Table of Contents

Welcome to Nessus Agent 8.2.x ................................................................. 6
Agent Deployment Workflow ................................................................. 7
Benefits and Limitations ........................................................................ 9
  Traditional Active Scans (Non-credentialed) .................................... 10
  Traditional Active Scans (Credentialed) .............................................. 11
Agent Scans ............................................................................................ 13
Agent Use Cases ...................................................................................... 15
  Mobile, Distributed Workforce ........................................................... 16
  High Latency Networks ...................................................................... 17
  Hardened Systems ............................................................................. 18
Deployment Considerations ................................................................. 19
  General Considerations ................................................................... 20
  Large-scale Deployments ................................................................. 22
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
  Port Requirements ........................................................................... 36
  Licensing Requirements ................................................................... 38
Agent CPU Resource Control ............................................................... 40
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance Metrics</td>
<td>43</td>
</tr>
<tr>
<td>Nessus Agents Performance</td>
<td>44</td>
</tr>
<tr>
<td>Host System Utilization</td>
<td>45</td>
</tr>
<tr>
<td>Software Footprint</td>
<td>46</td>
</tr>
<tr>
<td>Agent Lifecycle &amp; Bandwidth</td>
<td>47</td>
</tr>
<tr>
<td>Nessus Manager Performance</td>
<td>48</td>
</tr>
<tr>
<td>Install Nessus Agent</td>
<td>50</td>
</tr>
<tr>
<td>Retrieve the Nessus Agents Linking Key</td>
<td>51</td>
</tr>
<tr>
<td>Install a Nessus Agent on Linux</td>
<td>53</td>
</tr>
<tr>
<td>Install a Nessus Agent on Windows</td>
<td>57</td>
</tr>
<tr>
<td>Configure and View the System Tray Application</td>
<td>62</td>
</tr>
<tr>
<td>Install a Nessus Agent on Mac OS X</td>
<td>65</td>
</tr>
<tr>
<td>Update a Nessus Agent</td>
<td>69</td>
</tr>
<tr>
<td>Downgrade Nessus Agent</td>
<td>72</td>
</tr>
<tr>
<td>Back Up Agent</td>
<td>74</td>
</tr>
<tr>
<td>Restore Agent</td>
<td>75</td>
</tr>
<tr>
<td>Remove Nessus Agent</td>
<td>77</td>
</tr>
<tr>
<td>Uninstall a Nessus Agent on Linux</td>
<td>78</td>
</tr>
<tr>
<td>Uninstall a Nessus Agent on Windows</td>
<td>80</td>
</tr>
<tr>
<td>Uninstall a Nessus Agent on Mac OS X</td>
<td>81</td>
</tr>
<tr>
<td>Manage Agents</td>
<td>82</td>
</tr>
<tr>
<td>Start or Stop a Nessus Agent</td>
<td>83</td>
</tr>
<tr>
<td>Agent Status</td>
<td>85</td>
</tr>
<tr>
<td>Filter Agents</td>
<td>86</td>
</tr>
</tbody>
</table>
Environment Variables ........................................................................................................... 142
Deploy Nessus Agent Using JSON .......................................................................................... 143
Nessus Service ......................................................................................................................... 146
Nessus CLI Agent Commands .................................................................................................. 149
Plugin Updates ......................................................................................................................... 156
Logging ..................................................................................................................................... 157
FAQ .......................................................................................................................................... 158
Appendix .................................................................................................................................... 160
Configure Nessus Agent for NIAP Compliance ........................................................................ 161
File and Process Allow List ....................................................................................................... 163
Nessus Agent Cheatsheet .......................................................................................................... 165
Benefits and Limitations of Using Nessus Agents .................................................................... 166
System Requirements for Nessus Agents .................................................................................. 168
Installing and Linking Nessus Agents ....................................................................................... 170
Customer Case Studies ........................................................................................................... 172
ACME Customer Case Study ..................................................................................................... 173
Nessus Agent Operational Tier (Tenable.io) ............................................................................ 175
Reporting Tier (Tenable.sc) ....................................................................................................... 177
Initech Customer Case Study .................................................................................................... 179
Agent Deployment (Nessus Manager & Tenable.io) ................................................................. 181
Reporting and Traditional Network Scanning (Tenable.sc) ...................................................... 183
Sprocket .................................................................................................................................... 185
Welcome to Nessus Agent 8.2.x

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 are missed by traditional scanning. Nessus Agents collect vulnerability, compliance, and system data, and report that information back to a manager for analysis. With Nessus Agents, you extend scan flexibility and coverage. You can scan hosts without using credentials, as well as offline assets and endpoints that intermittently connect to the internet. You can also run large-scale concurrent agent scans with little network impact.

About Nessus Agents

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 is executed. If laptops or other transient devices are not accessible when a scan is executed, they are excluded from the scan, leaving you blind to vulnerabilities on those devices. Nessus Agents help reduce your organization’s attack surface by scanning assets that are off the network or powered-down during scheduled assessments or by scanning other difficult-to-scan assets.

Once installed on servers, portable devices, or other assets found in today’s complex IT environments, 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 Nessus Agents with Nessus Manager or Tenable.io.

Nessus Agents Product Page
Agent Deployment Workflow

Before you begin:

- If you are using Nessus Manager to manage Nessus Agents, you must deploy and configure Nessus Manager before you deploy Nessus Agents.
- If you are using Tenable.io to manage your Nessus Agents, no preliminary deployment is required.

To deploy Nessus Agents

1. On each host, [install 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 blackout 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.io)](#)

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 Nessus Manager or Tenable.io, 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 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 (e.g., Nessus Manager or Tenable.io) when network connectivity resumes.

If traditional 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-Credentialled)
- Traditional Active Scans (Credentialled)
- 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 ports, protocols, and services that are exposed on a host 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 being tested.
- 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 in most cases; that is, does not have a negative effect on the network, device, or application being tested.
- 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

Nessus Agent scans use lightweight, low-footprint programs that you install locally on hosts. Nessus Agents collect vulnerability, compliance, and system data, and report that information back to Nessus Manager or Tenable.io for analysis. 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). 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 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 Nessus Agents on all major operating systems.
  ◦ You can install Nessus Agents anywhere, including transient endpoints like laptops.
  ◦ You can deploy 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 specifically 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 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 for their devices to be 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. Any scan requests or results submitted after the scan window are discarded, and the system is marked as not scanned.

This approach helps ensure accurate security data while also reducing the need for duplicative and irrelevant scanning. For example, an employee returning from a two-week vacation won’t 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 to 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 is transmitted 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 is completed, the agent only downloads incremental plugin updates. This approach drastically reduces the ongoing network traffic by only pulling content deltas across the network. Additionally, 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. In order to access key data sets, 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 should not be viewed as a step-by-step guide for deploying Nessus Agents. You should 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 prior to deploying Nessus Agents:

- What operating system do you plan to deploy the Nessus Agent on?
  - Linux (Debian/RHEL/Fedora/Ubuntu)
  - Windows (Win 7/8/10, Win Server 2008/2012/2016 R2)
  - OS X (10.8+)

- How many 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 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 Nessus Agent to the 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 Whitelist in the appendix for a list of files and processes to whitelist per operating system.

- How do you plan to deploy 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 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 in order to properly ensure successful activation or deactivation of the 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 will you maintain vulnerability information across agent and network scans? How will you 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.io or Nessus Manager.

Deployment Strategy

When deploying a large number of agents, consider using software to push agents through the network. For example:

For Nessus Agents prior to 7.4.2, you should deploy batches of agents over a period of 24 hours when deploying a large amount of agents. This prevents the agents from attempting a full plugin set update at the same time. After an agent is initially installed and gets its first plugin update, it sets its...
timer to attempt the next update 24 hours from that time. As a result, if you deploy 10,000 agents all at once, all of those agents would attempt a full plugin set download at the same time each day, resulting in an excessive amount of bandwidth utilization. Please refer to Plugin Updates for more information on plugin update time frames.

For Nessus Agents 7.4.2 and later, an agent links to Nessus Manager or Tenable.io after a random delay ranging from zero to five minutes. This delay occurs when the agent initially links, and also when the agent is restarted 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 Nessus Manager or Tenable.io.

Clustering

With Nessus Manager clustering, you can deploy and manage large numbers of agents from a single 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 Nessus Manager, rather than needing to link multiple instances of Nessus Manager to Tenable.sc.

A 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 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 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 Nessus Manager.

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

With clustering, you use one Nessus Manager instance to manage 100,000 agents. You enable clustering on 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 Nessus Manager parent node to Tenable.sc.

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 Nessus Manager or Tenable.io and then importing the scan data into Tenable.sc. You can size agent groups when you manage agents in Nessus Manager or Tenable.io.

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

<table>
<thead>
<tr>
<th>Product</th>
<th>Agents Assigned per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tenable.io</td>
<td>Unlimited agents per group if not sending to Tenable.sc</td>
</tr>
<tr>
<td></td>
<td>1,000 agents per group if sending to Tenable.sc</td>
</tr>
<tr>
<td>Nessus Manager</td>
<td>Unlimited agents per group if not sending to Tenable.sc</td>
</tr>
<tr>
<td></td>
<td>20,000 agents per group if sending to Tenable.sc</td>
</tr>
<tr>
<td>Nessus Manager Clusters</td>
<td>Unlimited since scans are automatically broken up as appropriate by separate child nodes.</td>
</tr>
</tbody>
</table>

**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 750 agents in Tenable.io, all in one scan, then data for 2,250 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
You can also add agents to more than one group if you have multiple scanning strategies.

Scan Profile Strategy

Once agents are deployed to all necessary assets, you can create scan profiles and tie them to existing agent groups. A few scan strategies are described below.

**Operating System Scan strategy**

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

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 will only scan these assets.

### 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.

### Basic Agent Scan - Production Servers

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

### 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 will only scan workstation assets.

<table>
<thead>
<tr>
<th>Name</th>
<th>Basic Agent Scan - Workstations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Folder</td>
<td>My Scans</td>
</tr>
<tr>
<td>Agent Groups</td>
<td>Workstations</td>
</tr>
<tr>
<td>Scan Window</td>
<td>3 hours</td>
</tr>
</tbody>
</table>

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

**Note:** Workstation scans may want to be configured for longer scan windows, as most organizations cannot guarantee when these systems will be online (as opposed to servers which are typically on 24/7).

**Scan Staggering**

While scans with the 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 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. Virtual environments that are over-subscribed or resource-limited 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 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 Nessus Agent data has been 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. You should limit the number of agents per scan in Nessus Manager or Tenable.io to 1,000 agents. If you import large amounts of data to Tenable.sc while parallel operations are occurring, Tenable.sc performance may be impacted.

- 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 Nessus Agents.

- **Hardware**
- **Software**
- **Dataflow**
- **Licensing**
- **Agent CPU Resource Control**
- **Performance Metrics**
  - **Nessus Agents Performance**
    - **Host System Utilization**
    - **Software Footprint**
    - **Agent Lifecycle & Bandwidth**
  - **Nessus Manager Performance**

**Note:** For more details about Nessus Agent installation limitations, see the Deployment Strategy video.
Hardware Requirements

Nessus Agents

Nessus Agents are designed to be lightweight and to use only minimal system resources. Generally, a Nessus Agent uses 40 MB of RAM (all pageable). A 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 Nessus Agent resource usage, see [Nessus Agents Performance](#).

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

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Minimum Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>1 Dual-core CPU</td>
</tr>
<tr>
<td>Processor Speed</td>
<td>&gt; 1 Ghz</td>
</tr>
<tr>
<td>RAM</td>
<td>&gt; 1 GB</td>
</tr>
<tr>
<td>Disk Space</td>
<td>Agents 7.7.x and earlier: &gt; 1 GB</td>
</tr>
<tr>
<td></td>
<td>Agents 8.0.x and later: &gt; 3 GB</td>
</tr>
<tr>
<td></td>
<td>More space may be required during certain processes, such as a plugins-code.db defragmentation operation.</td>
</tr>
<tr>
<td>Disk Speed</td>
<td>15-50 IOPS</td>
</tr>
</tbody>
</table>

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

Nessus Manager

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Minimum Recommended Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nessus Manager with 0-10,000 agents</td>
<td><strong>CPU:</strong> 4 2GHz cores</td>
</tr>
<tr>
<td></td>
<td><strong>Memory:</strong> 16 GB RAM</td>
</tr>
<tr>
<td>Scenario</td>
<td>Minimum Recommended Hardware</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td><strong>Disk space:</strong> 30 GB, not including space used by the host operating system.</td>
</tr>
<tr>
<td>Note</td>
<td>Scan results and plugin updates require more disk space over time.</td>
</tr>
<tr>
<td></td>
<td><strong>CPU:</strong> 8 2GHz cores</td>
</tr>
<tr>
<td></td>
<td><strong>Memory:</strong> 64 GB RAM</td>
</tr>
<tr>
<td></td>
<td><strong>Disk space:</strong> 30 GB, not including space used by the host operating system.</td>
</tr>
<tr>
<td>Note</td>
<td>Scan results and plugin updates require more disk space over time.</td>
</tr>
<tr>
<td>Note</td>
<td>Engage with your Tenable representative for large deployments.</td>
</tr>
</tbody>
</table>
Software Requirements

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

SE Linux Requirements


- Disabled and permissive mode policies typically do not require customization to interact with Nessus Agents and Nessus Manager.
- Enforcing mode policies require customization to interact with Nessus Agents and Nessus Manager. For more information, see Customize SE Linux Enforcing Mode Policies for Nessus Agents and Nessus Manager.

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

Nessus Agents

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Supported Versions</th>
</tr>
</thead>
</table>
| 32-bit Linux     | • Debian 9 and 10 / Kali Linux 2017.3  
                  | • Debian 9 and 10 / Kali Linux 2017.3, 2018, 2019, 2020  
                  | • Red Hat ES 6 / CentOS 6 / Oracle Linux 6 (including Unbreakable Enterprise Kernel)  
                  | • SUSE Enterprise 11  
                  | • Ubuntu 14.04 and 16.04 |
| 64-bit Linux     | • Amazon Linux 2015.03, Amazon Linux 2015.09, Amazon Linux 2017.09, Amazon Linux 2018.03, and Amazon Linux 2  
                  | • Debian 9 and 10 / Kali Linux 2017.3  
                  | • Debian 9 and 10 / Kali Linux 2017.3, 2018, 2019, 2020  
<pre><code>              | • Fedora 20, Fedora 21, Fedora 24, Fedora 25, Fedora 26, Fedora 27, |
</code></pre>
<table>
<thead>
<tr>
<th>Operating System</th>
<th>Supported Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fedora 31, Fedora 32</td>
</tr>
<tr>
<td></td>
<td>• Red Hat ES 6 / CentOS 6 / Oracle Linux 6 (including Unbreakable Enterprise Kernel)</td>
</tr>
<tr>
<td></td>
<td>• Red Hat ES 7 / CentOS 7 / Oracle Linux 7 (including Unbreakable Enterprise Kernel)</td>
</tr>
<tr>
<td></td>
<td>• Red Hat ES 8 / CentOS 8 / Oracle Linux 8 (including Unbreakable Enterprise Kernel)</td>
</tr>
<tr>
<td></td>
<td>• SUSE Enterprise 11, 12, 15</td>
</tr>
<tr>
<td></td>
<td>• Ubuntu 14.04, 16.04, 18.04, and 20.04</td>
</tr>
<tr>
<td>32-bit Windows</td>
<td>• Windows 7 SP1, Windows 8.1, and Windows 10</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 SP1, Windows 8.1 with April 2014 update, and Windows 10</td>
</tr>
<tr>
<td>Mac OS X</td>
<td>• 10.9 - 10.15, 11.x</td>
</tr>
<tr>
<td></td>
<td>Provides support for the ARM-based Apple M1 Processor.</td>
</tr>
<tr>
<td>AArch64 Linux</td>
<td>• Amazon Linux 2</td>
</tr>
<tr>
<td></td>
<td>Provides support for ARM platforms, including AWS Graviton2.</td>
</tr>
</tbody>
</table>

**Note:** Nessus Agent 8.2.0 and later requires Windows host systems to be running the Universal Microsoft C Runtime Library (UCRT). Some older versions of Microsoft Windows require a minimum update to work with Nessus Agent 8.2.0 and later. Current Agent installations on Windows that do not meet these requirements will not automatically upgrade past Agent 8.1.0.
## Port Requirements

<table>
<thead>
<tr>
<th>Port</th>
<th>Traffic from</th>
<th>Traffic to</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP 443</td>
<td>Standalone Nessus or Nessus Manager</td>
<td>Tenable (plugins.nessus.org, plugins-customers.nessus.org, or plugins-us.nessus.org)</td>
<td>Update plugins</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Note: Offline updates are also available if Nessus Manager does not have internet access.</td>
</tr>
<tr>
<td>TCP 443</td>
<td>Nessus Agents</td>
<td>Tenable.io (*.cloud.tenable.com)</td>
<td>Pull plugin updates and scan configurations; push scan results</td>
</tr>
<tr>
<td>TCP 443</td>
<td>Tenable.sc</td>
<td>Tenable.io (cloud.tenable.com, downloads-agent.cloud.tenable.com, uploads-agent.cloud.tenable.com)</td>
<td>Push scan configurations and pull scan results</td>
</tr>
<tr>
<td>TCP 8834 (customizable)</td>
<td>Management Workstation</td>
<td>Nessus or Nessus Manager</td>
<td>Nessus or Nessus Manager Administrative GUI</td>
</tr>
<tr>
<td>TCP 8834 (customizable)</td>
<td>Nessus Agents</td>
<td>Nessus Manager</td>
<td>Pull plugin updates and scan configurations; push scan results</td>
</tr>
<tr>
<td>TCP 8834 (customizable)</td>
<td>Tenable.sc</td>
<td>Nessus</td>
<td>Push plugin updates and scan configurations; pull scan results</td>
</tr>
<tr>
<td>TCP 8834 (customizable)</td>
<td>Tenable.sc</td>
<td>Nessus Manager</td>
<td>Pull scan results</td>
</tr>
<tr>
<td>Protocol</td>
<td>Tool</td>
<td>Organization DNS Servers</td>
<td>DNS lookups</td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
<td>---------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>UDP/TCP 53</td>
<td>Nessus</td>
<td>Organization DNS Servers</td>
<td>DNS lookups</td>
</tr>
</tbody>
</table>
Licensing Requirements

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

Nessus Manager

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

It is recommended that you obtain the Activation Code before starting the installation process, as it is required before you can set up Nessus.

Additionally, your activation code:

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

**Note:** For more information about managing Nessus offline, refer to the Nessus User Guide.

You may purchase a Nessus subscription through the Tenable, Inc. online store at https://store.tenable.com/ or via a purchase order through Authorized Nessus Partners. You will then receive an Activation Code from Tenable, Inc.. This code will be used when configuring your copy of Nessus for updates.

**Note:** See the Obtain an Activation Code page to obtain an Activation Code.
If you are using Tenable.sc to manage your Nessus scanners, the Activation Code and plugin updates are managed from Tenable.sc. You must start Nessus before it communicates with Tenable.sc, which it normally does not do without a valid Activation Code and plugins. To have 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 Nessus Agent will depend 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 table below.

<table>
<thead>
<tr>
<th>Preference Value</th>
<th>Windows - Priority Class</th>
<th>Mac OS - Nice Value</th>
<th>Linux - Nice Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal (default)</td>
<td>normal</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>low</td>
<td>below normal</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>high</td>
<td>above normal</td>
<td>-10</td>
<td>-5</td>
</tr>
</tbody>
</table>

Note: Setting your process_priority preference value to low could cause longer running scans. You may need to increase your scan-window time frame 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: Customers with a large number of 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.
<table>
<thead>
<tr>
<th>Name</th>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Valid Values</th>
</tr>
</thead>
</table>
| Plugin Compilation Performance | plugin_load_performance_mode | 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:  
  - low: Less than 50% of a single core used by nessusd during plugin compilation.  
  - medium: 100% of two cores used by nessusd during plugin compilation.  
  - high: Unchanged from previous versions. Will not | high    | low, medium, or high |
<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scan Performance</td>
<td>low, medium, or high</td>
<td>Use more than eight cores.</td>
</tr>
</tbody>
</table>
| scan_performance_mode | high          | 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:  
  - low: Less than 50% of a single core used by nessusd during a scan.  
  - medium: 100% of two cores used by nessusd during a scan.  
  - high: Unchanged from previous versions. Will not 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 Nessus Agents and Nessus Manager:

- **Nessus Agents Performance**
  - Host System Utilization
  - 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 Nessus Agents:

- **Host System Utilization**
- **Software Footprint**
- **Lifecycle & Bandwidth**
Host System Utilization

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

Generally, a Nessus Agent uses 40 MB of RAM (all pageable). A Nessus Agent uses almost no CPU while idle, but is designed to use up to 100% of CPU when available during jobs.

To measure network utilization when uploading results, Tenable monitored Agent uploads into Tenable.io over a 7 day period. Of over 36,000 uploads observed:

- The average size was 1.6 MB.
- The largest size was 37 MB.
- 90% of uploads were 2.2 MB or less.
- 99% of uploads were 5 MB or less.
- Nessus Agent consumes 40 MB of RAM when dormant.
- The Watchdog service consumes 3 MB.
- Plugins consume approximately 300 MB of disk space (varies based on operating system). However, under certain conditions, disk usage can spike up to 2GB, e.g. when a plugins-code.db defragmentation operation is in progress.
- Scan results from Nessus Agents to Nessus Manager and Tenable.io range between 2-3 MB.
- Check-in frequency starts at 30 seconds and is adjusted by Nessus Manager or Tenable.io based on the management system load (number of agents).
Software Footprint

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

<table>
<thead>
<tr>
<th>Agent Footprint on Disk</th>
<th>Total Software Footprint on Disk</th>
<th>RAM Usage While Not Scanning</th>
<th>Average RAM Usage While Scanning/Peak</th>
<th>Network Bandwidth Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.6 MB</td>
<td>800 MB including plugin updates*</td>
<td>&lt;10%</td>
<td>45 MB RAM</td>
<td>~1.5 MB/day**</td>
</tr>
</tbody>
</table>

* Under certain conditions, disk usage can spike up to 2GB. For example, when a plugins-code.db defragmentation operation is in progress.

**Assuming only one scan a day with no plugin updates. Used nethogs program to collect network usage (sent/received) of nessusd. After a single scan that detected 66 vulnerabilities on the agent host, 0.855 MB was sent and received (breakdown: .771 MB sent, .084 MB received). After two total scans, 1.551 MB was sent and 0.204 MB was received. Set to > 1 MB day as the polling for jobs adds up (~0.008 MB per poll).
## Agent Lifecycle & Bandwidth

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

<table>
<thead>
<tr>
<th></th>
<th>Windows</th>
<th>OSX</th>
<th>RedHat/CentOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Core Software Initial Install</td>
<td>24.5 MB</td>
<td>13.6 MB</td>
<td>14.9 MB</td>
</tr>
<tr>
<td>Agent Core Software Updates</td>
<td>17.8 MB</td>
<td>44 MB</td>
<td>11.5 MB</td>
</tr>
<tr>
<td>Initial Plugin Download</td>
<td>49 MB</td>
<td>31 MB</td>
<td>40 MB</td>
</tr>
<tr>
<td>Differential Plugin Updates*</td>
<td>0.1-30 MB</td>
<td>0.1-30 MB</td>
<td>0.1-30 MB</td>
</tr>
<tr>
<td>Report Size**</td>
<td>1-100+ MB</td>
<td>1-100+ MB</td>
<td>1-100+ MB</td>
</tr>
</tbody>
</table>

* 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

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

Testing Environments

The following testing environments were used 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 Nessus Manager and polling for jobs.

<table>
<thead>
<tr>
<th>Number of Agents</th>
<th>Number of Agents Sending Job Requests at a Time (2%)</th>
<th>MAX CPU Usage</th>
<th>Average CPU Usage</th>
<th>Average Agents Page Load Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,000</td>
<td>20</td>
<td>33%</td>
<td>5%</td>
<td>0.60 seconds</td>
</tr>
<tr>
<td>2,000</td>
<td>40</td>
<td>34%</td>
<td>5%</td>
<td>1.05 seconds</td>
</tr>
<tr>
<td>Number of Agents</td>
<td>Number of Agents Sending Job Requests at a Time (5%)</td>
<td>MAX CPU Usage</td>
<td>Average CPU Usage</td>
<td>Average Agents Page Load Time</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------</td>
<td>---------------</td>
<td>-------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>1,000</td>
<td>50</td>
<td>38%</td>
<td>7%</td>
<td>0.88 seconds</td>
</tr>
<tr>
<td>2,000</td>
<td>100</td>
<td>39%</td>
<td>7%</td>
<td>1.14 seconds</td>
</tr>
<tr>
<td>5,000</td>
<td>250</td>
<td>54%</td>
<td>6%</td>
<td>1.73 seconds</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Number of Agents</th>
<th>MAX CPU Usage</th>
<th>Average CPU Usage</th>
<th>Average Agents Page Load Time</th>
<th>Scan Report Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,000</td>
<td>65%</td>
<td>52%</td>
<td>1.16 seconds</td>
<td>363 MB</td>
</tr>
<tr>
<td>2,000</td>
<td>82%</td>
<td>53%</td>
<td>1.45 seconds</td>
<td>726 MB</td>
</tr>
<tr>
<td>3,000</td>
<td>82%</td>
<td>46%</td>
<td>1.67 seconds</td>
<td>1079 MB</td>
</tr>
<tr>
<td>4,000</td>
<td>100%</td>
<td>40%</td>
<td>1.70 seconds</td>
<td>1452 MB</td>
</tr>
<tr>
<td>5,000</td>
<td>86%</td>
<td>47%</td>
<td>1.73 seconds</td>
<td>1780 MB</td>
</tr>
</tbody>
</table>
Install Nessus Agent

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

- Linux
- Windows
- Mac OS X

Once installed, an agent links to Nessus Manager or Tenable.io 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 Nessus Manager or Tenable.io. Agents automatically download plugins from the manager upon linking; this process can take several minutes and is required before an agent can return scan results.
Retrieve the Nessus Agents Linking Key

Before you begin the Nessus Agents installation process, you must retrieve the Nessus Agent Linking Key from Nessus Manager or Tenable.io.

Use this procedure to retrieve the linking key in Nessus Manager or Tenable.io.

To retrieve the linking key in Nessus Manager:

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

   The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.

2. (Optional) In Nessus Manager 8.10 and later, to modify the **Linking Key**, click the **button next to the linking key.

   Examples of when you would want to modify a linking key include:
   
   - You regenerated your linking key and want to revert to a previous linking key.
   - You have a mass deployment script where you want to predefine your linking key.

   **Note:** The linking key must be a 64 character alphanumeric string.

3. Record or copy the **Linking Key**.

To retrieve the linking key in the new Tenable.io interface:

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 **Agents**.

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

5. In the upper-right corner of the page, click **Add Agent**.

   The **Add Agent** plane appears.

6. Click the **Copy** button to copy the **Linking Key**.

   A **Linking key copied to clipboard** confirmation message appears.

What to do next:

- **Install Nessus Agent**.
Install a Nessus Agent on Linux

Caution: If you install a Nessus Agent on a system where an existing Nessus Agent, Nessus Manager, or 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.
- If you previously had the 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.

Example Nessus Agent Package Names

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Example Package Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat, CentOS, and Oracle Linux</td>
<td>NessusAgent-&lt;version number&gt;-es5.x86_64.rpm</td>
</tr>
<tr>
<td></td>
<td>NessusAgent-&lt;version number&gt;-es6.i386.rpm</td>
</tr>
<tr>
<td></td>
<td>NessusAgent-&lt;version number&gt;-es7.x86_64.rpm</td>
</tr>
<tr>
<td>Fedora</td>
<td>NessusAgent-&lt;version number&gt;-fc20.x86_64.rpm</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>NessusAgent-&lt;version number&gt;-ubuntu1110_amd64.deb</td>
</tr>
<tr>
<td></td>
<td>NessusAgent-&lt;version number&gt;-ubuntu1110_i386.deb</td>
</tr>
<tr>
<td></td>
<td>NessusAgent-&lt;version number&gt;-ubuntu910_amd64.deb</td>
</tr>
<tr>
<td></td>
<td>NessusAgent-&lt;version number&gt;-ubuntu910_i386.deb</td>
</tr>
<tr>
<td>Debian</td>
<td>NessusAgent-&lt;version number&gt;-debian6_amd64.deb</td>
</tr>
<tr>
<td></td>
<td>NessusAgent-&lt;version number&gt;-debian6_i386.deb</td>
</tr>
</tbody>
</table>
Install Nessus Agent

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

Using the command line interface, install the Nessus Agent.

**Example Linux Install Commands**

**Red Hat, CentOS, and Oracle Linux**

```
# rpm -ivh NessusAgent-<version number>-es6.i386.rpm
# rpm -ivh NessusAgent-<version number>-es5.x86_64.rpm
```

**Fedora**

```
# rpm -ivh NessusAgent-<version number>-fc20.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 for the purpose of reducing the bandwidth impact during a mass installation. This is accomplished via the `nessuscli agent update` command with the `--file` parameter specifying the location the plugins set. This must be done prior to starting the 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 plugins set older than five days will force a full plugins 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 to Nessus Manager
At the command prompt, use the `nessuscli agent link` command. For example:

```
/opt/nessus_agent/sbin/nessuscli agent link
  --key=00abcd0000efgh1111i0k222lmopq3333st4455u66v777777w88xy9999zabc00
  --name=MyOSXAgent --groups="All" --host=yourcompany.com --port=8834
```

The supported arguments for this command are:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Required?</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--key</code></td>
<td>yes</td>
<td>Use the values you retrieved from the manager.</td>
</tr>
<tr>
<td><code>--host</code></td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td><code>--port</code></td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td><code>--name</code></td>
<td>no</td>
<td>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.</td>
</tr>
<tr>
<td><code>--groups</code></td>
<td>no</td>
<td>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 Nessus Manager or Tenable.io.</td>
</tr>
<tr>
<td><code>--offline-install</code></td>
<td>no</td>
<td>For Nessus Agents 7.0.3 or later, you can install the Nessus Agent on a system even if it is offline. Add the command line option <code>offline-install=&quot;yes&quot;</code> to the command line input. The Nessus Agent will periodically attempt to link itself to either Tenable.io or 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.</td>
</tr>
<tr>
<td><code>--cloud</code></td>
<td>no</td>
<td>Specify the <code>--cloud</code> argument to link to Tenable.io.</td>
</tr>
</tbody>
</table>

**Note:** The agent group name is case-sensitive and must match exactly.
The --cloud argument is a shortcut to specifying --host=t=cloud.tenable.com --port=443.

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 Nessus Manager or Tenable.io, a 409 error may appear. This error appears because another machine has been 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.

**Verify a Linked Agent**

To verify a linked agent in Nessus Manager or Tenable.io:

1. In the top navigation bar, click **Scans**.
   
The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.
   
The **Agents** page appears.

3. Locate the new agent in the linked agents table.
Install a Nessus Agent on Windows

**Caution:** If you install a Nessus Agent on a system where an existing Nessus Agent, Nessus Manager, or 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 Nessus Agents via the command line. You can also deploy 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](#).
- Consider the following if you are reinstalling Nessus Agent after uninstalling it:
  - If you previously had the Nessus Agent installed on your system, see the knowledge base article on how to avoid linking errors.
  - On Windows, the Nessus Agent uninstall process automatically creates a backup file in the %TEMP% directory. If you reinstall Nessus Agent within 24 hours, Nessus Agent uses that backup file to restore the installation. If you want to reinstall 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 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=00abcd0000efgh1111i0k222lmopq3333st4455u66v777777w88xy9999zabc00 /qn
```

**Note:** For more information, see the knowledge base article.
The following are available linking parameters:

- You can install the Nessus Agent on a system even if it is offline. Add the command line option `NESSUS_OFFLINE_INSTALL="yes"` to the command line input. The Nessus Agent will periodically attempt to link itself to either Tenable.io or 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.

- To install the **Nessus Agent System Tray Application**, add the command line option ADDLOCAL-L=ALL to the command line.

- Additionally, you can install a full plugins set before linking for the purpose of reducing the bandwidth impact during a mass installation. Add the command line option `NESSUS_PLUGINS_FILEPATH="C:\path\to\plugins_set.tgz"` where `plugins_set.tgz` 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 [Nessus Agents download page](#).

- **NESSUS_GROUPS** - 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 Nessus Manager or Tenable.io.

  **Note:** The agent group name is case-sensitive and must match exactly.

  **Note:** Quotation marks ("") are necessary when listing multiple groups, or one group with spaces in its name. For example:

  - GroupName
  - "Group Name"
  - "Group, Another Group"

- **NESSUS_NAME**

- **NESSUS_PROXY_AGENT**

- **NESSUS_PROXY_PASSWORD**

- **NESSUS_PROXY_SERVER**

- **NESSUS_PROXY_USERNAME**
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 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 be required to restart your computer to complete installation on Windows.

Note: If you want to include the system tray application in your installation, see the procedure described in 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 Nessus Agents.

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

<table>
<thead>
<tr>
<th>Field</th>
<th>Required?</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>yes</td>
<td>Use the values you <strong>retrieved</strong> from the manager.</td>
</tr>
<tr>
<td>Server (host)</td>
<td>yes</td>
<td>Specify existing agent groups(s) where you want to add the agent.</td>
</tr>
<tr>
<td>Groups</td>
<td>no</td>
<td>If you do not specify an agent group during the install process, you can later add your linked agent to an agent group.</td>
</tr>
</tbody>
</table>

**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 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 Nessus Manager or Tenable.io, a 409 error may appear. This error appears because another machine has been 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.

---

**Verify a Linked Agent**

To verify a linked agent in Nessus Manager or Tenable.io:
1. In the top navigation bar, click **Scans**.
   
   The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.
   
   The **Agents** page appears.

3. Locate the new agent in the linked agents table.
Configure and View the System Tray Application

The installation package for Nessus Agents 7.0.3 and later on Windows includes an optional system tray application that displays 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**.

![System Tray Application Setup Type]

Copyright © 2021 Tenable, Inc. All rights reserved. Tenable, Tenable.io, Tenable Network Security, Nessus, SecurityCenter, SecurityCenter Continuous View and Log Correlation Engine are registered trademarks of Tenable, Inc. Tenable.sc, Tenable.ot, Lumin, Indegy, Assure, and The Cyber Exposure Company are trademarks of Tenable, Inc. All other products or services are trademarks of their respective owners.
The **Custom Setup** page appears. By default, the system tray application is excluded from the installation package.

![Image of Custom Setup page]

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

![Image of 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 Nessus Agent**.

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

- **Nessus agent status**: Displays the state of the Tenable Nessus Agent Service (**Starting**, **Running**, **Paused**, or **Stopped**).

- **Scan status**: Displays whether the agent is running a scan job on the endpoint (**Scanning** or **Idle**).

- **Nessus agent version**: Displays the version of the agent running on the endpoint (e.g., 7.0.3).

- **Pluginset ID**: Displays the current plugin set available on the endpoint (e.g., 201802271350).
Install a Nessus Agent on Mac OS X

Caution: If you install a Nessus Agent on a system where an existing Nessus Agent, Nessus Manager, or 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](#).
- If you previously had the 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: The following steps require root privileges.

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

**GUI Installation:**

1. Double-click the Nessus Agent .dmg (Mac OS X 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.
2. Open Terminal.

3. At the command prompt, enter the following command:

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

You can install a full plugins set before linking for the purpose of reducing the bandwidth impact during a mass installation. This is accomplished via the `nessuscli agent update` command with the `--file` parameter specifying the location the plugins set. This must be done prior to starting the 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 plugins set older than five days will force a full plugins download to occur. You can download a recent plugins set from the Nessus Agents download page.

### Link Agent Using Command Line Interface

To link an agent on a Mac OS X:

1. Open Terminal.

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

For example:

```
# /Library/NessusAgent/run/sbin/nessuscli agent link
--key=00abcd0000efgh1111i0k222lmopq3333st4455u66v777777w88xy9999zabc00
--name=MyOSXAgent --groups=All --host=yourcompany.com --port=8834
```

The supported arguments for this command are:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Required?</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>Required</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>--key</td>
<td>yes</td>
<td>Use the values you retrieved from the manager.</td>
</tr>
<tr>
<td>--host</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>--port</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>--name</td>
<td>no</td>
<td>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.</td>
</tr>
<tr>
<td>--groups</td>
<td>no</td>
<td>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 Nessus Manager or Tenable.io.</td>
</tr>
<tr>
<td>--offline-install</td>
<td>no</td>
<td>For Nessus Agents 7.0.3 or later, you can install the Nessus Agent on a system even if it is offline. Add the command line option <code>NESSUS_OFFLINE_INSTALL-L=&quot;yes&quot;</code> to the command line input. The Nessus Agent will periodically attempt to link itself to either Tenable.io or 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.</td>
</tr>
<tr>
<td>--cloud</td>
<td>no</td>
<td>Specify the <code>--cloud</code> argument to link to Tenable.io. The <code>--cloud</code> argument is a shortcut to specifying <code>--host=cloud.tenable.com --port=443</code>.</td>
</tr>
</tbody>
</table>

**Verify a Linked Agent**

To verify a linked agent in Nessus Manager or Tenable.io:

Note: The agent group name is case-sensitive and must match exactly.
1. In the top navigation bar, click **Scans**.
   
   The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.

   The **Agents** page appears.

3. Locate the new agent in the linked agents table.
Update a Nessus Agent

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

Agent Update Plan

For Tenable.io-linked agents, you can set an agent update plan to determine the version that the agent automatically updates to. Set the agent update plan from the command line interface.

To set the agent update plan for Tenable.io-linked agents:

1. Log in to the agent from a command terminal.
2. Enter the command:

   ```
   nessuscli fix --set agent_update_channel=<value>
   ```

Use one of the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ga</td>
<td>Automatically updates to the latest Agent version when it is made generally available (GA).</td>
</tr>
<tr>
<td>ea</td>
<td>Automatically updates to the latest Agent version as soon as it is released for Early Access (EA), typically a few weeks before general availability.</td>
</tr>
<tr>
<td>stable</td>
<td>Does not automatically update to the latest Nessus Agent version. Remains on an earlier version of Nessus Agent set by Tenable, usually one release older than the current generally available version, but no earlier than 7.7.0. When Nessus Agent releases a new version, your agent updates software versions, but stays on a version prior to the latest release.</td>
</tr>
</tbody>
</table>

Note: For Tenable.io-linked agents, this date is usually one week after the version is made generally available. For versions that address critical security issues, Tenable may make the version available immediately.
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.io-linked agents update to the generally available (GA) version one week after the version is GA. Therefore, if you manually update a Tenable.io-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 **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.
5. Do one of the following, depending on your operating system:

   **Windows**

   **Note:** Administrator-level privileges are required.

   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:

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

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

```
# rpm -Uvh <path-to>/NessusAgent-<version>.rpm
```

**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 Nessus Agent.
Downgrade Nessus Agent

Nessus Agent 7.7.0 and later supports the ability to downgrade Nessus to a previous version of Nessus. You cannot downgrade to a version prior to 7.7.0.

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, 8.0.0, and now want to downgrade to the previous version, 7.7.0.

Solution:
1. Turn off automatic software updates by doing any of the following:
   - On Nessus Manager, disable the advanced setting Automatically Download Agent Updates, or agent_updates_from_feed.
   - On Tenable.io, 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 7.7.0.

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

Scenario:
Your agent update plan determines what version Nessus Agent updates to, if you have automatic updates enabled. In this scenario, your update plan is set to ga, meaning the agent automatically
updates to the latest generally available (GA) release. You are currently on a GA version of Nessus Agent; for example, 8.0.0.

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. Nessus Agent automatically downgrades to 7.7.0, one release prior to the latest GA version.
Back Up Agent

Using **Nessus CLI Agent Commands**, you can back up your Nessus Agent to restore it later on any system, even if it is a different operating system. When you back up Nessus Agent, your settings are preserved. 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 Nessus Agent, you must reconfigure any Nessus Agent configurations that use schedules (for example, scan schedules or agent blackout windows). Schedules do not transfer correctly across these platforms because the operating systems use different timezone names.

To back up Nessus Agent:

1. **Access Nessus Agent from a command terminal.**

2. **Create the Nessus Agent backup file:**

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

   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 Nessus Agent to restore later on any system, even if it is a different operating system. When you back up Nessus Agent, your settings are preserved. 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 Nessus Agent, you must reconfigure any Nessus Agent configurations that use schedules (for example, scan schedules or agent blackout windows). Schedules do not transfer correctly across these platforms because the operating systems use different timezone names.

Before you begin:

- [Back Up Agent](#)

To restore Nessus Agent:

1. **Access** Nessus Agent from a command terminal.
2. **Stop** your Nessus Agent service.
   
   For example:
   
   ```
   # /sbin/service nessusagent stop
   ```
   
   Nessus Agent terminates all processes.
3. **Restore** Nessus Agent from the backup file you previously saved:
   
   ```
   > nessuscli backup --restore path/to/<backup_filename>
   ```
   
   Nessus Agent restores your backup.
4. **Stop and start** your Nessus Agent service.
   
   For example:
   
   ```
   # /sbin/service nessusagent stop
   ```
# /sbin/service nessusagent start

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

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

- [Uninstall a Nessus Agent on Linux](#)
- [Uninstall a Nessus Agent on Windows](#)
- [Uninstall a Nessus Agent on Mac OS X](#)

**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 Nessus Agent on Linux:

1. From a command prompt, determine your package name.

   Example Nessus Package Name Determination Commands

   Red Hat, CentOS, Oracle Linux, Fedora, SUSE, FreeBSD
   
   ```
   # rpm -qa | grep -i NessusAgent
   
   Debian/Kali and Ubuntu
   
   # dpkg -l | grep -i NessusAgent
   
   FreeBSD
   
   # pkg_info | grep -i NessusAgent
   ```

2. Using the package name identified, type the remove command specific to your Linux-style operating system.

   Example Nessus Agent Remove Commands

   Red Hat, CentOS, Oracle Linux, Fedora, SUSE
   
   ```
   # rpm -e <Agent package name>
   
   Debian/Kali and Ubuntu
   
   # dpkg -r <Agent package name>
   
   FreeBSD
   
   # pkg delete <Agent package name>
   ```

What to do next:
• If you plan on reinstalling the 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 Nessus Agent on Windows:

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 Nessus Agent.
4. Click [Yes](#).

   Windows deletes all Nessus related files and folders.

   **Note:** On Windows, the Nessus Agent uninstall process automatically creates a backup file in the %TEMP% directory. If you reinstall Nessus Agent within 24 hours, Nessus Agent uses that backup file to restore the installation. If you want to reinstall Nessus Agent within 24 hours without using the backup, manually delete the backup file in the %TEMP% directory beforehand.

What to do next:

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

Before you begin:

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

To uninstall Nessus Agent on Mac OS X:

1. Remove the Nessus directories. Using [Finder](#), locate and delete the following items.
   - `/Library/NessusAgent`
   - `/Library/LaunchDaemons/com.tenablesecurity.nessusagent.plist`
   - `/Library/PreferencePanes/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 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

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

Mac OS X

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.

<table>
<thead>
<tr>
<th>Start or Stop</th>
<th>Mac OS X Command Line Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td><code># launchctl load -w /Library/LaunchDae-mons/com.tenablesecurity.nessusagent.plist</code></td>
</tr>
<tr>
<td>Stop</td>
<td><code># launchctl unload -w /Library/LaunchDae-mons/com.tenablesecurity.nessusagent.plist</code></td>
</tr>
</tbody>
</table>

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**.
### Windows Command Line Operation

<table>
<thead>
<tr>
<th>Start or Stop</th>
<th>Windows Command Line Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td><code>C:\Windows\system32&gt;net start &quot;Tenable Nessus Agent&quot;</code></td>
</tr>
<tr>
<td>Stop</td>
<td><code>C:\Windows\system32&gt;net stop &quot;Tenable Nessus Agent&quot;</code></td>
</tr>
</tbody>
</table>

### Linux Command Line Operation

**Use the following commands:**

<table>
<thead>
<tr>
<th>Start or Stop</th>
<th>Linux Command Line Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RedHat, CentOS, and Oracle Linux</strong></td>
<td></td>
</tr>
<tr>
<td>Start</td>
<td><code>#/sbin/service nessusagent start</code></td>
</tr>
<tr>
<td>Stop</td>
<td><code>#/sbin/service nessusagent stop</code></td>
</tr>
<tr>
<td><strong>SUSE</strong></td>
<td></td>
</tr>
<tr>
<td>Start</td>
<td><code>#/etc/rc.d/nessusagent start</code></td>
</tr>
<tr>
<td>Stop</td>
<td><code>#/etc/rc.d/nessusagent stop</code></td>
</tr>
<tr>
<td><strong>FreeBSD</strong></td>
<td></td>
</tr>
<tr>
<td>Start</td>
<td><code>#/etc/init.d/nessusagent start</code></td>
</tr>
<tr>
<td>Stop</td>
<td><code>#/etc/init.d/nessusagent stop</code></td>
</tr>
<tr>
<td><strong>Debian, Kali, and Ubuntu</strong></td>
<td></td>
</tr>
<tr>
<td>Start</td>
<td><code>#/etc/init.d/nessusagent start</code></td>
</tr>
<tr>
<td>Stop</td>
<td><code>#/etc/init.d/nessusagent stop</code></td>
</tr>
</tbody>
</table>
Agent Status

Nessus Agents can be in one of the following statuses:

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Online</td>
<td>The host that contains the Nessus Agent is currently connected and in communication with Nessus Manager or Tenable.io.</td>
</tr>
<tr>
<td>Offline</td>
<td>The host that contains the Nessus Agent is currently powered down or not connected to a network.</td>
</tr>
<tr>
<td>Initializing</td>
<td>The Nessus Agent is in the process of checking in with Nessus Manager or Tenable.io.</td>
</tr>
<tr>
<td>Unlinked</td>
<td>(Nessus Manager only) The agent is in an unlinked state. Agents with this status are only present if <strong>Track unlinked agents</strong> is enabled.</td>
</tr>
</tbody>
</table>

**Note:** Agents that are automatically unlinked via the **Unlink inactive agents after X days** setting can automatically relink to Nessus Manager if they come back online. Agents that are manually unlinked must be manually relinked.
Filter Agents

Use this procedure to filter agents in Nessus Manager or Tenable.io.

To filter agents in the agents table in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.
   The **Agents** page appears.

3. Above the agents table, click the **Filter** button.
   The **Filter** window appears.

4. Configure the filters as necessary. For more information, see **Agent Filters**.

5. Click **Apply**.
   The manager filters the list of agents to include only those that match your configured options.

To filter agents in the agents table in Nessus Manager 8.12 and later:

1. In the top navigation bar, click **Sensors**.
   The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.

2. Above the agents table, click the **Filter** button.
   The **Filter** window appears.

3. Configure the filters as necessary. For more information, see **Agent Filters**.

4. Click **Apply**.
   Nessus Manager filters the list of agents to include only those that match your configured options.

Agent Filters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Operator</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Address</td>
<td>is equal to, is not equal to, contains, does not contain</td>
<td>In the text box, type the IPv4 or IPv6 addresses on which you want to filter.</td>
</tr>
<tr>
<td>Last Connection</td>
<td>earlier than, later than on, not on</td>
<td>In the text box, type the date on which you want to filter.</td>
</tr>
<tr>
<td>Last Plugin Update</td>
<td>is equal to, is not equal to, contains, does not contain</td>
<td>From the drop-down list, select from your existing agent groups.</td>
</tr>
<tr>
<td>Last Scanned</td>
<td>is equal to, is not equal to, contains, does not contain</td>
<td>In the text box, type the agent name on which you want to filter.</td>
</tr>
<tr>
<td>Platform</td>
<td>contains, does not contain</td>
<td>In the text box, type the platform name on which you want to filter.</td>
</tr>
<tr>
<td>Status</td>
<td>is equal to, is not equal</td>
<td>In the drop-down list, select an agent status.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Operator</td>
<td>Expression</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>------------</td>
</tr>
<tr>
<td>to</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>is equal to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>is not equal to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>contains</td>
<td></td>
</tr>
<tr>
<td></td>
<td>does not contain</td>
<td>In the text box, type the version you want to filter.</td>
</tr>
</tbody>
</table>
Export Agents

To export agents data in Nessus Manager or Tenable.io:

1. In the top navigation bar, click **Scans**.
   
The My Scans page appears.

2. In the left navigation bar, click **Agents**.
   
The Agents page appears.

3. (Optional) Click the **Filter** button to apply a filter to the agents list.
   
   **Note:** You can only filter agent data for export in Nessus Manager. Tenable.io will export all data even if filters are applied.

4. In the upper right corner, click **Export**. If a drop-down appears, click **CSV**.
   
   Your browser's download manager appears.

5. Click **OK** to save the agents.csv file.

The agents.csv file exported from Nessus Manager contains the following data:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Name</td>
<td>The name of the agent</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the agent at the time of export. Possible values are unlinked, online, or offline.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IPv4 or IPv6 address of the agent.</td>
</tr>
<tr>
<td>Platform</td>
<td>The platform the agent is installed on.</td>
</tr>
<tr>
<td>Groups</td>
<td>The names of any groups the agent belongs to.</td>
</tr>
<tr>
<td>Version</td>
<td>The version of the agent.</td>
</tr>
<tr>
<td>Last Plugin Update</td>
<td>The date (in ISO-8601 format) the agent's plugin set was last updated.</td>
</tr>
<tr>
<td>Last Scanned</td>
<td>The date (in ISO-8601 format) the agent last performed a scan of the host.</td>
</tr>
</tbody>
</table>
The `agents.csv` file exported from Tenable.io contains the following data:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Name</td>
<td>The name of the agent.</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the agent at the time of export. Possible values are unlinked, online, or offline.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IPv4 or IPv6 address of the agent.</td>
</tr>
<tr>
<td>Platform</td>
<td>The platform the agent is installed on.</td>
</tr>
<tr>
<td>Groups</td>
<td>The names of any groups the agent belongs to.</td>
</tr>
<tr>
<td>Group IDs</td>
<td>The group IDs of any groups the agent belongs to.</td>
</tr>
<tr>
<td>Version</td>
<td>The version of the agent.</td>
</tr>
<tr>
<td>Last Plugin Update</td>
<td>The date (in ISO-8601 format) the agent's plugin set was last updated.</td>
</tr>
<tr>
<td>Agent ID</td>
<td>The ID of the agent.</td>
</tr>
<tr>
<td>Agent UUID</td>
<td>The UUID of the agent.</td>
</tr>
<tr>
<td>Linked On</td>
<td>The date (in ISO-8601 format) the agent was linked to Tenable.io.</td>
</tr>
<tr>
<td>Last Connect</td>
<td>The date (in ISO-8601 format) of the agent's last check-in.</td>
</tr>
<tr>
<td>Last Scanned</td>
<td>The date (in ISO-8601 format) the agent was last scanned.</td>
</tr>
</tbody>
</table>
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 Nessus Manager or Tenable.io.

Tip: You can configure agents to automatically unlink if they are inactive for a certain number of days, as described in agent settings.

To manually unlink a single agent:

1. In the top navigation bar, click Scans.
   The My Scans page appears.

2. In the left navigation bar, click Agents.
   The Agents page appears.

3. Do one of the following:
   • To unlink a single agent:
     a. In Tenable.io: In the agents table, in the row for the agent that you want to unlink, click the button.

     Note: For Nessus Manager 7.1.0 and earlier, in the row for the agent that you want to unlink, click the button.

     b. In Tenable.io: In the agents table, in the row for the agent that you want to unlink, click the button.

     A confirmation window appears.
   • To unlink multiple agents:
     a. In the agents table, select the check box in each row for each agent you want to unlink.
b. In the upper-right corner, click **Unlink**.

A confirmation window appears.

4. Click the **Unlink** button.

The manager unlinks the agent.
Agent Groups

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

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

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.

To manage agent groups, use the following procedures:

- Create a New Agent Group
- Configure User Permissions for an Agent Group
- Modify an Agent Group
- Delete an Agent Group
Create a New Agent Group

You can use agent groups to organize and manage the agents linked to your account. You can add an agent to more than one group, and configure scans to use these groups as targets.

Use this procedure to create an agent group in Nessus Manager or Tenable.io.

To create a new agent group in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   
   The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.
   
   The **Agents** page appears.

3. Click the **Agent Groups** tab.

4. In the upper right corner, click the **New Group** button.
   
   The **New Agent Group** window appears.

5. In the **Name** box, type a name for the new agent group.

6. (Optional) Add agents to the group:
   
   a. Click the **Add Members** toggle.
      
      A list of linked agents appears.

   b. Click each agent you want to add to the group.

7. Click **Save**.

   The manager adds the agent group and it appears in the table.

To create a new agent group in Nessus Manager 8.12 and later:

1. In the top navigation bar, click **Sensors**.
   
   The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.

2. In the left navigation bar, click **Agent Groups**.
The Agent Groups page appears.

3. In the upper right corner, click the New Group button.

   The New Agent Group window appears.

4. In the Name box, type a name for the new agent group.

5. Click Add.

   Nessus Manager adds the agent group and it appears in the table.

What to do next:

- Configure user permissions for the agent group.
- Use the agent group in an agent scan configuration.
Configure User Permissions for an Agent Group

You can share an agent group with other users or user groups in your organization.

User permissions for agent groups include the following:

- **No access** – (Default user only) The user or user group cannot add the agent group to an agent scan. If a user or user group with this permission attempts to launch an existing scan that uses the agent group, the scan fails.

- **Can use** – The user or user group can add the agent group to an agent scan and can launch existing scans that use the agent group.

Use this procedure to configure permissions for an agent group in Nessus Manager or Tenable.io.

To configure user permissions for an agent group:

1. Create or modify an agent group.
2. In the agent groups table, click the agent group for which you want to configure permissions.
   
   The agent group details page appears.
3. Click the Permissions tab.
   
   The Permissions tab appears.
4. Do any of the following:

   **Tip:** Tenable recommends assigning permissions to user groups, rather than individual users, to minimize maintenance as individual users leave or join your organization.

   - Add permissions for a new user or user group:
     a. In the Add users or groups box, type the name of a user or group.
        
        As you type, a filtered list of users and groups appears.
     b. Select a user or group from the search results.
        
        Tenable.io adds the user to the permissions list, with a default permission of Can Use.
• Change the permissions for an existing user or user group:

  **Note:** The Default user represents any users who have not been specifically added to the agent group.

  a. Next to the permission drop-down for the Default user, click the ▼ button.
  
  b. Select a permissions level.
  
  c. Click **Save**.

• Remove permissions for a user or user group:
  
  • For the Default user, set the permissions to **No Access**.
  
  • For any other user or user group, click the × button next to the user or user group for which you want to remove permissions.

5. Click **Save**.

  Tenable.io saves the changes you made to the agent group.
Modify an Agent Group

Use this procedure to modify an agent group in Nessus Manager or Tenable.io.

To modify an agent group in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   The *My Scans* page appears.

2. In the left navigation bar, click **Agents**.
   The *Agents* page appears.

3. Click the **Agent Groups** tab.

4. Do any of the following:

   • **Modify the group name.**
     a. In the row for the agent group that you want to modify, click the **button.
        The *Edit Agent Group* window appears.
     b. In the **Name** box, type a new name for the agent group.
     c. Click **Save**.
        The manager saves your changes.

   • **Add agents to the agent group.**
     a. In the agent groups table, click the agent group you want to modify.
        The agent group details page appears.
     b. In the upper-right corner of the page, click the **Add Agents** button.
        The *Add Agents* window appears. This window contains a table of available agents.
     c. (Optional) In the **Search** box, type the name of an agent, then click **Enter**.
        The table of agents refreshes to display the agents that match your search criteria.
d. Click the check box next to each agent you want to add to the group.

e. Click **Add**.

The manager adds the selected agent or agents to the group.

• **Remove agents from the agent group.**

  a. In the agent groups table, click the agent group you want to modify.

      The agent group details page appears. By default, the tab is active.

  b. (Optional) Filter the agent groups in the table.
  c. (Optional) Search for an agent by name.

  d. Select the agent or agents you want to remove:

      • For an individual agent, click the **X** button next to the agent.

      • For multiple agents, select the check box next to each, then click the **Remove** button in the upper-right corner of the page.

      A confirmation window appears.

  e. In the confirmation window, confirm the removal.

• **Modify the user permissions for the agent group.**

  a. In the agent groups table, click the agent group you want to modify.

      The agent group details page appears.

  b. Click the **Permissions** tab.

      The **Permissions** tab appears.

  c. **Configure** the user permissions for the group.

To modify an agent group in Nessus Manager 8.12 and later:

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

      The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.
2. In the left navigation bar, click **Agent Groups**.

   The **Agent Groups** page appears.

3. Do any of the following:

   - **Modify the group name.**
     a. In the row for the agent group that you want to modify, click the button.

        The **Edit Agent Group** window appears.

     b. In the **Name** box, type a new name for the agent group.

     c. Click **Save**.

        The manager saves your changes.

   - **Add agents to the agent group.**
     a. In the agent groups table, click the agent group you want to modify.

        The agent group details page appears.

     b. In the upper-right corner of the page, click the **Add Agents** button.

        The **Add Agents** window appears. This window contains a table of available agents.

     c. (Optional) In the **Search** box, type the name of an agent, then click **Enter**.

        The table of agents refreshes to display the agents that match your search criteria.

     d. Click the check box next to each agent you want to add to the group.

     e. Click **Add**.

        The manager adds the selected agent or agents to the group.

   - **Remove agents from the agent group.**
     a. In the agent groups table, click the agent group you want to modify.

        The agent group details page appears. By default, the tab is active.
b. (Optional) Filter the agent groups in the table.

c. (Optional) Search for an agent by name.

d. Select the agent or agents you want to remove:
   
   - For an individual agent, click the ✗ button next to the agent.
   
   - For multiple agents, select the check box next to each, then click the Remove button in the upper-right corner of the page.

   A confirmation window appears.

e. In the confirmation window, confirm the removal.

   • Modify the user permissions for the agent group.
      
      a. In the agent groups table, click the agent group you want to modify.

         The agent group details page appears.

      b. Click the Permissions tab.

         The Permissions tab appears.

      c. Configure the user permissions for the group.
Delete an Agent Group

Use this procedure to delete an agent group in Nessus Manager or Tenable.io.

To delete an agent group in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   
   The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.
   
   The **Agents** page appears.

3. Click the **Groups** tab.

4. In the row for the agent group that you want to delete, click the **X** button.
   
   A confirmation window appears.

5. To confirm, click **Delete**.
   
   The manager deletes the agent group.

To modify an agent group in Nessus Manager 8.12 and later:

1. In the top navigation bar, click **Sensors**.
   
   The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.

2. In the left navigation bar, click **Agent Groups**.
   
   The **Agent Groups** page appears.

3. In the row for the agent group that you want to delete, click the **X** button.
   
   A confirmation window appears.

4. To confirm, click **Delete**.
   
   The manager deletes the agent group.
Scans

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

See the following topics:

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

To create an agent scan:

1. In the top navigation bar, click **Scans**.
   
   The **My Scans** page appears.

2. In the upper right corner, click the **New Scan** button.
   
   The **Scan Templates** page appears.

3. Click the **Agent** tab.
   
   The **Agent** scan templates page appears.

4. Click the **scan template** that you want to use.

   **Tip:** Use the search box in the top navigation bar to filter templates on the tab currently in view.

5. Configure the scan’s settings.

   For more information about scan settings in Nessus Manager, see [Scan and Policy Settings](#).

   For more information about scan settings in Tenable.io, see [Scan and Policy Settings (Classic Interface)](#).

6. (Optional) Configure **compliance checks** for the scan.

7. (Optional) Configure security checks by **plugin family or individual plugin**.

8. Do one of the following:

   - If you want to launch the scan later, click the **Save** button.
     
     Nessus saves the scan.

   - If you want to launch the scan immediately:
     
     a. Click the ☐ button.

     b. Click **Launch**.

     Nessus saves and launches the scan.
Agent Scan and Policy Templates

You can use templates to create an agent scan or policy.

In both Nessus Manager and Tenable.io, default templates for agent scans appear in the **Agent** tab. The manager interface provides brief explanations of each default template.

**Note:** If you create custom policies for agent scans, those templates appear in the **User Defined** tab.

The table below briefly describes the settings for the default agent scan templates. You may also have access to **special templates**.

For a comprehensive explanation of template settings, see the documentation for Nessus Manager or Tenable.io.

Agent Templates

Agent templates fall into two categories: **Vulnerabilities** and **Compliance**.

<table>
<thead>
<tr>
<th>Template</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Vulnerabilities</strong></td>
<td></td>
</tr>
<tr>
<td>Advanced Agent Scan</td>
<td>Scans without any recommendations.</td>
</tr>
<tr>
<td><strong>Note:</strong> When you create an agent scan using the Advanced Agent Scan template, you must also select the plugins you want to use for the scan.</td>
<td></td>
</tr>
<tr>
<td>Basic Agent Scan</td>
<td>Scans systems connected via Nessus Agents.</td>
</tr>
<tr>
<td>Malware Scan</td>
<td>Scans for malware on systems connected via Nessus Agents.</td>
</tr>
<tr>
<td><strong>Compliance</strong></td>
<td></td>
</tr>
<tr>
<td>Policy Compliance Auditing</td>
<td>Audits systems connected via Nessus Agents.</td>
</tr>
<tr>
<td>Template</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>SCAP and OVAL Auditing</td>
<td>Audits systems using SCAP and OVAL definitions.</td>
</tr>
</tbody>
</table>
Compliance

For the **Advanced Agent Scan** templates, **Policy Compliance Auditing** templates, and certain **Special Use Templates**, you can configure one or more of the following compliance checks:

<table>
<thead>
<tr>
<th>Audit Capability</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unix</td>
<td>A predefined or custom audit policy file to be specified to test Unix servers against compliance standards.</td>
</tr>
<tr>
<td>Unix File Contents</td>
<td>A predefined or custom audit policy file to be specified to test Unix servers for sensitive content such as SSN, credit cards etc.</td>
</tr>
<tr>
<td>Windows</td>
<td>A predefined or custom audit policy file to be specified to test Windows servers against compliance standards.</td>
</tr>
<tr>
<td>Windows File Contents</td>
<td>A predefined or custom audit policy file to be specified to test Windows servers for sensitive content such as SSN, credit cards etc.</td>
</tr>
</tbody>
</table>
Plugins

For Advanced Agent Scan templates, you can select security checks by Plugin Family or by individual plugin.

**Note:** If a plugin requires authentication or settings to communicate with another system, the plugin is not available for Nessus Agents. This includes, but is not limited to, plugins in the following areas:

- Patch management
- Mobile device management
- Cloud infrastructure audit
- Database checks that require authentication

To disable or enable an entire plugin family for a scan, click the Status toggle next to the family in the left column of the table in the Plugins tab.

To disable or enable an individual plugin for a scan:

a. In the left column of the table in the Plugins tab, click the family where the individual plugin belongs.

   The plugins for that family appear in the right column of the table.

b. In the right column of the table, click the Status toggle next to an individual plugin.

   If you disable one or more of the plugins in the right column, the status toggle for the family changes to **Mixed** and appears blue.

   If you disable all of the plugins in the right column, the status toggle for the family changes to **Disabled** and appears grey.

**Tip:** To view details of an individual plugin, click on the plugin in the right column of the table. The plugin details include a **Synopsis**, **Description**, **Solution**, **Plugin Information**, and **Risk Information**.
Special Use Templates

**Note:** For more information about performing custom audits with Nessus, see the [Custom Auditing video](#).

## Compliance

Nessus compliance auditing can be configured using one or more of the following **Scanner** and **Agent** templates.

- Audit Cloud Infrastructure
- MDM Config Audit
- Offline Config Audit
- SCAP and OVAL Auditing
- Policy Compliance Auditing

## Mobile Device

With Nessus Manager, the Nessus Mobile Devices plugin family provides the ability to obtain information from devices registered in a Mobile Device Manager (MDM) and from Active Directory servers that contain information from Microsoft Exchange Servers.

- To query for information, the Nessus scanner must be able to reach the Mobile Device Management servers. You must ensure no screening devices block traffic to these systems from the Nessus scanner. In addition, Nessus must be given administrative credentials (e.g., domain administrator) to the Active Directory servers.

- To scan for mobile devices, Nessus must be configured with authentication information for the management server and the mobile plugins. Since Nessus authenticates directly to the management servers, a scan policy does not need to be configured to scan specific hosts.

- For ActiveSync scans that access data from Microsoft Exchange servers, Nessus will retrieve information from phones that have been updated in the last 365 days.

## Payment Card Industry (PCI)
Tenable offers two Payment Card Industry Data Security Standard (PCI DSS) templates: one for testing internal systems (11.2.1) and one for Internet facing systems (11.2.2). Also, these scan templates may also be used to complete scans after significant changes to your network, as required by PCI DSS 11.2.3.

<table>
<thead>
<tr>
<th>Template</th>
<th>Product</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCI Quarterly External Scan</td>
<td>Tenable.io Only</td>
<td>The PCI Quarterly External Scan template is only available in Tenable.io. Using this template, Tenable.io tests for all PCI DSS external scanning requirements, including web applications. The scan results obtained using the PCI Quarterly External Scan template may be submitted to Tenable, Inc. (an Approved Scanning Vendor) for PCI validation. Refer to the Scan Results section for details on creating, reviewing, and submitting PCI scan results.</td>
</tr>
<tr>
<td>PCI Quarterly External Scan (Unofficial)</td>
<td>Nessus Manager</td>
<td>For Nessus Manager and Nessus Professional versions, Tenable provides the PCI Quarterly External Scan (Unofficial) template. This template can be used to simulate an external scan (PCI DSS 11.2.2) to meet PCI DSS quarterly scanning requirements. However, the scan results from the Unofficial template cannot be submitted to Tenable, Inc. for PCI Validation. The PCI Quarterly External Scan (Unofficial) Template performs the identical scanning functions as the Tenable.io version of this template.</td>
</tr>
<tr>
<td>PCI Quarterly External Scan (Unofficial)</td>
<td>Nessus Professional</td>
<td>The Internal PCI Network Scan template can be used to meet PCI DSS Internal scanning requirement (11.2.1).</td>
</tr>
</tbody>
</table>
SCAP and OVAL

The National Institute of Standards and Technology (NIST) Security Content Automation Protocol (SCAP) is a set of policies for managing vulnerabilities and policy compliance in government agencies. It relies on multiple open standards and policies, including OVAL, CVE, CVSS, CPE, and FDCC policies.

- SCAP compliance auditing requires sending an executable to the remote host.
- Systems running security software (e.g., McAfee Host Intrusion Prevention), may block or quarantine the executable required for auditing. For those systems, an exception must be made for the either the host or the executable sent.
- When using the SCAP and OVAL Auditing template, you can perform Linux and Windows SCAP CHECKS to test compliance standards as specified in NIST’s Special Publication 800-126.
Settings

You can configure Nessus Agents settings in Nessus and Tenable.io.

See the following topics:

- **Modify Agent Settings** - Configure settings for Nessus Agents linked to Nessus Manager or Tenable.io.
- **Blackout Windows** - Create, modify, and delete blackout windows for Nessus Agents in Nessus Manager and Tenable.io.
- **Advanced Settings** - Configure advanced settings for Nessus Agents linked to Nessus Manager.
Proxy Settings

Configure Proxy Settings

You can configure an agent to connect to its manager (i.e., Nessus Manager or Tenable.io) 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 the agent is already installed and linked.
  After initial linking, you can configure a proxy or change existing proxy settings via the command line. For more information, see 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 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 this setting is enabled, 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.
Agent Settings

To change agent settings, use the nessuscli utility on the command line interface to set available settings. For more information, see [Nessus CLI Agent Commands](#).

Agent Manager Settings

On your agent's manager, you can configure system-wide agent settings to specify agent and black-out window settings for all your linked agents. For more information on creating, modifying, and deleting blackout windows, see [Blackout Windows](#).

In Nessus Manager:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage Agents</td>
<td></td>
</tr>
<tr>
<td>Track unlinked agents</td>
<td>When this setting is enabled, agents that are unlinked are preserved in the manager along with the corresponding agent data. This option can also be set using the nessuscli utility.</td>
</tr>
<tr>
<td>Unlink inactive agents after X days</td>
<td>Specifies the number of days an agent can be inactive before the manager unlinks the agent. Inactive agents that were automatically unlinked by Nessus Manager automatically relink if they come back online. Requires that Track unlinked agents is enabled.</td>
</tr>
<tr>
<td>Remove agents that have been inactive for X days</td>
<td>Specifies the number of days an agent can be inactive before the manager removes the agent.</td>
</tr>
<tr>
<td>Override Blackout Windows (Nessus Manager 8.3 and earlier)</td>
<td></td>
</tr>
<tr>
<td>Exclude all agents from software updates</td>
<td>When enabled, Tenable.io or Nessus Manager prevents agents from updating software. This setting does not</td>
</tr>
</tbody>
</table>
Option | Description
--- | ---
 | affect the agent update plan. Agents continue to receive plugin updates and perform scans.

Blackout Windows (Nessus Manager 8.4 - 8.11)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enforce a permanent blackout window schedule</td>
<td>When enabled, Nessus Manager prevents agents from updating software. The following blackout window settings also apply during this window. <strong>Tip:</strong> This setting acts as a permanent blackout window while it is enabled.</td>
</tr>
<tr>
<td>Prevent software updates</td>
<td>When enabled, agents do not receive software updates during scheduled blackout windows.</td>
</tr>
<tr>
<td>Prevent plugin updates</td>
<td>When enabled, agents do not receive plugin updates during scheduled blackout windows.</td>
</tr>
<tr>
<td>Prevent agent scans</td>
<td>When enabled, the system does not run agent scans during scheduled blackout windows.</td>
</tr>
</tbody>
</table>

Configure global blackout/freeze window settings as described in [Modify Blackout Window Settings](Nessus Manager 8.12 and 8.13) or [Modify Freeze Window Settings](Nessus Manager 8.14 and later).

In Tenable.io:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inactive Agents</td>
<td>Specifies the number of days an agent can be inactive before the manager unlinks the agent. After the specified number of days, the agent is unlinked, but the corresponding agent data is not removed from the manager. Tenable.io automatically tracks unlinked agents and related data for the num-</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>ber of days specified in this option. You cannot turn off this tracking.</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Note:</strong> Inactive agents that were automatically unlinked by Tenable.io do not automatically relink if they come back online.</td>
<td></td>
</tr>
<tr>
<td><strong>Override Blackout Windows</strong></td>
<td></td>
</tr>
<tr>
<td>Exclude all agents from software updates</td>
<td>When enabled, this option overrides scheduled blackout windows. It prevents agents from receiving software updates at any time. Agents continue to receive plugin updates and perform scheduled scans.</td>
</tr>
</tbody>
</table>
Modify Agent Settings

Use this procedure to modify agent settings in Nessus Manager or Tenable.io.

To modify agent settings in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   The **My Scans** page appears.
2. In the left navigation bar, click **Agents**.
   The **Agents** page appears.
3. Click the **Agent Settings** tab.
4. Modify the **settings** as necessary.
5. Click **Save**.
   The manager saves your changes.

To modify agent settings in Nessus Manager 8.12 and later:

1. In the top navigation bar, click **Sensors**.
   The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.
2. Do any of the following:
   - To modify system-wide agent settings:
     a. Click the **Settings** tab.
     b. Modify the settings as described in *System-Wide Agent Settings*.
     c. Click **Save**.
   - To modify agent settings remotely, see
   - To modify agent blackout window settings, see *Modify Blackout Window Settings*.
3. Modify the **settings** as necessary.
4. Click **Save**.

   Nessus Manager saves your changes.
Advanced Settings

You can manually configure agents by setting advanced settings from the agent command line interface. Some system-wide agent settings can be modified from Nessus Manager Advanced Settings. Nessus Agent validates your input values to ensure only valid configurations are allowed.

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 a large number of 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.

<table>
<thead>
<tr>
<th>Name</th>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Valid Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Update Plan</td>
<td>agent_update_channel</td>
<td>(Tenable.io-linked agents only)</td>
<td>ga</td>
<td><code>ga</code>: Automatically updates to the latest Agent version when it is made generally available (GA). Note: For Tenable.io-linked agents, this date is usually one week after the version is made generally available. For versions that address critical security issues, Tenable may make the ver-</td>
</tr>
</tbody>
</table>


version available immediately.

**ea:** Automatically updates to the latest Agent version as soon as it is released for Early Access (EA), typically a few weeks before general availability.

**stable:** Does not automatically update to the latest Nessus Agent version. Remains on an earlier version of Nessus Agent set by Tenable, usually one release older than the current generally available version, but no earlier than 7.7.0. When Nessus Agent releases a new version, your agent updates software versions, but stays on a version prior to the latest release.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Always Validate SSL Server Certificates</td>
<td>strict_certificate_validation</td>
<td>When enabled, always validate SSL server certificates, even during initial remote link (requires manager to use a trusted root CA).</td>
<td>no</td>
<td>yes or no</td>
</tr>
<tr>
<td>Automatic Hostname Update</td>
<td>update_hostname</td>
<td>When enabled, when the hostname on the endpoint is modified the new hostname will be updated in the agent's manager. This feature is disabled by default to prevent custom agent names from being overridden.</td>
<td>no</td>
<td>yes or no</td>
</tr>
<tr>
<td>Detect Duplicate Agents</td>
<td>detect_duplicates</td>
<td>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. Addi-</td>
<td>no</td>
<td>yes or no</td>
</tr>
</tbody>
</table>
tionally, for agents linked to Tenable.io or Nessus Manager 8.11.1 and later, the manager performs the same check to identify duplicate agents.

When disabled, the agent automatically logs duplicates in `backend.log`, but no action is taken.

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 `backend.log`. You must manually relink the agent.
<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Allowed Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable Core Updates</td>
<td>When set to yes, the agent does not request automatic core updates. You can still upgrade software versions manually. The agent can still receive plugin updates.</td>
<td>no</td>
<td>yes or no</td>
</tr>
<tr>
<td>Nessus Dump File Max Files</td>
<td>Sets the maximum number of the nessusd.dump files kept on disk. If the number exceeds the specified value, the oldest dump file is deleted.</td>
<td>100</td>
<td>Integers 1-1000.</td>
</tr>
<tr>
<td>Nessus Dump File Max Size</td>
<td>Sets the maximum size of the nessusd.dump files in megabytes. If file size exceeds the maximum size, a new dump file is created.</td>
<td>512</td>
<td>Integers 1-2048</td>
</tr>
<tr>
<td>Plugin Compilation Performance</td>
<td>Sets plugin compilation performance, which affects</td>
<td>high</td>
<td>low, medium, or high</td>
</tr>
<tr>
<td>Scan Performance</td>
<td>scan_performance_mode</td>
<td>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. For more information, see <a href="#">Agent CPU Resource Control</a>.</td>
<td>high</td>
</tr>
</tbody>
</table>
quickly, but the agent consumes more CPU. For more information, see [Agent CPU Resource Control](#).

<table>
<thead>
<tr>
<th>SSL Cipher List</th>
<th>ssl_cipher_list</th>
<th>Sets the cipher list to use for Agent outbound connections.</th>
<th>compatible</th>
</tr>
</thead>
<tbody>
<tr>
<td>legacy – A list of ciphers that can integrate with older APIs.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>compatible – A list of secure ciphers. May not include all the latest ciphers.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>modern – A list of the latest and most secure ciphers.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>custom – A custom OpenSSL cipher list. For more information on valid cipher list formats, see the OpenSSL</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SSL Mode | ssl_mode | Minimum supported version of TLS. | tls_1_2 |
|----------|---------|---------------------------------|--------|
|          |         |                                 | compat – TLS v1.0+.
|          |         |                                 | ssl_3_0 – SSL v3+
|          |         |                                 | tls_1_1 – TLS v1.1+
|          |         |                                 | tls_1_2 – TLS v1.2+

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.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Valid Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ignore_proxy</td>
<td>If enabled, the agent attempts a direct connection to the manager instead of using the set proxy, until it fails 10 times. If disabled, the agent attempts to connect using the set proxy, until it fails three times. This setting changes automatically, as described in Proxy Connection Fallback. You can also set this setting manually, however, if at any point the agent meets one of the conditions described in Proxy Connection Fallback, the agent auto-</td>
<td>yes or no</td>
</tr>
</tbody>
</table>
### matically changes the setting.

<table>
<thead>
<tr>
<th>Name</th>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Valid Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ms_proxy</td>
<td></td>
<td>When enabled, the agent uses a proxy to connect to its manager.</td>
<td>true or false</td>
<td></td>
</tr>
<tr>
<td>proxy</td>
<td></td>
<td>The hostname or IP address of your proxy server.</td>
<td>String</td>
<td></td>
</tr>
<tr>
<td>proxy_port</td>
<td></td>
<td>The port number of the proxy server.</td>
<td>String</td>
<td></td>
</tr>
<tr>
<td>proxy_auth</td>
<td></td>
<td>(Optional) If you want to use authentication to connect to the proxy, specify the authentication scheme.</td>
<td>basic, digest, ntlm, or auto</td>
<td></td>
</tr>
<tr>
<td>proxy_username</td>
<td></td>
<td>If using authentication to connect to the proxy, the name of a user account that has permissions to access and use the proxy server.</td>
<td>String. If there are spaces, use quotes (&quot;).</td>
<td></td>
</tr>
<tr>
<td>proxy_password</td>
<td></td>
<td>If authenticating with the proxy, password associated with the username.</td>
<td>String</td>
<td></td>
</tr>
</tbody>
</table>

### Nessus Manager Advanced Settings

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

<table>
<thead>
<tr>
<th>Name</th>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Valid Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agents Progress</td>
<td>agents_progress_viewable</td>
<td>When a scan gathers information from agents, Nessus Manager does not show detailed agents</td>
<td>100</td>
<td>Integers. If set to 0, this defaults to 100.</td>
</tr>
<tr>
<td>Name</td>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Valid Values</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>Automatically Download Agent Updates</td>
<td>agent_updates_from_feed</td>
<td>When enabled, new Nessus Agent software updates are automatically downloaded.</td>
<td>yes</td>
<td>yes or no</td>
</tr>
<tr>
<td>Concurrent Agent Software Updates</td>
<td>cloud.manage.download_max</td>
<td>The maximum concurrent agent update downloads.</td>
<td>10</td>
<td>Integers</td>
</tr>
<tr>
<td>Include Audit Trail Data</td>
<td>agent_merge_audit_trail</td>
<td>Controls whether or not agent scan result audit trail data is included in the main agent database. Excluding</td>
<td>false</td>
<td>true or false</td>
</tr>
<tr>
<td>Name</td>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Valid Values</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
<td>--------------</td>
</tr>
<tr>
<td>Include KB Data</td>
<td>agent_merge_kb</td>
<td>Includes the agent scan result KB data in the main agent database. Excluding KB data can significantly improve agent result processing performance. If this setting is false, the Audit Trail Verbosity setting in an individual scan or policy defaults to No audit trail. Available in Nessus 8.3 and later.</td>
<td>false</td>
<td>true or false</td>
</tr>
<tr>
<td>Name</td>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Valid Values</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>--------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>set to false, the <strong>Include the KB</strong> setting in an individual scan or policy defaults to <strong>Exclude KB</strong>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available in Nessus 8.3 and later.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Result Processing</td>
<td>agent_merge_journal_</td>
<td>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 sqlite3 documentation.</td>
<td>DELETE</td>
<td>MEMORY, TRUNCATE, DELETE</td>
</tr>
<tr>
<td>Journal Mode</td>
<td>mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Available in Nessus 8.3 and later.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Valid Values</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>----------------</td>
</tr>
<tr>
<td>Result Processing Sync Mode</td>
<td>agent_merge_synchronous_setting</td>
<td>Sets the filesystem sync mode to use when processing agent results. Turning this off will significantly 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.</td>
<td>FULL</td>
<td>OFF, NORMAL, FULL</td>
</tr>
<tr>
<td>Track Unique Agents</td>
<td>track_unique_agents</td>
<td>When enabled, Nessus Manager checks if MAC addresses of agents trying</td>
<td>no</td>
<td>yes or no</td>
</tr>
<tr>
<td>Name</td>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Valid Values</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>-------------</td>
<td>---------</td>
<td>--------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to link match MAC addresses of currently linked agents with the same hostname, platform, and distro. Nessus Manager deletes duplicates that it finds.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Blackout Windows

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

In Tenable.io:

Blackout windows allow you to schedule times where certain agent activities are suspended for all linked agents. This activity includes:

- Receiving and applying software updates

Blackout 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 where certain agent activities are suspended for all linked agents. This activity includes:

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

You can configure a permanent freeze window and global settings for how freeze windows work for linked agents. To configure global blackout freeze window settings, see Agent Settings.

To manage blackout freeze windows, use the following procedures:

- Create a Blackout Window
- Modify a Blackout Window
- Delete a Blackout Window
Create a Blackout Window

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

**In Tenable.io:**

Blackout windows allow you to schedule times where certain agent activities are suspended for all linked agents. This activity includes:

- Receiving and applying software updates

Blackout 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 where certain agent activities are suspended for all linked agents. This activity includes:

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

To create a blackout window for linked agents in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   
   The **My Scans** page appears.

2. In the left navigation bar, click **Agents**.
   
   The **Agents** page appears.

3. Click the **Blackout Windows** tab.

4. In the upper-right corner, click the **New Window** button.

   The **New Blackout Window** page appears.
5. Configure the options as necessary.

6. Click **Save**.

   The blackout window goes into effect and appears on the **Blackout Windows** tab.

To create a freeze window for linked agents in Nessus Manager 8.12 and later:

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

   The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.

2. In the left navigation bar, click **Freeze Windows**.

   The **Freeze Windows** page appears.

3. In the upper-right corner, click the **New Window** button.

   The **New Freeze Window** page appears.

4. Configure the options as necessary.

5. Click **Save**.

   The freeze window goes into effect and appears on the **Freeze Windows** tab.
Modify a Blackout Window

Use this procedure to modify a blackout window in Nessus Manager or Tenable.io.

To configure global blackout window settings, see Agent Settings.

To modify a blackout window in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click Scans.
   The My Scans page appears.
2. In the left navigation bar, click Agents.
   The Agents page appears.
3. Click the Blackout Windows tab.
4. In the blackout windows table, click the blackout window you want to modify.
   The blackout window details page appears.
5. Modify the options as necessary.
6. Click Save to save your changes.

To modify a freeze window in Nessus Manager 8.12 and later:

1. In the top navigation bar, click Sensors.
   The Linked Agents page appears. By default, Linked Agents is selected in the left navigation menu and the Linked Agents tab is active.
2. In the left navigation bar, click Freeze Windows.
   The Freeze Windows page appears.
3. In the freeze windows table, click the freeze window you want to modify.
   The freeze window details page appears.
4. Modify the options as necessary.
5. Click Save to save your changes.
   Nessus Manager updates your changes.
Delete a Blackout Window

Use this procedure to delete a blackout or freeze window in Tenable.io or Nessus Manager.

To delete a blackout window for linked agents in Tenable.io or Nessus Manager 8.11 and earlier:

1. In the top navigation bar, click **Scans**.
   - The **My Scans** page appears.
2. In the left navigation bar, click **Agents**.
   - The **Agents** page appears.
3. Click the **Blackout Windows** tab.
4. In the blackout window table, in the row for the blackout window that you want to delete, click the **X** button.
   - A dialog box appears, confirming your selection to delete the blackout window.
5. Click **Delete** to confirm the deletion.
   - The manager deletes the blackout window.

To delete a freeze window for linked agents in Nessus Manager 8.12 and later:

1. In the top navigation bar, click **Sensors**.
   - The **Linked Agents** page appears. By default, **Linked Agents** is selected in the left navigation menu and the **Linked Agents** tab is active.
2. In the left navigation bar, click **Freeze Windows**.
   - The **Freeze Windows** page appears.
3. In the freeze window table, in the row for the freeze window that you want to delete, click the **X** button.
   - A dialog box appears, confirming your selection to delete the freeze window.
4. Click **Delete** to confirm the deletion.

   Nessus Manager deletes the freeze window.
Additional Resources

This section contains the following resources:

- [Agent Settings](#)
- [Agent Status](#)
- [Nessus CLI Agent Commands](#)
- [Agent Scan and Policy Template](#)
  - [Compliance](#)
  - [Plugins](#)
  - [Special Use Templates](#)
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 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 Nessus Manager or Tenable.io. 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 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.

Location of config.json file

Place the `config.json` file in the following location:

- **Linux**: `/opt/nessus_agent/var/nessus/config.json`
- **Windows**: `C:\ProgramData\Tenable\Nessus Agent\nessus\config.json`
- **Mac OS X**: `/Library/NessusAgent/run/var/nessus/config.json`

Example Agent config.json file format:

```json
{
    "link": {
        "name": "sensor name",
        "host": "hostname or IP address",
        "port": 443,
        "key": "abcdefghijklmnopqrstuvwxyz",
        "ms_cert": "CA certificate for linking",
        "retry": 1,
        "groups": ["group 1", "group 2"],
        "proxy": {
            "proxy": "proxyhostname",
            "proxy_port": 443,
            "proxy_username": "proxyusername",
            "proxy_password": "proxypassword",
            "user_agent": "proxyagent",
            "proxy_auth": "NONE"
        }
    },
    "preferences": {
        "update_hostname": "yes"
    }
}
```
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.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>(Optional) 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.</td>
</tr>
<tr>
<td>host</td>
<td>The hostname or IP address of the manager you want to link to.</td>
</tr>
<tr>
<td></td>
<td>To link to Tenable.io, use cloud.tenable.com.</td>
</tr>
<tr>
<td>port</td>
<td>The port for the manager you want to link to.</td>
</tr>
<tr>
<td></td>
<td>For Nessus Manager: 8834 or your custom port.</td>
</tr>
<tr>
<td></td>
<td>For Tenable.io: 443</td>
</tr>
<tr>
<td>key</td>
<td>The linking key that you retrieved from the manager.</td>
</tr>
<tr>
<td>ms_cert</td>
<td>(Optional) A custom CA certificate to use to validate the manager's server certificate.</td>
</tr>
<tr>
<td>groups</td>
<td>(Optional) One or more existing agent groups where you want to add the</td>
</tr>
</tbody>
</table>
If you do not specify an agent group during the install process, you can add your linked agent to an agent group later in Nessus Manager or Tenable.io.

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"

**Note:** The agent group name is case-sensitive and must match exactly.

### proxy

((Optional)

If you are using a proxy server, include the following:

- **proxy:** The hostname or IP address of your proxy server.
- **proxy_port:** The port number of the proxy server.
- **proxy_username:** The name of a user account that has permissions to access and use the proxy server.
- **proxy_password:** The password of the user account that you specified as the username.
- **user_agent:** The user agent name, if your proxy requires a preset user agent.
- **proxy_auth:** The authentication method to use for the proxy.

---

**Preferences**

The preferences section configures any advanced settings. For more information, see [Advanced 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

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux</td>
<td># /opt/nessus/sbin/nessus-service [-vhD] [-c &lt;config-file&gt;] [-p &lt;port-number&gt;] [-a &lt;address&gt;] [-S &lt;ip,[ip,..]&gt;]</td>
</tr>
<tr>
<td>FreeBSD</td>
<td># /usr/local/nessus/sbin/nessus-service [-vhD] [-c &lt;config-file&gt;] [-p &lt;port-number&gt;] [-a &lt;address&gt;] [-S &lt;ip,[ip,..]&gt;]</td>
</tr>
<tr>
<td>Mac OS X</td>
<td># /Library/Nessus/run/sbin/nessus-service [-vhD] [-c &lt;config-file&gt;] [-p &lt;port-number&gt;] [-a &lt;address&gt;] [-S &lt;ip,[ip,..]&gt;]</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-c &lt;config-file&gt;</td>
<td>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.</td>
</tr>
<tr>
<td>-S &lt;ip [,ip2,...]&gt;</td>
<td>When starting the nessusd server, force the source IP of the connections established by Nessus during scanning to &lt;ip&gt;. 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.</td>
</tr>
<tr>
<td>-D</td>
<td>When starting the nessusd server, this option forces the server to run in the background (daemon mode).</td>
</tr>
<tr>
<td>-v</td>
<td>Display the version number and exit.</td>
</tr>
<tr>
<td>-l</td>
<td>Display a list of those third-party software licenses.</td>
</tr>
<tr>
<td>-h</td>
<td>Show a summary of the commands and exit.</td>
</tr>
<tr>
<td>--ipv4-only</td>
<td>Only listen on IPv4 socket.</td>
</tr>
<tr>
<td>--ipv6-only</td>
<td>Only listen on IPv6 socket.</td>
</tr>
<tr>
<td>-q</td>
<td>Operate in &quot;quiet&quot; mode, suppressing all messages to stdout.</td>
</tr>
<tr>
<td>-R</td>
<td>Force a re-processing of the plugins.</td>
</tr>
<tr>
<td>-t</td>
<td>Check the time stamp of each plugin when starting up to only compile newly updated plugins.</td>
</tr>
<tr>
<td>-K</td>
<td>Set a master password for the scanner.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>If your master password is set and then lost, it cannot be recovered by your administrator nor Tenable Support.</td>
</tr>
</tbody>
</table>
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 Nessus Agent functions through a command line interface.

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

## Nessuscli Syntax

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux</td>
<td># /opt/nessus_agent/sbin/nessuscli &lt;cmd&gt; &lt;arg1&gt; &lt;arg2&gt;</td>
</tr>
<tr>
<td>Mac OS X</td>
<td># /Library/NessusAgent/run/sbin/nessuscli &lt;cmd&gt; &lt;arg1&gt; &lt;arg2&gt;</td>
</tr>
<tr>
<td>Windows</td>
<td>C:\Program Files\Tenable\Nessus Agent\nessuscli.exe &lt;cmd&gt; &lt;arg1&gt; &lt;arg2&gt;</td>
</tr>
</tbody>
</table>

## Nessuscli Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Informational Commands</strong></td>
<td></td>
</tr>
<tr>
<td># nessuscli help</td>
<td>Displays a list of <code>nessuscli</code> commands.</td>
</tr>
<tr>
<td># nessuscli -v</td>
<td>Displays your current version of Nessus Agent.</td>
</tr>
<tr>
<td><strong>Bug Reporting Commands</strong></td>
<td>Generates an archive of system diagnostics.</td>
</tr>
<tr>
<td># nessuscli bug-report-generator</td>
<td>If you run this command without arguments, the utility prompts you for values.</td>
</tr>
<tr>
<td></td>
<td><strong>Optional arguments:</strong></td>
</tr>
<tr>
<td></td>
<td>--quiet: Run the bug report generator without prompting user for feed-</td>
</tr>
</tbody>
</table>
### Command Table

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>back.</td>
<td></td>
</tr>
<tr>
<td>--scrub:</td>
<td>The bug report generator sanitizes the last two octets of the IPv4 address.</td>
</tr>
<tr>
<td>--full:</td>
<td>The bug report generator collects extra data.</td>
</tr>
</tbody>
</table>

### Local Agent Commands

**Used to link, unlink, and display agent status**

```bash
# nessuscli agent
link --key=<key> --host=<host> --port=<port>
```

Using the [Nessus Agent Linking Key](https://tenable.com), this command links the agent to the Nessus Manager or Tenable.io.

**Required arguments:**

- `--key`: The linking key that you retrieved from the manager.
- `--host`: To link to Nessus Manager: The static IP address or hostname you set during the Nessus Manager installation. To link to Tenable.io: `cloud.tenable.com`
- `--port`: To link to Nessus Manager: 8834 or your custom port. To link to Tenable.io: 443

**Tenable.io arguments:**

- `--cloud`: To link to Tenable.io, set the argument `--cloud="yes"`.

  The `--cloud` argument is a shortcut to specifying `--host=t=cloud.tenable.com --port=443`. If you use `--cloud`, you do not need to set `--host` and `--port`.

**Optional arguments:**

- `--name`: 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.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>- --groups:</td>
<td>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 Nessus Manager or Tenable.io. List multiple groups in a comma-separated list. If any group names have spaces, use quotes around the whole list. For example: &quot;Atlanta, Global Headquarters&quot;</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The agent group name is case-sensitive and must match exactly.</td>
</tr>
<tr>
<td></td>
<td>- --ca-path: A custom CA certificate to use to validate the manager's server certificate.</td>
</tr>
<tr>
<td></td>
<td>- --offline-install: When enabled (set to &quot;yes&quot;), installs Nessus Agent on the system, even if it is offline. Nessus Agent periodically attempts to link itself to its manager.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>- --proxy-host: The hostname or IP address of your proxy server.</td>
</tr>
<tr>
<td></td>
<td>- --proxy-port: The port number of the proxy server.</td>
</tr>
<tr>
<td></td>
<td>- --proxy-password: The password of the user account that you specified as the username.</td>
</tr>
<tr>
<td></td>
<td>- --proxy-username: The name of a user account that has permissions to access and use the proxy server.</td>
</tr>
<tr>
<td></td>
<td>- --proxy-agent: The user agent name, if your proxy requires a preset user agent.</td>
</tr>
<tr>
<td># nessuscli agent</td>
<td>Unlinks agent from the Nessus Manager or Tenable.io.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>unlink</td>
<td>Displays the status of the agent, jobs pending, and if the agent is linked or not linked to server.</td>
</tr>
</tbody>
</table>

**Optional arguments:**

- `--local`: (Default behavior) Provides the status, current jobs count, and jobs pending. This option prevents the agent from contacting the management software that it is linked with to fetch the status. Instead, it displays the last known information from its most recent sync.

- `--remote`: Fetches the job count from the manager and displays the status.

**Note:** Tenable does not recommend running frequent status checks with the `--remote` option (for example, when using automation).

- `--offline`: Provides the most recently cached agent status when it cannot connect to Nessus Manager or Tenable.io.

- `--show-token`: Displays the agent's token that is used to identify and authenticate with its manager.

- `--show-uuid`: Displays the agent's Tenable UUID.

### Update Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code># nessuscli agent update --file=&lt;plugins_set.tgz&gt;</code></td>
<td>Manually installs a plugins set.</td>
</tr>
</tbody>
</table>

`nessuscli fix --set agent_update_channel=<value>` *(Tenable.io-linked agents only)*

Sets the agent update plan to determine what version the agent automatically updates to.

**Values:**
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>• <strong>ga</strong>: Automatically updates to the latest Agent version when it is made generally available (GA). <strong>Note:</strong> For Tenable.io-linked agents, this date is usually <em>one week after</em> the version is made generally available. For versions that address critical security issues, Tenable may make the version available immediately.</td>
<td></td>
</tr>
<tr>
<td>• <strong>ea</strong>: Automatically updates to the latest Agent version as soon as it is released for Early Access (EA), typically a few weeks before general availability.</td>
<td></td>
</tr>
<tr>
<td>• <strong>stable</strong>: Does not automatically update to the latest Nessus Agent version. Remains on an earlier version of Nessus Agent set by Tenable, usually one release older than the current generally available version, but no earlier than 7.7.0. When Nessus Agent releases a new version, your agent updates software versions, but stays on a version prior to the latest release.</td>
<td></td>
</tr>
</tbody>
</table>

### Fix Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>nessuscli fix</strong> --list</td>
<td>Displays a list of agent settings and their values.</td>
</tr>
<tr>
<td><strong>nessuscli fix</strong> --set <code>&lt;setting&gt;=&lt;value&gt;</code></td>
<td>Set an agent setting to the specified value. For a list of agent settings, see <strong>Advanced Settings</strong>.</td>
</tr>
<tr>
<td># <strong>nessuscli fix</strong> --set update_hostname=&quot;&lt;value&gt;&quot;</td>
<td>Updates agent hostnames automatically in Tenable.io or Nessus Manager 7.1.1 or later. The <strong>update_hostname</strong> parameter can be set to yes or no. By default, this preference is disabled. <strong>Note:</strong> Restart the agent service for the change to take effect in Nessus Manager.</td>
</tr>
<tr>
<td># <strong>nessuscli fix</strong> --set track_</td>
<td>Tracks unique agent assets by MAC address to prevent duplicates and outdated agents from appearing in Nessus Manager if a system is rein-</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>unique_agents=&quot;&lt;value&gt;&quot;</td>
<td>stalled. The track_unique_agent parameter is available in Nessus 7.1.1 and can be set to yes or no. By default, this preference is enabled.</td>
</tr>
<tr>
<td># nessuscli fix --set max_retries=&quot;&lt;value&gt;&quot;</td>
<td>Sets the maximum number of times an agent should retry in the event of a failure when executing the agent link, agent status, or agent unlink commands. The commands retry, the specified number of times, consecutively, sleeping increasing increments of time set by retry_sleep_milliseconds between attempts. The default value for max_retries is 0. For example, if max_retries is set to 4, and retry_sleep_milliseconds is set 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. <strong>Note:</strong> This setting does not affect offline updates or the agent’s normal 24 hour check-in after it is linked.</td>
</tr>
<tr>
<td># nessuscli fix --set retry_sleep_milliseconds=&quot;&lt;value&gt;&quot;</td>
<td>Sets the number of milliseconds that an agent sleeps for between retries in event of a failure when executing the agent link, agent status, or agent unlink commands. The default is 1500 milliseconds (1.5 seconds).</td>
</tr>
</tbody>
</table>

**Fix Secure Settings**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td># nessuscli fix --secure --set &lt;setting&gt;=&lt;value&gt;</td>
<td>Set secure settings on the agent. <strong>Caution:</strong> Tenable does not recommend changing undocumented --secure settings as it may result in an unsupported configuration. For a list of supported secure settings, see Advanced Settings.</td>
</tr>
</tbody>
</table>

**Resource Control Commands**
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td># nessuscli fix -set process_priority=&quot;&lt;value&gt;&quot;</td>
<td>Commands</td>
</tr>
<tr>
<td># nessuscli fix -get process_priority</td>
<td>Set, get, or delete the process_priority setting.</td>
</tr>
<tr>
<td># nessuscli fix -delete process_priority</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>For valid values and more information on how the setting works, see See <a href="#">Agent CPU Resource Control</a>.</td>
</tr>
</tbody>
</table>
Plugin Updates

The table below describes the behavior of differential plugin updates for agents linked to either Tenable.io or Nessus Manager.

<table>
<thead>
<tr>
<th>Linked</th>
<th>Differential Update</th>
<th>Full Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tenable.io</td>
<td>When the agent plugin set is 15 days or less behind the Tenable.io plugin set.</td>
<td>When the agent plugin set is more than 15 days behind the Tenable.io plugin set.</td>
</tr>
<tr>
<td>Nessus Manager</td>
<td>When the agent plugin set is 5 days or less behind the Nessus Manager plugin set.</td>
<td>When the agent plugin set is more than 5 days behind the Nessus Manager plugin set.</td>
</tr>
</tbody>
</table>
Logging

Logs for the Nessus Agent are located in the following locations.

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Log Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>C:\ProgramData\Tenable\Nessus Agent\nessus\logs</td>
</tr>
<tr>
<td>Linux</td>
<td>/opt/nessus_agent/var/nessus/logs</td>
</tr>
<tr>
<td>macOS</td>
<td>/Library/NessusAgent/run/var/nessus/logs</td>
</tr>
</tbody>
</table>
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 Nessus Network Monitor in every network segment?
- Would it be easier to install a small number of Nessus Managers (for example, 1 or 3) and just 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 a large number of systems?
- Does your network include a number of laptops that predominantly work remotely and which cannot be credential-scanned 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, please visit: [https://www.tenable.com/plugins](https://www.tenable.com/plugins).

The majority of plugins work with 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.
Appendix

- File and Process Whitelist
- Nessus Agent Cheatsheet
- Customer Case Studies
- Expanded Agent Large Scale Deployment Guide (PDF)
Configure Nessus Agent for NIAP Compliance

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

Before you begin:

- Ensure you are running Nessus Agent version 8.0.0.
- If Nessus Agent is linked to Nessus Manager, verify that the CA certificate of Nessus Manager is found 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 Nessus Agent is installed.

To configure 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/sbin/nessuscli fix --set niap_mode=enforcing
     ```

Nessus Agent does the following:

**Note:** When Nessus Agent is in NIAP mode, Nessus Agent overrides the following settings as long as Nessus Agent remains in NIAP mode. If you disable NIAP mode, 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 response is that the certificate is revoked, then the certificate will be marked as invalid. If there is no response, then the certificate will not be marked as invalid, and its use will be 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.io and plugins.nessus.org are already in known_CA.inc in the plugins directory.

• If linked to Nessus Manager, verifies that the CA certificate of Nessus Manager is found in custom_CA.inc or known_CA.inc.
You should allow certain 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

<table>
<thead>
<tr>
<th>Folder Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Program Files\Tenable\Nessus Agent*</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files (x86)\Tenable\Nessus Agent*</td>
<td></td>
</tr>
<tr>
<td>C:\ProgramData\Tenable\Nessus Agent*</td>
<td></td>
</tr>
</tbody>
</table>

#### Processes

<table>
<thead>
<tr>
<th>Process Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Program Files\Tenable\Nessus Agent\nasl.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\Tenable\Nessus Agent\nessuscli.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\Tenable\Nessus Agent\nessusd.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\Tenable\Nessus Agent\nessus-service.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files\Tenable\Nessus Agent\tenable_ovaldi_3ef350e0435440418f7d33232f74f260.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files (x86)\Tenable\Nessus Agent\nasl.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files (x86)\Tenable\Nessus Agent\nessuscli.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files (x86)\Tenable\Nessus Agent\nessusd.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files (x86)\Tenable\Nessus Agent\nessus-service.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Program Files (x86)\Tenable\Nessus Agent\tenable_ovaldi_3ef350e0435440418f7d33232f74f260.exe</td>
<td></td>
</tr>
</tbody>
</table>
### 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`

### 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/nessusmgt`
Benefits and Limitations of Using 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). 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 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 Nessus Agents on all major operating systems.
  - You can install Nessus Agents anywhere, including transient endpoints like laptops.
You can deploy 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 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 Nessus Agents

For dataflow and licensing requirements, please refer to the System Requirements section.

Hardware

Nessus Agents are designed to be lightweight and to use only minimal system resources. Generally, a Nessus Agent uses 40 MB of RAM (all pageable). A 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 Nessus Agent resource usage, refer to Software Footprint and Host System Utilization.

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

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Minimum Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>1 Dual-core CPU</td>
</tr>
<tr>
<td>Processor Speed</td>
<td>&lt; 1 Ghz</td>
</tr>
<tr>
<td>RAM</td>
<td>&lt; 1 GB</td>
</tr>
<tr>
<td>Disk Space</td>
<td>&lt; 1 GB</td>
</tr>
<tr>
<td>Disk Speed</td>
<td>15-50 IOPS</td>
</tr>
</tbody>
</table>

Software

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Supported Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux</td>
<td>Debian 7, 8, and 9- i386</td>
</tr>
<tr>
<td></td>
<td>Debian 7, 8, and 9 - AMD64</td>
</tr>
<tr>
<td></td>
<td>Red Hat ES 6 / CentOS 6 / Oracle Linux 6 (including Unbreakable Enterprise Kernel) - i386</td>
</tr>
<tr>
<td></td>
<td>Red Hat ES 6 / CentOS 6 / Oracle Linux 6 (including Unbreakable Enterprise Kernel) - x86_64</td>
</tr>
<tr>
<td>Operating System</td>
<td>Supported Versions</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>Red Hat ES 7 / CentOS 7 / Oracle Linux 7 - x86_64</td>
<td>Fedora 24 and 25 - x86_64</td>
</tr>
<tr>
<td></td>
<td>Ubuntu 12.04, 12.10, 13.04, 13.10, 14.04, and 16.04 - i386</td>
</tr>
<tr>
<td></td>
<td>Ubuntu 12.04, 12.10, 13.04, 13.10, 14.04, and 16.04 - AMD64</td>
</tr>
<tr>
<td>Windows</td>
<td>Windows 7, 8, and 10 - i386</td>
</tr>
<tr>
<td>Mac OS X</td>
<td>Mac OS X 10.8 - 10.13</td>
</tr>
</tbody>
</table>
Installing and Linking Nessus Agents

The following installation instructions are for the command line. To install using the user interface, see Install Nessus Agents.

Linux

Install the package:

Red Hat, CentOS, and Oracle Linux

# rpm -ivh NessusAgent-<version number>-es6.i386.rpm
# rpm -ivh NessusAgent-<version number>-es5.x86_64.rpm

Fedora

# rpm -ivh NessusAgent-<version number>-fc20.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 Nessus Manager or Tenable.io:

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

```
/opt/nessus_agent/sbin/nessuscli_agent link
--key=00abcd00000efgh1111i0k222lmopq3333st4455u66v777777w88xy9999zabc00
--name=MyOSXAgent --groups="All" --host=yourcompany.com --port=8834
```

Windows

You can deploy and link 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=00abcd00000efgh1111i0k222lmopq3333st4455u66v777777w88xy9999zabc00 /qn

Mac OS X

**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:

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

**Link Agent to Nessus Manager or Tenable.io:**

1. Open Terminal.

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

   For example:

   ```
   # /Library/NessusAgent/run/sbin/nessuscli agent link
   --key=00abcd00000efgh1111i0k222lmopq3333st4455u66v777777w88xy9999zabc00
   --name=MyOSXAgent --groups=All --host=yourcompany.com --port=8834
   ```
Customer Case Studies

The customer case studies describe 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.io 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 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.io and Nessus Manager solution for managing Nessus Agents. Tenable.io was used for user workstation Nessus Agent scan operations, and Nessus Manager was used for servers and other permanent on-premise infrastructure. Initech then imported all 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.io for Nessus Agent management and local scan and audit information, remote network scan functionality, and integration with their third-party applications via the Tenable.io 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 Nessus Manager and Tenable.io, 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 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.io was required for user workstation Nessus Agent scan operations, and Nessus Manager was used for servers and other permanent on-premise infrastructure.

- Initech used the scaling ability, uptime guarantee, and cloud flexibility of Tenable.io to meet the dynamic requirements of a constantly changing workstation environment.
- Initech used 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 Nessus Manager and Tenable.io 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

- 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).

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 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.io to manage all aspects of their environment. The solution used 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.io also provided the needed API to utilize their third-party and customized applications.

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