enterprise nodes are visible on the **Resources** dashboard in the Remediate UI.

We do not recommend discovering PE nodes using other sources. Although Remediate deduplicates these nodes in most cases, you might see some duplications of PE nodes discovered using other sources. In particular, Remediate is not able to deduplicate PE nodes without agents in these instances.

Parameter	Description
Name	A unique and descriptive name to identify this source.
Puppet Enterprise URL	<p>The hostname or URL of your Puppet Enterprise server e.g. <i>puppet.domain.com</i>, or <i>https://puppet.domain.com</i>.</p> <p>Remediate uses the Puppet Enterprise Orchestrator API to run tasks on your Puppet Enterprise nodes.</p>
Puppet authentication token	<p>The Puppet authentication token. This is generated by using the <code>puppet-access</code> command in Puppet Enterprise.</p> <p>For more information on using the <code>puppet-access</code> command, see https://puppet.com/docs/pe/latest/rbac_token_auth_intro.html</p>

Adding node credentials

Add credentials to authenticate with nodes and fix vulnerabilities or gain insights into discovered resources. Each credential is encrypted and stored securely in the vault.

The two authentication methods are Secure Shell (SSH) authentication with Linux nodes on port 22, and Windows Remote Management (WinRM) authentication with Windows nodes on ports 5986 and 5985.



CAUTION: Using each node credential you provide, Remediate attempts to authenticate with each discovered node until a successful authentication is achieved. This process repeats every four hours, using previously successful credentials first. Depending on the configuration of your network management and security sensors, Remediate activities might trigger alerts or an active response.

- [Secure Shell \(SSH\)](#) on page 58

The two types of SSH authentication are username and password, using negotiated encryption, and private key files, using asymmetric encryption.

- [Windows Remote Management \(WinRM\)](#) on page 59

To authenticate with Windows nodes, Puppet Remediate uses NTLM authentication over HTTPS on port 5986. When enabled, Remediate falls back to using NTLM authentication over HTTP on port 5985, if the default authentication fails.

Secure Shell (SSH)

The two types of SSH authentication are username and password, using negotiated encryption, and private key files, using asymmetric encryption.

Tip: As a dual-factor authentication with nodes, it's recommended to use SSH private key files and to include a username and passphrase for each file. Using an SSH private key file, rather than an SSH username and password, is considered more secure against potential compromises on remote nodes because the password is not sent over the network.

When using SSH authentication to discover resources running on Linux nodes, there are a number of prerequisites:

- To install the Puppet agent on nodes, your SSH credentials must be for the root account.
- To discover containers on nodes, your SSH credentials must be for the root account or an account that is a member of the Docker group.

Add SSH private key files

Upload an SSH private key file to discover resources, and to run tasks on your Linux hosts.

1. On the sidebar, click **Manage credentials**.
2. Click **Add credentials** and then click **SSH private key file**.
3. Click **Browse**, select your files, and then click **Open**.
4. Click **Configure keys**.
5. In the **Name** field enter a unique and descriptive name.
6. Assign an individual scope, or both, to the credential:
 - **Discover resources on nodes:** This credential scope is valid only for discovering resources on your Linux nodes.
 - **Run tasks:** This credential is valid only for running tasks on your Linux nodes. When this individual scope is selected, no attempts are made to discover resources.
 - **Escalate privileges to root:** When required to run tasks on nodes, sudo escalate `non-root` account privileges to `root`. Privilege escalation occurs if the first attempt to run a task fails when using `non-root` privileges.
7. In the **Username** field, enter your SSH username.
8. In the **Passphrase** field, enter your SSH passphrase, or leave it blank if your key is not encrypted.
9. Click **Add keys**.

Add SSH username and password

Add an SSH username and password to discover resources, and to run tasks on your Linux hosts.

1. On the sidebar, click **Manage credentials**.
2. Click **SSH credential**.
3. In the **Name** field, enter a unique and descriptive name.
4. Assign an individual scope, or both, to the credential:
 - **Discover resources on nodes:** This credential scope is valid only for discovering resources on your Linux nodes.
 - **Remediate vulnerabilities:** This credential is valid only for running tasks on your Linux nodes. When this individual scope is selected, no attempts are made to discover resources.
 - **Escalate privileges to root:** When required to run tasks on nodes, sudo escalate non-root account privileges to root. Privilege escalation occurs if the first attempt to run a task fails when using non-root privileges
5. In the **Username** field, enter your SSH username.
6. In the **Password** field, enter your SSH password, and then click **Add credential**.

Windows Remote Management (WinRM)

To authenticate with Windows nodes, Puppet Remediate uses NTLM authentication over HTTPS on port 5986. When enabled, Remediate falls back to using NTLM authentication over HTTP on port 5985, if the default authentication fails.

To discover resources on your Windows hosts, you must enable [WinRM](#) access on each host by running the following commands:

```
winrm quickconfig
y
winrm set winrm/config/winrs '@{MaxMemoryPerShellMB="1024"}'
```



CAUTION:

To enable the HTTP fallback option (see step 6 below), include the `winrm set winrm/config/service '@{AllowUnencrypted="true"}'` command. This means that all Puppet Remediate commands and responses, not including credentials, are sent in plain text over the network.

Note:

Your Windows user account must be a member of either the local administrator group or the `WinRMRemoteWMIUsers_` group. For more information, see the [Default Group Access](#) section in the Microsoft Windows Remote Management documentation.

Add WinRM credentials

1. On the sidebar, click **Manage credentials**.
2. Click **WinRM credential**.
3. In the **Name** field, enter a unique and descriptive name.

4. Assign an individual scope, or both, to the credential:
 - **Discover resources on nodes:** This credential scope is valid only for discovering resources on your Windows nodes.
 - **Remediate vulnerabilities:** This credential is valid only for running tasks on your Windows nodes. When this individual scope is selected, no attempts are made to discover resources.
5. Select **HTTP fallback** to permit using authentication over HTTP, if the default authentication over HTTPS fails.
6. Click **Add credential**.

Tagging nodes

Tagging nodes allows you to group and run tasks on tagged nodes according to criteria that you define.

Tags are key/value pairs you can assign to nodes in order to identify and group them in different ways. You can then run tasks against nodes grouped by whatever criteria you set (e.g. operating system, physical location, environment etc.).

The **Tags** column on the **Nodes** page table contains information on the tags assigned to a particular node. Mouse over the tag icon for more information on the tags assigned to any node.

Note: Tags are only usable after a discovery run is completed.

Where can I assign tags?

Important: Tags can be assigned to one or more *accessible* nodes. If Remediate does not have the credentials to access a node, it cannot be tagged.

Tags can assigned in the following places:

- Any **Node Details** page.
- Any of the **Nodes** pages accessible from the **Resources** dashboard.
- The **Vulnerable Nodes** page accessible from the **Metrics bar** on the **Resources** dashboard.
- Any **Vulnerability Details** page.

Imported tags

You can also elect to import tags that are assigned to nodes by Qualys and Tenable.io by selecting the **Import Tags** option when adding them as a source. Imported tags keys have the `imported_` prefix. Imported tags cannot be edited or deleted via the Remediate UI.

- [Assign a tag to a node](#) on page 61

You can assign tags to a single nodes on the **Node Details** pages.

- [Assign a tag to multiple nodes](#) on page 61

You can add a tag to multiple nodes at once.

- [Add multiple tag values](#) on page 62

Adding multiple tag values to a tag key can help you group nodes more efficiently.

- [Edit tag values](#) on page 62

To change a tag key, you simply create a new tag, and delete the old one. To change an existing tag value you can edit the tag itself.

- [Remove tags from nodes](#) on page 63

You can remove tags from single or multiple nodes.

- [Filter nodes by tag](#) on page 63

You can filter nodes to see which nodes share the same tag.

Assign a tag to a node

You can assign tags to a single nodes on the **Node Details** pages.

Assigning a tag involves creating a tag key and, optionally, a tag value for that key. For example, if you want to group all nodes by in your production environment you would assign environment as the tag key and *production* as the tag value.

A node can have multiple tags. The tags assigned to a node are displayed as badges at the top of the node's **Node Details** page.

1. On a **Node Details** page, click **Assign tags**.
2. On the side panel that appears on the right of the screen:
 - a) Enter a name for the tag in the **Tag key** field.
 - b) Next, enter a value for the tag in the **Tag value** field.

This step is optional but it's a good idea to add a tag value at this point. A tag can have more than one value.

- c) Click **Confirm**.

The new tag badge appears at the top of the **Node Details** page.

Note: To remove a tag from a node, click the **x** on the tag badge at the top of the **Node Details** page.

Related information

[Assign a tag to multiple nodes](#) on page 61

You can add a tag to multiple nodes at once.

[Edit tag values](#) on page 62

To change a tag key, you simply create a new tag, and delete the old one. To change an existing tag value you can edit the tag itself.

[Remove tags from nodes](#) on page 63

You can remove tags from single or multiple nodes.

[Filter nodes by tag](#) on page 63

You can filter nodes to see which nodes share the same tag.

Assign a tag to multiple nodes

You can add a tag to multiple nodes at once.

Tags can be assigned to multiple nodes on the following pages in Remediate:

- Any of the **Nodes** pages accessible from the **Resources** dashboard.
- The **Vulnerable Nodes** page accessible from the **Metrics bar** on the **Resources** dashboard.
- Any **Vulnerability Details** page.

To sign a tag to multiple nodes:

1. On your selected page (see above), select the available node or nodes you want to tag.
2. Click **Assign tags**.
3. On the side panel that appears on the right of the screen:
 - a) Enter a name for the tag in the **Tag key** field.
 - b) Next, enter a value for the tag in the **Tag value** field.

This step is optional but it's a good idea to add a tag value at this point. A tag can have more than one value.

- c) Click **Close**.

The new tag badge appears at the top of the **Node** page.

Related information

[Assign a tag to a node](#) on page 61

You can assign tags to a single nodes on the **Node Details** pages.

[Edit tag values](#) on page 62

To change a tag key, you simply create a new tag, and delete the old one. To change an existing tag value you can edit the tag itself.

[Remove tags from nodes](#) on page 63

You can remove tags from single or multiple nodes.

[Filter nodes by tag](#) on page 63

You can filter nodes to see which nodes share the same tag.

Add multiple tag values

Adding multiple tag values to a tag key can help you group nodes more efficiently.

By assigning multiple values to tag, you can create sub-groups of nodes grouped by a particular tag. For example, if you have an existing tag, environment with a value of *production*, you might wish also to add values of *test* and *pre-production* to define nodes in those environments too.

You cannot add new values to existing tags by editing them. The following procedure explains how to add new values correctly.

1. On the **Nodes** page, select the nodes to which you want to add the new tag.

Important: Do not select nodes that already use the tag key whose value you want to change. If you do that, the existing tag value is changed, no additional value is added for the tag.

2. Click **Assign tags**.
3. On the side panel that appears on the right of the screen:
 - a) In the **Tag key** field, type the name of the key to which you want to add a new value.
 - b) Next, enter a new value for the tag in the **Tag value** field. This action does not overwrite any existing values for the selected tag key provided that none of the selected nodes already use the selected tag key.
 - c) Click **Close**.

The tag with the new value is assigned to the selected nodes.

Edit tag values

To change a tag key, you simply create a new tag, and delete the old one. To change an existing tag value you can edit the tag itself.

To edit a tag value:

1. On the **Nodes** page, click **Create Filter** and select the **Filter by tags** tab on the dialog box that opens.
2. Click **Add filter** and select the **Tag Key** from the drop-down list.
3. The **Operator** drop-down list gives you two options:
 - Select **Is assigned** to retrieve a list of nodes that use the selected **Tag Key**.
 - Select **Equals** and add a tag **Value** to search for nodes where the selected tag key has a specific value. This is useful in cases where you have assigned more than one tag value to a tag key.
4. Click **Update** and then **Apply all** to see your search results.
5. Select the nodes that use the tag whose value you want to edit in the search results table.
6. Click **Assign tags**, and enter the **Tag key** for the tag you want to change in the side panel that appears.
7. Enter a new value in the **Key value** field, and click **Add tag**.

The new value is updated on all nodes that use the selected tag.

Related information

[Assign a tag to multiple nodes](#) on page 61

You can add a tag to multiple nodes at once.

[Assign a tag to a node](#) on page 61

You can assign tags to a single nodes on the **Node Details** pages.

[Remove tags from nodes](#) on page 63

You can remove tags from single or multiple nodes.

[Filter nodes by tag](#) on page 63

You can filter nodes to see which nodes share the same tag.

Remove tags from nodes

You can remove tags from single or multiple nodes.

You can remove a tag from a single node by removing the tag badge on the node's **Node Details** page.

To remove a tag from several nodes at one time:

1. On the **Nodes** page, click **Create Filter** and select the **Filter by tags** tab on the dialog box that opens.
2. Click **Add filter** and select the **Tag Key** from the drop-down list.
3. The **Operator** drop-down list gives you two options:
 - Select **Is assigned** to retrieve a list of nodes that use the selected **Tag Key**.
 - Select **Equals** and add a tag **Value** to search for nodes where the selected tag key has a specific value. This is useful in cases where you have assigned more than one tag value to a tag key.
4. Click **Update** and then **Apply all** to see your search results.
5. Select the nodes from which you want to remove the tag in the search results table.
6. Click **Assign tags** and remove the relevant tag badge in the **Common tags** area of the **Assign tags** side panel that appears.
7. Click **Close** to remove the tag from the selected nodes.

Related information

[Edit tag values](#) on page 62

To change a tag key, you simply create a new tag, and delete the old one. To change an existing tag value you can edit the tag itself.

[Assign a tag to multiple nodes](#) on page 61

You can add a tag to multiple nodes at once.

[Assign a tag to a node](#) on page 61

You can assign tags to a single nodes on the **Node Details** pages.

[Filter nodes by tag](#) on page 63

You can filter nodes to see which nodes share the same tag.

Filter nodes by tag

You can filter nodes to see which nodes share the same tag.

Filtering by tag allows you to quickly see all the nodes to which a tag has been applied. It's then easy to run a task against the tagged nodes.

1. On the **Nodes** page, click **Create Filter** and select the **Filter by tags** tab on the dialog box that opens.
2. Click **Add filter** and select the **Tag Key** from the drop-down list.

3. The **Operator** drop-down list gives you two options:
 - Select **Is assigned** to retrieve a list of nodes that use the selected **Tag Key**.
 - Select **Equals** and add a tag **Value** to search for nodes where the selected tag key has a specific value. This is useful in cases where you have assigned more than one tag value to a tag key.
4. Click **Update** and then **Apply all** to see your search results.
5. To run a task against the nodes returned by your search, select them all in the **Resource Name** column, click **Run Task** and create a task workflow.

Note: Remember, you can only run tasks on accessible nodes.

Related information

[Edit tag values](#) on page 62

To change a tag key, you simply create a new tag, and delete the old one. To change an existing tag value you can edit the tag itself.

[Assign a tag to multiple nodes](#) on page 61

You can add a tag to multiple nodes at once.

[Assign a tag to a node](#) on page 61

You can assign tags to a single nodes on the **Node Details** pages.

[Add multiple tag values](#) on page 62

Adding multiple tag values to a tag key can help you group nodes more efficiently.

[Run tasks](#) on page 70

Run ad hoc tasks on target nodes to upgrade packages, restart services, execute shell commands, or perform any other type of single-action executions on your nodes.

Prioritizing vulnerabilities

To improve your infrastructure security, it is crucial to analyze the vulnerabilities detected during a vulnerability scan. By monitoring the number of vulnerabilities and affected nodes, the risk score, and the analysis of each vulnerability, you can prioritize its remediation and manage your security resources efficiently.

To help you examine the severity of the vulnerabilities in the context of each node, the **Vulnerabilities** dashboard provides a summary view of the most critical ones detected on your infrastructure.

Tip: By default, the dashboard automatically refreshes every 15 minutes. To change to a manual refresh, click **Manage sources > Automatically refresh dashboard**. The manual refresh dashboard option appears at the top of the dashboard.

Vulnerable nodes needing immediate attention

It's a top priority to fix a critical vulnerability in a node that's considered extremely important. However, remediating the same vulnerability might not be a top priority if it's present on a node of medium or low importance.

There are several ways you can explore vulnerable nodes from the **Vulnerabilities** dashboard:

- From the metrics bar at the top of the **Vulnerabilities** dashboard:
 - The **Vulnerable Nodes** area of the metrics bar displays how many nodes on your network are vulnerable. Click this area to go to the **Most vulnerable nodes** page. The **Most vulnerable nodes** page lists and orders nodes ranked by the number of vulnerabilities detected on each node. You can further filter by resource name, IP address, operating system, and OS version, or any combination of these. Select an individual node to go its **Most vulnerable nodes** page where you can run tasks to remediate the vulnerabilities affecting it.

- The **Most Vulnerable Node** area of the metrics bar displays the number of vulnerabilities affecting the most vulnerable node on your system. Click this area to go to the **Most vulnerable nodes** page for this particular node. You can run task directly from this page to remediate the vulnerabilities affecting this node.
- From the **Nodes** tab:
 - Sort the **Nodes** table by the total number of **Vulnerabilities** column to list the nodes with most vulnerabilities.

Vulnerabilities posing the highest risk to my infrastructure

A risk is a potential for loss, damage, or destruction of a node as a result of a threat exploiting a vulnerability. The risk score assigned to each vulnerability indicates the potential danger it poses to each node. It is based on the impact and possibility of exploit.

There are several ways you can explore high risk vulnerabilities from the **Vulnerabilities** dashboard:

- On the metrics bar at the top of the **Vulnerabilities** dashboard:
 - The **Highest Risk Score** area displays the risk score (as defined by your scanner) for the most serious vulnerability affecting your network. Click this area to go to the **Vulnerability detail** page for this vulnerability, where you can information on the number of nodes with the vulnerability, an analysis of the threat, and information on the steps needed to remediate the problem (if provided by your scanner).
- On the **Vulnerabilities** tab:
 - The **Criticality breakdown** chart displays the percentage of vulnerabilities in each criticality division affecting nodes on your network. Mouse over each section in the chart to see more information on the number of vulnerabilities in that category affecting nodes in your network. Click on a section to view all vulnerabilities in this criticality division listed in the **Vulnerabilities** table.
 - Sort the **Vulnerabilities** table by **Risk score** to list vulnerabilities by highest risk score.

The **Highest risk vulnerabilities** page lists and orders vulnerabilities ranked by the assigned risk score and by the number of nodes the vulnerability was detected on.

Important: The risk score is assigned by your vulnerability scanner.

Vulnerabilities affecting a large percentage of my infrastructure

On the **Vulnerabilities** tab of the Vulnerabilities dashboard, the **Top 5 common vulnerabilities** chart lists the top 5 vulnerabilities detected on nodes running on your infrastructure. Each vulnerability on this chart is ranked by the number of nodes it was detected on. Click a bar in this area to see all the nodes affected by the selected vulnerability in the **Vulnerabilities** table.

The **Hosts affected** column in the **Vulnerabilities** table lists vulnerabilities ranked by the number of nodes the vulnerability is detected on. Sort this column to order vulnerabilities by the number of nodes they affect.

- [Viewing vulnerability details](#) on page 66

Use the **Vulnerability details** page to identify which nodes are affected by the vulnerability, and using the analysis and remediation details determine which task to run on the nodes to help fix the specific vulnerability.

- [Latest events for vulnerabilities](#) on page 67

The **Vulnerability Latest Events** page outlines the latest events that have occurred to mitigate an individual vulnerability.

- [Latest event vulnerability task details](#) on page 67

The **Vulnerability Task Detail** page for vulnerabilities provides information on a particular task that has been run to mitigate an individual vulnerability.

- [Accepting vulnerability risk](#) on page 67

You may not want or need to deal with every vulnerability on every node in your network. You can also tell Puppet Remediate not to report on selected vulnerabilities on individual hosts.

- [Filtering and exporting data](#) on page 68

Create custom filters and customize table views to view vulnerability data most important to you, or for backup purposes, export data to a CSV file.

Viewing vulnerability details

Use the **Vulnerability details** page to identify which nodes are affected by the vulnerability, and using the analysis and remediation details determine which task to run on the nodes to help fix the specific vulnerability.

Metrics

Displayed on each vulnerability details page are metrics relating to the vulnerability:

- Risk score - the risk score value assigned to the vulnerability by your vulnerability scanner.
- Nodes affected - the number of nodes the vulnerability was detected on.
- Infrastructure affected - the percentage of your entire infrastructure the vulnerability affects.
- Accepted risks - The number of vulnerabilities you have instructed Remediate not to report on for selected nodes.

CVE details

If provided by your vulnerability scanner, the following CVE details are displayed for each vulnerability.

CVE detail	Description
CVE ID	The CVE ID is a unique identifier for a security vulnerability and is the number sequence of a CVE entry. For example, CVE-2019-0601 includes the CVE prefix, the year that the CVE ID was assigned or made public, and the sequence of numbers.
CVSS base score	Depending on how easy it is to exploit the vulnerability and how damaging it can be, each vulnerability is assigned a base score in the Common Vulnerability Scoring System (CVSS) which ranges from zero to ten.
CVSS V3 base score	
CVSS temporal score	The temporal score is calculated using metrics on how likely the vulnerability is to be exploited, on how easy the vulnerability is to fix, and on how confidently it can be said that a vulnerability exists.
CVSS V3 temporal score	

Latest events

The **Latest events** area lists the last three tasks carried out to remediate the vulnerability. Mouse over each task to see who initiated the task, any comments they made, how many nodes were affected, and how long ago the task was run. Click an individual task to go to the **Vulnerability Task Detail** page for more information about the chosen task.

Click **View all** to go to the **Latest events** page for the vulnerability in question, where information on all remediation tasks carried out for the vulnerability is given.

Analysis

The analysis provided by your vulnerability scanner is a description of the vulnerability threat, and the possible consequences that can occur if the vulnerability is successfully exploited.

Remediation

The remediation details provided by your vulnerability scanner is a brief summary of how you can fix the vulnerability. For information on how to fix a vulnerability, see [Remediating Vulnerabilities](#).

Latest events for vulnerabilities

The **Vulnerability Latest Events** page outlines the latest events that have occurred to mitigate an individual vulnerability.

The **Vulnerability Latest Events** page contains a metrics bar that give you at-a-glance information on:

- The number of tasks run as part of remediation events for this vulnerability
- The number of tasks run that failed
- The time and date the last task ran.

The **Vulnerability Events** table contains the following information:

- The event type.
- Comments added to the event by its initiator.
- Event status (Succeeded/Failed/Running/Pending)
- The number of nodes on which event tasks failed
- The user who initiated the event.
- The date and time the event started.

Latest event vulnerability task details

The **Vulnerability Task Detail** page for vulnerabilities provides information on a particular task that has been run to mitigate an individual vulnerability.

You access the **Vulnerability Task Detail** page by clicking the task name in the **Event** column in the table on the **Latest events** page for the vulnerability.

You can also access the **Vulnerability Task Detail** page by clicking the task in the **Latest events** area on the vulnerability's **Vulnerability details** page.

The **Vulnerability Task Detail** page contains a metrics bar that give you at a glance information on:

- The status of the task (Succeeded/Failed/Running/Pending).
- The number of nodes targeted.
- The number of nodes on which the task failed.
- The ID of the user who initiated the task.
- The date and time the task was initiated.

Information is also displayed on:

- Any comments made on the task by the initiator.
- Any parameters passed to the tasks.
- The credentials used by the task to access nodes.

Vulnerability tasks table shows status information and task completion time information for each node included in the task. Click the IP address for a node to go to the **Node Detail** page.

Accepting vulnerability risk

You may not want or need to deal with every vulnerability on every node in your network. You can also tell Puppet Remediate not to report on selected vulnerabilities on individual hosts.

If a vulnerability is low-risk, you might decide to deal with the vulnerability at a later time. If you decide to accept the risk a reported vulnerability poses on a given node, you can instruct Remediate to stop reporting on that vulnerability on the selected node on via the **Vulnerability detail** page.

1. Go to the **Vulnerability detail** page for the vulnerability you want to work on.

2. Select the nodes you want Remediate to pause reporting on for the chosen vulnerability, and click **Accept Risk**. Information on the nodes you selected is provided in the table on this page. You can click **Edit nodes** to further refine your selection.
3. On the **Review and accept risk** page that appears, enter a name in the **Identifier** field to help you identify the accepted risk later, and click **Confirm accepted risk**.

Your acceptance of the risk for the given vulnerability on the selected nodes is displayed on the **Accepted risks** page.

Use the **Accepted risks** page to view more information on the vulnerability and nodes associated with the accepted risk entries you have made. You can also delete accepted risks from this page.

On deletion of an accepted risk, Remediate recommences reporting on that vulnerability on the relevant nodes.

Note: You can also access the **Accepted risks** page by clicking **Accepted Risks** on the **Metrics bar** at the **Vulnerabilities** dashboard page.

Filtering and exporting data

Create custom filters and customize table views to view vulnerability data most important to you, or for backup purposes, export data to a CSV file.

Create custom filters

You can create a list of nodes or vulnerabilities for further investigation by creating a custom filter with multiple criteria.


1. On any listing or details page, click **Create filter**.
2. Select a **Field**.
3. Select an **Operator**:

Operator	Filter for resources where the value of the field:
Equals	is an exact match to the value you specify.
Not equal to	is anything except an exact match to the value you specify.
Contains	contains the value you specify.
Does not contain	does not contain the value you specify.

4. Enter a **Value**.
5. Click **Add filter**.
6. To add criteria to the filter, click **Add another filter**.
7. As needed, repeat these steps to add additional filters.
8. To display the filtered list, click **Apply all**.

Customize table views

Customize each table view by showing or hiding table columns on any of the node or vulnerability pages.

1. On any listings or details page, click **Columns +/-**.
By default, all available table columns are selected.
2. To hide a table column, click the column header to remove the selection indicator ().
3. To confirm your selections, click **Apply**.

Export data

To keep a backup of data relating to nodes or vulnerabilities, download a CSV file that contains the same information as the page you are currently viewing, including all filter selections.

To download resource data as a .csv file, click **Export**. The filename is <ui_page>_<timestamp>.csv, for example pd_export_highest_risk_vulnerabilities_2019-06-17_1054.csv.

Remediating vulnerabilities

To help fix vulnerabilities, run ad hoc tasks on target Linux and Windows nodes to install Puppet agents, manage packages or system services, or to execute shell commands. Upload your scripts to be converted into tasks, or upload Puppet module containing a task.

- [Upload scripts](#) on page 69

Uploaded Linux and Windows scripts are converted into tasks to run on vulnerable nodes and help fix vulnerabilities.

- [Upload modules](#) on page 70

To help fix vulnerabilities on nodes, upload modules published on the Puppet Forge. The module must contain a task.

- [Run tasks](#) on page 70

Run ad hoc tasks on target nodes to upgrade packages, restart services, execute shell commands, or perform any other type of single-action executions on your nodes.

- [Delete a task](#) on page 74

You can delete tasks from the **Manage Tasks** page.

- [Remediate your top vulnerabilities](#) on page 74

The **Top vulnerabilities** table on the **Vulnerabilities** page lists the vulnerabilities that you need to address most urgently. The ranking on this table is based on a combination of risk score and number of network nodes affected.

- [Remediate high risk vulnerabilities](#) on page 75

The **Highest risk vulnerabilities** chart on the Remediate dashboard gives you visibility of the most serious vulnerabilities affecting your system.

Upload scripts

Uploaded Linux and Windows scripts are converted into tasks to run on vulnerable nodes and help fix vulnerabilities.

You can upload any script that can be executed from PowerShell on Windows, or a shell on Linux. You must ensure the scripting language is installed on the nodes where you want to run your task as Remediate does not check for this, and you get an error if the relevant language interpreter is not present.

1. On the sidebar, click **Manage tasks**, and then click **Add tasks**.
2. On the **Upload file** page, select **Add Linux script** or **Add Windows script**.
3. Select your script, and then click **Open**.

Tip: If you upload both a Linux and Windows script, they are converted and combined into one task that can run on either operating system.

4. Click **Confirm details**.
5. On the **Add details** page, enter a name and description for the task.
6. Click **Save task**.

The **Settings** page appears, and your task is now available and is listed in the **Tasks** table.

Note: If you want to update a previously uploaded script, you must delete the existing script on the **Manage Tasks** page first, before uploading the updated version using the method described above.

Related information

[Run tasks](#) on page 70

Run ad hoc tasks on target nodes to upgrade packages, restart services, execute shell commands, or perform any other type of single-action executions on your nodes.

[Delete a task](#) on page 74

You can delete tasks from the **Manage Tasks** page.

Upload modules

To help fix vulnerabilities on nodes, upload modules published on the Puppet Forge. The module must contain a task.

Modules are self-contained bundles of code and data. Each module manages a specific task in your infrastructure, such as installing and configuring a piece of software. For more information, see [Module fundamentals](#).

1. Download a module from the [Forge](#).
2. On the sidebar, click **Manage tasks**, and then click **Add tasks**.
3. Click **Upload a module**, choose the module you just downloaded from the Forge.

Important: The name of your module needs to follow the default download format.

4. Click **Confirm details**.

Note: If the module name is the same as a previously uploaded module, a warning message appears. Select the check box on this page if you want to overwrite the tasks in the pre-existing module with those in the new module.

5. Click **Save task**.

The **Settings** page appears, and your tasks from the module are now available and are listed in the **Tasks** table.

Important: Your module must not exceed 1Mb in size.

Related information

[Run tasks](#) on page 70

Run ad hoc tasks on target nodes to upgrade packages, restart services, execute shell commands, or perform any other type of single-action executions on your nodes.

[Delete a task](#) on page 74

You can delete tasks from the **Manage Tasks** page.

Run tasks

Run ad hoc tasks on target nodes to upgrade packages, restart services, execute shell commands, or perform any other type of single-action executions on your nodes.

Tasks are similar to scripts, but they are kept in modules and can have metadata. For more information, see [Bolt tasks](#).

When selecting to run a task from either a details or listing page, some of the selected nodes might not be eligible to run the task on. An eligible node must be accessible using the SSH or WinRM credentials you added, or if running the install Puppet agent task, an eligible node doesn't have the agent installed.

1. Select the nodes you want to run tasks against on a **Vulnerability detail** page, and click **Run task**.
2. On the **Select a task** page, choose the task you want to run.

Note: You can only run tasks that are available on all the nodes you selected.

3. For Puppet Enterprise nodes only, choose the environment where you want the tasks to run in the **Environment** column.

4. Click **Select this task** to proceed.
5. On the **Configure task** page, configure the task as required. Instructions on remediation for the selected vulnerability are visible on this page. Click **Confirm details** when you are done.
6. On the **Select credentials** page, select the credentials that allow you to run the task on the selected nodes, and click **Confirm credentials**.

Note: If you are running the task solely on nodes where Puppet Enterprise is installed, the **Select credentials** page is skipped entirely. Credentials are not needed for PE nodes.

7. On the **Review and run task** page, verify that the task summary information is correct, and click **Run task**.

A confirmation message appears at the top of the page, confirming that the task type that is now running and how many nodes it affects.

Tip: To view the status of the task run, on the left hand pane, click **Latest events**.

Note: The changes made by the task if successful are only reflected here after your next security scan, so don't worry if you see no updates at this point.

Installing Puppet agents

Install a Puppet Enterprise agent to regularly pull configuration catalogs from a Puppet master, and apply them to your target Linux or Windows nodes. The agent maintains the node configuration you want.

Although Puppet Remediate is not integrated with Puppet or Puppet Enterprise, and you do not need to have Puppet or Puppet Enterprise installed to use Remediate, you can use Remediate to install a Puppet Enterprise agent to work with a Puppet master.

Parameter	Description
cacert_content	The master CA certificate content (optional). If not specified, the master's identity is not verified during the agent installation.
certname	The unique certificate name for the Puppet agent (optional).
custom_attribute	The custom attribute setting added to <code>puppet.conf</code> and included in the <code>custom_attributes</code> section of <code>csr_attributes.yaml</code> . Important: Values must be entered as an array.
dns_alt_names	The alternative DNS names for generating the agent certificate.
environment	The environment to install with the Puppet agent (optional).
extension_request	The extension attribute setting added to <code>puppet.conf</code> and included in the <code>extension_requests</code> section of <code>csr_attributes.yaml</code> . Important: Values must be entered as an array.
master	The required hostname for the Puppet master. The FQDN must be fully resolvable by the node on which you're installing the agent.
Comment	Any comments that provide context to this action.

Running shell commands

Execute an arbitrary shell command on discovered nodes without installing an agent.

Note: If you are using Remediate on Linux, the Remediate user must be added without a password to the `/etc/sudoers` file and configured to not require a `tty`. For example:

```
Defaults:myuser !requiretty
```

Parameter	Description
command	<p>The command to execute on the target nodes.</p> <p>Linux example:</p> <pre>echo "Hello, World \${USER}" > /hello.txt</pre> <p>Windows example:</p> <pre>echo "Hello, World \$env:UserName" > C:\hello.txt</pre> <p>To execute commands, Windows tasks use the command prompt. To run PowerShell commands, you must invoke PowerShell. For example:</p> <pre>powershell Get-Process</pre>
failonfail	<p>By default, the task fails when the command returns a non-zero.</p> <p>If you do not configure this parameter, its value is set to the default value (true). Deselect the check box to set the parameter value to false. Select it again to reset the parameter to true.</p>
interleave	<p>By default, content from stdout and stderr is interleaved.</p> <p>If you do not configure this parameter, its value is set to the default value (true). Deselect the check box to set the parameter value to false. Select it again to reset the parameter to true.</p>
Comment	Any comments that provide context to this action.

Managing packages

Install, upgrade, or uninstall packages on discovered nodes without installing an agent.

Remember:

To run the manage package task on target hosts, the following package management systems are required:

- APT or YUM for Linux hosts.
- Chocolatey for Windows hosts.

Parameter	Description
Action	<p>The action to be applied to the package:</p> <ul style="list-style-type: none"> • <code>install</code> the package. To install a specific version of the package, specify the value in the <code>version</code> parameter. If installing the package for the first time, the package repository on each target node must have the package stored. • <code>status</code> of the package. Whether the package is currently installed or not. • <code>uninstall</code> the package. To uninstall a specific version of the package, specify the value in the <code>version</code> parameter. • <code>upgrade</code> the version of the package. This is particularly useful for upgrading vulnerable packages to secure versions. To upgrade to a specific version, choose <code>install</code>, and specify the value in the <code>version</code> parameter.
Name	The name of the package.
Provider	The name of the provider to use for managing or inspecting the package.
Version	<p>The version, and if applicable, the release value of the package. A version number range or a semver pattern are not permitted. For example, to install the <code>bash-4.1.2-29.el6.x86_64.rpm</code> package, enter <code>4.1.2-29.el6</code>.</p> <p>Tip: To install or upgrade to the latest version of a package, leave the <code>version</code> parameter blank.</p>
Comment	Any comments that provide context to this action.

Managing system services

Manage system services on discovered hosts without installing an agent.

Parameter	Description
Action	The action to be applied to the service: <ul style="list-style-type: none"> • stop the service. • start the service. • restart the service. • enable the service. • disable the service. • View the current status of the service (Windows only).
Force	Force a Windows service to restart even if it has dependent services. This optional parameter is passed for Windows services only. If you do not configure this parameter, its value is set to the default value (false). Select the check box to set to true. Deselect it again to reset the parameter to false.
Name	The name of the service.
Provider	The name of the provider to use for managing or inspecting the service.
Comment	Any comments that provide context to this action.

Delete a task

You can delete tasks from the **Manage Tasks** page.

To delete an existing task:

1. On the sidebar, click **Manage tasks**.
2. Select the check box on the row assigned to the task you want to delete in the **Tasks** table.
3. Click **Actions**, and then **Remove**.

Remediate your top vulnerabilities

The **Top vulnerabilities** table on the **Vulnerabilities** page lists the vulnerabilities that you need to address most urgently. The ranking on this table is based on a combination of risk score and number of network nodes affected.

To remediate a top vulnerability:

1. On the Remediate Vulnerabilities dashboard **Vulnerability overview** tab, click the relevant bar in the **Top 5 common vulnerabilities** chart.
Information on the selected vulnerability appears in the **Vulnerabilities** table.

Tip: Alternatively, sort the **Vulnerabilities** table by **Hosts affected** to list the vulnerabilities affecting the greatest number of nodes in the **Vulnerabilities** table.

2. Select the vulnerability you want to remediate in the **Vulnerabilities** table..
3. On the **Vulnerability detail** page, review the analysis and remediation information that is displayed for the selected vulnerability.
 - The **Analysis** section provides you with information on the nature of the vulnerability and the threat it poses.
 - The **Remediation** section gives practical information on the remediation task you need to carry out to block the threat (where provided by your vulnerability scanner).
4. In the **Nodes affected** table, select the nodes to which you want the remediation task to apply.

Note:

If Remediate does not have the credentials to apply a task to a node, it is not selectable in the **Nodes affected** table.

5. Click **Run Task**.
6. On the **Select a task** page, choose the task you want to run.

Note: You can only run tasks that are available on all the nodes you selected.

7. For Puppet Enterprise nodes only, choose the environment where you want the tasks to run in the **Environment** column.
8. Click **Select this task** to proceed.
9. On the **Configure task** page, configure the task as required. Instructions on remediation for the selected vulnerability are visible on this page. Click **Confirm details** when you are done.
10. On the **Select credentials** page, select the credentials that allow you to run the task on the selected nodes, and click **Confirm credentials**.

Note: If you are running the task solely on nodes where Puppet Enterprise is installed, the **Select credentials** page is skipped entirely. Credentials are not needed for PE nodes.

11. On the **Review and run task** page, verify that the task summary information is correct, and click **Run task**.

A confirmation message appears at the top of the page, confirming that the task type that is now running and how many nodes it affects.

Tip: To view the status of the task run, on the left hand pane, click **Latest events**.

Note: The changes made by the task if successful are only reflected here after your next security scan, so don't worry if you see no updates at this point.

Remediate high risk vulnerabilities

The **Highest risk vulnerabilities** chart on the Remediate dashboard gives you visibility of the most serious vulnerabilities affecting your system.

To remediate a high risk vulnerability:

1. On the Remediate Vulnerabilities dashboard **Vulnerability overview** tab, click the **Critical** (or **High** if **Critical** does not exist) segment in the **Criticality breakdown** chart.

Tip: Alternatively, select the appropriate criticality level from the filter drop-down menu at the top of the **Vulnerabilities** table. You can also sort the **Vulnerabilities** table by **Risk score** to discover the highest risk vulnerabilities.

A list of critical vulnerabilities appears in the **Vulnerabilities** table.

2. Select the vulnerability you want to remediate in the **Vulnerabilities** table.
3. On the **Vulnerability detail** page, review the analysis and remediation information that is displayed for the selected vulnerability.
 - The **Analysis** section provides you with information on the nature of the vulnerability and the threat it poses.
 - The **Remediation** section gives practical information on the remediation task you need to carry out to block the threat (where provided by your vulnerability scanner).
4. In the **Nodes affected** table, select the nodes to which you want the remediation task to apply.

Note:

If Remediate does not have the credentials to apply a task to a node, it is not selectable in the **Nodes affected** table.

5. Click **Run Task**.
6. On the **Select a task** page, choose the task you want to run.

Note: You can only run tasks that are available on all the nodes you selected.

7. For Puppet Enterprise nodes only, choose the environment where you want the tasks to run in the **Environment** column.
8. Click **Select this task** to proceed.
9. On the **Configure task** page, configure the task as required. Instructions on remediation for the selected vulnerability are visible on this page. Click **Confirm details** when you are done.
10. On the **Select credentials** page, select the credentials that allow you to run the task on the selected nodes, and click **Confirm credentials**.

Note: If you are running the task solely on nodes where Puppet Enterprise is installed, the **Select credentials** page is skipped entirely. Credentials are not needed for PE nodes.

11. On the **Review and run task** page, verify that the task summary information is correct, and click **Run task**.

A confirmation message appears at the top of the page, confirming that the task type that is now running and how many nodes it affects.

Tip: To view the status of the task run, on the left hand pane, click **Latest events**.

Note: The changes made by the task if successful are only reflected here after your next security scan, so don't worry if you see no updates at this point.

Discovering and managing resources

The **Resources** dashboard provides a high-level summary view of your infrastructure, and consists of a number of dashboard cards to drill down from for detailed information about nodes, packages, and containers.

Remember: To discover resources running on each node, you must provide SSH or WinRM credentials. See [Adding node credentials](#).

Related information

[Add sources](#) on page 51

Add your vulnerability scanner to detect and fix vulnerabilities across your infrastructure. To discover nodes, packages, and containers running on your entire infrastructure, add multiple infrastructure sources.

Discovering resources

Discover node resources running on your infrastructure source account. Add your node credentials and discover node attributes, the system services, the users and groups belonging to each node, along with packages, tags, and containers.


What resources can I discover?

Depending on the level of credentials you enter, this table lists the depth of resources that you can discover.


Sources and credentials	Discovered resources
<ul style="list-style-type: none"> Infrastructure source. 	<ul style="list-style-type: none"> Nodes, including node attributes.
<ul style="list-style-type: none"> Infrastructure source. SSH or WinRM credentials. 	<ul style="list-style-type: none"> Nodes, including node attributes. Services Users Groups Tags Packages, including package attributes. Container images, including container attributes. Vulnerabilities
<ul style="list-style-type: none"> Puppet Enterprise source. 	<ul style="list-style-type: none"> Nodes, including node attributes. Tasks configured on the PE source.

Manually triggering a discovery scan

In addition to the scheduled scanning of resources, you can also manually trigger a rescan of all or selected resources. On the **Manage Sources** page, click **Discover All** to rescan all sources listed in the **Sources** table.


Alternatively, to trigger a rescan for an individual resource, click the  icon on the row assigned to it in the **Sources** table.

Canceling a discovery scan

On the **Manage Sources** page, click the  icon on the row assigned to the resource whose ongoing discovery scan you want to stop.

Important: When you cancel the discovery run on the selected source, only the items discovered thus far on the aborted scan are retained. To keep the selected source in sync, you must trigger another discovery scan or remove and re-add the source again.

Deleting a resource

On the **Manage Sources** page, click the  icon on the row assigned to the resource you want to delete in the **Sources** table.



CAUTION: After you delete a resource by this method, it is not automatically rediscovered by the scheduled discovery. You must manually add it again. For information on how to add a resource, see [Add sources](#) on page 51.

Nodes

A number of nodes summary pages give you information about node instances across your entire infrastructure, and each page queries the data platform for specific attributes belonging to each node. For more information on nodes, see [Node attributes](#) on page 79.

Tip: Click a card to drill down and view the detailed list.

Node summary page	Description
Nodes	Click the Nodes dashboard card to see a list of on-premises and cloud nodes running on your infrastructure.
Packages	The total number of packages installed on discovered hosts.
Inaccessible nodes	The total number of discovered nodes running on your infrastructure that could not be accessed due to entering invalid credentials or due to configuration issues.
AWS nodes	Click the AWS nodes dashboard card to see a list of EC2 instances running on your AWS account.
VMware nodes	The total number of nodes running on your VMware vSphere account.
OpenStack nodes	The total number of nodes running on your OpenStack account.
Microsoft Azure nodes	The total number of nodes running on your Microsoft Azure account.
Google Cloud Platform nodes	The total number of nodes running on your GCP account.
Nodes with Puppet	The percentage and total number of discovered nodes with or without Puppet installed, as well as unknown hosts.
Uptime less than 24 hours	The percentage and total number of nodes with an uptime with less than, or greater than, 24 hours.
Top operating systems	The top six operating systems installed on discovered hosts.
Linux distributions	The percentage and total number of Linux nodes categorized by the distribution installed.

Node summary page	Description
Windows versions	The percentage and total number of Windows nodes categorized by the version of Windows installed.
Top cloud instances by region	The top six cloud instances categorized by region.
Top containers by image	The top six container images used by running containers.

Packages

Click the Packages dashboard card to see a list of packages in use across your infrastructure by name, version, and manager, as well as the number of instances of each package. For more information about packages, see [Package attributes](#).

Containers

Click the Containers, or the Top containers by image, dashboard card to see a list of container instances running on your infrastructure. For more information about containers, see [Container attributes](#).

Related information

[Add sources](#) on page 51

Add your vulnerability scanner to detect and fix vulnerabilities across your infrastructure. To discover nodes, packages, and containers running on your entire infrastructure, add multiple infrastructure sources.

Viewing resource details

Puppet Remediate groups attributes associated with each discovered resource into a number of different facets to give you even more insights. A facet represents a set of related attributes, each one independently maintained, that describe a certain aspect of a discovered node.

Node attributes

Puppet Remediate considers each node as a network accessible resource, whether it's physical or virtual, and discovers attributes that define the host's state and properties.

Amazon Web Services

Discover EC2 instances running on your AWS account, including instance attributes, packages, and containers.

Attribute	Facet::attribute	Description
AMI Name	aws_ec2Instance::name	The name of the Amazon Machine Image (AMI).
Availability zone	aws_ec2Instance::PlacementAvailabilityZone	The availability zone of the instance.
Creation Date	aws_ec2Instance::CreationDate	The date and time the resource was created.
Description	aws_ec2Instance::Description	The description of the image.

Attribute	Facet::attribute	Description
Image type	aws_ec2Instance::imageType	The type of image: <ul style="list-style-type: none"> • machine • kernel • ramdisk
Instance type	aws_ec2Instance::InstanceType	The instance type.
Key name	aws_ec2Instance::KeyName	The name of the key pair.
Launch time	aws_ec2Instance::LaunchTime	The date and time the resource was launched.
Name	host::name	The name of the host.
Operating system	computeHost::os	The operating system running on the instance.
OS version	computeHost::osVersion	The version of the operating system running on the instance.
Owner ID	aws_ec2Instance::OwnerId	The account ID of the image owner.
Private DNS Name	aws_ec2Instance::PrivateDnsName	The private DNS name.
Private IP address	aws_ec2Instance::PrivateIpAddress	The private IP version 4 address.
Public DNS Name	aws_ec2Instance::PublicDnsName	Fully qualified public DNS hostname.
Public IP address	aws_ec2Instance::PublicIpAddress	The public IP version 4 address.
Region	cloudResource::region	The region where the instance exists.
Security groups	aws_ec2Instance::SecurityGroups	The security group associated with the instance.
State	aws_ec2Instance::MonitoringEnabled	Indicates whether monitoring is enabled: <ul style="list-style-type: none"> • disabled • disabling • enabled • pending
Status	cloudResource_status	The current status of the instance.
State transition reason	aws_ec2Instance::StateTransitionReason	Describes the state change.
Subnet ID	aws_ec2Instance::SubnetId	The ID of the subnet.
Tags	tags::name tags::value	The tag assigned to the AWS resource.

Attribute	Facet::attribute	Description
Uptime	computeHost::uptime	The uptime for the instance, in seconds.
VM Image	vm::image	The image identification number.
VM key name	vm::keyName	The unique identifier for the file in storage.
VM virtualization type	vm::virtualizationType	The virtualization type: <ul style="list-style-type: none"> pv hvm
VPC ID	aws_ec2Instance::VpcId	The identification for the Virtual Private Cloud (VPC).

Microsoft Azure

Discover compute instances running on your Azure account, including instance attributes, packages, and containers.

Attribute	Facet::attribute	Description
Created at	azure_computeInstance::CreatedAt	The date and time the resource was created.
Data disks	azure_computeInstance::DataDisks	The parameters used to add the data disk to the virtual machine.
ID	azure_computeInstance::ID	The resource ID.
Image	azure_computeInstance::Image	The virtual machine image.
Instance type	azure_computeInstance::InstanceType	The type of instance.
IP address	host::privateIPv4	The private IP version 4 address.
Key name	azure_computeInstance::KeyName	The keypair name.
Last scan		The Last scan of the host represented by the data platform timestamp in UTC.
Location	azure_computeInstance::Location	The location of the resource.
MAC address	azure_computeInstance::MacAddress	The MAC address assigned to the resource.
Name	host::name	The name of the host.
Operating system	computeHost::os	The operating system running on the host.
OS disk	azure_computeInstance::OSDisk	The name of the operating system used by the virtual machine.
OS version	computeHost::osVersion	The version of the operating system running on the host.
Puppet version	computeHost::puppetVersion	The version of Puppet installed.
Resource group	azure_computeInstance::ResourceGroup	The name of the resource group.
Size	azure_computeInstance::VMSize	The size type of the virtual machine.

Attribute	Facet::attribute	Description
Status	azure_computeInstance::Status	The status of the virtual machine.
Subscription ID	azure_computeInstance::SubscriptionID	The subscription ID.
Tags	azure_computeInstance::Tags	A list of tags relevant to the content of the image.
Uptime	computeHost::Uptime	The uptime for the host, in seconds.

OpenStack

Discover hosts running on your OpenStack account, including host attributes, packages, and containers.

Attribute	Facet::attribute	Description
Server ID	os_compute::ServerID	The UUID of the server.
Name	host::hostname	
IP address	host::privateIPv4	The private IP version 4 address.
Operating system	computeHost::os	The operating system running on the host.
OS version	computeHost::osVersion	The version of the operating system running on the host.
Uptime	computeHost::uptime	The uptime for the host, in seconds.
Last scan		The Last scan of the host represented by the data platform timestamp in UTC.
Puppet version	computeHost::puppetVersion	The version of Puppet installed.
Created on	os_compute::ServerCreated	The date and time the resource was created.
Server Name	os_compute::ServerName	The server name.
Flavor	os_compute::FlavorName	The display name of the flavor.
Image name	os_compute::ImageName	The display name of the image.
Status	os_compute::ServerStatus	The current server state.
Security groups	os_compute::security_group	The security group names.
VCPUs	os_compute::vcpus	The number of virtual CPUs in use.
RAM (MB)	os_compute::ram	The RAM (MB) on the virtual machine.
Disk (GB)	os_compute::disk	The disk size (GB) on the virtual machine.
Key name	os_compute::key_name	The keypair name.
IP addresses	os_compute::ServerAddresses	The IP addresses assigned to the virtual machine.
Metadata	os_compute::ServerMetadata	Custom server metadata at server launch time.

Google Cloud Platform

Discover compute engine instances running on your GCP account, including instance attributes, packages, and containers.

Attribute	Description
createTime	The date and time the resource was created.
hostname	The name of the host.
id	The ID of the instance.
image	The image identification number used by your cloud source.
name	The resource name.
os	The operating system running on the host.
osKernel	The type of kernel running in the operating system: Linux or Windows.
osVersion	The version of the operating system running on the host.
preemptible	Indicates whether it is an instance that can be temporarily interrupted: true or false.
privateIPv4	The private IP version 4 address.
provider	The name of the cloud provider: Google Cloud Platform.
publicIPv4	The IP version 4 address.
puppetInstalled	The indicator on whether Puppet is installed.
puppetVersion	The version of Puppet installed.
region	The region where the host exists.
status	The current status of the host.
tags	The tags, containing metadata, assigned to the instance.
uptime	The uptime for the host, in seconds.

VMware vSphere

Discover ESXi nodes running on your VMware vSphere account, including node attributes, packages, and containers.

Field	Facet::attribute	Description
Annotation	vmware_computeInstance::Configuration	Description for the virtual machine.
Disk	vmware_computeInstance::Disk	Guest information about disks; used space and total space available.

Field	Facet::attribute	Description
Fault tolerance state	vmware_computeInstance::RunState	Defines a set of states for fault tolerant virtual machine: <ul style="list-style-type: none"> • disabled • enabled • needSecondary • notConfigured • running • starting
Guest full name	vmware_computeInstance::GuestName	The full name of the guest operating system for the virtual machine.
Guest hostname	vmware_computeInstance::HostName	The FQDN of the guest host.
Guest heartbeat status	vmware_computeInstance::GuestStatus	The status of the guest operating system: <ul style="list-style-type: none"> • gray - VMware tools not installed or running. • red - No heartbeat. • yellow - Intermittent heartbeat. • green - Guest operating system responding normally.
Guest state	vmware_computeInstance::GuestState	Operation mode of the guest operating system: <ul style="list-style-type: none"> • running • shuttingdown • resetting • standby • notrunning • unknown
ID	vmware_computeInstance::ID	The guest operating system identifier.
IP address	host::privateIPv4	The private IP version 4 address.

Field	Facet::attribute	Description
IP stack	vmware_computeInstance::IpStack	Guest information about the IP networking stack: <ul style="list-style-type: none"> • DomainName • IPAddress
Last scan		The last scan of the host represented by the data platform timestamp in UTC.
Max memory usage (MB)	vmware_computeInstance::RunTimeMaxMemoryUsage	The maximum memory usage based on the memory configuration of the virtual machine, as well as limits configured on the virtual machine, or any parent resource pool.
Memory size (MB)	vmware_computeInstance::ConfigMemorySize	The memory size of the virtual machine, in megabytes.
Name	host::name	The name of the host.
Net	vmware_computeInstance::NetworkAdapters	Guest information about network adapters: <ul style="list-style-type: none"> • Network • MacAddress • IPAddress
Number of CPU's	vmware_computeInstance::ConfigNumCPUs	The number of processors in the virtual machine.
Number of virtual disks	vmware_computeInstance::ConfigNumVirtualDisks	The number of virtual disks attached to the virtual machine.
Operating system	computeHost::os	The operating system running on the host.
OS version	computeHost::osVersion	The version of the operating system running on the host.
Overall status	vmware_computeInstance::OverallStatus	The overall status on this host.
Power state	vmware_computeInstance::RuntimePowerState	Defines a set of states for a virtual machine: <ul style="list-style-type: none"> • poweredOff • poweredOn • suspended

Field	Facet::attribute	Description
Tools running status	vmware_computeInstance::ToolsRunningStatus	The current running status of VMware tools in the guest operating system.
Tools version	vmware_computeInstance::ToolsVersion	The VMware tools version.
Uptime	computeHost::uptime	The uptime for the instance, in seconds.

Network nodes

Discover nodes running on your network, including node attributes, packages, and containers.

Attribute	Description
dnsName	The private DNS name.
hostname	The name of the host.
name	The resource name.
os	The operating system running on the host.
osKernel	The type of kernel running in the operating system: Linux or Windows.
osVersion	The version of the operating system running on the host.
privateIPv4	The private IP version 4 address.
privateIPv6	The private IP version 6 address.
publicDnsName	Fully qualified public DNS hostname.
publicIPv4	The IP version 4 address.
puppetInstalled	The indicator on whether Puppet is installed.
puppetVersion	The version of Puppet installed.
uptime	The uptime for the host, in seconds.

Package attributes

Discover various types of information about a package instance, its attributes, along with a list of hosts and containers it's installed on.

Discover attributes that describe the characteristics of the package. The UI queries the data platform to discover the name, version, and packageManager attributes.

Attribute	Description
name	The package name.
version	The package version.
packageManager	The name of the package manager: <ul style="list-style-type: none"> apk Chocolatey dpkg rpm

Attribute	Description
	<ul style="list-style-type: none"> msi

Container attributes

Discover various types of information about a container image, its attributes, packages in use, and its label information.

Discover attributes that describe the characteristics of the container. The UI queries the data platform for the discovered `container` facet.

Attribute	Description
name	The container name for processes running inside the container.
dockerAPIVersion	The version of the Docker Engine API.
dockerMinAPIVersion	The minimum version of the Docker Engine API.
dockerVersion	The version of Docker.
id	The Docker container ID for processes running inside the container.
imageName	The Docker image name for processes running inside the container.

Discover the container label names and values:

- **Label** - the label name, derived from the `name` attribute.
- **Value** - the label value, derived from the `value` attribute.

Review Latest events

View a list of latest events and drill down to see useful information about each one.

The **Latest events** page provides a quick glance at the status of each event, organized by the type of event, and sorted by the date and time of when each event succeeded or failed. Access the **Latest events** page by clicking **Latest events** on the navigation sidebar.

The **Latest events metrics bar** at the top of the page shows:

- The total number of task and discovery runs.
- The total number of discovery runs.
- The total number of task runs.

The **Latest events table** lists the following information:

Column	Description
Event	The type of event: Discovery or Tasks. Click the name of the event in this column to go to dedicated page for this event.

Column	Description
Status	The completion status of the event: <ul style="list-style-type: none"> • Succeeded • Failed • Pending • Running
Initiated by	The name of the user who started the job.
Submission time	When the month, day, year and time the event was started.

Filter events by status by using the drop-down list at the top of the table.

- [Discovery events](#) on page 88

The **Latest events** page provides the following information about each discovery run which, by default, occurs every 4 hours.

- [Tasks events](#) on page 89

The Task event page provides the following information about the task that was ran on nodes.

Discovery events

The **Latest events** page provides the following information about each discovery run which, by default, occurs every 4 hours.

To view details on all task or discovery events, select **Discovery** from the drop-down menu on the **Latest events table**.

The **Metrics bar** on the **Discover data** page that opens, provides information on:

- The total number of jobs relating to the discovery event. Each discovery event has two jobs:
 - Discover the node.
 - Discover resources on the node.
- The total number of jobs that failed.
- The latest discovery event status.
- Information on who initiated the last discovery run event.
- The date and time the discovery run event started.

The **Discover data table** provides the following information:

Event detail	Description
Status	The job status: <ul style="list-style-type: none"> • Succeeded • Failed • Pending • Running

Event detail	Description
Source	Your configured source: vulnerability scanner or infrastructure source.
Job details	<p>Further information on the job, confirming successful discovery or reason for failure:</p> <ul style="list-style-type: none"> • Successfully connected: Remediate was able to connect to the node and access information on <i>x</i> number of facets (packages and/or containers). • Unable to connect: Remediate was not able to reach the specified • No valid credentials available: Remediate was able to connect to the node but did not have valid credentials for further discovery.
Completion time	The time the discovery run completed.

Filter jobs by status by using the drop-down list at the top of the table.

Tasks events

The Task event page provides the following information about the task that was ran on nodes.

To view details on all task or discovery events, select **Tasks** from the drop-down menu on the **Latest events table**.

The **Metrics bar** on the **Discover data** page that opens, provides information on:

- The total number of jobs relating to the task event. Each task event has two jobs:
 - Run task on node.
 - Discover resources on the node.
- The total number of jobs that failed.
- The latest task event status.
- Information on who initiated the last task run event.
- The date and time the task run event started.

The **Task data table** provides the following information:

Event detail	Description
Status	<p>The job status:</p> <ul style="list-style-type: none"> • Succeeded • Failed • Pending • Running

Event detail	Description
Source	Your configured source: vulnerability scanner or infrastructure source.
Job details	The command output from running the task and the number of facets discovered. Each facet represents a set of related attributes, each one independently maintained, that describe a certain aspect of a discovered resource; node, package, or container.
Completion time	The time the task completed.

Filter jobs by status by using the drop-down list at the top of the table.

Integration status

The **Latest scan results** page displays information on the latest discovery scan carried out by your scanner.

Click the **Integration status** link in the navigation sidebar to access the **Latest scan results**. The **Integration status** link is accompanied by an icon indicating if the latest scan is in progress, completed or returned an error.

The **Latest scan results** page provides the following information:

Information	Value
Vulnerability scanner status	Either in progress, completed, or returned an error.
Vulnerability scanner	The name of the scanner used for the scan.
Latest communication time	The time of the latest status response on the scan.
Completion time of previous update	The time when the last scan was completed.
Vulnerability scanner error	Error message from the vulnerability scanner, if any.

Remediate audit log

The Remediate audit logs key events in the system in a central location.

The audit log records the following events:

- Failed login attempts
- Successful login attempts
- Creation of a new user
- Removal of a user
- Addition of a data source
- Addition of credentials
- Removal of a data source or credential
- Discovery Events - this includes discovery events from VR sources.
- Running of a Task
- Upload of a custom task
- Removal of a custom task
- Accept Risk events

Accessing the audit log

The audit log can be accessed by running the mayday command (see [Configuring Remediate](#) on page 43 for more details). Alternatively, you can copy the log files to the Docker host by using the following command:

```
docker cp `docker ps -f name=remediate_audit --format "{{.ID}}"`:/app/log_vol/. ./
```

Log file format and samples

The log file uses the following format:

```
{ "msg": "<Message", "source": "<Service Name>", "timestamp": "<Timestamp", "type": "<Event Type", "user": "<username>" }
```

For example:

```
{ "msg": "SSH credential (e405e192-d73d-4a74-8e98-635208155cc6) added", "source": "controller", "timestamp": "2020-05-01 10:36:49.7111308 +0000 UTC m=+759.485022701", "type": "CREDENTIALS_ADDED", "user": "admin" }
```

Or:

```
{ "msg": "Accepting Risk for vuln '38738' on 1 hosts", "source": "controller", "timestamp": "2020-09-10 15:10:52.4962584 +0000 UTC m=+8173.421309601", "type": "RISK_ACCEPTED", "user": "admin" }
{ "msg": "Risk Acceptance '1' created for vuln '38738'", "source": "controller", "timestamp": "2020-09-10 15:10:52.5188046 +0000 UTC m=+8173.443854301", "type": "RISK_ACCEPTED", "user": "admin" }
```

Related information

[Configuring Remediate](#) on page 43

After installing Puppet Remediate, configure and administer the application by using the command line interface (CLI).

Troubleshooting

Use this section to troubleshoot issues with your Puppet Remediate installation.

Forgotten password

We do not currently support resetting individual passwords. If you forget the admin password, you must reset the entire system and re-enter any provider and credential information. To reset, run the following command:

```
docker-compose run remediate reset
```

How to generate and send logs to support

To generate service logs, create a tarball by running the mayday command:

```
docker-compose run remediate mayday
```

The tarball contains debug information which the Puppet support team uses for troubleshooting. You can send this to support via a Zendesk ticket or email.

For general help

For a list of helpful CLI commands, run the following:

```
docker-compose run remediate --help
```