Tenable.sc Large Enterprise Deployment Guide

Last Revised: October 27, 2022

Table of Contents

Welcome to the Tenable.sc Large Enterprise Deployment Guide

You may have a number of unique technical and business requirements to consider when planning a large enterprise deployment of Tenable.sc. If your organization scans 100,000 or more IP addresses, consider the information in this guide when planning, executing, and operationalizing your Tenable.sc deployment.

This guide can help you plan your deployment, but it does not cover all deployment scenarios or network architectures. Contact Tenable Support or Tenable Professional Services for more assistance.

> **Tip:** Tenable recommends using this guide as a companion to the Tenable.sc User Guide.

- Plan Your Deployment
- Plan Your Scanning Strategy
- Operationalize Your Established Deployment

Plan Your Deployment

Consider the following when planning your Tenable.sc deployment:

- Air-Gapped Environments
- Tiered Deployments
- Access Control
- Integrations
- API Usage

Air-Gapped Environments

Related Reading: Offline Repositories in the *Tenable.sc User Guide*

Consider the following when deploying Tenable.sc in an air-gapped (offline) environment.

Architecture

You must deploy a Tenable.sc and a set of scanners within each air-gapped network.

If you want to consolidate data from other networks with the data generated in your air-gapped network, you can use offline repositories to export data from your air-gapped Tenable.sc to your other instance of Tenable.sc. This supports both consolidated and federated reporting structures.

Upgrades and Updates

Tenable recommends performing Tenable.sc upgrades at least once a year (quarterly preferred) and plugin/feed updates at least once a month. After you perform a plugin update, run comprehensive scans to take advantage of the new vulnerability data and generate current scan results.

Note: A few plugins require internet access and cannot run in an air-gapped environment. For example, Nessus plugin 52669 checks to see if a host is part of a botnet.

After you perform a plugin update or feed update, verify the files as described in the knowledge base article.

To perform a Tenable.sc upgrade or a plugin/feed update offline:

Tip: You can use the API to automate some Tenable.sc upgrade and plugin update process.

1. Download the files in a browser or via the API.
2. Verify the integrity of the files.
    - Tenable.sc upgrade: Compare the download checksum with the checksum on the Tenable downloads page
    - Plugin/feed update: Download and compare the checksums.

3. Move the files to your Tenable.sc instance.
4. Upload the files to Tenable.sc.
    - Tenable.sc upgrade: via the CLI.
    - Plugin/feed update: in a browser or via the API.

Nessus Agents

If you deployed Nessus Manager to manage Nessus Agents in an air-gapped environment, perform an offline software update (`nessus-agent-updates-X.X.X.tar.gz` on the Tenable Downloads site) on your Nessus Manager. Nessus Manager pushes the update to the managed Nessus Agents.

For more information, see the knowledge base article.

Tiered Deployments

Related Reading: Tiered Remote Repositories in the *Tenable.sc User Guide* and Hardware Requirements in the *General Requirements Guide*

A *tiered remote repository* configuration uses remote repositories to share data between multiple Tenable.sc instances.

- If you plan to support 100,000-249,999 hosts, Tenable recommends a tiered remote repository configuration.
- If you plan to support 250,000 or more hosts, Tenable **requires** a tiered remote repository configuration.

Tiered Tenable.sc instances perform informal roles in your overall Tenable.sc deployment. Tenable recommends at least one designated reporting Tenable.sc and an additional Tenable.sc instance for every 100,000 to 150,000 hosts on your network.

- A *scanning tier* Tenable.sc optimizes scanning by managing scan jobs across your attached scanners. Scanning tier Tenable.sc instances prioritize efficient collection of scan data.
- A *reporting tier* Tenable.sc optimizes dashboards and reporting by centralizing the data collected by scanning tier Tenable.sc instances.

Note: Your scanning tier and reporting tier Tenable.sc instances must be running the same Tenable.sc version.

Without a tiered remote repository configuration, enterprise-scale scanning and analysis may cause performance issues on a single Tenable.sc. Tiered remote repositories optimize your analysis and report generation without negatively impacting scanning performance.

Tip: While you could connect two Tenable.sc instances as offline repositories, offline repositories do not establish a true connection between the instances. All data must be transferred manually between offline repositories.

Connect Tiers Using Repositories

Connect your scanning tiers to your reporting tiers as read-only repositories in your reporting tier Tenable.sc deployments.

To configure a tiered remote repository deployment:

1. On the scanning tier Tenable.sc instance, create one or more repositories for storing scan result data.

   > **Note:** To view trend data for scanning tier Tenable.sc instances on your reporting tier Tenable.sc instance, enable the **Generate Trend Data** option for each repository on your scanning tier Tenable.sc instances. For more information, see Agent Repositories and IPv4/IPv6 Repositories.

2. On the scanning tier Tenable.sc instance, run scans to populate the repositories with data.

3. On the reporting tier Tenable.sc instance, create a remote repository for each repository on your scanning tier Tenable.sc instance.

   The reporting tier Tenable.sc syncs scan result data from the scanning tier Tenable.sc repositories.

By default, remote repositories synchronize daily. You can use the Tenable.sc API to initiate more frequent data refreshes.

Version and Upgrade Considerations

Your scanning tier and reporting tier Tenable.sc instances must be running the same Tenable.sc version. When upgrading to a new version of Tenable.sc, update your reporting tier instance before your scanning tier instances.

Hardware Considerations

For optimal performance, customize the hardware on your scanning tier and reporting tier instances.

Scanning Tier Instance	Reporting Tier Instance
Scanning tier instances benefit from: • High CPU speeds • High disk I/O speeds Consider adding additional CPU and disk	Reporting tier instances benefit from: • High capacity, high-speed RAM • High capacity disk space Consider adding additional RAM and disk space to

Scanning Tier Instance	Reporting Tier Instance
I/O resources to support your active scanning and sensor management.	support your reporting, user management, and data queries. Tenable recommends 128 GB of RAM for every 100,000 active IP addresses (for example, for 150,000 IP addresses, allocate 192 GB of RAM).

For more information, see Performance.

Plan User Access Control

Grant users access to match the purpose of your scanning tier and reporting tier instances.

Scanning Tier Instance	Reporting Tier Instance
Create accounts for: • Technical users who need to configure administrative settings on the instance • Technical users who need to configure and run scans • Technical users who need to generate reports for organization-wide analysis	Create accounts for: • Technical users who need to manage your repositories and tiered configuration. • Business users who need a centralized view of cumulative and trend data for vulnerability analysis.

Access Control

Related Reading: User Access in the *Tenable.sc User Guide*

The Tenable.sc user access model supports role-based access control (RBAC) principles. Each user has a defined *group* membership (for data access) and *role* (for application access) so that users on a team access the same data (by shared group) but with different levels of access (by role) to perform different functions. You configure *organizations* to contain a set of groups and the users within them. Organizations allow for a distinct set of users and groups with unique resources assigned to them. You can use this functionality to mirror your company's organizational structure in Tenable.sc.

For example, you could:

- Grant complete Security Manager access to a Senior Vulnerability Management Engineer
- Grant no access to C-level executives, but instruct Security Managers to export ARCs and share them
- Grant API export access to a Security Engineer
- Grant API integrations access to a Security Engineer

Access Control and the API

Tenable.sc API access is user-based; this allows for both pre-built and custom integrations to utilize the RBAC user model. For more information, see API Usage.

Access Control and Repositories

You configure *repositories* to store scan result data in Tenable.sc. Tenable recommends breaking up large sets of data (tens of thousands of IP addresses) into multiple repositories to:

- Perform faster data import and queries
- Increase control and flexibility of user access
- Increase control and flexibility of reporting
- Manage potential issues related to maximum repository size (32 GB)

Repository Organization

There are many ways to organize your repositories, depending on your needs. For example:

- By division or department in your organization to simplify reporting across an organization's structure
- By logical network definition to accommodate a centralized IT department or specific needs in a non-federated organization

Repository Capacity

A single repository can store 32 GB of data, which is around 30,000 to 100,000 IP addresses depending on your asset types and whether you are running credentialed scans.

When you plan your repository organization, estimate the number of IP addresses that will be stored by each repository. If any of your repository estimates approach the maximum, break the repository into two or more repositories. Tenable recommends sizing your repositories conservatively since you cannot move data to another repository after it has been imported.

Integrations

Tenable.sc supports third-party product integrations of various types to maximize operation inside your organization's network. For information about Tenable-supported integrations, see https://www.tenable.com/partners/technology or the documentation.

Most integrations use the Tenable.sc API to enhance the data within Tenable.sc and to share Tenable.sc data with other platforms used by your organization.

Consider the following best practices when using integrations with large deployments of Tenable.sc:

- Confirm the integration is Tenable-supported. Tenable Support does not provide assistance with Tenable.sc integrations maintained by third-party vendors.
- Confirm that your Tenable.sc meets the environment requirements.
- Confirm that your third-party product configuration can handle the size of your Tenable.sc deployment and your expected data flow.
- Maintain test instances of Tenable.sc and the third-party product to minimize upgrade risk. Test upgrades and configuration changes on your test instance before deploying the changes to your production environment.

For information about custom integrations for in-house platforms or tools, contact Tenable Professional Services.

API Usage

The Tenable.sc API is a RESTful interface to Tenable.sc functions that provides data in JSON format. Developers often use the REST APIs to integrate Tenable.sc with other standalone or web applications. Administrators often use the REST APIs to script interactions with the Tenable.sc server.

For more information, see:

- The Tenable.sc API guide
- The Tenable.sc API best practices guide
- The Python SDK guide for common functions

Consider the following best practices when using the API with large deployments of Tenable.sc:

- From a processing perspective, tasks initiated via the user interface or the API take the same amount of time to complete.
- Tenable.sc uses the same RBAC system for user API access and user interface access.
- Tenable does not recommend multi-threading API calls to speed up access.
- Tenable generally recommends pulling data from the `/analysis` endpoint instead of parsing individual results from the `/scanResult` endpoint.
- Consider the frequency that data is likely to change when setting the frequency for an API call to submit or request data from Tenable.sc. For example, you do not need to pull data every hour if you are only performing weekly scans.

Note: Tenable may not maintain backward compatibility when extending a protocol or implementation. Consequently, some APIs may change in either structure or function. **The API comes with no guarantee of future compatibility.**

Tenable Support does not assist with custom implementations using the API. For assistance with custom designs or implementations, contact Tenable Professional Services.

Plan Your Scanning Strategy

Consider the following when planning your scanning strategy for your Tenable.sc deployment:

- Network Scan Coverage
- Assessment Scanning Methods
- Variables Impacting Scan Time
- Data Flow

Network Scan Coverage

Related Reading: Tenable.sc Hardware Requirements and License Requirements in the *General Requirements Guide*

Most organizations have many types of technology on their network, which can complicate getting a clear picture (and total number) of the assets on your network. Your network may include assets with diverse hardware, operating systems, software, and infrastructure purposes.

Tenable.sc is primarily an IP address-based tool; most Tenable.sc data, scans, queries, and reports are based on asset IP addresses. The IP address count of assets on your network is the primary measure of data when discussing network size and licensing.

If you are new to Tenable.sc, you should consider deploying Tenable.sc to support more assets than you are currently tracking on your network. If you have an asset inventory from a different product, Tenable generally recommends increasing your total by 20-30% to account for previously unseen assets (e.g., unknown systems, untracked systems, and systems with multiple IP addresses in use). The exact increase varies, but 20-30% is a good starting point to estimate your network size.

Tip: You can also run discovery scans (for example, a scan configured with the Host Discovery template or an NNM instance in discovery mode) to get a more accurate estimate of your actual IP address count.

Tenable.sc Instance Configurations

After you estimate your network size, consider that a single instance of Tenable.sc can support 150,000 to 200,000 IP addresses if properly deployed and scaled.

A *tiered remote repository* configuration uses remote repositories to share data between multiple Tenable.sc instances.

- If you plan to support 100,000-249,999 hosts, Tenable recommends a tiered remote repository configuration.
- If you plan to support 250,000 or more hosts, Tenable **requires** a tiered remote repository configuration.

Tiered Tenable.sc instances perform informal roles in your overall Tenable.sc deployment. Tenable recommends at least one designated reporting Tenable.sc and an additional Tenable.sc instance for every 100,000 to 150,000 hosts on your network.

For more information, see Tiered Deployments.

Active Scans

If you intend to perform active scanning, consider that Nessus scanner deployments are designed to be flexible to meet the unique needs of your network architecture. There are many ways to optimize Nessus coverage. For example, you could configure:

- One scanner dedicated for one scan zone that covers a remote, low-bandwidth network area containing 50 IP addresses
- Ten scanners dedicated for many scan zones that cover a flat network area containing 50,000 IP addresses

Tenable recommends customizing your Nessus scanner deployment to meet the unique needs of your network architecture. For more information, see Deployment Considerations in the *Nessus User Guide*.

For information about placing scanners, see Assessment Scanning Methods.

Assessment Scanning Methods

Related Reading: Scanning Overview in the *Tenable.sc User Guide*

There are two primary methods for assessing your assets: *active* network scans and *agent* scans.

- Active — use Nessus or Tenable.io scanners to assess defined networks and targets and send scan data back to Tenable.sc
- Agent — use lightweight agents installed on endpoints to send scan data back to Nessus Manager or Tenable.io

For more information about the benefits and limitations of each type, see Benefits and Limitations in the *Nessus Agent Deployment and User Guide*.

Choose your assessment scanning method based on your targets. You may decide to perform both methods (scanning different target types by different methods) to ensure complete coverage and to properly assess your organizational risk.

Examples

Agent scans are a good choice for a system that is only occasionally on the network (or one that hops between multiple networks). Nessus Agents can report in from anywhere and do not need to stay within expected networks.

Active network scans are a good choice in most environments to assess systems connected in a data center. These systems usually have numerous listening network services and are always running. Network-based assessment scans assess each service individually and can be scheduled for specific times when the systems are not being heavily utilized.

Tip: For other needs, Tenable.sc Continuous View also supports passive scanning via NNM and event logging with LCE.

Scan Zones in Active Scanning

Related Reading: Scan Zones in the *Tenable.sc User Guide*

A complete active scan configuration includes a *scan zone*, which associates one or more scanners with a specific area of your network. Scans of IP addresses within a zone are load balanced between the scanners assigned to that zone. You can customize this to support your unique network topology. For example, you could:

- Create one zone per business unit and add one scanner to each zone.
- Create one large zone and add multiple scanners to the zone.
- Create a zone for an isolated network (a network isolated by a low bandwidth or high latency connection), add one scanner to the zone, and deploy the scanner inside the isolated network.

Scan zones are crucial to the success of an enterprise Tenable.sc deployment. Assigning scanners to scan zones restricts the scanners to scanning their own limited portion of the network, avoiding issues created by scanning through firewalls or across WAN links.

Deployment Examples

You can specify scan zone IP addresses as a single IP address, a range of IP addresses, or subnets in CIDR notation so that you can segment scanning on your network by logical group, physical location, or IP address range.

In general, multiple scanners are most efficient in large, flat networks where Tenable.sc can automatically distribute the scan load across your scanners. Large organizations commonly deploy several scanners in their core network and additional scanners in more segregated or remote networks. You can also design a mixed architecture to suit your unique network infrastructure.

Optimal deployments vary depending on your network and the needs of your organization; there is no one-size-fits-all deployment methodology.

For example, two regional banks with 30 physical sites may have different optimal deployments:

- Bank A: deploys five scanners internally at a data center and performs scans only over the network links.
- Bank B: deploys one scanner at each physical site.

Furthermore, there is no optimal recommendation based on network size:

- Customer A: deploys 40 Nessus scanners to scan a total of 300,000 IP addresses
- Customer B: deploys 300 Nessus scanners at 300 physical sites with local scanner requirements to scan a total of 37,000 IP addresses

Recommendations for Large Enterprise Deployments

In large enterprise deployments, Tenable recommends:

- Adding, at minimum, one scanner for every 5,000 active IP addresses in a zone
- Adding a single scanner to a single zone. Tenable does not recommend adding a scanner to multiple zones.
- Disabling automatic scan distribution if your scan zones contain overlapping IP addresses
- Disabling automatic scan distribution if you are scanning any of your IP addresses from scanners located both inside and outside your network and storing the IP address data in multiple repositories

Agent Scanning

Related Reading: Agent Use Cases (High Latency Networks, Mobile/Distributed Workforces, and Hardened Systems) and Large-Scale Deployment Considerations in the *Nessus Agent Deployment and User Guide*

Nessus Agents can increase the flexibility of your Tenable.sc deployment since agent scanners are not limited by the same network architecture considerations as active scanners. Nessus Agents are also a good solution for high latency networks, unreachable networks, and hardened systems.

You can deploy Nessus Agents to communicate through an intermediary manager: Tenable.io (cloud-based) or Nessus Manager (on-premises). If you deploy large numbers of Nessus Agents, review the large-scale deployment considerations.

Variables Impacting Scan Time

There are many variables in your configurations and environment that can impact your scan performance. The following list summarizes the most common variables to consider when planning your deployment.

Tip: Tenable recommends contacting Professional Services to jointly architect a successful large deployment of Tenable.sc.

Variable	Impact
Your rate of simultaneous assessment	The number of IP addresses you can assess simultaneously depends on two things: • The number of available Nessus scanners • Your **Max Simultaneous Hosts Per Scan** setting in the scan policy Increasing one or both of these is the fastest way to improve your rate of simultaneous assessment and overall scan time. However, large enterprise networks often have infrastructure or technology limitations that prohibit increasing these values beyond a certain maximum. Since Tenable.sc sends jobs to Nessus scanners in chunks and there are eight IP scan segments, you may want to consider setting **Max Simultaneous Hosts Per Scan** to a multiple of eight. **Note:** Real-world performance is highly dependent on your local environment.
Your Nessus environment specifications	Nessus scanners should meet the hardware requirements whenever possible. In rare cases, you may need to install a Nessus scanner on in an underpowered environment. In this case, limit the scan targets the underpowered Nessus scanner is responsible for. Similarly, when deploying Nessus on a virtual machine, assume a 20% decrease in performance and adjust your specifications. Do not deploy Nessus on an over-utilized or over-subscribed virtual infrastructure, as scan per-

Variable	Impact
	formance will suffer and you may experience data corruption.
Your Nessus scan settings	The scan engine has many parameters that are used to modify the scan engine runtime operation. These parameters range from the number of simultaneous hosts scanned to the number of concurrent open TCP sessions. These parameters are meant to allow customers to individually tune the engine parameters to best fit their network by tuning the performance up or down.
Your Tenable.sc scan policy configuration	Your scan policy configuration specifies the depth of your scan. In general, increasing the depth of your scan increases the time to run the scan. Consider the following when evaluating your scan depth: • What type of port scanning is being performed? • What ports are being scanned? • What vulnerabilities are you scanning for? • Are you running credentialed scans? • Are you performing malware checks, filesystem checks, or configuration audits? You can use Tenable-provided templates to perform targeted checks. You can create custom policies to customize all possible policy settings.
Your scanner's proximity to your targets	Tenable recommends placing your scanners close to your targets, connected with minimum latency. Latency has an additive effect on every packet exchanged between a scanner and its target. The largest impacts tend to be network latency and simultaneous plugin checks. For example: • Scanning through routers, VPNs, load balancers, and firewalls can impact the fidelity of your scan results by blocking ports that should be open or by auto-responding to closed ports.

Variable	Impact
	• Scanning numerous hosts behind a single piece of network infrastructure can increase the load on your equipment, given the large number of sessions exchanged between scanner and host.
Your number of live hosts	Scanning a dead host takes less time than scanning a live host. A distribution of IP addresses with a low number of associated hosts takes less time to scan than a distribution of IP addresses with a higher number of hosts.
Your target configurations	Scanning a locked-down system with few exposed network services takes less time than complicated target configurations. For example, a Windows server with a web server, database, and host intrusion prevention software takes more time to scan.
Your target resources	The resources available to the scan target can impact scan time as well. A public-facing system (a system with load) takes longer to scan than an idle backup system.

Data Flow

For information about data flow in Tenable.sc and Tenable.sc Continuous View, see the Tenable Continuous Network Monitoring Architecture Overview.

Operationalize Your Established Deployment

Consider the following when operationalizing your established Tenable.sc deployment:

- Upgrades and Tenable Product Lifecycles
- Backup and Failover
- Logging
- Security
- Performance

Upgrades and Tenable Product Lifecycles

In most large environments, Tenable recommends updating your Tenable products quarterly to take advantage of the feature and security updates in the latest versions of Tenable products.

To plan and prepare for a Tenable.sc upgrade:

- Review the Tenable.sc Release Notes for information about new features, bug fixes, supported upgrade paths, and integrated product version requirements.

  If your upgrade path skips versions of Tenable.sc (e.g., upgrading from 5.6.2.1 to 5.12.0), Tenable recommends reviewing the release notes for all skipped versions. You may need to update your configurations because of features and functionality added in skipped versions.

  Tenable.sc versions sometimes require:

  - A specific minimum version of a downstream product (for example, Nessus) for complete feature support.
  - An updated set of minimum hardware requirements.
  - A separate installation or configuration for a third-party product integration.
- Perform a backup and validate it before beginning the upgrade, as described in Backup and Fail-over.
- Test the upgrade in a test environment before deploying it to your production environment.
- Tenable recommends performing upgrades on your highest tier Tenable.sc instances first. For example, upgrade your reporting tier Tenable.sc instance, then your scanning tier Tenable.sc instance, then your individual scanners.

Architecture Review and Hardware Refresh

Tenable recommends performing an architecture review and considering a hardware refresh every three to five years. You may want to do this more frequently if your underlying environment changes or increases in size, or if your vulnerability policies change (for example, you increase your data retention from 180 to 365 days).

Tenable Product Lifecycles

For information about end-of-support (EOS) and end-of-life (EOL) dates for Tenable products, see the Tenable Software Release Lifecycle Matrix.

Backup and Failover

Related Reading: Backup and Restore in the *Tenable.sc User Guide*

Tenable.sc encourages all organizations, but especially large organizations, to maintain Tenable.sc backups for disaster recovery.

Consider the following when planning and performing your backup:

- In general, linked scanners (e.g., Nessus scanners) do not need to be backed up since they do not permanently store vulnerability data.
- In general, vulnerability trending snapshots consume the most storage in Tenable.sc deployments. Consider creating a separate backup policy for this data; once Tenable.sc creates the nightly vulnerability trending snapshot, the data does not change.
- Tenable does not recommended backing up volatile directories (for example, `/opt/sc/admin/tmp` and `/opt/sc/data/scans`).
- Since Tenable.sc does not encrypt most data on disk, consider encrypting your backups.

Tenable.sc does not support high availability failover scenarios, but you can maintain a cold standby system using system backups.

Logging

Related Reading: System Logs in the *Tenable.sc User Guide*

You may need to monitor a variety of log sources related to your Tenable.sc deployment.

Tenable.sc

Log Location	Description
/opt/sc/admin/logs/<*yyyymm*>.log	Contains detailed information about functionality to troubleshoot unusual system or user activity. You can view the same log activity in the Tenable.sc interface.
/opt/sc/admin/logs/install.log	Written at installation. Review this log only if instructed by Tenable Support.
/opt/sc/admin/logs/upgrade.log	Written during upgrades. Review this log only if instructed by Tenable Support.

Nessus

Note: Your Nessus *data_directory* location depends on your operating system, as described in Data Directories in the *Nessus User Guide*.

Log Location	Description
data_directory/logs/nessusd.messages	Contains Nessus startup and scan parameters, as well as start and stop times for individual IP addresses. You can enable troubleshooting logs using touch debugging, but Tenable does not recommend leaving touch debugging enabled in a production environment.
data_directory/logs/backend.log	Contains backend Nessus application processes. Review this log only if instructed by Tenable Support.

Security

Related Reading: User Access (including LDAP Authentication, Certificate Authentication, SAML Authentication, and WebSeal) and Encryption Strength in the *Tenable.sc User Guide*

Review the following information about Tenable.sc security features and considerations.

Tenable.sc

At its core, Tenable.sc is a web application served with Apache and written in PHP. While controls have been put in place to secure the user interface, Tenable recommends deploying Tenable.sc on a secure, internal-facing network. In high security environments, you may want to restrict the interface only to authorized networks and systems. For more information, see the port requirements.

From a user perspective, Tenable.sc supports a role-based access control model for user data interaction and separation of duties. This allows you to grant application administrators control over management tasks without exposing organizational vulnerability data. Users can authenticate to Tenable.sc in a variety of ways, including local authentication, LDAP/AD authentication, certificate/smart card authentication, SAML authentication, and WebSeal authentication. All user interface interaction, including user authentication, takes place over HTTPS.

You can customize the default Tenable.sc HTTPS certificate to meet your organizational requirements.

Nessus and Nessus Manager

From a network interface perspective, Nessus only requires a connection to Tenable.sc for operational usage; you may want to consider restricting interface access to only the Tenable.sc server. Before restricting access, consider:

- You may need user interface access to Nessus for setup or troubleshooting.
- You need user interface access to Nessus Manager for operational usage.

When connected to Tenable.sc, Nessus does not store any vulnerability or credential data. Nessus runs the scan and transmits the scan data to Tenable.sc using an HTTPS connection. Then, Nessus deletes the scan data.

If you are using Nessus Agents with Tenable.sc, vulnerability data is stored in Nessus Manager or Tenable.io.

Data Storage Encryption

Credentials are stored encrypted on the Tenable.sc server, while vulnerability and application data is not encrypted. Tenable.sc also integrates with PAM solutions, allowing Nessus to access a centralized password store during a network scan.

If your organization requires data at rest encryption for vulnerability data or backup data, Tenable recommends hardware-level disk encryption. Tenable Support does not assist with hardware-level disk encryption.

Communications Encryption

Tenable.sc encrypts all communications over the network. This includes user interaction with the user interface and API as well as all scanner communications and communications with Tenable. You can customize these encryptions to meet specific organizational requirements.

By default, Nessus uses encrypted protocols to authenticate to targets, but the security of this traffic is based on the protocols that the targets support for authentication.

Product Upgrades

In most large environments, Tenable recommends updating your Tenable products quarterly to take advantage of the feature and security updates in the latest versions of Tenable products.

In addition, you can:

- View security-related product updates in our Tenable Product Security Advisories and RSS feed.
- Report vulnerabilities in Tenable products. Tenable releases detections for Tenable product vulnerabilities in our plugin feeds to ensure visibility for outstanding issues.

Performance

Use the following sections to begin optimizing your performance. Tenable strongly recommends using Professional Services Health Checks to optimize Tenable.sc for your specific environment and organizational processes.

Before beginning performance optimization, confirm that your Tenable.sc and scanner deployments meet the environment requirements described in the General Requirements Guide.

Tenable.sc

- Very large deployments should designate instances as scanning tier or reporting tier instances. For more information, see Tiered Deployments.
- If you have complex reporting requirements, consider offloading certain functions to applications designed to handle very large amounts of data with frequent access requests (for example, a SIEM).
- For standalone instances and reporting tier instances, allocate 128 GB of RAM for every 100,000 active IP addresses (for example, for 150,000 IP addresses, allocate 192 GB of RAM).
- If you do not use specific static disk locations (for example, trend data), you can use mount points to offload them to larger, slower storage.
- Unless specially recommended or assisted by Tenable Support or Professional Services, comply with these resource recommendations for all of your Tenable.sc instances:
    - 500 or fewer Tenable.sc user accounts
    - 50 or fewer concurrent Tenable.sc user account sessions
    - 50 or fewer organizations
    - 250 or fewer attached scanners
    - 200 or fewer repositories

    > **Note:** Generally, several smaller repositories perform better than one large repository (for example, five repositories with 5000 IP addresses each generally perform better than a single repository with 25,000 IP addresses).

- In Tenable.sc 5.11 or later, disable creation of sample content (for example, sample dashboards and assets) if they are not needed.

Scanners

- Confirm your Nessus scanner network placement is optimal for the scanner's environment, considering the information in Assessment Scanning Methods.
- Enable Nessus scanner event logging and monitor the logs for signs of performance issues related to overloaded scans.
- In high performance environments (for example, environments where scans must finish by specific deadlines), dedicate hardware resources to Nessus either through physical systems or with dedicated resource pools in virtual environments.
- Review and consider the implications described in Variables Impacting Scan Time.