



# Tenable Nessus Agent 10.4.x User Guide

---

Last Updated: May 19, 2023



# Table of Contents

<b>Welcome to Tenable Nessus Agent 10.4.x</b> .....	<b>6</b>
Agent Deployment Workflow .....	7
Benefits and Limitations .....	8
Traditional Active Scans (Non-credentialed) .....	9
Traditional Active Scans (Credentialed) .....	10
Agent Scans .....	12
Agent Use Cases .....	14
Mobile, Distributed Workforce .....	15
High Latency Networks .....	16
Hardened Systems .....	17
Deployment Considerations .....	18
General Considerations .....	19
Large-scale Deployments .....	21
Best Practices for Nessus Agents .....	28
General Best Practices .....	29
Data Aggregation in a Hybrid Environment .....	30
System Requirements .....	31
Hardware Requirements .....	32
Software Requirements .....	34
Customize SELinux Enforcing Mode Policies .....	35
Port Requirements .....	36
Licensing Requirements .....	39
Agent CPU Resource Control .....	41



Performance Metrics .....	45
Nessus Agents Performance .....	46
Software Footprint .....	47
Agent Lifecycle & Bandwidth .....	48
Nessus Manager Performance .....	49
Install Tenable Nessus Agent .....	51
Retrieve the Nessus Agent Linking Key .....	52
Install a Nessus Agent on Linux .....	53
Install a Nessus Agent on Windows .....	57
Configure and View the System Tray Application .....	63
Install a Nessus Agent on macOS .....	66
Update a Nessus Agent .....	71
Downgrade Nessus Agent .....	73
Back Up Agent .....	75
Restore Agent .....	76
Remove Nessus Agent .....	77
Uninstall a Nessus Agent on Linux .....	78
Uninstall a Nessus Agent on Windows .....	79
Uninstall a Nessus Agent on macOS .....	80
<b>Manage Agents .....</b>	<b>81</b>
Start or Stop a Nessus Agent .....	82
Agent Status .....	84
Filter Agents .....	85
Export Agents .....	86



Unlink an Agent .....	87
Agent Groups .....	88
<b>Scans .....</b>	<b>89</b>
Create an Agent Scan .....	90
Agent Scan and Policy Templates .....	91
<b>Settings .....</b>	<b>92</b>
Modify Agent Settings .....	93
Advanced Settings .....	94
Freeze Windows .....	111
Modify log.json Settings .....	112
Proxy Settings .....	113
<b>Additional Resources .....</b>	<b>115</b>
Mass Deployment Support .....	116
Environment Variables .....	117
Deploy Tenable Nessus Agent Using JSON .....	118
Create Windows or Linux Master Image with Tenable Nessus Agent Installed .....	124
Manage Logs .....	127
Nessus Service .....	132
Nessus CLI Agent Commands .....	135
Plugin Updates .....	145
Rule-based Trigger File Location .....	146
<b>FAQ .....</b>	<b>147</b>
<b>Appendix .....</b>	<b>150</b>
Configure Tenable Nessus Agent for NIAP Compliance .....	151



---

File and Process Allow List .....	153
Nessus Agent Cheatsheet .....	156
Benefits and Limitations of Using Tenable Nessus Agents .....	157
System Requirements for Tenable Nessus Agents .....	159
Installing and Linking Tenable Nessus Agents .....	161
Customer Case Studies .....	163
ACME Customer Case Study .....	164
Nessus Agent Operational Tier (Tenable.io) .....	166
Reporting Tier (Tenable.sc) .....	168
Initech Customer Case Study .....	170
Agent Deployment (Nessus Manager & Tenable.io) .....	172
Reporting and Traditional Network Scanning (Tenable.sc) .....	174
Sprocket .....	176



## Welcome to Tenable Nessus Agent 10.4.x

---

Tenable Nessus Agents are lightweight, low-footprint programs that you install locally on hosts to supplement traditional network-based scanning or to provide visibility into gaps that traditional scanning misses. Tenable Nessus Agents collect vulnerability, compliance, and system data, and report that information back to a manager for analysis. With Tenable Nessus Agents, you extend scan flexibility and coverage. You can scan hosts and endpoints that intermittently connect to the internet without using credentials. You can also run large-scale concurrent agent scans with little network impact.

### About Tenable Nessus Agents

Tenable Nessus Agents help you address the challenges of traditional network-based scanning, specifically for the assets where it's impossible or nearly impossible to consistently collect information about your organization's security posture. Traditional scanning typically occurs at selected intervals or during designated windows and requires systems to be accessible when a scan executes. If laptops or other transient devices are not accessible when a scan executes, they are excluded from the scan, leaving you unaware of vulnerabilities on those devices.

Once installed on servers, portable devices, or other assets found in today's complex IT environments, Tenable Nessus Agents identify vulnerabilities, policy violations, misconfigurations, and malware on the hosts where they are installed and report results back to the managing product. You can manage Tenable Nessus Agents with Tenable Nessus Manager or Tenable Vulnerability Management.

[Nessus Agents Product Page](#)



# Agent Deployment Workflow

Before you begin:

- If you are using Tenable Nessus Manager to manage Tenable Nessus Agents, you must deploy and configure Tenable Nessus Manager before you deploy Tenable Nessus Agents.
- If you are using Tenable Vulnerability Management to manage your Tenable Nessus Agents, you do not need to execute a preliminary deployment.

To deploy Tenable Nessus Agents

1. On each host, [install Tenable Nessus Agents](#).

As part of this step, you link the agent to the manager and verify that link. The link must be successful before you continue to the next step.

2. On the manager, create an agent group.
3. (Optional) Configure a freeze window.
4. (Optional) [Modify the default agent settings](#).
5. Create a scan targeting the agent group. For more information, see:
  - [Create a Scan \(Nessus\)](#)
  - [Create a Scan \(Tenable Vulnerability Management\)](#)

As part of this step, you configure the type of scan you want the agents to perform and the scan window during which agents communicate with the manager.

**Note:** The next time an agent in the specified agent group checks in during the scan window, it will download the scan policy from Tenable Nessus Manager or Tenable Vulnerability Management, run the scan, and upload the scan results back to the manager.



## Benefits and Limitations

Agent scans and traditional active network-based scans each have their own benefits and limitations when discovering assets and analyzing vulnerabilities on your network.

In a nutshell, traditional active scans originate from a Tenable Nessus scanner that reaches out to the hosts targeted for scanning, while agent scans run on hosts regardless of network location or connectivity and then report the results back to the manager (for example, Tenable Nessus Manager or Tenable Vulnerability Management) when network connectivity resumes.



If traditional Tenable Nessus scanning is adequate for your environment and requirements, you may not need to use agents. However, for most organizations, Tenable recommends a combination of agents and traditional scanning to ensure full visibility into the entire network.

As you design the optimal scanning strategy for your organization's technology infrastructure, it is important to understand the differences between each scanning technology available to you. The following sections describe the benefits and limitations of each scanning method:

- [Traditional Active Scans \(Non-Credentialed\)](#)
- [Traditional Active Scans \(Credentialed\)](#)
- [Agent Scans](#)





---

## Traditional Active Scans (Non-credentialed)

---

A traditional active non-credentialed scan, also known as an unauthenticated scan, is a common method for assessing the security of systems without system privileges. Non-credentialed scans enumerate a host's exposed ports, protocols, and services and identifies vulnerabilities and misconfigurations that could allow an attacker to compromise your network.

### Benefits

- Ideal for large-scale assessments in traditional enterprise environments.
- Discovers vulnerabilities that an outside attacker can use to compromise your network (provides a malicious adversary's point of view).
- Runs network-based plugins that an agent is restricted from performing.
- Can perform targeted operations like the brute forcing of credentials.

### Limitations

- Can be disruptive; that is, can sometimes have a negative effect on the network, device, or application you are testing.
- Misses client-side vulnerabilities such as detailed patch information.
- Can miss transient devices that are not always connected to the network.



---

## Traditional Active Scans (Credentialed)

---

A traditional active credentialed scan, also known as an authenticated scan, provides a deeper insight than a non-credentialed scan. The scan uses credentials to log into systems and applications and can provide a definitive list of required patches and misconfigurations.

Because a credentialed scan looks directly at the installed software, including at the version numbers, it can assess items such as:

- Identifying vulnerabilities in the software.
- Evaluating password policies.
- Enumerating USB devices.
- Checking anti-virus software configurations.

It performs all these tasks with minimal to no impact on the device.

### Benefits

- Consumes far fewer resources than non-credentialed scanning because the scan executes on hosts themselves rather than across the network.
- Non-disruptive; that is, does not have a negative effect on the network, device, or application you are testing.
- Provides more accurate results—a complete enumeration of software and patches installed on the host.
- Uncovers client-side software vulnerabilities.

### Limitations

- Requires credentials management for each scanned host.
  - Large organizations can potentially struggle with creating service accounts with the proper rights and access needed to safely conduct a credentialed scan.
  - Password rotation requirements can add to management complexity.



**Note:** Tenable integrates with leading password vaults and password managers to alleviate this limitation for traditional active credentialed scanning.

- Misses transient devices that are not always connected to the network.



---

## Agent Scans

---

Tenable Nessus Agent scans use lightweight, low-footprint programs that you install locally on hosts. Tenable Nessus Agents collect vulnerability, compliance, and system data, and report that information back to Tenable Nessus Manager or Tenable Vulnerability Management for analysis. Tenable Nessus Agents are designed to have minimal impact on the system and the network, giving you the benefit of direct access to all hosts without disrupting your end users.

### Benefits

- Provides extended scan coverage and continuous security:
  - Can deploy where it's not practical or possible to run network-based scans.
  - Can assess off-network assets and endpoints that intermittently connect to the internet (such as laptops). Tenable Nessus Agents can scan the devices regardless of network location and report results back to the manager.
- Eliminates the need for credential management:
  - Doesn't require host credentials to run, so you don't need to manually update credentials in scan configurations when credentials change, or share credentials among administrators, scanning teams, or organizations.
  - Can deploy where remote credentialed access is undesirable, such as Domain Controllers, DMZs, or Certificate Authority (CA) networks.
- Efficient:
  - Can reduce your overall network scanning overhead.
  - Relies on local host resources, where performance overhead is minimal.
  - Reduces network bandwidth need, which is important for remote facilities connected by slow networks.
  - Removes the challenge of scanning systems over segmented or complex networks.
  - Minimizes maintenance, because Tenable Nessus Agents can update automatically without a reboot or end-user interaction.
  - Large-scale concurrent agent scans can run with little network impact.



- Easy deployment and installation:
  - You can install and operate Tenable Nessus Agents on all major operating systems.
  - You can install Tenable Nessus Agents anywhere, including transient endpoints like laptops.
  - You can deploy Tenable Nessus Agents using software management systems such as Microsoft's System Center Configuration Manager (SCCM).

## **Limitations**

- Network checks—Agents are not designed to perform network checks, so certain plugins items cannot be checked or obtained if you deploy only agent scans. Combining traditional scans with agent-based scanning eliminates this gap.
- Remote connectivity—Agents miss things that can only be performed through remote connectivity, such as logging into a DB server, trying default credentials (brute force), traffic-related enumeration, etc.



---

## Agent Use Cases

---

The following sections describe various use cases for Tenable Nessus Agents.

- [Mobile, Distributed Workforce](#)
- [Tactical / Satellite / High Latency Networks](#)
- [Hardened Systems](#)



---

## Mobile, Distributed Workforce

---

Tenable recommends deploying agents for a mobile workforce, because agents eliminate the need for your employees to VPN into your organization's headquarters to have their devices scanned. In this scenario, active scanning over WAN or VPN connections incurs risks of low link speed, high encryption overhead, and possible problems with link stability. Agents can reduce scan times from hours to minutes.

To support a mobile workforce, Tenable recommends that you:

- Deploy the manager in the DMZ and assign it a publicly facing IP address that the agents can use to communicate. All communication between agent and manager occurs via TLS encrypted communication.
- Configure appropriate scan windows for agent scans. The scan window is the period of time where agents conduct their scans and report their results back to the manager. The agent discards any scan requests or results submitted after the scan window is discarded, and marks the system as not scanned.

This approach helps ensure accurate security data while also reducing the need for duplicate and irrelevant scanning. For example, an employee returning from a two-week vacation will not have to endure 14 queued scans (one for each day his/her system was offline).



---

## High Latency Networks

---

In traditional Nessus Scanning, a best practice is to put the scanner close to the assets targeted for scanning and never scan across a Wide Area Network (WAN). This strategy has proven difficult for deployment scenarios where the targeted assets do not have the luxury of a local Nessus server. These scenarios include ships underway, mobile military operations, and areas with high latency and low bandwidth. These networks typically rely on satellite connections for connectivity. The network burden that a ports, protocols, and services scan produces when running a full active scan can easily take down a satellite connection.

Nessus Agents help solve this problem by significantly minimizing network traffic related to scanning.

There are three types of data transmitted when using Nessus Agents:

- Command and control data – Transmitted from the manager to Nessus Agents, this data represents the who, what, when, where and how needed to complete the task of local scanning. This data is the smallest set of data that traverses the network.
- Results data – Result data varies in size due to the scan configuration. Historically, compliance scans are larger than vulnerability scans. This data transmits back to the manager for aggregation. Update data is the largest data type transmitted using Nessus Agents.
- Updates – When you install a Nessus Agent and link it to a Nessus Manager, the agent downloads a full set of plugins. Once that first full download completes, the agent only downloads incremental plugin updates. This approach drastically reduces the ongoing network traffic by only pulling content deltas across the network. Also, you can handle code updates by patch management systems like System Center Configuration Manager (SCCM) or Yellowdog Updater Modified (YUM), or via the manager itself.





---

## Hardened Systems

---

Traditional active scanning using scanners such as Nessus Professional has long been the preferred method for scanning systems in the enterprise environment. Active scanning is done remotely and requires access to key services that are typically disabled as part of system hardening (for example, Remote Registry access). The hardening of systems can actually limit the data collected by active scanning. Compounding this problem is that enumeration of key services requires credential scanning. To access key datasets, elevated privileges are required (that is, root, local admin, or domain admin). Many security professionals are hesitant to use these elevated privileges across the network. On high-value targets such as domain controllers, this caution is further elevated.

Nessus Agents do not require elevated privileges or extra accounts because they operate at the system level. The use of Nessus Agents allows a low-risk approach to scanning hardened systems without requiring that you reduce security. You can effectively eliminate the need for credentials while scanning at the system level.



---

## Deployment Considerations

---

All organizations face their own unique challenges for deploying technology, and as such, these deployment considerations are not a step-by-step guide for deploying Nessus Agents. Consult the Tenable technical support team to address specific product issues. You can also contact the Tenable Professional Services team for product integration requirements, complex deployment scenarios, and product training.

The following sections contain deployment guidance:

- [General Considerations](#)
- [Large-scale Deployments](#) (More than 10,000 hosts)



## General Considerations

The following are some common questions that you should answer before deploying Tenable Nessus Agents:

- What operating system do you plan to deploy the Tenable Nessus Agent on?
  - Linux (Debian/RHEL/Fedora/Ubuntu)
  - Windows (Win 10, Win Server 2012/2016 R2)
  - OS X (10.8+)
- How many Tenable Nessus Agents do you plan to deploy?
  - Fewer than 1,000
  - More than 1,000 and fewer than 5,000
  - More than 5,000 and fewer than 10,000
  - More than 10,000

**Note:** In deployment scenarios with more than 10,000 agents you should consider optimizing performance with agent group sizing and scan staggering as discussed in [Large-Scale Deployments](#).

- What are the typical hardware specifications of the hosts where you want to install Tenable Nessus Agents? For example, consider disk space, disk type and speed, CPU, cores, and RAM.
- Are there any countermeasures that exist on the host that would prevent the egress communications from the Tenable Nessus Agent to the Tenable Nessus Manager (DST: TCP/8834 [default, customizable])?
- Are there any countermeasures that exist on the host that would prevent the agent process from executing?

**Note:** See [File and Process Allow List](#) in the [appendix](#) for a list of files and processes to allow per operating system.

- How do you plan to deploy Tenable Nessus Agents across the enterprise? For example, do you want to use an enterprise deployment technology such as Active Directory, SMS, Microsoft SCCM, and/or Red Hat Satellite?



- Do you want to deploy Tenable Nessus Agents to virtual or non-persistent systems? If so, consider adding the agent to your base device template. Tenable recommends that you review your organization's process for commissioning and decommissioning virtual/non-persistent hosts to ensure successful activation or deactivation of the Tenable Nessus Agents.
- How do you plan to track the ratio of potentially deployable agent assets to actual assets with deployed agents?
- How do you plan to track the health and status of the agent on the host? For example, you might want to monitor for condition x (where x is the status of the service or the registration status of the agent); if that condition is present, you might then trigger an action or notification.
- What naming schema would best fit the infrastructure where deployed agents exist? It is important to plan how you would like to organize the breakdown of hosts running agents.
- Do you plan to supplement agent-based scanning with traditional network scans? How do you plan to maintain vulnerability information across agent and network scans? How do you plan to manage multiple repositories?

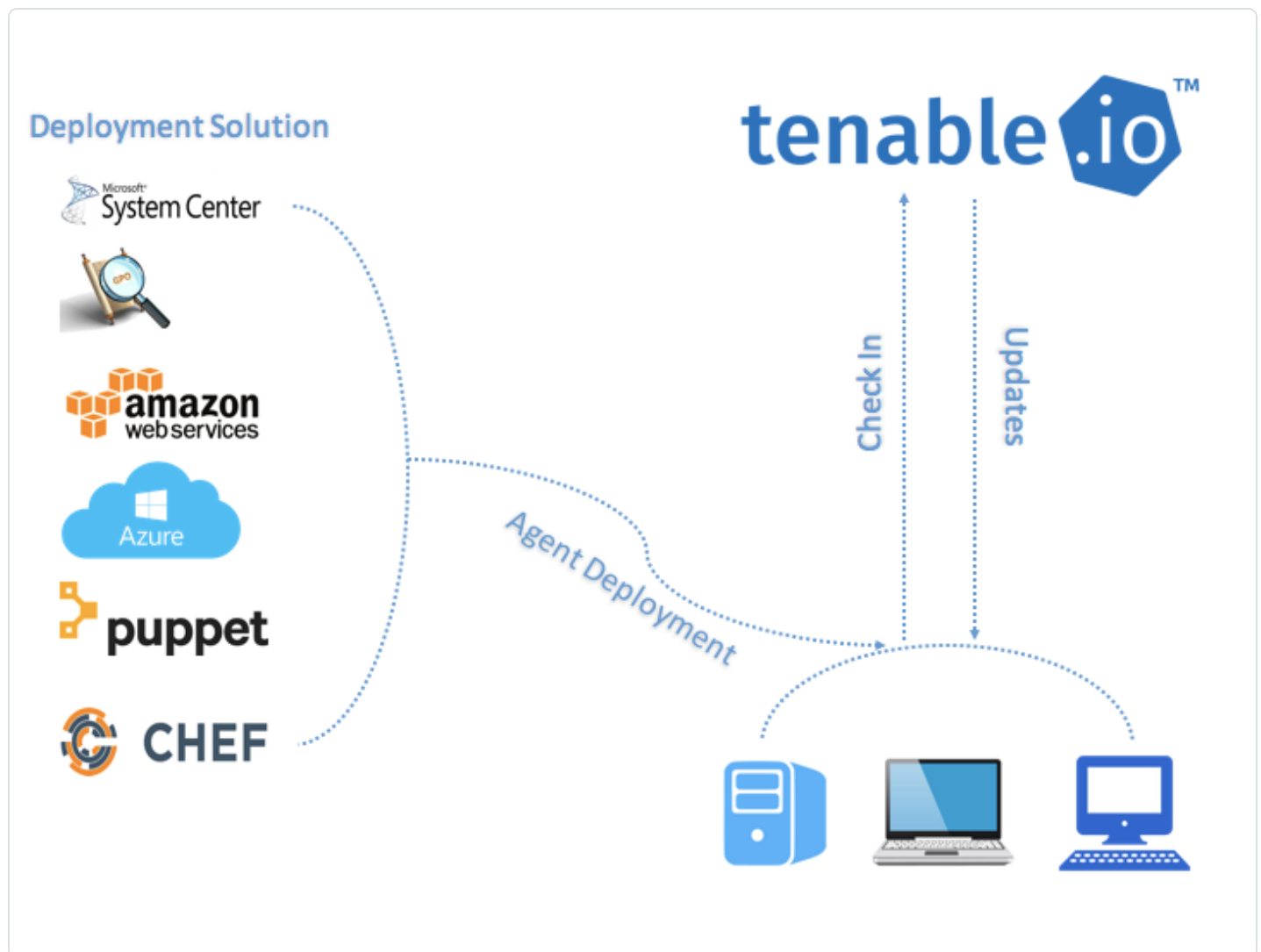


## Large-scale Deployments

If you want to deploy agents across a large-scale environment, your deployment strategy must ensure that all agents are continuously active and stay connected to Tenable Vulnerability Management or Tenable Nessus Manager.

### Deployment Strategy

When deploying many agents, consider using software to push agents through the network. For example:



Tenable recommends that you deploy batches of agents over a 24-hour period when deploying a large number of agents. This is especially helpful if you have a limited network bandwidth and need to limit the amount of data your network is downloading at one time.



After you install an agent, it receives its first plugin update once it receives instructions to run an assessment. The agent sets a timer to attempt the next update 24 hours from the initial plugin update time (and update the plugin update date on subsequent successful plugin downloads). Deploying your agents in batches also prevents too many agents from checking for product updates at one time and consuming too much bandwidth.

An agent links to Tenable Nessus Manager or Tenable Vulnerability Management after a random delay ranging from zero to five minutes. This delay occurs when the agent initially links, and also when the agent restarts either manually or through a system reboot. Enforcing a delay reduces network traffic when deploying or restarting large amounts of agents, and reduces the load on Tenable Nessus Manager or Tenable Vulnerability Management.

## Clustering

With Tenable Nessus Manager clustering, you can deploy and manage large numbers of agents from a single Tenable Nessus Manager instance. For Tenable.sc users with over 10,000 agents and up to 200,000 agents, you can manage your agent scans from a single Tenable Nessus Manager, rather than needing to link multiple instances of Tenable Nessus Manager to Tenable.sc.

A Tenable Nessus Manager instance with clustering enabled acts as a *parent node* to *child nodes*, each of which manage a smaller number of agents. Once a Tenable Nessus Manager instance becomes a parent node, it no longer manages agents directly. Instead, it acts as a single point of access where you can manage scan policies and schedules for all the agents across the child nodes. With clustering, you can scale your deployment size more easily than if you had to manage several different Tenable Nessus Manager instances separately.

### Example scenario: Deploying 100,000 agents

You are a Tenable.sc user who wants to deploy 100,000 agents, managed by Tenable Nessus Manager.

*Without clustering*, you deploy 10 Tenable Nessus Manager instances, each supporting 10,000 agents. You must manually manage each Tenable Nessus Manager instance separately, such as setting agent scan policies and schedules, and updating your software versions. You must separately link each Tenable Nessus Manager instance to Tenable.sc.

*With clustering*, you use one Tenable Nessus Manager instance to manage 100,000 agents. You enable clustering on Tenable Nessus Manager, which turns it into a parent node, a management point for child nodes. You link 10 child nodes, each of which manages around 10,000 agents. You



can either link new agents or migrate existing agents to the cluster. The child nodes receive agent scan policy, schedule, and plugin and software updates from the parent node. You link only the Tenable Nessus Manager parent node to Tenable.sc.

**Note:** All Tenable Nessus nodes in a cluster must be on the same version (for example, using the clustering example above, the Tenable Nessus Manager parent node and 10 children nodes need be on the same Tenable Nessus version). Otherwise, the cluster deployment is unsupported.

For more information, see [Clustering](#) in the *Nessus User Guide*.

## Agent Groups

Tenable recommends that you size agent groups appropriately, particularly if you are managing scans in Tenable Nessus Manager or Tenable Vulnerability Management and then importing the scan data into Tenable.sc. You can size agent groups when you manage agents in Tenable Nessus Manager or Tenable Vulnerability Management.

The more agents that you scan and include in a single agent group, the more data that the manager must process in a single batch. The size of the agent group determines the size of the `.nessus` file that must be imported into Tenable.sc. The `.nessus` file size affects hard drive space and bandwidth.

### Group Sizing

Product	Agents Assigned per Group
Tenable Vulnerability Management	Unlimited agents per group if not sending to Tenable.sc 20,000 agents per group if sending to Tenable.sc
Tenable Nessus Manager	Unlimited agents per group if not sending to Tenable.sc 20,000 agents per group if sending to Tenable.sc
Tenable Nessus Manager Clusters	Unlimited since scans are automatically broken up as appropriate by separate child nodes.

**Caution:** If you scan multiple groups of agents in a single scan, the total number of agents per scan might not match the total number of agents per group. For example, if you have three groups of 7,500 agents in Tenable Vulnerability Management, all in one scan, then data for 22,500 agents would be imported into



Tenable.sc at one time and may overwhelm it.

## Group Types

Before you deploy agents to your environment, create groups based on your scanning strategy.

The following are example group types:

### Operating System

<input type="checkbox"/> Name ^	Agents	Last Modified		
<input type="checkbox"/> <small>Shared</small> Amazon Linux	0	11:53 AM		
<input type="checkbox"/> <small>Shared</small> CentOS	0	11:53 AM		
<input type="checkbox"/> <small>Shared</small> Red Hat	0	11:53 AM		
<input type="checkbox"/> <small>Shared</small> Windows	0	11:53 AM		

### Asset Type or Location

<input type="checkbox"/> Name ^	Agents	Last Modified		
<input type="checkbox"/> <small>Shared</small> Production Servers	0	11:56 AM		
<input type="checkbox"/> <small>Shared</small> Servers in External DMZ	0	11:57 AM		
<input type="checkbox"/> <small>Shared</small> Servers in internal DMZ	0	11:57 AM		
<input type="checkbox"/> <small>Shared</small> Workstations	0	11:57 AM		

You can also add agents to more than one group if you have multiple scanning strategies.

<input type="checkbox"/> Name ^	Agents	Last Modified		
<input type="checkbox"/> <small>Shared</small> Production Servers	0	11:56 AM		
<input type="checkbox"/> <small>Shared</small> Servers in External DMZ	0	11:57 AM		
<input type="checkbox"/> <small>Shared</small> Servers in internal DMZ	0	11:57 AM		
<input type="checkbox"/> <small>Shared</small> Workstations	0	11:57 AM		

## Scan Profile Strategy

Once you deploy agents to all necessary assets, you can create scan profiles and tie them to existing agent groups. The section below describes a few scan strategies.

### Operating System Scan strategy





The following strategy is useful if your scanning strategy is based off of the operating system of an asset.

<input type="checkbox"/> Name	Schedule	Last Modified
<input type="checkbox"/> Basic Agent Scan - Windows	On Demand	N/A
<input type="checkbox"/> Basic Agent Scan - Linux	On Demand	N/A

### Basic Agent Scan - Linux

In this example, a scan is created based on the **Basic Agent Scan** template, and is assigned the group *Amazon Linux, CentOS, and Red Hat*. This scan only scans these assets.

Name:

Description:

Folder:

Agent Groups:

Scan Window:

Agents must report within this timeframe to be visible in scan results.

### Asset Type or Location Scan Strategy

The following strategy is useful if your scanning strategy is based off of the asset type or location of an asset.

<input type="checkbox"/> Name	Schedule	Last Modified
<input type="checkbox"/> Basic Agent Scan - Production Servers	On Demand	N/A
<input type="checkbox"/> Basic Agent Scan - Internal DMZ	On Demand	N/A
<input type="checkbox"/> Basic Agent Scan - Workstations	On Demand	N/A
<input type="checkbox"/> Basic Agent Scan - External DMZ	On Demand	N/A

### Basic Agent Scan - Production Servers

In this example, a scan is created a scan based on the **Basic Agent Scan** template, and is assigned the group *Production Servers*. This scan only scans production server assets.



Name	<input type="text" value="Basic Agent Scan - Production Servers"/>
Description	<input type="text"/>
Folder	<input type="text" value="My Scans"/>
Agent Groups	<input type="text" value="Production Servers x"/>
Scan Window	<input type="text" value="3 hours"/>

Agents must report within this timeframe to be visible in scan results.

## Basic Agent Scan - Workstations

In this example, a scan is created based on the **Basic Agent Scan** template, and is assigned the group *Workstations*. This scan only scans workstation assets.

Name	<input type="text" value="Basic Agent Scan - Workstations"/>
Description	<input type="text"/>
Folder	<input type="text" value="My Scans"/>
Agent Groups	<input type="text" value="Workstations x"/>
Scan Window	<input type="text" value="3 hours"/>

Agents must report within this timeframe to be visible in scan results.

**Note:** You may want to configure workstation scans with longer scan windows, as most organizations cannot guarantee when these systems are online (as opposed to servers which are typically on 24/7).

## Scan Staggering

While scans with the Tenable Nessus Agents are more efficient in many ways than traditional network scans, scan staggering is something to consider on certain types of systems.

For example, if you install Tenable Nessus Agents on virtual machines, you may want to distribute agents among several groups and have their associated scan windows start at slightly different times.

Staggering scans limits the one-time load on the virtual host server, because agents run their assessments as soon as possible at the start of the scan window. Oversubscribed or resource-lim-



ited virtual environments may experience performance issues if agent assessments start on all systems at the same time.



---

## Best Practices for Nessus Agents

---

The following sections contain best practice guidance:

- [General Best Practices](#)
- [Data Aggregation in a Hybrid Environment](#)



---

## General Best Practices

---

- With traditional network scans, never scan through or try to bypass devices such as firewalls, switches, etc., that are designed to obfuscate or impede scans (for example, network address translation).
- Either put Nessus scanners in every segment, closest to the host, *or* run agents locally on the system, which does not require explicitly making an overage of firewall rules. Both solutions require minimal firewall rules to provide connectivity when implemented correctly.
- For full visibility into your network, Tenable recommends that you combine agent-based and traditional scanning to identify risk across your entire network. This approach is especially important for organizations in the United States Federal Government as there are specific laws and acts that mandate you evaluate the entire spectrum of your risk.



---

## Data Aggregation in a Hybrid Environment

---

This section briefly identifies areas to consider when aggregating Tenable Nessus Agent data from Nessus Manager into Tenable.sc repositories. It is important to note that communications to the Nessus Manager for data retrieval initiate from Tenable.sc. Once Tenable Nessus Agent data is imported, all normal Tenable.sc operations such as vulnerability analysis, compliance, and workflow automation apply.

- Carefully consider agent group size to reduce the volume of data being imported into Tenable.sc at a given time. Tenable recommends limiting the number of agents per scan in Nessus Manager or Tenable.io to 1,000 agents. Importing large amounts of data to Tenable.sc while parallel operations are occurring impacts Tenable.sc performance.
- Properly plan the number of Nessus scanners and Nessus Managers connected to Tenable.sc, seeking guidance from Tenable technical support staff if needed.
- Properly plan the number of concurrent scans to include agent scans (agent data retrieval process), concurrent users, number of dashboards configured, and frequency/type of reports operating on a Tenable.sc, seeking guidance from Tenable technical support staff if needed.



---

# System Requirements

---

This section includes information related to the requirements necessary to install Tenable Nessus Agents.

- [Hardware](#)
- [Software](#)
- [Dataflow](#)
- [Licensing](#)
- [Agent CPU Resource Control](#)
- [Performance Metrics](#)
  - [Nessus Agents Performance](#)
    - [Software Footprint](#)
    - [Agent Lifecycle & Bandwidth](#)
  - [Nessus Manager Performance](#)



## Hardware Requirements

### Nessus Agents

Tenable Nessus Agents are lightweight and only minimal system resources. Generally, a Tenable Nessus Agent uses 40 MB of RAM (all pageable). A Tenable Nessus Agent uses almost no CPU while idle, but is designed to use up to 100% of CPU when available during jobs.

For more information on Tenable Nessus Agent resource usage, see [Nessus Agents Performance](#).

The following table outlines the minimum recommended hardware for operating a Tenable Nessus Agent. You can install Tenable Nessus Agents on a virtual machine that meets the same requirements specified.

Hardware	Minimum Requirement
Processor	1 Dual-core CPU
Processor Speed	> 1 GHz
RAM	> 1 GB
Disk Space	<ul style="list-style-type: none"><li>• Agents 7.7.x and earlier: &gt; 1 GB, not including space used by the host operating system</li><li>• Agents 8.0.x and later: &gt; 3 GB, not including space used by the host operating system</li><li>• Agents 10.0.x and later: &gt; 2 GB, not including space used by the host operating system</li></ul> <p>The agent may require more space during certain processes, such as a <code>plugins-code.db</code> defragmentation operation.</p>
Disk Speed	15-50 IOPS

**Note:** You can control the priority of the Tenable Nessus Agent relative to the priority of other tasks running on the system. For more information see [Agent CPU Resource Control](#).

### Nessus Manager





Scenario	Minimum Recommended Hardware
Nessus Manager with 0-10,000 agents	<p><b>CPU:</b> 4 2GHz cores</p> <p><b>Memory:</b> 16 GB RAM</p> <p><b>Disk space:</b> 5 GB per 5,000 agents per concurrent scan</p> <div data-bbox="703 453 1479 569" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> Scan results and plugin updates require more disk space over time.</p></div>
Nessus Manager with 10,001-20,000 agents	<p><b>CPU:</b> 8 2GHz cores</p> <p><b>Memory:</b> 32 GB RAM</p> <p><b>Disk space:</b> 5 GB per 5,000 agents per concurrent scan</p> <div data-bbox="703 810 1479 926" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> Scan results and plugin updates require more disk space over time.</p></div> <div data-bbox="703 947 1479 1062" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> Engage with your Tenable representative for large deployments.</p></div>



---

## Software Requirements

---

Tenable Nessus Agents and Tenable Nessus Manager support Mac, Linux, and Windows operating systems.

### Nessus Agents

For Tenable Nessus Agent software requirements, see the [Agent Software Requirements](#) in the *General Requirements User Guide*.

### SELinux Requirements

Tenable Nessus Agents and Tenable Nessus Manager supports disabled, permissive, and enforcing mode Security-Enhanced Linux (SELinux) policy configurations.

- Disabled and permissive mode policies typically do not require customization to interact with Tenable Nessus Agents and Tenable Nessus Manager.
- Enforcing mode policies require customization to interact with Tenable Nessus Agents and Tenable Nessus Manager. For more information, see [Customize SELinux Enforcing Mode Policies](#).

**Note:** Tenable recommends testing your SELinux configurations before deploying on a live network.



---

## Customize SELinux Enforcing Mode Policies

---

Security-Enhanced Linux (SELinux) enforcing mode policies require customization to interact with Nessus Agents.

Tenable Support does not assist with customizing SELinux policies, but Tenable recommends monitoring your SELinux logs to identify errors and solutions for your policy configuration.

Before you begin:

- Install the SELinux `sealert` tool in a test environment that resembles your production environment.

To monitor your SELinux logs to identify errors and solutions:

1. Run the `sealert` tool, where `/var/log/audit/audit.log` is the location of your SELinux audit log:

```
sealert -a /var/log/audit/audit.log
```

The tool runs and generates a summary of error alerts and solutions. For example:

```
SELinux is preventing /usr/sbin/sshd from write access on the sock_file /dev/log
SELinux is preventing /usr/libexec/postfix/pickup from using the rlimitinh access
on a process.
```

2. Execute the recommended solution for each error alert.
3. Restart Nessus Agents.
4. Run the `sealert` tool again to confirm you resolved the error alerts.



## Port Requirements

Tenable Nessus Agent port requirements include Tenable Nessus Agent-specific requirements and manager-specific requirements. Depending on your deployment setup, see the [Nessus Manager](#) and [Tenable.sc](#) port requirements.

### Nessus Agent

Your Tenable Nessus Agents require access to specific ports for outbound traffic.

#### Outbound Traffic

You must allow outbound traffic to the following ports.

Port	Traffic
TCP 443	Communicating with Tenable Vulnerability Management.
TCP 8834	Communicating with Tenable Nessus Manager. <b>Note:</b> The default Tenable Nessus Manager port is TCP 8834. However, this port is configurable and may be different for your organization.
UDP 53	Performing DNS resolution.

### Nessus Manager

Your Tenable Nessus instances require access to specific ports for inbound and outbound traffic.

#### Inbound Traffic

You must allow inbound traffic to the following ports.

Port	Traffic
TCP 8834	Accessing the Tenable Nessus interface. Communicating with Tenable.sc. Interacting with the API.



## Outbound Traffic

You must allow outbound traffic to the following ports.

Port	Traffic
TCP 25	Sending SMTP email notifications.
TCP 443	Communicating with Tenable Vulnerability Management. Communicating with the <code>plugins.nessus.org</code> server for plugin updates.
UDP 53	Performing DNS resolution.

## Tenable.sc

Your Tenable.sc instances require access to specific ports for inbound and outbound traffic.

### Inbound Traffic

You must allow inbound traffic to the following ports.

Port	Traffic
TCP 22	Performing remote repository synchronization with another Tenable.sc.
TCP 443	Accessing the Tenable.sc interface. Communicating with Tenable Security Center Director instances. Communicating with Tenable OT Security instances. Performing the initial key push for remote repository synchronization with another Tenable.sc. Interacting with the API.

## Outbound Traffic

You must allow outbound traffic to the following ports.



Port	Traffic
TCP 22	Communicating with Log Correlation Engine for event query.
TCP 25	Sending SMTP email notifications.
TCP 443	Communicating with Tenable Vulnerability Management. Communicating with Tenable Lumin for synchronization. Communicating with the <code>plugins.nessus.org</code> server for plugin updates.
TCP 1243	Communicating with Tenable Log Correlation Engine.
TCP 8834	Communicating with Tenable Nessus.
TCP 8835	Communicating with Tenable Nessus Network Monitor.
UDP 53	Performing DNS resolution.

## Agent Content Distribution Network (CDN)

Dependent on rule logic in place, you may need to adjust your firewall or proxy rules in order to utilize the Agent Content Distribution Network (CDN) introduced with Nessus Agent 7.1.2.

### FQDN Updates

The CDN leverages `downloads-agent.cloud.tenable.com` for downloading plugins and binary updates, `uploads-agent.cloud.tenable.com` for uploading scan results, and `sensor.cloud.tenable.com` for linking and communicating with Tenable Vulnerability Management. If you have a firewall or proxy rule configured for `*.cloud.tenable.com` then you should not encounter issues. However, if there are stricter rules in place then you will need to update your rule set.

### IP Allowlisting

The IP addresses associated with `downloads-agent.cloud.tenable.com` and `uploads-agent.cloud.tenable.com` are dynamic and dependent on the locale of the Nessus Agent and its connectivity to the internet. If you currently have IP-based rules configured for proxies and firewalls you must update the rules based on IP ranges utilized by Amazon CloudFront. Amazon's documentation [Locations and IP Address Ranges of CloudFront Edge Servers](#) has a list of the IP ranges available for download.



## Licensing Requirements

Agents are licensed through the product that manages them: Nessus Manager or Tenable Vulnerability Management.

### Tenable Nessus Manager

Nessus is available to operate either as a subscription or managed by Tenable.sc. Tenable Nessus requires a plugin feed activation code to operate in subscription mode. This code identifies which version of Tenable Nessus that Tenable licensed you to install and use, and if applicable, how many IP addresses you can scan, how many remote scanners you can link to Tenable Nessus, and how many Nessus Agents you can link to Tenable Nessus Manager. Tenable Nessus Manager licenses are specific to your deployment size, especially for large deployments or deployments with multiple Tenable Nessus Manager instances. Discuss your requirements with your Tenable Customer Success Manager.

You must obtain the activation code before starting the installation process and setting up Tenable Nessus.

Your activation code:

- is a **one-time** code, unless your license or subscription changes, at which point Tenable issues you a new activation code.
- must be used with the Tenable Nessus installation within 24 hours.
- cannot be shared between scanners.
- is not case-sensitive.
- is required to manage Tenable Nessus offline.

**Note:** For more information about managing Tenable Nessus offline, refer to the [Nessus User Guide](#).

**Note:** See the [Obtain an Activation Code page](#) to obtain an activation code.

For managed Tenable Nessus scanners, the activation code and plugin updates are managed from Tenable.sc. You must start Tenable Nessus before it communicates with Tenable.sc, which it normally does not do without a valid activation code and plugins. To have Tenable Nessus ignore this



requirement and start (so that it can get the information from Tenable.sc), when you register your scanner, select **Managed by SecurityCenter**.





## Agent CPU Resource Control

You can control the priority of the Nessus Agent relative to the priority of other tasks running on the system by using the `process_priority` preference. Due to the relative nature of this preference, the amount of system resources consumed by the Tenable Nessus Agent depends not only on the value of the `process_priority` preference, but also on the overall load on the system. This may reflect on system monitors as if the agent is consuming resources over the higher priority processes. For resource control commands see [Nessus CLI Agent Commands](#).

**Note:** There may be a slight delay between setting a value for `process_priority` and seeing the change reflected in Linux, Mac OS nice values, or Windows Priority Class.

To see the effect of the `process_priority` preference, see the following table.

Preference Value	Windows - Priority Class	Mac OS - Nice Value	Linux - Nice Value
normal (default)	normal	0	0
low	below normal	10	10
high	above normal	-10	-5

**Note:** Setting your `process_priority` preference value to low could cause longer running scans. You may need to increase your scan-window timeframe to account for this value.

## Agent CPU Resource Control Advanced Settings

You can configure the following agent settings in the command line interface using the `nessuscli` utility.

Use the command `# nessuscli fix --set setting=value`. For more information, see [Nessus CLI Agent Commands](#).

For more information, and a complete list of CLI-configurable settings, see [Advanced Settings](#).

**Tip:** If you have many agents (10,000+), you may want to configure the `agent_merge_audit_trail`, `agent_merge_kb`, `agent_merge_journal_mode`, and `agent_merge_synchronous_setting` settings. Modifying these settings can dramatically lower the amount of time it takes to merge agent scan results. Review the descriptions in the following table for suggested configurations.



Name	Setting	Description	Default	Valid Values
Plugin Compilation Performance	plugin_load_performance_mode	<p>Sets plugin compilation performance, which affects CPU usage. Low performance slows down plugin compilation, but reduces the agent's CPU consumption. Setting the performance to medium or high means that plugin compilation completes more quickly, but the agent consumes more CPU. Target ranges for each setting value are:</p> <ul style="list-style-type: none"><li>• <b>low:</b> Less than 50% of a single core used by nessusd during plugin compilation.</li><li>• <b>medium:</b> 100% of two cores used by nessusd during plugin compilation.</li><li>• <b>high:</b> Unchanged from</li></ul>	high	low, medium, or high



		previous versions. Does not use more than eight cores.		
Scan Performance	scan_performance_mode	<p>Sets scan performance, which affects CPU usage. Low performance slows down scans, but reduces the agent's CPU consumption. Setting the performance to medium or high means that scans complete more quickly, but the agent consumes more CPU. Target ranges for each setting value are:</p> <ul style="list-style-type: none"><li>• <b>low:</b> Less than 50% of a single core used by nessusd during a scan.</li><li>• <b>medium:</b> 100% of two cores used by nessusd during a scan.</li><li>• <b>high:</b> Unchanged from previous versions. Does not</li></ul>	high	low, medium, or high



---

		use more than eight cores.		
--	--	-------------------------------	--	--



---

## Performance Metrics

---

Tenable transparently provides performance metrics based on internal performance testing. Performance varies by environment and you may or may not see similar results.

The following sections describe performance metrics for Tenable Nessus Agents and Tenable Nessus Manager:

- [Nessus Agents Performance](#)
  - [Software Footprint](#)
  - [Agent Lifecycle & Bandwidth](#)
- [Nessus Manager Performance](#)



---

## Nessus Agents Performance

---

Tenable transparently provides performance metrics based on internal performance testing. Performance varies by environment and you may or may not see similar results.

The following sections describe various performance metrics for Tenable Nessus Agents:

- [Software Footprint](#)
- [Lifecycle & Bandwidth](#)



## Software Footprint

**Note:** Performance varies by environment and you may or may not see similar results.

### Agents running standard agent scans

Agent Footprint on Disk	Total Agent Software Footprint on Disk	Average RAM Usage While Not Scanning	Average RAM Usage While Scanning	Average RAM Usage During Plugin Compilation	Average Network Bandwidth Usage
~40 MB	~550 MB including plugin updates *	~50 MB RAM	~85 MB RAM	~150 MB RAM	~8 MB/day

\*Under certain conditions, disk usage can spike up to 1 GB.

### Agents running inventory scans

Agent Footprint on Disk	Total Agent Software Footprint on Disk	Average RAM Usage While Not Scanning	Average RAM Usage While Scanning	Average RAM Usage During Plugin Compilation	Average Network Bandwidth Usage
~40 MB	~150 MB including plugin updates *	~50 MB RAM	~80 MB RAM	~105 MB RAM	~8 MB/day

\*Under certain conditions, disk usage can spike up to 200 MB.

For more information about inventory scanning, see [Tenable-Provided Nessus Agent Templates](#) in the *Tenable.io User Guide*.



## Agent Lifecycle & Bandwidth

**Note:** Performance varies by environment and you may or may not see similar results.

	Windows	Linux	macOS
Agent Core Software Initial Install	24.5 MB	14.9 MB	13.6 MB
Agent Core Software Updates	17.8 MB	11.5 MB	44 MB
Initial Plugin Download	150 MB	121 MB	118 MB
Differential Plugin Updates*	0.1-30 MB	0.1-30 MB	0.1-30 MB
Report Size**	1-100+ MB	1-100+ MB	1-100+ MB

\* Plugin update sizes vary depending on the difference between the new plugins available and the last date the agent updated its plugins.

\*\* Report size can vary greatly depending on the scan. Compliance audit scans can be especially large.





# Nessus Manager Performance

Tenable tested Nessus Manager performance in two scenarios. **Scenario 1** is when Nessus Agents are connected to Tenable Nessus Manager and polling for jobs. **Scenario 2** is when Nessus Agents are actively scanning and uploading scan results.

## Testing Environments

Tenable used the following testing environments for the two scenarios.

### Scenario 1

- OS: Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-75-generic x86\_64)
- RAM: 16 GB
- CPU: Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.60GHz
- Cores: 2

### Scenario 2

- OS: Windows 10 v. 1703 (OS Build: 15063.447)
- RAM: 16 GB
- CPU: Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.59GHz
- Cores: 2

Scenario 1: When Nessus Agents are connected to Tenable Nessus Manager and polling for jobs

Number of Agents	Number of Agents Sending Job Requests at a Time (2%)	MAX CPU Usage	Average CPU Usage	Average Agents Page Load Time
1,000	20	33%	5%	0.60 seconds
2,000	40	34%	5%	1.05 seconds
5,000	100	43%	6%	1.7 seconds



7,500	150	92%	7%	3.22 seconds
10,000	200	100%	7%	3.26 seconds
Number of Agents	Number of Agents Sending Job Requests at a Time (5%)	MAX CPU Usage	Average CPU Usage	Average Agents Page Load Time
1,000	50	38%	7%	0.88 seconds
2,000	100	39%	7%	1.14 seconds
5,000	250	54%	6%	1.73 seconds

Scenario 2: When Nessus agents are actively scanning and uploading scan results

Number of Agents	MAX CPU Usage	Average CPU Usage	Average Agents Page Load Time	Scan Report Size
1,000	65%	52%	1.16 seconds	363 MB
2,000	82%	53%	1.45 seconds	726 MB
3,000	82%	46%	1.67 seconds	1079 MB
4,000	86%	40%	1.70 seconds	1452 MB
5,000	99%	47%	1.73 seconds	1780 MB



---

## Install Tenable Nessus Agent

---

This section describes how to install a Tenable Nessus Agent on the following operating systems:

- [Linux](#)
- [Windows](#)
- [macOS](#)

Once installed, an agent links to Tenable Nessus Manager or Tenable Vulnerability Management after a random delay ranging from zero to five minutes. Enforcing a delay reduces network traffic when deploying or restarting large amounts of agents, and reduces the load on Tenable Nessus Manager or Tenable Vulnerability Management. Agents automatically download plugins from the manager upon linking; this process can take several minutes and you before an agent can return scan results.



---

## Retrieve the Nessus Agent Linking Key

---

Before you begin the Tenable Nessus Agents installation process, you must retrieve the agent linking key from Tenable Nessus Manager or Tenable Vulnerability Management.

To retrieve the agent linking key, use the procedures described in the [Tenable Nessus Manager](#) and [Tenable Vulnerability Management](#) user guides. Once you retrieve the linking key, you can [install](#) your agent or agents.



# Install a Nessus Agent on Linux

**Caution:** If you install a Tenable Nessus Agent on a system where an existing Tenable Nessus Agent, Tenable Nessus Manager, or Tenable Nessus scanner is running `nessusd`, the installation process kills all other `nessusd` processes. You may lose scan data as a result.

## Before You Begin

- Retrieve the Nessus Agents linking key. For more information, see the [Tenable Nessus User Guide](#) or the [Tenable Vulnerability Management User Guide](#), depending on what manager you use.
- If you previously had the Tenable Nessus Agent installed on your system, see the [knowledge base](#) article on how to avoid linking errors.

## Download the Nessus Agent

On the [Nessus Agents Download Page](#), download the package specific to your operating system.

## Install Nessus Agent

**Note:** The following procedure requires root privileges.

Using the command line interface, install the Tenable Nessus Agent.

### Example Linux Install Commands

#### Red Hat, CentOS, and Oracle Linux

```
# dnf install NessusAgent-10.3.1-es8.x86_64.rpm
```

#### Fedora

```
# dnf install NessusAgent-10.3.1-fc34.x86_64.rpm
```

#### Ubuntu

```
# dpkg -i NessusAgent-<version number>-ubuntu1110_i386.deb
```



## Debian

```
# dpkg -i NessusAgent-<version number>-debian6_amd64.deb
```

You can install a full plugins set before linking to reduce the bandwidth impact during a mass installation. You can accomplish this by using the `nessuscli agent update` command with the `--file` parameter, which specifies the location the plugins set. You must do this before [starting](#) the Tenable Nessus Agent. For example:

```
/opt/nessus_agent/sbin/nessuscli agent update --file=./plugins_set.tgz
```

The plugins set must be less than five days old. A stale plugin set older than five days forces a full plugin download to occur. You can download a recent plugins set from the [Nessus Agents download page](#).

**Note:** After installing a Nessus Agent, you must manually start the service using the command `/sbin/service nessusagent start`.

## Link Agent Using Command Line Interface

At the command prompt, use the `nessuscli agent link` command. For example:

```
/opt/nessus_agent/sbin/nessuscli agent link  
--key=00abcd0000efgh1111i0k222lmopq3333st4455u66v77777w88xy9999zabc00  
--name=MyOSXAgent --groups="All" --host=yourcompany.com --port=8834
```

The supported arguments for this command are:

Argument	Required?	Value
<code>--key</code>	yes	Use the values you <a href="#">retrieved</a> from the manager.
<code>--host</code>	yes	
<code>--port</code>	yes	
<code>--name</code>	no	Specify a name for your agent. If you do not specify a name for your agent, the name defaults to the name of the computer where you are installing the agent.



<code>--groups</code>	no	<p>Specify existing agent group or groups where you want to add the agent. If you do not specify an agent group during the install process, you can add your linked agent to an agent group later in Tenable Nessus Manager or Tenable Vulnerability Management.</p> <div data-bbox="630 436 1479 554" style="border: 1px solid blue; padding: 5px;"><p><b>Note:</b> The agent group name is case-sensitive and must match exactly.</p></div>
<code>--offline-install</code>	no	<p>For Nessus Agents 7.0.3 or later, you can install the Tenable Nessus Agent on a system even if it is offline. Add the command line option <code>offline-install="yes"</code> to the command line input. The Tenable Nessus Agent periodically attempts to link itself to either Tenable Vulnerability Management or Tenable Nessus Manager.</p> <p>If the agent cannot connect to the controller then it retries every hour, and if the agent can connect to the controller but the link fails then it retries every 24 hours.</p>
<code>--cloud</code>	no	<p>Specify the <code>--cloud</code> argument to link to Tenable Vulnerability Management.</p> <p>The <code>--cloud</code> argument is a shortcut to specifying <code>--host-t=sensor.cloud.tenable.com --port=443</code> (or <code>--host-t=cloud.tenable.com --port=443</code> for agents 8.0.x and earlier).</p> <div data-bbox="630 1415 1479 1730" style="border: 1px solid blue; padding: 5px;"><p><b>Note:</b> Starting with Tenable Nessus Agent 8.1.0, Tenable Vulnerability Management-linked agents communicate with Tenable Vulnerability Management using <code>sensor.cloud.tenable.com</code>. If agents are unable to connect to <code>sensor.cloud.tenable.com</code>, they use <code>cloud.tenable.com</code> instead. Agents with earlier versions continue to use the <code>cloud.tenable.com</code> domain.</p></div>
<code>--network</code>	no	<p>For Tenable.io-linked agents, add the agent to a custom network. If you do not specify a network, the agent belongs to the default network.</p>



If the information that you provide is incorrect, a "Failed to link agent" error appears.

**Note:** If you attempt to clone an agent and link it to Tenable Nessus Manager or Tenable Vulnerability Management, a 409 error may appear. This error appears because another machine was linked with the same UUID value in the `/etc/machine_id` or `/etc/tenable_tag` file. To resolve this issue, replace the value in the `/etc/tenable_tag` file with a valid UUIDv4 value. If the `/etc/machine_id` file does not exist, you can delete `/etc/tenable_tag` to generate a new value.

**Note:** For more information about linking agents to Tenable Vulnerability Management, see [Link a Sensor](#) in the *Tenable.io User Guide*.

## Verify a Linked Agent

To verify a linked agent in Tenable Vulnerability Management:

1. In the upper-left corner, click the ☰ button.

The left navigation plane appears.

2. In the left navigation plane, click **Settings**.

The **Settings** page appears.

3. Click the **Sensors** tile.

The **Sensors** page appears. By default, **Nessus Scanners** is selected in the left navigation menu and the **Cloud Scanners** tab is active.

4. In the left navigation menu, click **Nessus Agents**.

The **Nessus Agents** page appears and the **Linked Agents** tab is active.

5. Locate the new agent in the linked agents table.

To verify a linked agent in Tenable Nessus Manager:

1. In the top navigation bar, click **Sensors**.

The **Linked Agents** page appears.

2. Locate the new agent in the linked agents table.





## Install a Nessus Agent on Windows

**Caution:** If you install a Tenable Nessus Agent on a system where an existing Tenable Nessus Agent, Tenable Nessus Manager, or Tenable Nessus scanner is running `nessusd`, the installation process kills all other `nessusd` processes. You may lose scan data as a result.

**Note:** This procedure describes deploying Tenable Nessus Agents via the command line. You can also deploy Tenable Nessus Agents with a standard Windows service such as Active Directory (AD), Systems Management Server (SMS), or other software delivery system for MSI packages. For more information on deploying via these methods, see the appropriate vendor's documentation.

**Note:** You may be required to restart your computer to complete installation.

### Before You Begin

- Retrieve the Nessus Agents linking key. For more information, see the [Tenable Nessus User Guide](#) or the [Tenable Vulnerability Management User Guide](#), depending on what manager you use.
- Consider the following if you are reinstalling Tenable Nessus Agent after uninstalling it:
  - If you previously had the Tenable Nessus Agent installed on your system, see the [knowledge base](#) article on how to avoid linking errors.
  - On Windows, the Tenable Nessus Agent uninstall process automatically creates a [backup](#) file in the %TEMP% directory. If you reinstall Tenable Nessus Agent within 24 hours, Tenable Nessus Agent uses that backup file to [restore](#) the installation. If you want to reinstall Tenable Nessus Agent within 24 hours without using the backup, manually delete the backup file in the %TEMP% directory beforehand.

### Deploy and Link via the Command Line

You can deploy and link Tenable Nessus Agents via the command line. For example:

**Note:** You must have administrator-level privileges to deploy and link via the command line.

```
msiexec /i NessusAgent-<version number>-x64.msi NESSUS_GROUPS="Agent Group Name"  
NESSUS_SERVER="192.168.0.1:8834" NESSUS_
```



KEY=00abcd0000efgh11111i0k222lmopq3333st4455u66v77777w88xy9999zabc00 /qn

**Note:** For more information, see the [knowledge base](#) article.

The following are available linking parameters:

Parameter	Description
NESSUS_OFFLINE_INSTALL	You can install the Tenable Nessus Agent on a system even if it is offline. Add the command line option <code>NESSUS_OFFLINE_INSTALL="yes"</code> to the command line input. The Tenable Nessus Agent will periodically attempt to link itself to either Tenable Vulnerability Management or Tenable Nessus Manager. If the agent cannot connect to the controller then it retries every hour, and if the agent can connect to the controller but the link fails then it retries every 24 hours.
NESSUS_SERVICE_AUTOSTART=false	Prevents the Tenable Nessus Agent from starting up after installation.  This parameter can be useful for streamlined deployment options (for example, deploying using a JSON file).
ADDLOCAL=ALL	Install the <a href="#">Tenable Nessus Agent System Tray Application</a> .
NESSUS_PLUGINS_FILEPATH-H="C:\path\to\plugins_set.tgz"	Install a full plugins set before linking to reduce the bandwidth impact during a mass installation. Add the command line option <code>NESSUS_PLUGINS_FILEPATH="C:\path\to\plugins_set.tgz"</code> where <i>plugins_set.tgz</i> is a recent plugins set tarball less than five days old. A stale plugins set older than five days will force a full plugins download to occur. You can download a recent plugins set from



	the <a href="#">Tenable downloads</a> page.
NESSUS_GROUPS	<p>Specify existing agent group or groups where you want to add the agent. If you do not specify an agent group during the install process, you can add your linked agent to an agent group later in Tenable Nessus Manager or Tenable Vulnerability Management.</p> <p><b>Note:</b> The agent group name is case-sensitive and must match exactly.</p> <p><b>Note:</b> Quotation marks (") are necessary when listing multiple groups, or one group with spaces in its name. For example:</p> <ul style="list-style-type: none"><li>• GroupName</li><li>• "Group Name"</li><li>• "Group, Another Group"</li></ul>
NESSUS_PROCESS_PRIORITY	Determine the priority of the agent relative to the priority of other tasks running on the system. For valid values and more information on how the setting works, see <a href="#">Agent CPU Resource Control</a> in the <i>Tenable Nessus Agent Deployment and User Guide</i> .
NESSUS_NAME	Specify the name for your agent. If you do not specify a name for your agent, the name defaults to the name of the computer where you are installing the agent.
NESSUS_CA_PATH	Specify a custom CA certificate to use to validate the manager's server certificate.
NESSUS_PROXY_SERVER	Specify the hostname or IP address of your proxy server.
NESSUS_PROXY_USERNAME	Specify the name of a user account that has per-



	missions to access and use the proxy server.
NESSUS_PROXY_PASSWORD	Specify the password of the user account that you specified as the username.
NESSUS_PROXY_AGENT	Specify the user agent name, if your proxy requires a preset user agent.

## Download Nessus Agent

On the [Nessus Agents Download Page](#), download the package specific to your operating system.

### Example: Nessus Agent package file

NessusAgent-<version number>-Win32.msi

Windows Server 7, and 8 (32-bit)

## Start Nessus Agent Installation

1. Navigate to the folder where you downloaded the Tenable Nessus Agent installer.
2. Next, double-click the file name to start the installation process. The **Welcome to the InstallShield Wizard for Nessus Agent** window appears.

## Complete the Windows InstallShield Wizard

**Note:** You may have to restart your computer to complete installation on Windows.

**Note:** If you want to include the system tray application in your installation, see [System Tray Application](#).

1. In the **Welcome to the InstallShield Wizard for Nessus Agent** window, click **Next** to continue.
2. In the **License Agreement** window, read the terms of the Tenable, Inc. Nessus software license and subscription agreement.
3. Click **I accept the terms of the license agreement**.
4. Click **Next**.



5. In the **Destination Folder** window, click **Next** to accept the default installation folder.

-or-

Click **Change** to browse and select a different folder where you want to install Tenable Nessus Agents.

6. In the **Configuration Options** window, type the **Agent Key** values:

Field	Value
Key	(Required) Use the value you <a href="#">retrieved</a> from the manager.
Server	(Required) Use the value you <a href="#">retrieved</a> from the manager. <ul style="list-style-type: none"><li>• To link to Tenable Vulnerability Management, enter <b>cloud.tenable.com:443</b>.</li><li>• To link to Tenable Nessus Manager, enter the IP/hostname of the manager with the appended port <b>8834</b>; for example, <b>192.0.2.0:8834</b>.</li></ul>
Groups	Specify existing agent group(s) where you want to add the agent.  If you do not specify an agent group during the installation process, you can later add your linked agent to an agent group.

**Note:** The agent name defaults to the name of the computer where you are installing the agent.

7. Click **Next**.

8. In the **Ready to Install the Program** window, click **Install**.

9. If presented with a **User Account Control** message, click **Yes** to allow the Tenable Nessus Agent to install.

10. In the **InstallShield Wizard Complete** window, click **Finish**.

**Note:** If you attempt to clone an Agent and link it to Tenable Nessus Manager or Tenable Vulnerability Management, a 409 error may appear. This error appears because another machine was linked with the same UUID value in the **HKLM/Software/Tenable/TAG** file. To resolve this issue, replace the value in the **HKLM/Software/Tenable/TAG** file with a valid UUIDv4 value.



**Note:** For more information about linking agents to Tenable Vulnerability Management, see [Link a Sensor](#) in the *Tenable.io User Guide*.

## Verify a Linked Agent

To verify a linked agent in Tenable Vulnerability Management:

1. In the upper-left corner, click the ☰ button.

The left navigation plane appears.

2. In the left navigation plane, click **Settings**.

The **Settings** page appears.

3. Click the **Sensors** tile.

The **Sensors** page appears. By default, **Nessus Scanners** is selected in the left navigation menu and the **Cloud Scanners** tab is active.

4. In the left navigation menu, click **Nessus Agents**.

The **Nessus Agents** page appears and the **Linked Agents** tab is active.

5. Locate the new agent in the linked agents table.

To verify a linked agent in Tenable Nessus Manager:

1. In the top navigation bar, click **Sensors**.

The **Linked Agents** page appears.

2. Locate the new agent in the linked agents table.



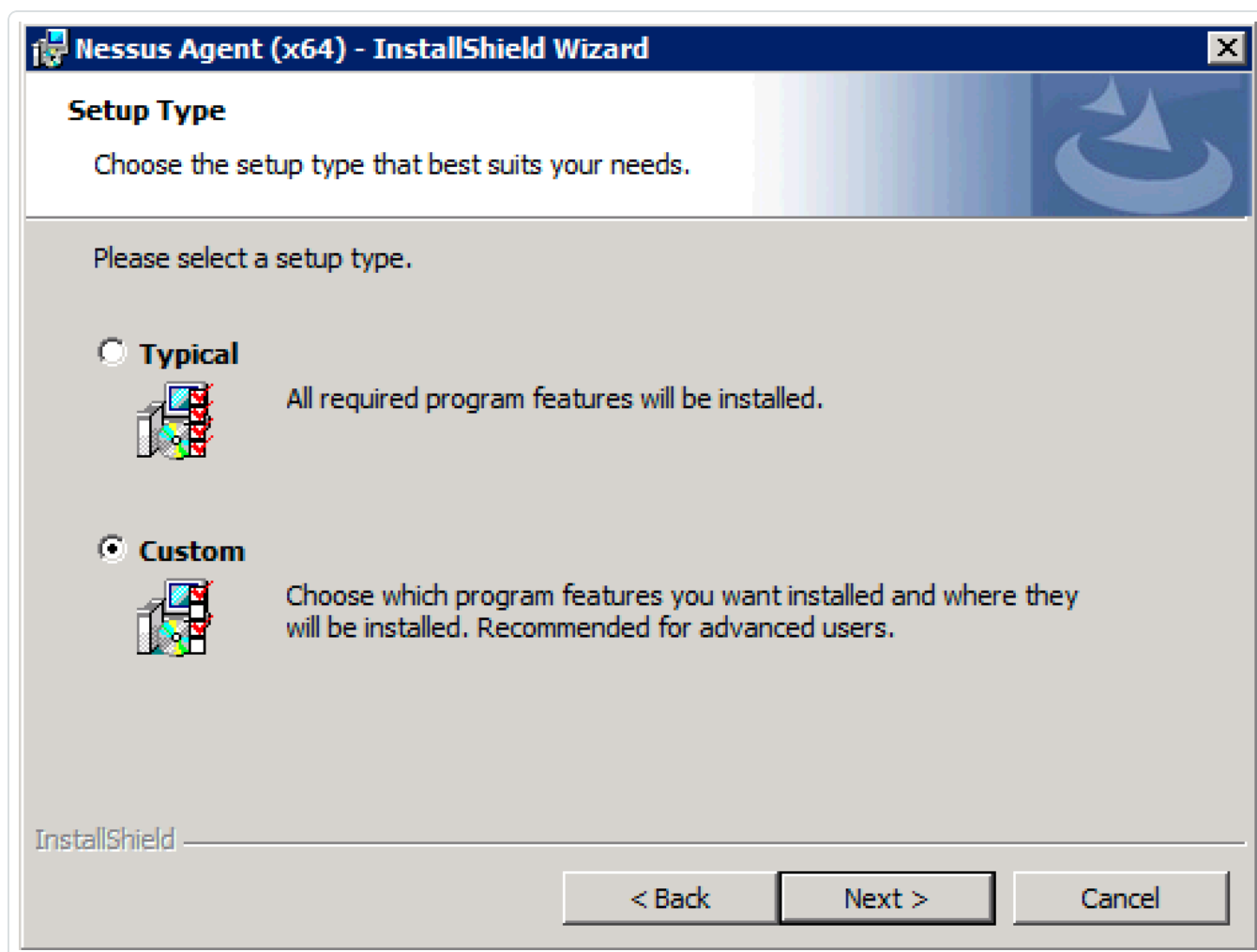
## Configure and View the System Tray Application

The Windows Tenable Nessus Agent installation package includes an optional system tray application that shows the status of your agent. You can enable the system tray application when you perform a custom installation.

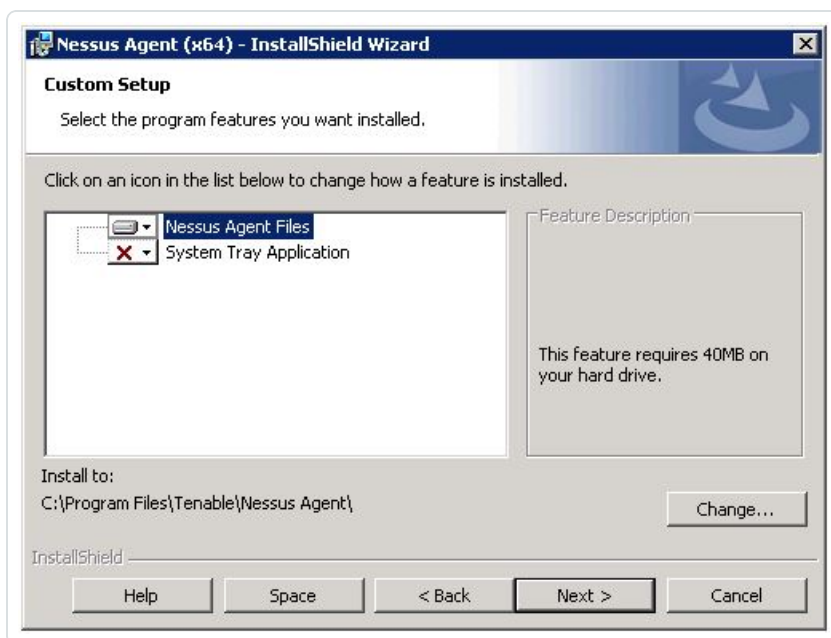
**Note:** You can also install the System Tray Application from the [command line](#).

To include the system tray application in your installation:

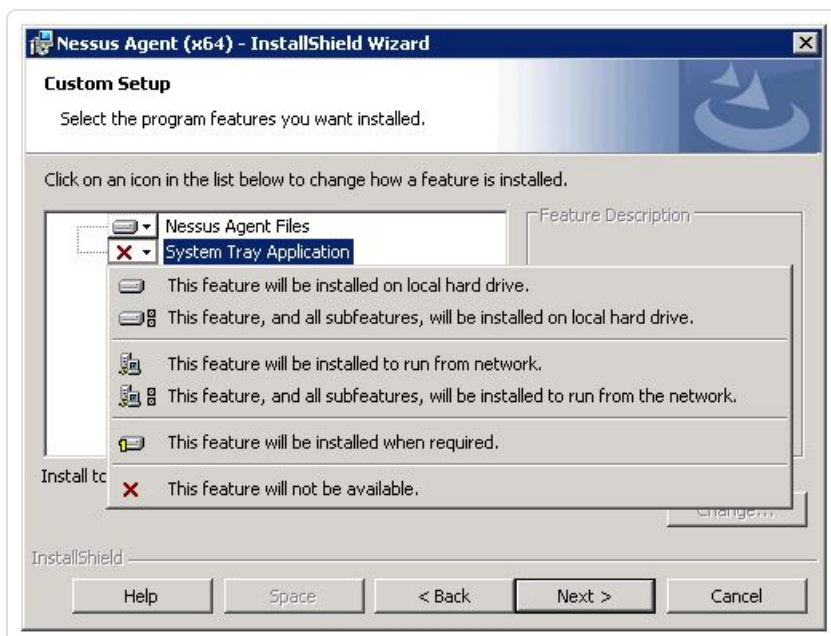
1. During the [installation process](#), on the **Setup Type** page, select **Custom**.



The **Custom Setup** page appears. By default, the system tray application is excluded from the installation package.



2. Click the **System Tray Application** drop-down box.




3. Click **This feature will be installed on local hard drive.**

4. Continue with the InstallShield Wizard to complete the installation.

To view the system tray application:





1. In your system tray, right-click the  button.
2. In the drop-down menu, click **About Tenable Nessus Agent**.

The **About Tenable Nessus Agent** dialog box appears, which displays the following information about your agent:

- **Nessus agent status:** Shows the state of the agent service (**Starting, Running, Paused, or Stopped**).
- **Scan status:** Shows whether the agent is running a scan job on the endpoint (**Scanning or Idle**).
- **Nessus agent version:** Shows the version of the agent running on the endpoint (for example, 7.0.3).
- **Pluginset ID:** Displays the current plugin set available on the endpoint (for example, 201802271350).





# Install a Nessus Agent on macOS

**Caution:** If you install a Tenable Nessus Agent on a system where an existing Tenable Nessus Agent, Tenable Nessus Manager, or Tenable Nessus scanner is running `nessusd`, the installation process kills all other `nessusd` processes. You may lose scan data as a result.

**Note:** Nessus Agents may need Full Disk Access when using some audits for full directory access. Therefore, Tenable recommends granting Full Disk Access to Nessus Agents installed on macOS.

## Before You Begin

- Retrieve the Nessus Agents linking key. For more information, see the [Tenable Nessus User Guide](#) or the [Tenable Vulnerability Management User Guide](#), depending on what manager you use.
- If you previously had the Tenable Nessus Agent installed on your system, see the [knowledge base](#) article on how to avoid linking errors.

## Download Nessus Agent

From the [Nessus Agents Download Page](#), download the package specific to your operating system.

### Example: Compressed Nessus Installer File

NessusAgent-<version number>.dmg

## Install Nessus Agent

**Note:** You need root privileges to perform the following steps.

To install the Tenable Nessus Agent, you can use either the GUI installation wizard or the command line.

### GUI Installation:

1. Double-click the Nessus Agent `.dmg` (macOS disk image) file.
2. Double-click `Install Nessus Agent.pkg`.
3. Complete the **Nessus Agent Install Wizard**.



## Command Line Installation:

1. Extract `Install Nessus Agent.pkg` and `.NessusAgent.pkg` from `NessusAgent-<version number>.dmg`.

**Note:** The `.NessusAgent.pkg` file is normally invisible in macOS Finder.

2. Open Terminal.
3. From the command line, enter the following command:

```
# sudo installer -pkg /<path-to>/Install Nessus Agent.pkg -target /
```

You can install a full plugins set before linking to reduce the bandwidth impact during a mass installation. You can accomplish this by using the `nessuscli agent update` command with the `--file` parameter, which specifies the location the plugins set. You must do this before [starting](#) the Tenable Nessus Agent. For example:

```
/opt/nessus_agent/sbin/nessuscli agent update --file=./plugins_set.tgz
```

The plugins set must be less than five days old. A stale plugin set older than five days forces a full plugins download to occur. You can download a recent plugin set from the [Nessus Agents download page](#).

## Link Agent Using Command Line Interface

To link an agent on macOS:

1. Open Terminal.
2. From the command line, use the `nessuscli agent link` command.

For example:

```
# sudo /Library/NessusAgent/run/sbin/nessuscli agent link  
--key=00abcd0000efgh1111i0k222lmopq3333st4455u66v77777w88xy9999zabc00  
--name=MyOSXAgent --groups=All --host=yourcompany.com --port=8834
```

The supported arguments for this command are:



Argument	Required?	Value
--key	yes	Use the values you <a href="#">retrieved</a> from the manager.
--host	yes	
--port	yes	
--name	no	Specify a name for your agent. If you do not specify a name for your agent, the name defaults to the name of the computer where you are installing the agent.
--groups	no	<p>Specify existing agent group or groups where you want to add the agent. If you do not specify an agent group during the install process, you can add your linked agent to an agent group later in Tenable Nessus Manager or Tenable Vulnerability Management.</p> <div style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> The agent group name is case-sensitive and must match exactly.</p></div>
--offline-install	no	<p>For Nessus Agents 7.0.3 or later, you can install the Tenable Nessus Agent on a system even if it is offline. Add the command line option <code>NESSUS_OFFLINE_INSTALL="yes"</code> to the command line input. The Tenable Nessus Agent periodically attempts to link itself to either Tenable Vulnerability Management or Tenable Nessus Manager.</p> <p>If the agent cannot connect to the controller then it retries every hour, and if the agent can connect to the controller but the link fails then it retries every 24 hours.</p>
--cloud	no	<p>Specify the <code>--cloud</code> argument to link to Tenable Vulnerability Management.</p> <p>The <code>--cloud</code> argument is a shortcut to specifying <code>--</code></p>



		<code>host=cloud.tenable.com --port=443.</code>
<code>--network</code>	<code>no</code>	For Tenable.io-linked agents, add the agent to a custom network. If you do not specify a network, the agent belongs to the default network.

**Note:** If you attempt to clone an agent and link it to Tenable Nessus Manager or Tenable Vulnerability Management, a 409 error may appear. This error appears because another machine was linked with the same UUID value in the `/private/etc/tenable_tag` file. To resolve this issue, replace the value in the `/private/etc/tenable_tag` file with a valid UUIDv4 value.

**Note:** For more information about linking agents to Tenable Vulnerability Management, see [Link a Sensor](#) in the *Tenable.io User Guide*.

## Verify a Linked Agent

To verify a linked agent in Tenable Vulnerability Management:

1. In the upper-left corner, click the  button.

The left navigation plane appears.

2. In the left navigation plane, click **Settings**.

The **Settings** page appears.

3. Click the **Sensors** tile.

The **Sensors** page appears. By default, **Nessus Scanners** is selected in the left navigation menu and the **Cloud Scanners** tab is active.

4. In the left navigation menu, click **Nessus Agents**.

The **Nessus Agents** page appears and the **Linked Agents** tab is active.

5. Locate the new agent in the linked agents table.

To verify a linked agent in Tenable Nessus Manager:



1. In the top navigation bar, click **Sensors**.

The **Linked Agents** page appears.

2. Locate the new agent in the linked agents table.



## Update a Nessus Agent

After you install an agent, its manager (either Tenable Vulnerability Management or Tenable Nessus Manager) automatically updates the agent software.

In either manager's user interface, you can set an agent update plan to determine the version that the agent automatically updates to. For more information, following the procedures described in the [Tenable Vulnerability Management](#) and [Tenable Nessus Manager](#) user guides.

### Manual Updates

In certain cases, such as air-gapped or Internet restricted networks, you may want to download application updates manually.

**Note:** By default, Tenable Vulnerability Management-linked agents update to the generally available (GA) version one week after the version is GA. Therefore, if you manually update a Tenable Vulnerability Management-linked agents to the latest version prior to that date, you should either disable automatic updates or set your update plan to opt in to Early Access releases. This ensures that the agent does not automatically downgrade to the previous version (GA).

To manually download agent updates:

1. Visit the [Tenable Downloads](#) page.
2. Click **Tenable Nessus Agents**.

The latest application update files for agents are available.

3. Click the application update file that you want to download.

The **License Agreement** window appears.

4. Click **I Agree**.

The download begins automatically.

Do one of the following, depending on your operating system:

#### Windows

**Note:** You need administrator-level privileges for the following steps.



Do one of the following:

- Double-click the .msi file you downloaded and follow the on-screen instructions.
- In the command line interface, enter the following command, using the location and file name of the package you downloaded:

```
> msixexec /i <path-to>\NessusAgent-<version>.msi /qn
```

## Linux

- In the command line interface, enter the following command, using the location and file name of the package you downloaded:

```
# yum upgrade <version>.rpm
```

or

```
# dpkg -i <path-to>/NessusAgent-<version>.deb
```

## macOS

- a. Mount the .dmg file you downloaded:

```
# sudo hdiutil attach <path-to>/NessusAgent-<version>.dmg
```

- b. Install the package:

```
# sudo installer -package /Volumes/Nessus\ Install/Install\ <path-to>/NessusAgent-<version>.dmg -target /
```

Your operating system installs Tenable Nessus Agent.





---

## Downgrade Nessus Agent

---

Tenable Nessus Agents support the ability to downgrade Tenable Nessus to a previous version of Tenable Nessus.

The following examples describe two scenarios: one scenario where you manually downgrade the agent software, and one scenario where the agent automatically downgrades because of your agent update plan setting.

### Example 1: Manually downgrade agent

#### Scenario:

You are currently running an Early Access release, 10.0.0, and now want to downgrade to the previous version, 8.3.0.

#### Solution:

1. Turn off automatic software updates by doing any of the following:
  - On Tenable Nessus Manager, disable the [advanced setting](#) **Automatically Download Agent Updates**, or `agent_updates_from_feed`.
  - On Tenable Vulnerability Management, enable the [agent setting](#) **Exclude all agents from software updates**.
  - On the agent, enable the [advanced setting](#) `disable_core_updates`.
2. [Uninstall](#) the agent.
3. Manually download and [install](#) the package of the previous version; in this example, Agent 8.3.0.

### Example 2: Agent automatically downgrades to align with your update plan

#### Scenario:

Your [agent update plan](#) determines what version Tenable Nessus Agent updates to, if you have automatic updates enabled. In this scenario, your update plan is set to `ga`, meaning the agent auto-



matically updates to the latest generally available (GA) release. You are currently on a GA version of Tenable Nessus Agent; for example, 10.0.0.

However, you change your update plan setting to `stable`, meaning the agent delays updates and stays on an older release.

**Result:**

According to your new agent update plan setting, your agent version should be an older release than the latest GA version (which you are currently on). Therefore, to align your agent version with this setting, the next time agent checks for an update, the agent automatically updates to be on an older version. Tenable Nessus Agent automatically downgrades to 8.3.0, one release before the latest GA version.



## Back Up Agent

Using [Nessus CLI Agent Commands](#), you can back up your Tenable Nessus Agent to restore it later on any system, even if it is a different operating system. When you back up Tenable Nessus Agent, your settings are preserved. Tenable Nessus Agent does not back up scan results.

**Note:** If you perform a cross-platform backup and restore between Linux and Windows systems, after you restore Tenable Nessus Agent, you must reconfigure any Tenable Nessus Agent configurations that use schedules (for example, scan schedules). Schedules do not transfer correctly across these platforms because the operating systems use different timezone names.

To back up Tenable Nessus Agent:

1. Access Tenable Nessus Agent from a command terminal.
2. Create the Tenable Nessus Agent backup file:

```
> nessuscli backup --create <backup_filename>
```

Tenable Nessus Agent creates the backup file in the following directory:

- Linux: /opt/nessus\_agent/var/nessus
- Windows: C:\ProgramData\Tenable\Nessus Agent\nessus\
- Mac: /Library/NessusAgent/run/var/nessus/

What to do next:

- [Restore Agent](#)



## Restore Agent

Using [Nessus CLI Agent Commands](#), you can use a previous backup of Tenable Nessus Agent to restore later on any system, even if it is a different operating system. When you back up Tenable Nessus Agent, you preserve your settings. Tenable Nessus Agent does not restore scan results.

**Note:** If you perform a cross-platform backup and restore between Linux and Windows systems, after you restore Tenable Nessus Agent, you must reconfigure any Tenable Nessus Agent configurations that use schedules (for example, scan schedules). Schedules do not transfer correctly across these platforms because the operating systems use different timezone names.

Before you begin:

- [Back Up Agent](#)

To restore Tenable Nessus Agent:

1. Access Tenable Nessus Agent from a command terminal.
2. [Stop](#) your Tenable Nessus Agent service.

For example:

```
# /sbin/service nessusagent stop
```

Tenable Nessus Agent terminates all processes.

3. Restore Tenable Nessus Agent from the backup file you previously saved:

```
> nessuscli backup --restore path/to/<backup_filename>
```

Tenable Nessus Agent restores your backup.

4. [Stop and start](#) your Tenable Nessus Agent service.

For example:

```
# /sbin/service nessusagent stop  
# /sbin/service nessusagent start
```

Tenable Nessus Agent begins initializing and uses settings from the backup.



---

## Remove Nessus Agent

---

This section includes information for uninstalling a Tenable Nessus Agent from hosts.

- [Uninstall a Nessus Agent on Linux](#)
- [Uninstall a Nessus Agent on Windows](#)
- [Uninstall a Nessus Agent on macOS](#)

**Note:** For instructions on how to remove an agent from a manager while leaving the agent installed on the host, see [Unlink an Agent](#).



---

# Uninstall a Nessus Agent on Linux

---

Before you begin:

- [Unlink the agent](#) from the manager.

To uninstall Tenable Nessus Agent on Linux:

1. Type the remove command specific to your Linux-style operating system.

## Example Nessus Agent Remove Commands

### Red Hat 6 and 7, CentOS 6 and 7, Oracle Linux 6 and 7

```
# yum remove Nessus
```

### Red Hat 8 and later, CentOS 8 and later, Oracle Linux 8 and later, Fedora, SUSE

```
# dnf remove Nessus
```

### Debian/Kali and Ubuntu

```
# dpkg -r NessusAgent
```

### FreeBSD

```
# pkg delete NessusAgent
```

What to do next:

- If you plan on reinstalling the Tenable Nessus Agent on the system, see the [knowledge base](#) article on how to avoid linking errors.



## Uninstall a Nessus Agent on Windows

Before you begin:

- [Unlink the agent](#) from the manager.

To uninstall Tenable Nessus Agent from the Windows user interface:

1. Navigate to the portion of Windows where you can **Add or Remove Programs** or **Uninstall or change a program**.
2. In the list of installed programs, select the **Tenable Nessus** product.
3. Click **Uninstall**.

A dialog box appears, prompting you to confirm your selection to remove Tenable Nessus Agent.

4. Click **Yes**.

Windows deletes all Nessus related files and folders.

To uninstall Tenable Nessus Agent from the Windows CLI:

1. Open PowerShell with administrator privileges.
2. Run the following command:

```
msiexec.exe /x <path to Nessus Agent package>
```

**Note:** For information about optional `msiexec /x` parameters, see [msiexec](#) in the Microsoft documentation.

What to do next:

- If you plan on reinstalling the Tenable Nessus Agent on the system, see the [knowledge base](#) article on how to avoid linking errors.



---

## Uninstall a Nessus Agent on macOS

---

Before you begin:

- [Unlink the agent](#) from the manager.

To uninstall Tenable Nessus Agent on macOS:

1. Remove the Nessus directories. From a command prompt, type the following commands:

- `$ sudo rm -rf /Library/NessusAgent`
- `$ sudo rm /Library/LaunchDaemons/com.tenablesecurity.nessusagent.plist`
- `$ sudo rm -r /Library/PreferencePanels/Nessus Agent Preferences.prefPane`

2. Disable the Nessus Agent service:

a. From a command prompt, type the following command:

```
$ sudo launchctl remove com.tenablesecurity.nessusagent
```

b. If prompted, provide the administrator password.

What to do next:

- If you plan on reinstalling the Tenable Nessus Agent on the system, see the [knowledge base](#) article on how to avoid linking errors.





---

## Manage Agents

---

To manage agents, see the following topics:

- [Start or Stop a Nessus Agent](#)
- [Filter Agents](#)
- [Export Agents](#)
- [Unlink an Agent](#)
- [Agent Groups](#)



## Start or Stop a Nessus Agent

You can temporarily stop an agent from gathering data and restart the agent to resume gathering data. Stopping and starting an agent can be helpful for troubleshooting. Tenable also recommends stopping the agent whenever you perform a [manual update](#).

The following sections describe best practices for starting and stopping a Nessus Agent on a host.

### Windows

1. Navigate to **Services**.
2. In the **Name** column, click **Tenable Nessus Agent**.
3. To stop the service, right-click **Tenable Nessus Agent**, and then click **Stop**.

-or-

To restart the Nessus Agent service, right-click **Tenable Nessus Agent**, and then click **Start**.

Start or Stop	Windows Command Line Operation
Start	<code>C:\Windows\system32&gt;net start "Tenable Nessus Agent"</code>
Stop	<code>C:\Windows\system32&gt;net stop "Tenable Nessus Agent"</code>

### Linux



Use the following commands:

Start or Stop	Linux Command Line Operation
RedHat, CentOS, and Oracle Linux	
Start	<code># /sbin/service nessusagent start</code>
Stop	<code># /sbin/service nessusagent stop</code>
SUSE	
Start	<code># /etc/rc.d/nessusagent start</code>



Start or Stop	Linux Command Line Operation
Stop	<code># /etc/rc.d/nessusagent stop</code>
FreeBSD	
Start	<code># service nessusagent start</code>
Stop	<code># service nessusagent stop</code>
Debian, Kali, and Ubuntu	
Start	<code># /etc/init.d/service nessusagent start</code>
Stop	<code># /etc/init.d/service nessusagent stop</code>

## macOS

1. Navigate to **System Preferences**.
2. Click the  button.
3. Click the  button.
4. Type your username and password.
5. To stop the Nessus Agent service, click the **Stop Nessus Agent** button.

-or-

To start the Nessus Agent service, click the **Start Nessus Agent** button.

Start or Stop	macOS Command Line Operation
Start	<code># sudo launchctl start com.tenablesecurity.nessusagent</code>
Stop	<code># sudo launchctl stop com.tenablesecurity.nessusagent</code>



## Agent Status

Tenable Nessus Agents can be in one of the following statuses:

Status	Description
Online	The host that contains the Tenable Nessus Agent is currently connected and in communication with Tenable Nessus Manager or Tenable Vulnerability Management.
Offline	The host that contains the Tenable Nessus Agent is currently powered down or not connected to a network.
Initializing	The Tenable Nessus Agent is in the process of checking in with Tenable Nessus Manager or Tenable Vulnerability Management.
Unlinked	<p>(Tenable Nessus Manager only) The agent is in an unlinked state.</p> <p>Agents with this status are only present if <b>Track unlinked agents</b> is enabled.</p> <div style="border: 1px solid blue; padding: 5px;"><p><b>Note:</b> Agents that are automatically unlinked via the <b>Unlink inactive agents after X days</b> setting can automatically relink to Tenable Nessus Manager if they come back online. You must manually relink agents that were manually unlinked.</p></div>



---

## Filter Agents

---

- To filter agents in Tenable Nessus Manager, see [Filter Agents](#) in the *Tenable Nessus User Guide*.
- To filter agents in Tenable Vulnerability Management, see [Filter Agents](#) in the *Tenable Vulnerability Management User Guide*.



---

## Export Agents

---

- To export agent data in Tenable Nessus Manager, see [Export Agents](#) in the *Tenable Nessus User Guide*.
- To export agent data in Tenable Vulnerability Management, see [Export Agents](#) in the *Tenable Vulnerability Management User Guide*.



---

## Unlink an Agent

---

When you manually unlink an agent, the agent disappears from the **Agents** page, but the system retains related data for the period of time specified in agent settings. When you manually unlink an agent, the agent does *not* automatically relink to either Tenable Nessus Manager or Tenable Vulnerability Management.

- To unlink an agent in Tenable Nessus Manager, see [Unlink an Agent](#) in the *Tenable Nessus User Guide*.
- To unlink an agent in Tenable Vulnerability Management, see [Unlink an Agent](#) in the *Tenable Vulnerability Management User Guide*.



---

## Agent Groups

---

You can use agent groups to organize and manage the agents linked to Tenable Nessus Manager or Tenable Vulnerability Management. You can add an agent to more than one group, and configure scans to use these groups as targets.

- To manage agent groups in Tenable Nessus Manager, see [Agent Groups](#) in the *Tenable Nessus User Guide*.
- To manage agent groups in Tenable Vulnerability Management, see [Agent Groups](#) in the *Tenable Vulnerability Management User Guide*.





---

## Scans

---

You can create and configure Tenable Nessus Agents scans in Tenable Nessus Manager and Tenable Vulnerability Management.

See the following topics:

- [Create an Agent Scan](#)
- [Agent Scan and Policy Templates](#)



---

## Create an Agent Scan

---

- To create an agent scan in Tenable Nessus Manager, see [Create an Agent Scan](#) in the *Tenable Nessus User Guide*.
- To create an agent scan in Tenable Vulnerability Management, see [Create a Scan](#) in the *Tenable Vulnerability Management User Guide*.



---

## Agent Scan and Policy Templates

---

- For information about agent templates in Nessus Manager, see [Scan Templates](#) in the *Nessus User Guide*.
- For information about agent templates in Tenable Vulnerability Management, see [Scan Templates](#) in the *Tenable Vulnerability Management User Guide*.



# Settings

---

You can configure Tenable Nessus Agent settings in Tenable Nessus Manager and Tenable Vulnerability Management.

See the following topics:

- [Modify Agent Settings](#) – Configure settings for Tenable Nessus Agents linked to Tenable Nessus Manager or Tenable Vulnerability Management.
- [Advanced Settings](#) – Configure advanced settings for Tenable Nessus Agents from the agent [command line interface](#).
- [Freeze Windows](#) – Create, modify, and delete freeze windows for Tenable Nessus Agents in Tenable Nessus Manager and Tenable Vulnerability Management.
- [log.json Settings](#) – Modify agent log settings in Tenable Nessus Manager.
- [Proxy Settings](#) – Configure an agent to connect to Tenable Nessus Manager or Tenable Vulnerability Management through a proxy.



---

## Modify Agent Settings

---

You can modify agent settings from Tenable Nessus Manager or Tenable Vulnerability Management, and you can use the `nessuscli` utility on the command line interface to modify certain settings.

- To modify agent settings in Tenable Vulnerability Management, see [Agent Settings](#) in the *Tenable Vulnerability Management User Guide*:
- To modify agent settings in Tenable Nessus Manager, see [Modify Agent Settings](#) in the *Tenable Nessus User Guide*.
- To modify agent settings from the `nessuscli` utility, see [Nessus CLI Agent Commands](#).



## Advanced Settings

You can manually configure agents by setting advanced settings from the agent [command line interface](#). You can modify some system-wide agent settings from [Nessus Manager Advanced Settings](#) or the **Linked Agents** tab in Tenable Vulnerability Management (see [Agent Settings](#) in the *Tenable Vulnerability Management User Guide* for more information). Nessus Agent validates your input values to ensure only valid configurations are allowed.

### Tenable Nessus Agent Advanced Settings

You can configure the following agent settings in the command line interface using the `nessuscli` utility.

Use the command `# nessuscli fix --set setting=value`. For more information, see [Nessus CLI Agent Commands](#).

**Tip:** Customers with many agents (10,000+) may want to configure the `agent_merge_audit_trail`, `agent_merge_kb`, `agent_merge_journal_mode`, and `agent_merge_synchronous_setting` settings. Modifying these settings can dramatically lower the amount of time it takes to merge agent scan results. Review the descriptions in the following table for suggested configurations.

Name	Setting	Description	Default	Valid Values
Agent Update Plan	<code>agent_update_channel</code>	(Tenable Vulnerability Management-linked agents only)  Sets the agent update plan to determine what version the agent automatically updates to.  <b>Note:</b> For agents linked to	<code>ga</code>	<code>ga</code> : Automatically updates to the latest Agent version when it is made generally available (GA). <b>Note:</b> This date is usually <i>one week after</i> the version is made generally available. For versions that address critical security issues, Tenable may make the version available



		<p>Tenable Vulnerability Management, you need to run the <code>agent_update_channel</code> command from the <code>agent nessuscli</code> utility. For agents linked to Tenable Nessus Manager, you need to run the <code>agent_update_channel</code> command from the Tenable Nessus Manager <code>nessuscli</code> utility.</p>		<p>immediately.</p> <p><b>ea:</b> Automatically updates to the latest Agent version as soon as it is released for Early Access (EA), typically a few weeks before general availability.</p> <p><b>stable:</b> Does not automatically update to the latest Tenable Nessus Agent version. Remains on an earlier version of Tenable Nessus Agent set by Tenable, usually one release older than the current generally available version, but no earlier than 7.7.0. When Tenable Nessus Agent releases a new version, your agent updates software versions, but stays on a version prior to the latest release.</p>
--	--	--	--	--



Always Validate SSL Server Certificates	strict_certificate_validation	When enabled, always validate SSL server certificates, even during initial remote link (requires manager to use a trusted root CA).	no	yes or no
Automatic Hostname Update	update_hostname	When enabled, when someone modifies the endpoint hostname, the new hostname is updated in the agent's manager. This feature is disabled by default to prevent custom agent names from being overridden.	no	yes or no
Connection Status Check Time	connection_status_check_time	(Tenable.io-linked agents only) Determines how often the agent checks its connection status when offline in seconds.	900	Integers >299
Days To Keep Unused Plugins	days_to_keep_unused_plugins	(Tenable.io-linked agents only) Determines the duration of time	14	Integers >7





		<p>(in days) after which an agent deletes an unused plugin set.</p> <p>For example, if you set this setting to <b>14</b> and the agent has not used one of its plugin set for scanning in over 14 days, the agent deletes that plugin set.</p>		
Detect Duplicate Agents	detect_duplicates	Regardless of this setting, the agent automatically checks if it is a duplicate agent by comparing its current list of MAC addresses to the MAC addresses the agent had at link time. For agents linked to Tenable.io or Tenable Nessus Manager 8.11.1 and later, the manager performs the same check to identify duplicate	no	yes or no



		<p>agents.</p> <p>When disabled, the agent automatically logs duplicates in <code>backend.log</code>, but no action is taken.</p> <p>When enabled, if either the agent or the manager detects a duplicate agent, the agent automatically unlinks and regenerates its identifying information (for example, the Tenable UUID) so that it can be linked again. This event is logged in <code>backend.log</code>. You must manually relink the agent.</p>		
Disable Core Updates	<code>disable_core_updates</code>	When set to <code>yes</code> , the agent does not request automatic core updates. You can still upgrade software versions manually. The agent can still	<code>no</code>	<code>yes or no</code>



		receive plugin updates.		
Long Term Upload Interval Seconds	long_term_upload_interval_seconds	(Tenable.io-linked agents only) Determines the number of seconds the agent waits between attempting to upload smart scan results.	180	Integers >59
Maximum Scans Per Day	maximum_scans_per_day	Determines the maximum number of scans an agent can run per day.	10	Integers 1-10
Minimum Metadata Update Interval	min_metadata_update_interval	(Tenable.io-linked agents only) Determines the minimum number of minutes between the agent's attempts to push metadata to Tenable Vulnerability Management. <b>Note:</b> The agent only attempts to push metadata to Tenable Vulnerability Management if the	10	Integers >4



		metadata changes.		
Nessus Dump File Max Files	dumpfile_max_files	Sets the maximum number of the <code>nessusd.dump</code> files kept on disk. If the number exceeds the specified value, the setting deletes the oldest dump file.	100	Integers 1-1000
Nessus Dump File Max Size	dumpfile_max_size	Sets the maximum size of the <code>nessusd.dump</code> files in MB. If file size exceeds the maximum size, the setting creates a new dump file.	512	Integers 1-2048
Offline Agent Scan Trigger Execution Threshold	offline_agent_scan_trigger_execution_threshold_days	(Tenable.io-linked agents only)  Determines the number of days of being offline after which rule-based scans no longer launch.	14	Integers >0
Plugin Compilation Performance	plugin_load_performance_mode	Sets plugin compilation performance, which affects	high	low, medium, or high



		<p>CPU usage. Low performance slows down plugin compilation, but reduces the agent's CPU consumption. Setting the performance to medium or high means that plugin compilation completes more quickly, but the agent consumes more CPU. For more information, see <a href="#">Agent CPU Resource Control</a>.</p>		
Scan Performance	scan_performance_mode	<p>Sets scan performance, which affects CPU usage. Low performance slows down scans, but reduces the agent's CPU consumption. Setting the performance to medium or high means that scans complete more</p>	high	low, medium, or high



		quickly, but the agent consumes more CPU. For more information, see <a href="#">Agent CPU Resource Control</a> .		
Skip Asset Observation On Update	skip_asset_observation_on_update	Determines whether the agent only updates the asset metadata when linking to Tenable Vulnerability Management. When you set this setting to <b>no</b> , the agent updates Tenable Vulnerability Management with new asset metadata based on the <a href="#">Minimum Metadata Update Interval</a> .	<b>no</b>	<b>yes or no</b>
SSL Cipher List	ssl_cipher_list	Sets the cipher list to use for Agent outbound connections.	<b>compatible</b>	<ul style="list-style-type: none"><li>• <b>legacy</b> – A list of ciphers that can integrate with older APIs.</li><li>• <b>compatible</b> – A list of secure ciphers. May</li></ul>



				<p>not include all the latest ciphers.</p> <ul style="list-style-type: none"><li>• <b>modern</b> – A list of the latest and most secure ciphers.</li><li>• <b>custom</b> – A custom OpenSSL cipher list. For more information on valid cipher list formats, see the <a href="#">OpenSSL documentation</a>.</li></ul>
SSL Mode	ssl_mode	Minimum supported version of TLS.	tls_1_2	<ul style="list-style-type: none"><li>• <b>compat</b> – TLS v1.0+.</li><li>• <b>ssl_3_0</b> – SSL v3+.</li><li>• <b>tls_1_1</b> – TLS v1.1+.</li><li>• <b>tls_1_2</b> – TLS v1.2+.</li></ul>

## Tenable Nessus Agent Secure Settings

You can configure the following secure settings in the command line interface, using the `nessuscli` utility.



Use the command `# nessuscli fix --secure --set setting=value`. For more information, see [Fix Secure Settings](#).

**Caution:** Tenable does not recommend changing undocumented `--secure` settings as it may result in an unsupported configuration.

Setting	Description	Valid Values
auto_proxy	<p>(Windows-only) If enabled, the agent uses Web Proxy Auto Discovery (WPAD) to obtain a Proxy Auto Config (PAC) file for proxy settings. This setting overrides all other proxy configuration preferences.</p> <p>If disabled, the agent defaults to the remaining proxy settings.</p>	true or false
ignore_proxy	<p>If enabled, the agent attempts a direct connection to the manager instead of using the set proxy, until it fails 10 times.</p> <p>If disabled, the agent attempts to connect using the set proxy, until it fails three times.</p> <p>This setting changes automatically, as described in <a href="#">Proxy Connection Fallback</a>. You can also set this setting manually; however, if at any point the agent meets one of the conditions described in <a href="#">Proxy Connection Fallback</a>, the agent automatically changes the setting.</p>	yes or no
ms_proxy	When enabled, the agent uses a proxy to connect to its manager.	true or false
proxy	The hostname or IP address of your proxy server.	String
proxy_port	The port number of the proxy server.	String
proxy_auth	(Optional) If you want to use authentication to connect to the proxy, specify the authentication scheme.	basic, digest, ntlm, or auto





proxy_user-name	If using authentication to connect to the proxy, the name of a user account that has permissions to access and use the proxy server.	String. If there are spaces, use quotes (").
proxy_password	If authenticating with the proxy, password associated with the username.	String

## Nessus Manager Advanced Settings

You can configure the following system-wide agent settings in Tenable Nessus Manager, under the **Agents & Scanners** section. For more information, see [Advanced Settings](#) in the *Nessus User Guide*.

Name	Setting	Description	Default	Valid Values
Agent Auto Delete	agent_auto_delete	Controls whether agents are automatically deleted after they have been inactive for the duration of time set for agent_auto_delete_threshold.	no	yes or no
Agent Auto Delete Threshold	agent_auto_delete_threshold	The number of days after which inactive agents are automatically deleted if agent_auto_delete is set to yes.	30	Integers 1-365



Name	Setting	Description	Default	Valid Values
Agent Auto Unlink	agent_auto_unlink	Controls whether agents are automatically unlinked after they have been inactive for the duration of time set for agent_auto_unlink_threshold.	no	yes or no
Agent Auto Unlink Threshold	agent_auto_unlink_threshold	The number of days after which inactive agents are automatically unlinked if agent_auto_unlink is set to yes. <div style="border: 1px solid blue; padding: 5px; margin-top: 10px;"><b>Note:</b> This value must be less than the agent_auto_delete_threshold.</div>	30	Integers 30-90
Agents Progress	agents_progress_viewable	When a scan gathers information from agents, Tenable Nessus Manager does not show detailed agents	100	Integers. If set to 0, this defaults to 100.



Name	Setting	Description	Default	Valid Values
		information if the number of agents exceeds this setting. Instead, a message indicates that results are being gathered and will be viewable when the scan is complete.		
Automatically Download Agent Updates	agent_updates_from_feed	When enabled, new Tenable Nessus Agent software updates are automatically downloaded.	yes	yes or no
Concurrent Agent Software Updates	cloud.manage.download_max	The maximum concurrent agent update downloads.	10	Integers
Include Audit Trail Data	agent_merge_audit_trail	Controls whether or not agent scan result audit trail data is included in the main agent database. Excluding audit trail data can significantly	false	true or false



Name	Setting	Description	Default	Valid Values
		<p>improve agent result processing performance.</p> <p>If this setting is set to false, the <b>Audit Trail Verbosity</b> setting in an individual scan or policy defaults to <b>No audit trail</b>.</p> <p>Available in Nessus 8.3 and later.</p>		
Include KB Data	agent_merge_kb	<p>Includes the agent scan result KB data in the main agent database. Excluding KB data can significantly improve agent result processing performance.</p> <p>If this setting is set to false, the <b>Include the KB</b> setting in an individual scan or policy defaults to <b>Exclude KB</b>.</p>	false	true or false



Name	Setting	Description	Default	Valid Values
		Available in Nessus 8.3 and later.		
Result Processing Journal Mode	agent_merge_journal_mode	<p>Sets the journaling mode to use when processing agent results. Depending on the environment, this can somewhat improve processing performance, but also introduces a small risk of a corrupted scan result in the event of a crash. For more details, refer to the <code>sqlite3</code> documentation.</p> <p>Available in Nessus 8.3 and later.</p>	DELETE	MEMORY TRUNCATE DELETE
Result Processing Sync Mode	agent_merge_synchronous_setting	<p>Sets the filesystem sync mode to use when processing agent results. Turning this off will significantly</p>	FULL	OFF NORMAL FULL



Name	Setting	Description	Default	Valid Values
		<p>improve processing performance, but also introduces a small risk of a corrupted scan result in the event of a crash. For more details, refer to the sqlite3 documentation.</p> <p>Available in Nessus 8.3 and later.</p>		
Track Unique Agents	track_unique_agents	<p>When enabled, Tenable Nessus Manager checks if MAC addresses of agents trying to link match MAC addresses of currently linked agents with the same hostname, platform, and distro. Tenable Nessus Manager deletes duplicates that it finds.</p>	no	yes or no



---

# Freeze Windows

---

Agent freeze windows behave differently in Tenable.io and Nessus Manager.

## **In Tenable.io:**

Freeze windows allow you to schedule times when Tenable.io suspends certain agent activities for all linked agents. This activity includes:

- Receiving and applying software updates

Freeze windows do not prevent linked agents from:

- Receiving plugin updates
- Installing or executing agent scans

## **In Nessus Manager:**

Freeze windows allow you to schedule times when Nessus Manager suspends certain agent activities for all linked agents. This activity includes:

- Receiving and applying software updates
- Receiving plugin updates
- Installing or executing agent scans

To manage freeze windows, use the procedures described in the [Nessus](#) and [Tenable.io](#) user guides.



---

## Modify log.json Settings

---

For information on modifying Tenable Nessus Agent log.json settings, see the **Modify log.json** section of [Modify Log Settings](#) in the *Nessus User Guide*.

**Note:** Documentation related to `www.server.log` does not apply to Tenable Nessus Agents.





---

## Proxy Settings

---

### Configure Proxy Settings

You can configure an agent to connect to its manager (Tenable Nessus Manager or Tenable Vulnerability Management) via a proxy in one of the following ways:

- During initial installation and linking.

For more information, see the [Nessus CLI Agent Commands](#) linking command proxy settings.

- After you have installed and linked.

After initial linking, you can configure a proxy or change existing proxy settings via the command line. For more information, see [Tenable Nessus Agent Secure Settings](#).

### Proxy Connection Fallback

If an agent is using a proxy to connect to its manager, there is a built-in proxy fallback in case of a connection failure.

The automatic fallback process happens as follows:

1. If the agent is unable to access its manager through the proxy, and fails three times in a row, the agent tries connecting directly to the manager.
2. If the agent successfully connects directly to the manager, the agent automatically sets the [secure setting](#) `ignore_proxy` to `yes`. When you enable this setting, the agent will connect directly to the manager on future attempts, instead of using the proxy.
3. However, if the agent fails to connect directly to the manager 10 times in a row, the agent retries connecting via the proxy again. If the agent successfully connects via the proxy, the agent automatically sets `ignore_proxy` to `no`, meaning the agent will connect using the proxy on future attempts.
4. The process repeats as needed, depending on whether the agent fails to connect to the proxy or directly to the manager.

At any point, you can manually change the [secure setting](#) `ignore_proxy` to `yes` or `no` to interrupt the automatic fallback process. This forces the agent to attempt to connect either directly or via the proxy, depending on what you set. However, if at any point the agent meets one of the



conditions listed above (for example, fails to connect via proxy three times in a row), the automatic fallback process resumes.



---

## Additional Resources

---

This section contains the following resources:

- [Mass Deployment Support](#)
- [Create Windows or Linux Master Image with Tenable Nessus Agent Installed](#)
- [Manage Logs](#)
- [Nessus Service](#)
- [Nessus CLI Agent Commands](#)
- [Plugin Updates](#)
- [Rule-based Trigger File Location](#)



---

## Mass Deployment Support

---

You can automatically configure and deploy agents using environment variables or a configuration JSON file. This allows you to streamline a mass deployment.

When you first launch the agent after installation, the agent first checks for the presence of environment variables, then checks for the `config.json` file. When the agent launches for the first time, the agent uses that information to link to a manager and set preferences.

**Note:** If you have information in both environment variables and `config.json`, the agent uses both sources of information. If there is conflicting information (for example, environment variables and `config.json` contain a different linking key), the agent uses the information from the environment variables.

For more information, see:

- [Environment Variables](#)
- [Deploy Tenable Nessus Agent Using JSON](#)



---

## Environment Variables

---

If you want to configure based on environment variables, you can set the following environment variables in the shell environment that is running in.

When you first launch after installation, first checks for the presence of environment variables, then checks for the [config.json](#) file.

### Linking configuration

Use the following environment variables for linking configuration:

- `NCONF_LINK_HOST` - The hostname or IP address of the manager you want to link to. To link to Tenable.io, use `cloud.tenable.com`.
- `NCONF_LINK_PORT` - Port of the manager you want to link to.
- `NCONF_LINK_NAME` - Name of the to use when linking.
- `NCONF_LINK_KEY` - Linking key of the manager you want to link to.
- `NCONF_LINK_CERT` - (Optional) CA certificate to use to validate the connection to the manager.
- `NCONF_LINK_RETRY` - (Optional) Number of times should retry linking.
- `NCONF_LINK_GROUPS` - (Optional) One or more existing agent groups where you want to add the agent. If you do not specify an agent group during the install process, you can add your linked agent to an agent group later in Tenable Nessus Manager or Tenable Vulnerability Management. List multiple groups in a comma-separated list. If any group names have spaces, use quotes around the whole list. For example: `"Atlanta,Global Headquarters"`



## Deploy Tenable Nessus Agent Using JSON

When you first launch the agent after installation, the agent first checks for the presence of [environment variables](#), then checks for the `config.json` file. When the agent launches for the first time, the agent uses that information to link to a manager and set preferences.

To deploy Nessus Agent with the `config.json` file:

1. Configure the `config.json` file.

**Note:** `config.json` must be in ASCII format. Some tools, such as PowerShell, create test files in other formats by default.

Example Agent `config.json` file format:

```
{
  "link": {
    "name": "sensor name",
    "host": "hostname or IP address",
    "port": 443,
    "key": "abcdefghijklmnopqrstuvwxy",
    "ms_cert": "CA certificate for linking",
    "retry": 1,
    "proxy": {
      "proxy": "proxyhostname",
      "proxy_port": 443,
      "proxy_username": "proxyusername",
      "proxy_password": "proxypassword",
      "user_agent": "proxyagent",
      "proxy_auth": "NONE"
    }
  },
  "preferences": {
    "global.max_hosts": "500"
  }
}
```



```
}
```

Example Agent `config.json` file format (when using `auto_proxy`):

```
{
  "link": {
    "name": "sensor name",
    "host": "hostname or IP address",
    "port": 443,
    "key": "abcdefghijklmnopqrstuvwxy",
    "ms_cert": "CA certificate for linking",
    "retry": 1,
    "proxy": {
      "proxy": "proxyhostname",
      "proxy_port": 443,
      "auto_proxy": "true"
    }
  }
}
```

## `config.json` Details

The following describes the format of the different settings in each section of `config.json`.

**Note:** All sections are optional; if you do not include a section, it is not configured when you first launch the agent. You can manually configure the settings later.

### Linking

The `link` section sets preferences to link the agent to a manager.

Setting	Description
<code>name</code>	(Optional) A name for the scanner.



	<p>A name for your agent. If you do not specify a name for your agent, the name defaults to the name of the computer where you are installing the agent.</p>
host	<p>The hostname or IP address of the manager you want to link to.</p> <p>To link to Tenable.io, use cloud.tenable.com.</p>
port	<p>The port for the manager you want to link to.</p> <p>For Tenable Nessus Manager: 8834 or your custom port.</p> <p>For Tenable Vulnerability Management: 443</p>
key	<p>The linking key that you retrieved from the manager.</p>
network	<p>(Optional, Tenable Vulnerability Management-linked agents only)</p> <p>The custom <a href="#">network</a> you want to link to. If you do not specify a network, the agent belongs to the default network.</p>
ms_cert	<p>(Optional)</p> <p>A custom CA certificate to use to validate the manager's server certificate.</p>
groups	<p>(Optional)</p> <p>One or more existing scanner groups where you want to add the scanner. List multiple groups in a comma-separated list. If any group names have spaces, use quotes around the whole list.</p> <p>For example: "Atlanta,Global Headquarters"</p> <p>One or more existing agent groups where you want to add the agent. If you do not specify an agent group during the install process, you can add your linked agent to an agent group</p>





	<p>later in Tenable Nessus Manager or Tenable Vulnerability Management.</p> <p>List multiple groups in a comma-separated list. If any group names have spaces, use quotes around the whole list.</p> <p>For example: "Atlanta,Global Headquarters"</p> <div data-bbox="618 485 1479 600"><p><b>Note:</b> The agent group name is case-sensitive and must match exactly.</p></div>
retry	<p>(Optional)</p> <p>The number of times the agent attempts to link to the manager if it fails the first attempt.</p> <p>If you do not include the <code>retry</code> preference, the agent does not attempt to link after the first failure. The maximum accepted value is 10.</p> <div data-bbox="618 989 1479 1304"><p><b>Note:</b> If you set <code>retry</code> to 1, the agent tries to link to the manager 30 seconds after the initial failure. Every proceeding retry occurs twice as long after the prior retry. For example, if you set <code>retry</code> to 5, the agent attempts to link 30 seconds after the first failure, 60 seconds after the second failure, 120 seconds after the third failure, 240 seconds after the fourth failure, and 480 seconds after the fifth failure.</p></div>
proxy	<p>(Optional)</p> <p>If you are using a proxy server, include the following:</p> <ul data-bbox="662 1482 1463 1759" style="list-style-type: none"><li>• <code>proxy</code>: The hostname or IP address of your proxy server.</li><li>• <code>proxy_port</code>: The port number of the proxy server.</li><li>• <code>auto_proxy</code> (Windows only): If enabled, the agent uses Web Proxy Auto Discovery (WPAD) to obtain a Proxy</li></ul>



	<p>Auto Config (PAC) file for proxy settings. This setting overrides all other proxy configuration preferences. If disabled, the agent defaults to the remaining proxy settings.</p> <div data-bbox="695 390 1479 541" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> If you include <code>auto_proxy</code> in your configuration file, you must also provide the <code>proxy</code> and <code>proxy_port</code> parameters.</p></div> <ul style="list-style-type: none"><li>• <code>proxy_username</code>: The name of a user account that has permissions to access and use the proxy server.</li><li>• <code>proxy_password</code>: The password of the user account that you specified as the username.</li><li>• <code>user_agent</code>: The user agent name, if your proxy requires a preset user agent.</li><li>• <code>proxy_auth</code>: The authentication method to use for the proxy.</li></ul>
<code>aws_scanner</code>	<p>(Optional)</p> <p>Set <code>aws_scanner</code> to <b>true</b> to link the Tenable Nessus scanner as an AWS scanner.</p> <div data-bbox="613 1251 1479 1360" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> The Tenable Nessus scanner must already be running on an AWS instance for the flag to take effect.</p></div>

## Preferences

The preferences section configures any advanced settings. For more information, see [Advanced Settings](#).

2. [Download](#) the Tenable Nessus Agent package.
3. (Windows only) Before you install the package, you must modify the package so that Nessus Agent does not start automatically after installation. This is because Nessus Agent must read



the `config.json` file when you start the Nessus Agent service for the first time.

To modify the package, run the following command:

```
msiexec /i <Nessus Agent package>.msi NESSUS_SERVICE_AUTOSTART=false /qn
```

4. [Install Tenable Nessus Agent.](#)

5. (macOS only) Unlike Windows, there is no way to turn off autostart before installing Nessus Agent. Therefore, you need to reset the Nessus Agent to a fresh state before adding `config.json` and starting the Nessus Agent service.

To return Nessus Agent to a fresh state on macOS, validate `config.json`, and place `config.json` in the correct directory, run the following command:

```
/Library/NessusAgent/run/sbin/nessuscli prepare-image --json=<path to json file>
```

**Note:** Nessus Agent autostart is disabled by default in Linux packages. Therefore, if you are using Linux, you can ignore steps 3 and 5.

6. Place `config.json` in the Nessus Agent directory if it is not already there:

- Linux – `/opt/nessus_agent/var/nessus/config.json`
- Windows – `C:\ProgramData\Tenable\Nessus Agent\nessus\config.json`
- macOS – `/Library/NessusAgent/run/var/nessus/config.json`

7. [Start the Nessus Agent service.](#)

8. Depending on your operating system, run the following command to verify the `config.json` preferences:

- Linux – `/opt/nessus_agent/sbin/nessuscli fix --secure --list`
- Windows – `"C:\Program Files\Tenable\Nessus Agent\nessuscli.exe" fix --secure --list`
- macOS – `/Library/NessusAgent/run/sbin/nessuscli fix --secure --list`

Once you verify that the preferences were successfully applied, the linking process is complete.



# Create Windows or Linux Master Image with Tenable Nessus Agent Installed

When creating a master image for Windows or Linux, you may include an agent installation. However, there are files and registry settings that you must set per host. By removing and changing files, the agent generates new files once the agent reboots. If the host is imaged with these files and you attempt to link several imaged agents, you receive a [409 UUID error](#).

You only need to perform the following steps if the agent used in the image is already linked to Tenable Vulnerability Management or Tenable Nessus Manager.

**Note:** The following steps require administrative or root privileges.

**Note:** The process for preparing an agent to use in a golden image is different starting with Agent version 8.3.0.

## To create a master image using Agent 8.3+:

Nessus Agent 8.3.0 introduced a new `nessuscli` utility called `prepare-image`. This command creates a new agent installation to use in a machine/golden image (see the [Nessus Agent CLI guide](#) for more information).

1. [Stop the agent service](#).
2. Run the `prepare-image` command (using Linux syntax as an example):

```
./nessuscli prepare-image
```

**Note:** Do not restart the agent service on the host until you have taken the image. Restarting the agent service regenerates the UUIDs, tags, and files that the `prepare-image` command has purged.

The agent install should be ready to use in a machine image.

## To create a master image using Agent 8.2.5 and older on Windows:

Nessus agents older than 8.3.0 do not include the `prepare-image` utility, so you must manually prepare the agent to be used in a machine image. Do not restart the agent service until you have taken the image.



**Caution:** Back up the Windows Registry before proceeding.

1. [Stop the agent service.](#)
2. Run the agent unlink command:

```
nessuscli.exe agent unlink
```

3. Open the Registry Editor on the agent host.
4. Remove the Tenable tag file by deleting following registry key:

```
HKLM\SOFTWARE\Tenable\TAG
```

5. Delete the agent UUID from the file system:

```
C:\ProgramData\Tenable\Nessus Agent\nessus\uuid
```

The agent installation is ready to use in a machine image.

### To create a master image using Agent 8.2.5 and older on Linux:

Nessus agents older than 8.3.0 do not include the `prepare-image` utility, so you must manually prepare the agent to use in a machine image. Do not restart the agent service until you have taken the image.

1. [Stop the agent service.](#)
2. Run the agent unlink command:

```
./nessuscli agent unlink
```

3. Remove the Tenable tag and UUID files:

```
rm /etc/tenable_tag /opt/nessus_agent/var/nessus/uuid
```

The agent installation is ready to use in a machine image.

More resources:



- [Deploy Nessus Agent Using JSON](#)
- [Mass Deployment Support](#)



## Manage Logs

Tenable Nessus Agent has the following default log files:

- `nessusd.dump` – Nessus Agent dump log file used for debugging output.  
To configure `nessusd.dump`, see [Advanced Settings](#).
- `nessusd.messages` – Nessus Agent message log.  
To configure `nessusd.messages`, see [Advanced Settings](#).
- `backend.log` – Nessus Agent backend log.  
To configure `backend.log`, see [Modify backend.log](#).
- `nessuscli.log` – Nessus Agent CLI log.

### Default Log Locations

The following table describes the default log file locations for each operating system.

Operating System	Log Location
Windows	<code>C:\ProgramData\Tenable\Nessus Agent\nessus\logs</code>
Linux	<code>/opt/nessus_agent/var/nessus/logs</code>
macOS	<code>/Library/NessusAgent/run/var/nessus/logs</code>

### Modify backend.log

You can configure log locations and rotation strategies for `backend.log` by editing the `log.json` file. You can also configure custom logs by creating a new `reporters[x].reporter` section and creating a custom file name.

1. Using a text editor, open the `log.json` file, located in the corresponding directory:
  - **Windows** – `C:\ProgramData\Tenable\Nessus Agent\nessus\log.json`
  - **Linux** – `/opt/nessus_agent/var/nessus/log.json`
  - **macOS** – `/Library/NessusAgent/run/var/nessus/log.json`



2. For `backend.log`, edit or create a `reporters[x].reporter` section, and add or modify the following parameters:

Parameter	Default value	Can be modified?	Description
<code>tags</code>	<code>log, info, warn, error, trace</code>	yes	Determines what log information the log includes. <ul style="list-style-type: none"><li>• <code>response</code> – Web server activity logs</li><li>• <code>info</code> – Informational logs for a specific task</li><li>• <code>warn</code> – Warning logs for a specific task</li><li>• <code>error</code> – Error logs for a specific task</li><li>• <code>debug</code> – Debugging output</li><li>• <code>verbose</code> – debugging output with more information than <code>debug</code></li><li>• <code>trace</code> – Logs used to trace output</li></ul>





<code>type</code>	<code>file</code>	not recommended	Determines the type of the log file.
<code>rotation_strategy</code>	<code>size</code>	yes	Determines whether the log archives files based on maximum rotation size or rotation time.  Valid values: <ul style="list-style-type: none"><li>• <code>size</code> – Rotate the log based on size, as specified in <code>max_size</code>.</li><li>• <code>daily</code> – Rotate the log based on time, as specified in <code>rotation_time</code>.</li></ul>
<code>rotation_time</code>	86400 (1 day)	yes	Rotation time in seconds.  Only used if <code>rotation_strategy</code> is <code>daily</code> .
<code>max_size</code>	10485760 (10 MB)	yes	Rotation size in bytes.  Only used if <code>rotation_strategy</code> is <code>size</code> .
<code>max_files</code>	2	yes	Maximum number of



			<p>files allowed in the file rotation.</p> <p>The maximum number includes the main file, so 10 <code>max_files</code> is 1 main file and 9 backups. If you decrease this number, Tenable Nessus deletes the old logs.</p>
<code>file</code>	Depends on operating system and log file	yes	<p>The location and name of the log file.</p> <p>If you change the name of a default Tenable Nessus Agent log file, some advanced settings may not be able to modify the log settings.</p>
<code>context</code>	<code>true</code>	not recommended	Enables more context information for logs.
<code>format</code>	<code>system</code>	not recommended	<p>Determines the format of the output.</p> <ul style="list-style-type: none"><li>• <code>combined</code> – Presents output in a format used for web server logs.</li><li>• <code>system</code> –</li></ul>



			Presents output in the default operating system log format.
--	--	--	---

3. Save the `log.json` file.
4. [Restart](#) the Nessus Agent service.

The Nessus Agent updates the log settings.



## Nessus Service

If necessary, whenever possible, Nessus services should be started and stopped using Nessus service controls in the operating system's interface.

However, there are many **nessus-service** functions that can be performed through a command line interface.

Unless otherwise specified, the **nessusd** command can be used interchangeably with **nessus-service** server commands.

The # **killall nessusd** command is used to stop all Nessus services and in-process scans.

**Note:** All commands must be run by a user with administrative privileges.

## Nessus-Service Syntax

Operating System	Command
Linux	# /opt/nessus/sbin/nessus-service [-vhD][ <b>-c</b> <config-file>][ <b>-p</b> <port-number>][ <b>-a</b> <address>][ <b>-S</b> <ip[,ip,...]>]
FreeBSD	# /usr/local/nessus/sbin/nessus-service [-vhD][ <b>-c</b> <config-file>][ <b>-p</b> <port-number>][ <b>-a</b> <address>][ <b>-S</b> <ip[,ip,...]>]
macOS	# /Library/Nessus/run/sbin/nessus-service [-vhD][ <b>-c</b> <config-file>][ <b>-p</b> <port-number>][ <b>-a</b> <address>][ <b>-S</b> <ip[,ip,...]>]

## Suppress Command Output Examples

You can suppress command output by using the **-q** option.

Linux

```
# /opt/nessus/sbin/nessus-service -q -D
```

FreeBSD

```
# /usr/local/nessus/sbin/nessus-service -q -D
```



## Nessusd Commands

Option	Description
-c <config-file>	When starting the nessusd server, this option is used to specify the server-side nessusd configuration file to use. It allows for the use of an alternate configuration file instead of the standard db.
-S <ip [ip2,...]>	When starting the nessusd server, force the source IP of the connections established by Nessus during scanning to <ip>. This option is only useful if you have a multihomed machine with multiple public IP addresses that you would like to use instead of the default one. For this setup to work, the host running nessusd must have multiple NICs with these IP addresses set.
-D	When starting the nessusd server, this option forces the server to run in the background (daemon mode).
-v	Display the version number and exit.
-l	Display a list of those third-party software licenses.
-h	Show a summary of the commands and exit.
--ipv4-only	Only listen on IPv4 socket.
--ipv6-only	Only listen on IPv6 socket.
-q	Operate in "quiet" mode, suppressing all messages to stdout.
-R	Force a re-processing of the plugins.
-t	Check the time stamp of each plugin when starting up to only compile newly updated plugins.
-K	Set a master password for the scanner.  If a master password is set, Nessus encrypts all policies and credentials contained in the policy. When a password is set, the Nessus UI prompts you for the password.  If your master password is set and then lost, it cannot be recovered by your administrator nor Tenable Support.



## Notes

If you are running `nessusd` on a gateway and if you do not want people on the outside to connect to your `nessusd`, set your `listen_address` advanced setting.

To set this setting:

```
nessuscli fix --set listen_address=<IP address>
```

This setting tells the server to only listen to connections on the address `<address>` that is an IP address, not a machine name.



# Nessus CLI Agent Commands

Use the Agent `nessuscli` utility to perform some Tenable Nessus Agent functions through a command line interface.

**Note:** You must run all Agent `nessuscli` commands as a user with administrative privileges.

## Nessuscli Syntax

Operating System	Command
Linux	<code># /opt/nessus_agent/sbin/nessuscli &lt;cmd&gt; &lt;arg1&gt; &lt;arg2&gt;</code>
Windows	<code>C:\Program Files\Tenable\Nessus Agent\nessuscli.exe &lt;cmd&gt; &lt;arg1&gt; &lt;arg2&gt;</code>
macOS	<code># sudo /Library/NessusAgent/run/sbin/nessuscli &lt;cmd&gt; &lt;arg1&gt; &lt;arg2&gt;</code>

## Nessuscli Commands

Command	Description
Informational Commands	
<code># nessuscli help</code>	Displays a list of <code>nessuscli</code> commands.
<code># nessuscli -v</code>	Displays your current version of Tenable Nessus Agent.
Bug Reporting Commands	
<code># nessuscli bug-report-generator</code>	<p>Generates an archive of system diagnostics.</p> <p>If you run this command without arguments, the utility prompts you for values.</p> <p><b>Optional arguments:</b></p> <ul style="list-style-type: none"><li><code>--quiet</code> – Run the bug report generator without prompting user for feedback.</li></ul>



Command	Description
	<ul style="list-style-type: none"><li>• <code>--scrub</code> – The bug report generator sanitizes the last two octets of the IPv4 address.</li><li>• <code>--full</code> – The bug report generator collects extra data.</li></ul>
Image Preparation Commands	
<pre># nessuscli pre-prepare-image</pre>	<p>Performs pre-imaging cleanup, including the following:</p> <ul style="list-style-type: none"><li>• Unlinks the agent, if linked.</li><li>• Deletes any host tag on the agent. For example, the registry key on Windows or <code>tenable_tag</code> on Unix.</li><li>• Deletes any UUID file on the agent. For example, <code>/opt/nessus/var/nessus/uuid</code> (or equivalent on MacOS/Windows).</li><li>• Deletes <code>plugin dbs</code>.</li><li>• Deletes <code>global db</code>.</li><li>• Deletes <code>master.key</code>.</li><li>• Deletes the backups directory.</li></ul> <p><b>Optional arguments:</b></p> <ul style="list-style-type: none"><li>• <code>--json=&lt;file&gt;</code> – Validates an auto-configuration <code>.json</code> file and places it in the appropriate directory.</li></ul>
Local Agent Commands	
Used to link, unlink, and display agent status	
<pre># nessuscli agent link --key=&lt;key&gt; --host=&lt;host&gt; --port=&lt;port&gt;</pre>	<p>Using the <a href="#">Nessus Agent Linking Key</a>, this command links the agent to the Tenable Nessus Manager or Tenable Vulnerability Management.</p> <p><b>Required arguments:</b></p> <ul style="list-style-type: none"><li>• <code>--key</code> – The linking key that you <a href="#">retrieved</a> from the manager.</li></ul>





Command	Description
	<ul style="list-style-type: none"><li>• <code>--host</code> – To link to Tenable Nessus Manager: The static IP address or hostname you set during the Tenable Nessus Manager installation. To link to Tenable Vulnerability Management: <code>sensor.cloud.tenable.com</code> (for Tenable Nessus Agents 8.0.x and earlier, <code>cloud.tenable.com</code>)</li></ul> <div data-bbox="574 558 1479 835" style="border: 1px solid #0070C0; padding: 10px;"><p><b>Note:</b> Starting with Tenable Nessus Agent 8.1.0, Tenable Vulnerability Management-linked agents communicate with Tenable Vulnerability Management using <code>sensor.cloud.tenable.com</code>. If agents are unable to connect to <code>sensor.cloud.tenable.com</code>, they use <code>cloud.tenable.com</code> instead. Agents with earlier versions continue to use the <code>cloud.tenable.com</code> domain.</p></div> <ul style="list-style-type: none"><li>• <code>--port</code> – To link to Tenable Nessus Manager: 8834 or your custom port. To link to Tenable Vulnerability Management: 443</li></ul> <p><b>Tenable Vulnerability Management arguments:</b></p> <ul style="list-style-type: none"><li>• <code>--cloud</code>– To link to Tenable Vulnerability Management, pass the argument <code>--cloud</code>.</li></ul> <p>The <code>--cloud</code> argument is a shortcut to specifying <code>--host=t=sensor.cloud.tenable.com --port=443</code>. If you use <code>--cloud</code>, you do not need to set <code>--host</code> and <code>--port</code>.</p> <p><b>Optional arguments:</b></p> <ul style="list-style-type: none"><li>• <code>--auto-proxy</code> – (Windows-only) When set, the agent uses Web Proxy Auto Discovery (WPAD) to obtain a Proxy Auto Config (PAC) file for proxy settings. This setting overrides all other proxy configuration preferences.</li><li>• <code>--name</code> – A name for your agent. If you do not specify a name for your agent, the name defaults to the name of the computer where you are installing the agent.</li></ul>



Command	Description
	<ul style="list-style-type: none"><li>• <code>--groups</code> – One or more existing agent groups where you want to add the agent. If you do not specify an agent group during the install process, you can add your linked agent to an agent group later in Tenable Nessus Manager or Tenable Vulnerability Management. List multiple groups in a comma-separated list. If any group names have spaces, use quotes around the whole list. For example: <code>"Atlanta,Global Headquarters"</code></li></ul> <div data-bbox="574 611 1479 724" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> The agent group name is case-sensitive and must match exactly.</p></div> <ul style="list-style-type: none"><li>• <code>--ca-path</code> – A custom CA certificate to use to validate the manager's server certificate.</li><li>• <code>--offline-install</code> – When enabled (set to "yes"), installs Tenable Nessus Agent on the system, even if it is offline. Tenable Nessus Agent periodically attempts to link itself to its manager.  If the agent cannot connect to the controller, it retries every hour. If the agent can connect to the controller but the link fails, it retries every 24 hours.</li><li>• <code>--network</code> – For Tenable Vulnerability Management-linked agents, adds the agent to a custom <a href="#">network</a>. If you do not specify a network, the agent belongs to the default network.</li><li>• <code>--proxy-host</code> – The hostname or IP address of your proxy server.</li><li>• <code>--proxy-port</code> – The port number of the proxy server.</li><li>• <code>--proxy-password</code> – The password of the user account that you specified as the username.</li><li>• <code>--proxy-username</code> – The name of a user account that has permissions to access and use the proxy server.</li></ul>



Command	Description
	<ul style="list-style-type: none"><li>• <code>--proxy-agent</code> – The user agent name, if your proxy requires a preset user agent.</li></ul>
<code># nessuscli agent unlink</code>	Unlinks agent from the Tenable Nessus Manager or Tenable Vulnerability Management.
<code># nessuscli scan-triggers --list</code>	Lists details about the agent's rule-based scans: <ul style="list-style-type: none"><li>• Scan name</li><li>• Status (for example, <b>uploaded</b>)</li><li>• Time of last activity (shown next to the status)</li><li>• Scan description</li><li>• Time of last policy modification</li><li>• Time of last run</li><li>• Scan triggers</li><li>• Scan configuration template</li><li>• Command to launch the scan (<code>nessuscli scan-triggers --start --UUID=&lt;scan-uuid&gt;</code>)</li></ul>
<code># nessuscli scan-triggers --start --UUID=&lt;scan-uuid&gt;</code>	(Tenable Vulnerability Management-linked agents only) Manually executes a rule-based scan based on UUID.
<code># nessuscli agent status</code>	Displays the status of the agent, rule-based scanning information, jobs pending, and whether the agent is linked to the server. <b>Optional arguments:</b> <ul style="list-style-type: none"><li>• <code>--local</code> – (Default behavior) Provides the status, current jobs count, and jobs pending. This option prevents the agent from contacting its management software to fetch the status. Instead, it shows the last known information from its most</li></ul>



Command	Description
	<p>recent sync.</p> <ul style="list-style-type: none"><li>• <code>--remote</code> – Fetches the job count from the manager and displays the status.</li></ul> <div data-bbox="574 432 1479 548" style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> Tenable does not recommend running frequent status checks with the <code>--remote</code> option (for example, when using automation).</p></div> <ul style="list-style-type: none"><li>• <code>--offline</code> – Provides the most recently cached agent status when it cannot connect to Tenable Nessus Manager or Tenable Vulnerability Management.</li><li>• <code>--show-token</code> – Displays the agent's token that is used to identify and authenticate with its manager.</li><li>• <code>--show-uuid</code> – Displays the agent's Tenable UUID.</li></ul>
<pre># nessuscli plugins --info</pre>	<p>Lists details about the agent's <code>full</code> and <code>inventory</code> plugin sets:</p> <ul style="list-style-type: none"><li>• <code>Installed version</code></li><li>• <code>Last downloaded</code></li><li>• <code>Last needed</code></li><li>• <code>Expires in</code> – The plugin set's expiration time and date (that is, when the plugin set is no longer needed).</li><li>• <code>Plugins</code> – The total number of plugins in the plugin set.</li><li>• <code>Uncompressed source size</code></li></ul> <p>Lists details and statistics about the agent's plugins, such as:</p> <ul style="list-style-type: none"><li>• <code>Last plugin update time</code></li><li>• <code>Last plugin update check time</code></li><li>• <code>Total compressed plugins source size</code></li><li>• <code>Total compiled plugins size</code></li></ul>



Command	Description
	<ul style="list-style-type: none"><li>• Total plugins attributes data</li><li>• Total plugin size on disk</li></ul>
<code># nessuscli plugins --reset</code>	<p>Deletes all plugins and plugin-related data off the disk. The agent is able to download plugins immediately after the deletion completes.</p> <div style="border: 1px solid blue; padding: 5px;"><b>Note:</b> This command only triggers if the agent has plugin data on its disk.</div>
<code># install-relay - -linking- key=&lt;Tenable.ad relay linking key&gt;</code>	<p>Installs a Tenable Identity Exposure Secure Relay on the agent.</p> <p>To retrieve the Tenable Identity Exposure relay linking key, see <a href="#">Secure Relay</a> in the <i>Tenable.ad Administrator Guide</i>.</p>
Update Commands	
<code># nessuscli agent update --file- e=&lt;plugins_ set.tgz&gt;</code>	<p>Manually installs a plugin set.</p>
<code># nessuscli fix - -set agent_ update_ channel=&lt;value&gt;</code>	<p>(Tenable Vulnerability Management-linked agents only)</p> <p>Sets the agent update plan to determine what version the agent automatically updates to.</p> <p>Values:</p> <ul style="list-style-type: none"><li>• <b>ga</b> – Automatically updates to the latest Agent version when it is made generally available (GA). <b>Note:</b> This date is usually <i>one week after</i> the version is made generally available. For versions that address critical security issues, Tenable may make the version available immediately.</li><li>• <b>ea</b> – Automatically updates to the latest Agent version as soon as it is released for Early Access (EA), typically a few weeks before general availability.</li></ul>



Command	Description
	<ul style="list-style-type: none"><li>• <b>stable</b> – Does not automatically update to the latest Tenable Nessus Agent version. Remains on an earlier version of Tenable Nessus Agent set by Tenable, usually one release older than the current generally available version, but no earlier than 7.7.0. When Tenable Nessus Agent releases a new version, your agent updates software versions, but stays on a version prior to the latest release.</li></ul> <div style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> For agents linked to Tenable Vulnerability Management, you need to run the <code>agent_update_channel</code> command from the <code>agent nessuscli</code> utility. For agents linked to Tenable Nessus Manager, you need to run the <code>agent_update_channel</code> command from the Tenable Nessus Manager <code>nessuscli</code> utility.</p></div>
<pre># nessuscli fix - -set maximum_ scans_per_day- y=&lt;value&gt;</pre>	(Tenable Vulnerability Management-linked agents only)  Sets the maximum number of scans an agent can run per day. The minimum amount is <b>1</b> , the maximum amount is <b>48</b> , and the default amount is <b>10</b> .
Fix Commands	
<pre># nessuscli fix - -list</pre>	Displays a list of agent settings and their values.
<pre>nessuscli fix -- set &lt;setting&gt;=&lt;value&gt;</pre>	Set an agent setting to the specified value.  For a list of agent settings, see <a href="#">Advanced Settings</a> .
<pre># nessuscli fix - -set update_host- name="&lt;value&gt;"</pre>	Updates agent hostnames automatically in Tenable Vulnerability Management or Tenable Nessus Manager 7.1.1 or later.  You can set the <code>update_hostname</code> parameter to <code>yes</code> or <code>no</code> . By default, this preference is disabled.  <div style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> Restart the agent service for the change to take effect in Tenable Nessus Manager.</p></div>



Command	Description
<pre># nessuscli fix - -set max_ retries="<i>&lt;value&gt;</i>"</pre>	<p>Sets the maximum number of times an agent should retry in the event of a failure when executing the agent <code>link</code>, <code>agent status</code>, or <code>agent unlink</code> commands. The commands retry, the specified number of times, consecutively, sleeping increasing increments of time set by <code>retry_sleep_milliseconds</code> between attempts. The default value for <code>max_retries</code> is 0.</p> <p>For example, if you set <code>max_retries</code> to 4 and set <code>retry_sleep_milliseconds</code> to the default of 1500, then the agent will sleep for 1.5 seconds after the first try, 3 seconds after the second try, and 4.5 seconds after the third try.</p> <div style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> This setting does not affect offline updates or the agent's normal 24 hour check-in after it is linked.</p></div>
<pre># nessuscli fix - -set retry_sleep_ milliseconds=" <i>&lt;value&gt;</i>"</pre>	<p>Sets the number of milliseconds that an agent sleeps for between retries in event of a failure when executing the agent <code>link</code>, <code>agent status</code>, or <code>agent unlink</code> commands. The default is 1500 milliseconds (1.5 seconds).</p>
<pre># nessuscli fix - -set niap_mod- e=enforcing</pre>	<p>Enforces NIAP mode for Nessus Agent. For more information about NIAP mode, see <a href="#">Configure Nessus Agent for NIAP Compliance</a>.</p>
<pre># nessuscli fix - -set niap_mod- e=non-enforcing</pre>	<p>Disables NIAP mode for Nessus Agent. For more information about NIAP mode, see <a href="#">Configure Nessus Agent for NIAP Compliance</a>.</p>
<pre># nessuscli fix - -set fips_mod- e=enforcing</pre>	<p>Enforces the current validated FIPS module for Nessus Agent communication and database encryption. The FIPS module does not affect scanning encryption.</p> <div style="border: 1px solid #0070C0; padding: 5px;"><p><b>Note:</b> Nessus Agent also enforces the FIPS module when you enforce NIAP mode. For more information, see <a href="#">Configure Nessus Agent for NIAP Compliance</a>.</p></div>



Command	Description
<pre># nessuscli fix - -set fips_mod- e=non-enforcing</pre>	<p>Disables the FIPS module for Nessus Agent communication and database encryption.</p> <div style="border: 1px solid blue; padding: 5px;"><p><b>Note:</b> Nessus Agent also disables the FIPS module when you disable NIAP mode. For more information, see <a href="#">Configure Nessus Agent for NIAP Compliance</a>.</p></div>
Fix Secure Settings	
<pre># nessuscli fix - -secure --set &lt;setting&gt;=&lt;value&gt;</pre>	<p>Set secure settings on the agent.</p> <div style="border: 1px solid red; padding: 5px;"><p><b>Caution:</b> Tenable does not recommend changing undocumented --secure settings as it may result in an unsupported configuration.</p></div> <p>For a list of supported secure settings, see <a href="#">Advanced Settings</a>.</p>
<pre># nessuscli fix - -secure --get agent_linking_key</pre>	<p>(Nessus versions 10.4.0 and later only) Retrieve your unique agent linking key.</p> <div style="border: 1px solid blue; padding: 5px;"><p><b>Note:</b> You can only use this linking key to link an agent. You cannot use it to link a scanner or a child node.</p></div>
Resource Control Commands	
<pre># nessuscli fix - -set process_pri- ority="&lt;value&gt;"  # nessuscli fix - -get process_pri- ority  # nessuscli fix - -delete process_ priority</pre>	<p><b>Commands</b></p> <p>Set, get, or delete the <code>process_priority</code> setting.</p> <p>You can control the priority of the Nessus Agent relative to the priority of other tasks running on the system by using the <code>process_priority</code> preference.</p> <p>For valid values and more information on how the setting works, see <a href="#">Agent CPU Resource Control</a>.</p>





## Plugin Updates

The following table describes the behavior of differential plugin updates for agents linked to either Tenable Vulnerability Management or Tenable Nessus Manager.

Linked	Differential Update	Full Update
Tenable Vulnerability Management	The agent requests differential updates from Tenable Vulnerability Management once every 24 hours.	<p>The agent performs a full plugin update at scan time whenever the agent needs a new plugin set or sets.</p> <p>The agent also deletes unused plugin sets after a configurable amount of time. After the amount of time passes, the agent performs a full update and deletes the unused plugin sets. For more information, see the <a href="#">days to keep unused plugins</a> advanced setting.</p>
Tenable Nessus Manager	The agent performs a differential plugin update when the agent plugin set is 5 days or less behind the Tenable Nessus Manager plugin set.	The agent performs a full plugin update when the agent plugin set is more than 5 days behind the Tenable Nessus Manager plugin set.



---

## Rule-based Trigger File Location

---

You can find rule-based trigger information for Tenable Nessus Agents in the following locations:

Operating System	Location
Windows	C:\ProgramData\Tenable\Nessus Agent\nessus\triggers
Linux	/opt/nessus_agent/var/nessus/triggers
macOS	/Library/NessusAgent/run/var/nessus/triggers

For more information about triggered scans, see the [Triggered Scan description](#) in the *Tenable.io User Guide*.



## FAQ

### Are agents or network-based scans easier to run?

The ease or difficulty of each scanning method depends on your environment and your organizational needs.

Consider the following questions:

- Is it possible to install a Nessus scanner and possibly a Tenable Nessus Network Monitor in every network segment?
- Would it be easier to install fewer Nessus Managers (for example, one or three) and allow the agents to report back in over and through hops and firewalls, etc.?
- Are all your systems online, connected, and reporting back full results during your scan windows?
- Are all systems, when sleeping, configured correctly and respond appropriately to wake-on-lan?
- Do you spend time trying to keep track or obtain the current credentials for many systems?
- Does your network include laptops that work remotely that you cannot credential scan through VPN or when not connected to the organization network directly?

### What plugins work with agents / credentialed scans?

**Note:** The Tenable Research team is constantly adding and updating plugins. For a comprehensive list of plugins, see <https://www.tenable.com/plugins>.

Most plugins work with Tenable Nessus Agents. The exceptions include:

- Plugins that work based on remotely disclosed information or that detect activity performed through remote connectivity, such as logging into a DB server, trying default credentials (brute force), or traffic-related enumeration.
- Plugins related to network checks.

There are also cases where there is overlap in the intent of the check. For example, if you use OS fingerprinting without credentials in a network-based scan and query the system for the exact



version of its OS in a credentialed scan, this overlap heightens the credential findings over the network, since the network version tends to be a best guess.

### **What data does an agent send to Tenable Vulnerability Management/Tenable Nessus Manager?**

Agents send the following data to Tenable Vulnerability Management/Tenable Nessus Manager:

- Version information (agent version, host architecture)
- Versions of installed Tenable plugins
- OS information (for example, Microsoft Windows Server 2019 Enterprise Service Pack 1)
- Tenable asset IDs (for example, /etc/tenable\_tag on Unix, HKEY\_LOCAL\_MACHINE\SOFTWARE\Tenable\TAG on Windows)
- Network interface information (network interface names, MAC addresses, IPv4 and IPv6 addresses, hostnames and DNS information if available)
- Hostname if update\_hostname is set to yes (see [Advanced Settings](#) for more information)
- AWS EC2 instance metadata, if available:
  - privateIp
  - accountId
  - imageId
  - region
  - instanceType
  - availabilityZone
  - architecture
  - instanceId
  - local-hostname
  - public-hostname
  - public-ipv4



- mac
- iam/security-credentials/
- public-keys/0/openssh-key
- security-groups



---

## Appendix

---

- [File and Process Whitelist](#)
- [Nessus Agent Cheatsheet](#)
- [Customer Case Studies](#)
- [Expanded Agent Large Scale Deployment Guide \(PDF\)](#)



## Configure Tenable Nessus Agent for NIAP Compliance

If your organization requires that Tenable Nessus Agent meets National Information Assurance Partnership (NIAP) standards, you can configure Tenable Nessus Agent so that relevant settings are compliant with NIAP standards.

Before you begin:

- If Tenable Nessus Agent is linked to Tenable Nessus Manager, verify that the CA certificate of Tenable Nessus Manager is in `custom_CA.inc` or `known_CA.inc`.
- Confirm you have enabled the full disk encryption capabilities provided by the operating system on the host where Tenable Nessus Agent is installed.

To configure Tenable Nessus Agent for NIAP compliance:

1. Access the agent from the command line interface.
2. Enable NIAP mode using the command line interface:
  - In the command line, enter the following command:

```
nessuscli fix --set niap_mode=enforcing
```

Linux example:

```
/opt/nessus_agent/sbin/nessuscli fix --set niap_mode=enforcing
```

Tenable Nessus Agent does the following:

**Note:** When Tenable Nessus Agent is in NIAP mode, Tenable Nessus Agent overrides the following settings as long as Tenable Nessus Agent remains in NIAP mode. If you disable NIAP mode, Tenable Nessus Agent reverts to what you had set before.

- Overrides the SSL mode (`ssl_mode`) with TLS 1.2 (`niap`).
- Overrides the SSL cipher list (`ssl_cipher_list`) setting with NIAP compliant ciphers (`niap`), which sets the following ciphers:



- ECDHE-RSA-AES128-SHA256
  - ECDHE-RSA-AES128-GCM-SHA256
  - ECDHE-RSA-AES256-SHA384
  - ECDHE-RSA-AES256-GCM-SHA384
- Uses strict certificate validation:
    - Disallows certificate chains if any intermediate certificate lacks the CA extension.
    - Authenticates a server certificate, using the signing CA certificate.
    - Authenticates a client certificate when using client certificate authentication for login.
    - Checks the revocation status of a CA certificate using the Online Certificate Status Protocol (OCSP). If the certificate is revoked, then the certificate is marked as invalid. If there is no response, then the certificate is not marked as invalid, and its use is permitted if it is otherwise valid.
    - Ensures that the certificate has a valid, trusted CA that is in known\_CA.inc. CA Certificates for Tenable Vulnerability Management and plugins.nessus.org are already in known\_CA.inc in the plugins directory.
    - If linked to Tenable Nessus Manager, verifies that the CA certificate of Tenable Nessus Manager is found in custom\_CA.inc or known\_CA.inc.
  - Enforces the current validated FIPS module for Nessus Agent communication and database encryption. The FIPS module does not affect scanning encryption.

**Note:** You can enforce the FIPS module from the Agent `nessuscli` utility without enforcing NIAP mode. For more information, see [Fix Commands](#).





## File and Process Allow List

You should allow certain Tenable Nessus Agent files, folders, and processes in third-party endpoint security products such as anti-virus applications and host-based intrusion and prevention systems.

**Tip:** If your Windows installation uses a non-standard drive or folder structure, you can use the %PROGRAMFILES% and %PROGRAMDATA% environment variables.

The table below contains a list of files, folders, and processes to add to an allow list.

Windows
Folders
C:\Program Files\Tenable\Nessus Agent\*
C:\Program Files (x86)\Tenable\Nessus Agent\*
C:\ProgramData\Tenable\Nessus Agent\*
Processes
C:\Program Files\Tenable\Nessus Agent\nasl.exe
C:\Program Files\Tenable\Nessus Agent\nessuscli.exe
C:\Program Files\Tenable\Nessus Agent\nessusd.exe
C:\Program Files\Tenable\Nessus Agent\nessus-service.exe
C:\Program Files\Tenable\Nessus Agent\nessus-agent-module.exe
C:\Program Files\Tenable\Nessus Agent\openssl.exe
%SystemRoot%\tenable_ovaldi_2ef350e0435440418f7d33232f74f260.exe
%SystemRoot%\tenable_mw_scan_*.exe
%SystemRoot%\temp\nessus_*.bat
C:\Program Files (x86)\Tenable\Nessus Agent\nasl.exe
C:\Program Files (x86)\Tenable\Nessus Agent\nessuscli.exe



C:\Program Files (x86)\Tenable\Nessus Agent\nessusd.exe

C:\Program Files (x86)\Tenable\Nessus Agent\nessus-service.exe

C:\Program Files (x86)\Tenable\Nessus Agent\nessus-agent-module.exe

C:\Program Files (x86)\Tenable\Nessus Agent\openssl.exe

%SystemRoot%\tenable\_ovaldi\_2ef350e0435440418f7d33232f74f260.exe

%SystemRoot%\Tenable\Nessus Agent\tenable\_mw\_scan\_\*.exe

%SystemRoot%\Tenable\Nessus Agent\temp\nessus\_\*.bat

## Linux

### Folders

/opt/nessus\_agent/sbin/\*

/opt/nessus\_agent/bin/\*

/opt/nessus\_agent/lib/nessus/\*

### Files

/opt/nessus\_agent/bin/nasl

/opt/nessus\_agent/sbin/nessusd

/opt/nessus\_agent/sbin/nessuscli

/opt/nessus\_agent/sbin/nessus-service

/opt/nessus\_agent/sbin/nessus-agent-module

### Processes

/opt/nessus\_agent/bin/nasl

/opt/nessus\_agent/bin/openssl

/opt/nessus\_agent/sbin/nessusd

/opt/nessus\_agent/sbin/nessuscli



/opt/nessus\_agent/sbin/nessus-service

/opt/nessus\_agent/sbin/nessus-agent-module

## macOS

### Folders

/Library/NessusAgent/run/sbin/\*

/Library/NessusAgent/run/bin/\*

### Files

/Library/NessusAgent/run/bin/nasl

/Library/NessusAgent/run/sbin/nessusd

/Library/NessusAgent/run/sbin/nessuscli

/Library/NessusAgent/run/sbin/nessus-service

/Library/NessusAgent/run/sbin/nessus-agent-module

/Library/NessusAgent/run/sbin/nessusmgt

### Processes

/Library/NessusAgent/run/bin/nasl

/Library/NessusAgent/run/bin/openssl

/Library/NessusAgent/run/sbin/nessusd

/Library/NessusAgent/run/sbin/nessuscli

/Library/NessusAgent/run/sbin/nessus-service

/Library/NessusAgent/run/sbin/nessus-agent-module

/Library/NessusAgent/run/sbin/nessusmgt



---

# Nessus Agent Cheatsheet

---



---

# Benefits and Limitations of Using Tenable Nessus Agents

---

## Benefits

- Provides extended scan coverage and continuous security:
  - Can deploy where it's not practical or possible to run network-based scans.
  - Can assess off-network assets and endpoints that intermittently connect to the internet (such as laptops). Tenable Nessus Agents can scan the devices regardless of network location and report results back to the manager.
- Eliminates the need for credential management:
  - Doesn't require host credentials to run, so you don't need to manually update credentials in scan configurations when credentials change, or share credentials among administrators, scanning teams, or organizations.
  - Can deploy where remote credentialed access is undesirable, such as Domain Controllers, DMZs, or Certificate Authority (CA) networks.
- Efficient:
  - Can reduce your overall network scanning overhead.
  - Relies on local host resources, where performance overhead is minimal.
  - Reduces network bandwidth need, which is important for remote facilities connected by slow networks.
  - Removes the challenge of scanning systems over segmented or complex networks.
  - Minimizes maintenance, because Tenable Nessus Agents can update automatically without a reboot or end-user interaction.
  - Large-scale concurrent agent scans can run with little network impact.
- Easy deployment and installation:
  - You can install and operate Tenable Nessus Agents on all major operating systems.
  - You can install Tenable Nessus Agents anywhere, including transient endpoints like laptops.



- You can deploy Tenable Nessus Agents using software management systems such as Microsoft's System Center Configuration Manager (SCCM).

## Limitations

- Network checks—Agents are not designed to perform network checks, so certain plugin items cannot be checked or obtained if you deploy only agent scans. Combining traditional scans with agent-based scanning eliminates this gap.
- Remote connectivity—Agents miss things that can only specifically be performed through remote connectivity, such as logging into a DB server, trying default credentials (brute force), traffic-related enumeration, etc.



## System Requirements for Tenable Nessus Agents

For dataflow and licensing requirements, refer to the [System Requirements](#) section.

### Hardware

Tenable Nessus Agents are lightweight and only use minimal system resources. Generally, a Tenable Nessus Agent uses 40 MB of RAM (all pageable). A Tenable Nessus Agent uses almost no CPU while idle, but is designed to use up to 100% of CPU when available during jobs.

For more information on Tenable Nessus Agent resource usage, refer to [Software Footprint](#).

The following table outlines the minimum recommended hardware for operating a Tenable Nessus Agent. Tenable Nessus Agents can be installed on a virtual machine that meets the same requirements specified.

Hardware	Minimum Requirement
Processor	1 Dual-core CPU
Processor Speed	> 1 GHz
RAM	> 1 GB
Disk Space	<ul style="list-style-type: none"><li>• Agents 7.7.x and earlier: &gt; 1 GB, not including space used by the host operating system</li><li>• Agents 8.0.x and later: &gt; 3 GB, not including space used by the host operating system</li><li>• Agents 10.0.x and later: &gt; 2 GB, not including space used by the host operating system</li></ul> <p>The agent may require more space during certain processes, such as a <code>plugins-code.db</code> defragmentation operation.</p>
Disk Speed	15-50 IOPS

### Software



To view the Tenable Nessus Agent software requirements, see [Nessus Agent Software Requirements](#).





## Installing and Linking Tenable Nessus Agents

The following installation instructions are for the command line. To install using the user interface, see [Install Tenable Nessus Agents](#).

### Linux

**Install the package:**

#### Red Hat, CentOS, and Oracle Linux

```
# dnf install NessusAgent-10.3.1-es8.x86_64.rpm
```

#### Fedora

```
# dnf install NessusAgent-10.3.1-fc34.x86_64.rpm
```

#### Ubuntu

```
# dpkg -i NessusAgent-<version number>-ubuntu1110_i386.deb
```

#### Debian

```
# dpkg -i NessusAgent-<version number>-debian6_amd64.deb
```

**Note:** After installing a Nessus Agent, you must manually start the service using the command `/sbin/service nessusagent start`.

### Link Agent to Tenable Nessus Manager or Tenable Vulnerability Management:

At the command prompt, use the `nessuscli agent link` command. For example:

```
/opt/nessus_agent/sbin/nessuscli agent link
--key=00abcd0000efgh11111i0k222lmopq3333st4455u66v77777w88xy9999zabc00
--name=MyOSXAgent --groups="All" --host=yourcompany.com --port=8834
```

### Windows

You can deploy and link Tenable Nessus Agents via the command line. For example:



```
msiexec /i NessusAgent-<version number>-x64.msi NESSUS_GROUPS="Agent Group Name"  
NESSUS_SERVER="192.168.0.1:8834" NESSUS_  
KEY=00abcd0000efgh11111i0k222lmopq3333st4455u66v77777w88xy9999zabc00 /qn
```

## macOS

### Install the package:

1. Extract Install Nessus Agent.pkg and .NessusAgent.pkg from NessusAgent-<version number>.dmg.

**Note:** The .NessusAgent.pkg file is normally invisible in macOS Finder.

2. Open Terminal.
3. At the command prompt, enter the following command:

```
# sudo installer -pkg /<path-to>/Install Nessus Agent.pkg -target /
```

### Link Agent to Tenable Nessus Manager or Tenable Vulnerability Management:

1. Open Terminal.
2. At the command prompt, use the `nessuscli agent link` command.

For example:

```
# sudo /Library/NessusAgent/run/sbin/nessuscli agent link  
--key=00abcd0000efgh11111i0k222lmopq3333st4455u66v77777w88xy9999zabc00  
--name=MyOSXAgent --groups=All --host=yourcompany.com --port=8834
```



---

## Customer Case Studies

---

The customer case studies describe Tenable Nessus Agent deployments in real customer environments. The case studies highlight key configuration and deployment considerations.

1. ACME's environment consisted of 70,000 assets. ACME utilized the Tenable Vulnerability Management platform to manage agent scanning operations, and a single Tenable.sc instance to manage 40 scanners and to provide unified analytics of both network and Tenable Nessus Agent assessment results.

### [ACME](#)

- [Nessus Agent Operational Tier \(Tenable.io\)](#)
- [Reporting Tier \(Tenable.sc\)](#)

2. Initech is a global organization consisting of 30+ sub-organizations, 40,000 users, 60,000 devices, and 150,000+ active IP addresses. Initech used a hybrid Tenable Vulnerability Management and Tenable Nessus Manager solution for managing Tenable Nessus Agents. Tenable Vulnerability Management was used for user workstation Tenable Nessus Agent scan operations, and Tenable Nessus Manager was used for servers and other permanent on-premise infrastructure. Initech then imported all Tenable Nessus Agent scan data into Tenable.sc for unified reporting and analytics.

### [Initech](#)

- [Agent Deployment \(Nessus Manager & Tenable.io\)](#)
- [Reporting and Traditional Network Scanning \(Tenable.sc\)](#)

3. Sprocket utilized Tenable Vulnerability Management for Tenable Nessus Agent management and local scan and audit information, remote network scan functionality, and integration with their third-party applications via the Tenable Vulnerability Management API.

### [Sprocket](#)



## ACME Customer Case Study

---

A customer, ACME, was using a single Tenable Tenable.sc instance that managed 40 scanners to perform network vulnerability assessments of approximately 1,200 stores on a monthly basis.

ACME wished to update their existing operational model to leverage Nessus Agents to collect assessment results from approximately 70,000 assets. ACME implemented a hybrid approach using the Tenable.io platform to manage agent scanning operations and import agent scan results into Tenable.sc for unified analytics and reporting of both network and agent assessment results.

The intent of this case study is to highlight key configuration considerations that were implemented when ACME moved forward with deploying Nessus Agents.

### Objectives

The primary goal defined by ACME to measure the success of the Tenable Nessus Agent project was their ability to leverage Nessus Agents across their store infrastructure to collect in-depth asset data, while reducing the current network latency experienced by traditional remote network scans.

Scanning Coverage:

- To implement local host scanning using agents on assets across stores to provide more detailed vulnerability assessment results than the current unauthenticated network active scan to stores from headquarter datacenters.
- To utilize agent scans to reduce the impact to ACME's network and allow for more frequent scans.

### Solution

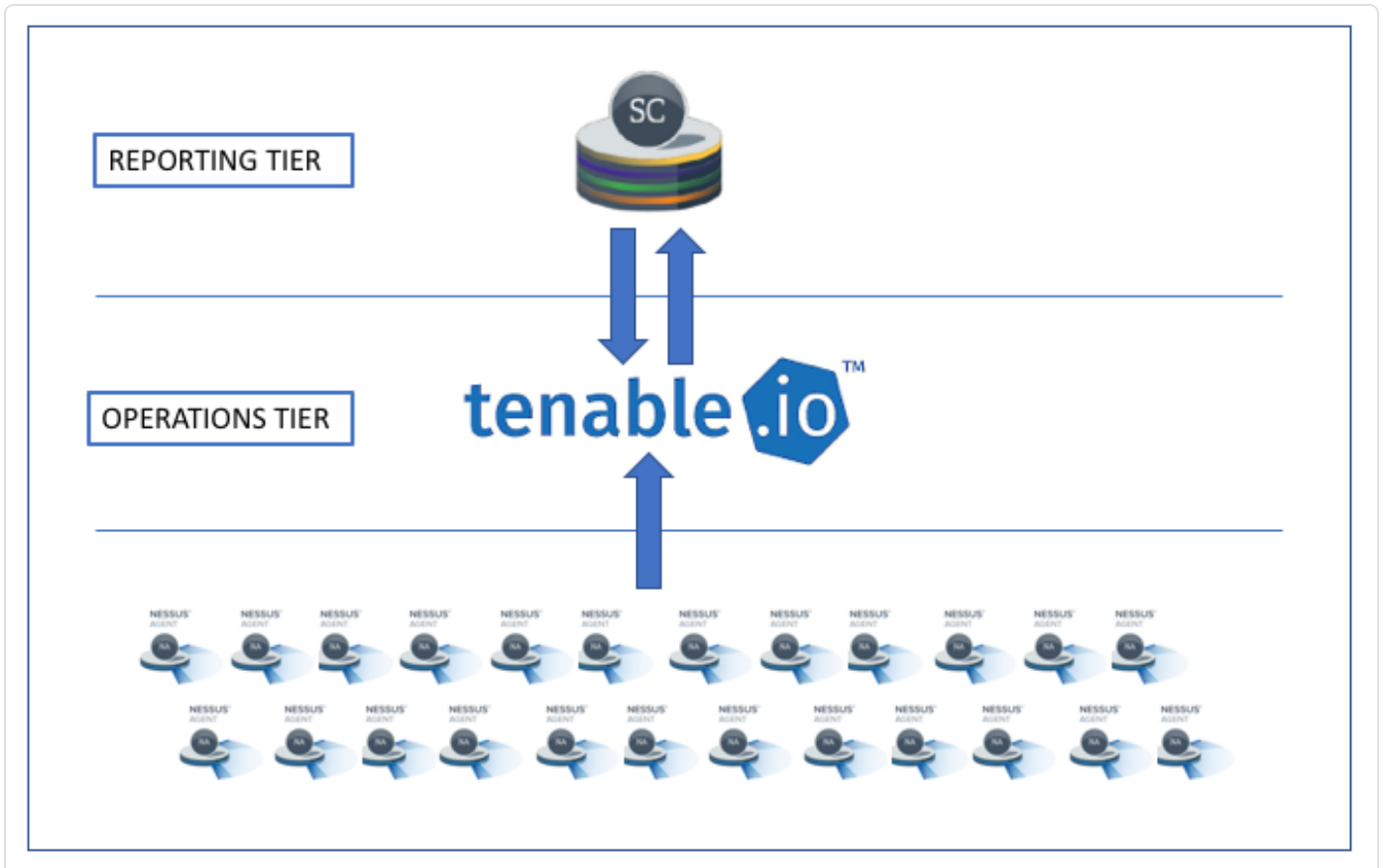
A Tenable.io and Tenable.sc hybrid deployment was used in their enterprise environment. Tenable.io was required for Nessus Agents scan operations, and the existing Tenable.sc infrastructure was used for advanced analytics and reporting. By leveraging Tenable.io for agent scan operations, ACME could automatically scale for large numbers of Nessus Agents and assets, without the need for on-prem software and hardware.

ACME leveraged their existing Tenable.sc infrastructure to achieve their vulnerability management program goals by importing agent scan data from Tenable.io into Tenable.sc for unified reporting



and analytics. This solution split the environment into two tiers, [Reporting \(Tenable.sc\)](#) and [Operational \(Tenable.io\)](#), so that ACME could optimize reporting experiences for its end users, while not impacting the data acquisition capabilities of the platform.

The hybrid deployment is illustrated here:



For more information on the tiered deployment, see:

- [Nessus Agent Operational Tier \(Tenable.io\)](#)
- [Reporting Tier \(Tenable.sc\)](#)



---

## Nessus Agent Operational Tier (Tenable.io)

---

The primary purpose for the Operational Tier (Tenable.io) was to perform agent management and agent scan operations.

### Functions Performed

The following processes and uses take place in the Operational Tier (Tenable.io).

- Deployed agents are linked to Tenable.io.
- Agents are organized in agent groups. Agents can be assigned to agent groups during the installation process.
- Agent scans are established to obtain assessment results from agents via agent groups.
- Agents automatically have plugin and version updates applied by Tenable.io.
- Customers can “opt-out” of having agent version updates automatically applied.

### Considerations

- Agents were deployed using ACME's internal software distribution processes (in this case, SCCM).
- Agent groups included no more than 2,000 agents per group (1,000 is recommended). Limiting the number of agents in each agent group ensures that Tenable.sc is able to successfully import scan results. This limitation only applies when Tenable.sc is part of the deployment.
- Agent scans were restricted to a single agent group each.
- Agent group membership was established by functional zones (by location, role, etc.) for organizational purposes.
- ACME monitored for agent deployment issues (failed installations, linking failures, etc.) out of band (logging client, scripts, etc.).
- Nessus Agents only performed local vulnerability assessments and did not perform network-based assessment (for example, SSL or CGI network based assessments).
- Network and firewalls were configured to allow Nessus Agents to communicate with <https://cloud.tenable.com>.

### Tier Design



Design assumptions included:

- ACME will leverage internal processes and tooling to deploy the Nessus Agent software.
- ACME will establish 50-70 agent groups.
- ACME will configure 50-70 agent scans.



---

## Reporting Tier (Tenable.sc)

---

The primary purpose of the reporting tier was to allow for centralized analytics and reporting of data collected from the Nessus Agent operational tier (Tenable.io). Dashboards, analytics, reports, and Assurance Report Cards are leveraged on this tier.

### Functions Performed

The following processes and uses take place in the Reporting Tier (Tenable.sc).

- Tenable.io was added to Tenable.sc as an “agent capable” scanner.
- Agent scans in Tenable.sc were configured to retrieve Nessus Agent scan results from Tenable.io.
- Analytics, dashboards, reports, and Assurance Report Cards in Tenable.sc were leveraged for all assessment types (Agent and Network Scanning).

### Considerations

- Tenable recommended that ACME configure Tenable.sc to retrieve Nessus Agent scan results from Tenable.io the same day Tenable.io collects assessment results from Nessus Agents. This configuration ensures that Tenable.sc captures proper detection dates.
- Tenable.sc required additional data repositories to support the Nessus Agent results. Tenable recommended that ACME establish two new repositories in Tenable.sc for Nessus Agent results, because repositories can only handle upwards of 50,000 assets each.
- Tenable.sc 5.7 introduced an agent-specific repository that leverages the Nessus Agent UUID to better track uniqueness when results are imported into Tenable.sc.
- ACME needed to perform a full analysis on their current Tenable.sc hardware configuration to determine if additional CPU/RAM/HDD was required for the additional data resulting from importing Nessus Agent scan results.

### Tier Design

Design assumptions included:

- ACME will establish two (2) repositories to store Nessus Agent scan results.
- ACME will establish 50-70 agent scans to retrieve Nessus Agent scan results from Tenable.io.





- ACME will balance each agent scan retrieval evenly across the two (2) new repositories.
- ACME will evaluate current infrastructure to determine if additional CPU/RAM/HDD is required.



---

## Initech Customer Case Study

---

A customer, Initech, was using a tiered Tenable Tenable.sc deployment across a large federated environment consisting of 30+ sub-organizations, 40,000 users, 60,000 devices, and 150,000+ active IPs. They performed weekly network vulnerability assessments with over 75 scanners at sites located around the United States.

Initech had a reporting requirement to perform more frequent assessments of their systems and to be able to remotely gather data from user laptops when they were off-site. Initech deployed over 50,000 Nessus Agents to accomplish this task, using a hybrid model with both Tenable Nessus Manager and Tenable Vulnerability Management, feeding data back into Tenable.sc for analytics and reporting.

The intent of this case study is to highlight key configuration considerations that were implemented when Initech moved forward with deploying Tenable Nessus Agents.

### Objectives

The primary goals defined by Initech to measure the success of the Tenable Nessus Agent project were to gather data more frequently, assess remote systems, and reduce the burden posed by managing credentials across a large disparate enterprise.

### Solution

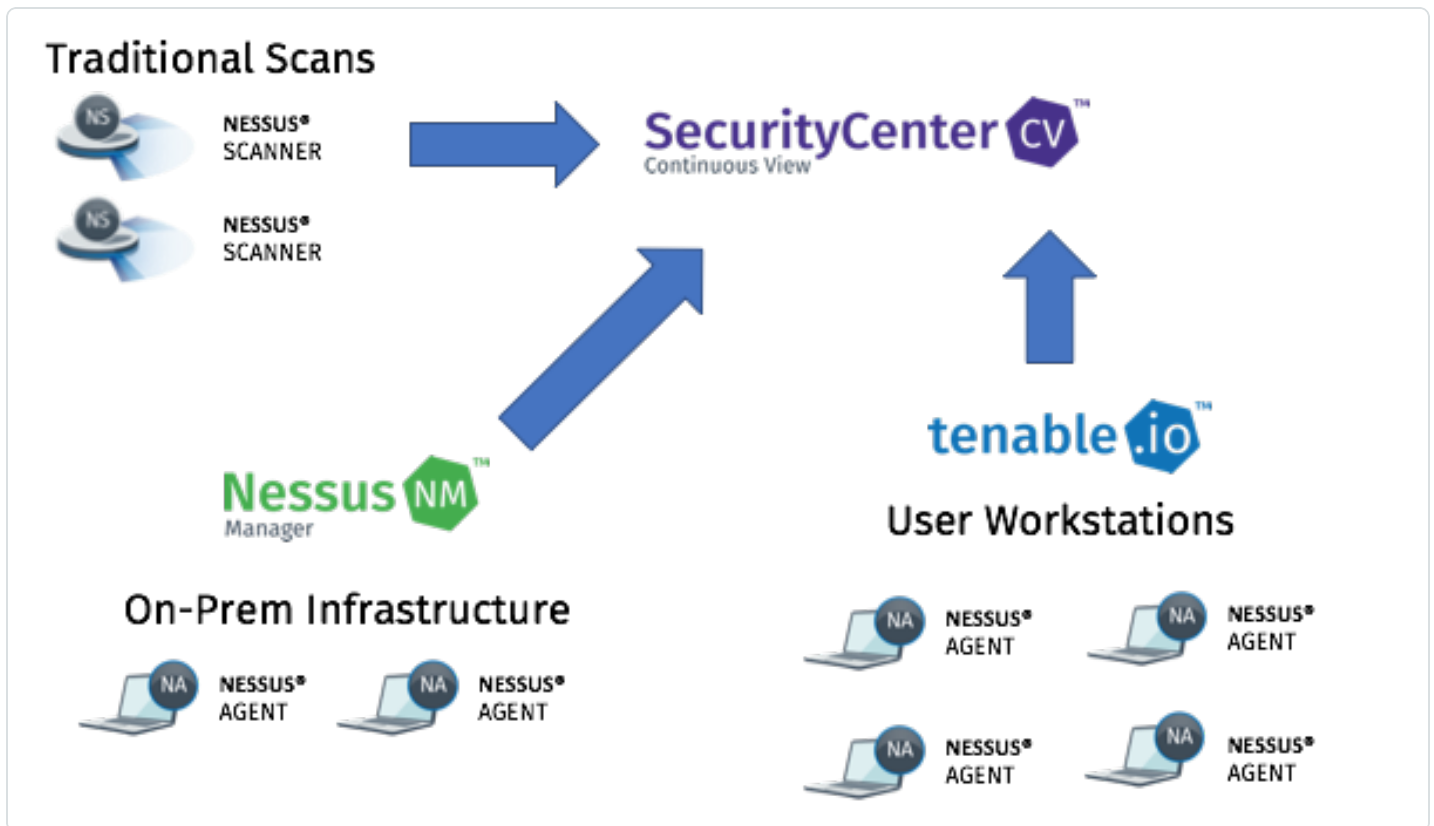
A Nessus Manager and Tenable.io hybrid deployment was used for Nessus Agents in their enterprise environment. Tenable Vulnerability Management was required for user workstation Tenable Nessus Agent scan operations, and Tenable Nessus Manager was used for servers and other permanent on-premise infrastructure.

- Initech used the scaling ability, uptime guarantee, and cloud flexibility of Tenable Vulnerability Management to meet the dynamic requirements of a constantly changing workstation environment.
- Initech used Tenable Nessus Manager, an on-premise solution, to provide more user control over the scan data for more sensitive systems, such as server infrastructure.

Initech leveraged their existing Tenable.sc infrastructure to achieve their vulnerability management program goals by importing agent scan data from Tenable Nessus Manager and Tenable Vulnerability Management into Tenable.sc for unified reporting and analytics.



The hybrid deployment is illustrated in the following diagram:



For more information on the tiered deployment, see:

- [Agent Deployment \(Nessus Manager & Tenable.io\)](#)
- [Reporting and Traditional Network Scanning \(Tenable.sc\)](#)



---

## Agent Deployment (Nessus Manager & Tenable.io)

---

The primary purpose for Nessus Manager was to perform agent management and agent scan operations for on-premise infrastructure (10,000 systems), while Tenable.io was used for agent management and scan operations of user workstations (40,000 systems).

### Functions Performed

- Deployed agents are linked to Nessus Manager or Tenable.io depending on system type.
- Agents are organized in agent groups. Agents can be assigned to agent groups during the installation process.
- Agent scans are established to obtain assessment results from agents via agent groups.
- Agents automatically have plugin and version updates applied by Nessus Manager or Tenable.io.

### Considerations

- Agents were deployed using Initech's internal software distribution processes (in this case, a large variety of platforms including Altiris, SCCM, Tivoli, Casper, and others).
- Agent groups included no more than 2,000 agents per group (1,000 is recommended). Limiting the number of agents in each agent group ensures that Tenable.sc is able to successfully import scan results. This limitation only applies when Tenable.sc is part of the deployment.
- Agent scans were restricted to a single agent group each.
- Agent scan policies were more thorough and verbose than the traditional network scans due to the increased efficiency of agent scan distribution.
- On-Premise/Server agent scan windows were restricted to custom time frames selected by each sub-org to meet individual organizational requirements.
- User workstation scan windows were set to ~24 hours and repeated daily to ensure full coverage regardless of when a system was turned on.
- Agent group membership was established by organization and in some cases, operational tier or other functional requirements.



- Initech monitored for agent deployment issues (failed installations, linking failures, etc.) out of band (logging client, scripts, etc.).
- Nessus Agents only performed local vulnerability assessments and did not perform network-based assessment (for example, SSL or CGI network based assessments).
- Network and firewalls were configured to allow infrastructure Nessus Agents to communicate with the on-premise Nessus Manager via a custom port, and user workstations to communicate with <https://cloud.tenable.com>.

## **Tier Design**

Design assumptions included:

- Initech will leverage internal processes and tooling to deploy the Nessus Agent software.
- Initech will establish 30-50 agent groups in both Nessus Manager and Tenable.io
- Initech will configure 30-50 agent scans in both Nessus Manager and Tenable.io
- Initech will configure and provision a Nessus Manager that can handle 10,000 agents connecting to it.



---

## Reporting and Traditional Network Scanning (Tenable.sc)

---

The primary purpose of the reporting tier was to allow for centralized analytics and reporting of data collected from the Nessus Agents and existing traditional network scans. Dashboards, analytics, reports, and Assurance Report Cards are leveraged on this tier.

### Functions Performed

The following processes and uses take place in Tenable.sc.

- Nessus Manager and Tenable.io were added to Tenable.sc as an “agent capable” scanners.
- Agent scans in Tenable.sc were configured to retrieve Nessus Agent scan results from Nessus Manager and Tenable.io.
- Agent data was placed in new repositories according to existing data models.
- Analytics, dashboards, reports, and Assurance Report Cards in Tenable.sc were leveraged for all assessment types (Agent and Network Scanning).

### Considerations

- Tenable.sc required additional data repositories to support the Nessus Agent results. Tenable recommended that Initech establish multiple new repositories in Tenable.sc for Nessus Agent results, because combining agent and network assessment results in the same repository can cause reporting challenges.
- Initech needed to perform a full analysis on their current Tenable.sc hardware configuration to determine if additional CPU/RAM/HD was required for the additional data resulting from importing Nessus Agent scan results.
- Initech needed to evaluate their existing traditional scan structures/policies to ensure limited data overlap once Nessus Agent assessments were implemented and data imported into Tenable.sc.

### Tier Design

Design assumptions included:



- Initech will establish multiple repositories to store Nessus Agent scan results.
- Initech will establish 60-100 agent jobs to retrieve Nessus Agent scan results from Tenable.io and Nessus Manager.
- Initech will evaluate current infrastructure to determine if additional CPU/RAM/HDD is required.
- Initech will evaluate existing scan structures/policies to limit data overlap.



## Sprocket

---

Sprocket Inc. is a global company with offices and employees in almost all countries. Sprocket's large and distributed workforce presented several challenges when selecting and designing a security solution. Sprocket required a solution that provided the following:

- Immediate and consistent local scans across all 330,000 assets including servers in company data centers, cloud servers (Azure and AWS), and transient devices like employee laptops.
- Minimized network load since their data centers were at capacity.
- Improved credential management due to their global distributed workforce and siloed organizations.
- The ability to integrate with third-party applications that are used to manage and monitor information across their IT landscape.
- A solution that could scale as their OT, WAS, and container environments increased.

### **Solution**

Sprocket leveraged Tenable Vulnerability Management to manage all aspects of their environment. The solution used Tenable Nessus Agents for all Windows, Linux, and macOS devices for local scan and audit information, and Nessus scanners located in private cloud instances in each organizational theater for remote network scanning. Tenable Vulnerability Management also provided the needed API to utilize their third-party and customized applications.

Sprocket deployed Tenable Nessus Agents using customized scripts for each operating system based on the asset function. The Tenable Nessus Agents were assigned to one of 130 groups based on the operating system and asset owner.