

The following content is part of the Graylog 6.0 documentation. If you are using another version of Graylog, please switch to your version.

Select Version ▼

Planning Your Deployment

Deploying Graylog effectively requires a thorough grasp of your security information and event management requirements and how Graylog can be leveraged to meet those needs effectively.

The architecture of Graylog plays a pivotal role in determining the success and efficiency of the deployment. Strategic architectural planning guarantees that Graylog has the necessary resources for log storage, processing, and forwarding, along with scalability, redundancy, and security measures. By incorporating these factors into your deployment strategy, you can enhance Graylog's dependability and overall performance. This deployment manual aims to provide a comprehensive roadmap for preparing and executing the deployment of Graylog.

The Graylog Stack

Graylog architecture comprises three key components, collectively known as the Graylog stack: Graylog, OpenSearch, and MongoDB. Let's start with the basics about the Graylog stack, what this comprises, and how it all works together.

Graylog Web Interface

The Graylog Web Interface is a user-friendly dashboard that acts as the control center for managing your log data. It enables users to efficiently query, filter, and analyze logs collected from various sources. Through its intuitive design, administrators can configure inputs for receiving data, set up alerts based on specific log patterns, and create custom dashboards to visualize data trends. This interface bridges the gap between complex log data and actionable insights, streamlining the process of log management and analysis. Furthermore, it allows for easy collaboration among team members by providing user access management and real-time data streaming.

OpenSearch

OpenSearch is a powerful open-source search engine that provides core functionality for the Graylog stack. It is responsible for indexing and storing ingested log data and enables fast retrieval of relevant data through its search capabilities. Given its continuous influx of read and write requests, it demands substantial system resources and storage capacity.

Additionally, OpenSearch provides support for distributed indexing and backup/restore of indices which can be used in disaster recovery scenarios.

Graylog leverages OpenSearch's advanced features such as data aggregation, anomaly detection, and machine learning algorithms to enhance log analysis and provide valuable insights into your IT environments. Its seamless integration with Graylog allows for efficient data processing and retrieval, making it an essential component of the Graylog stack.

MongoDB

MongoDB is a NoSQL database, responsible for storing metadata and configuration data for log messages in Graylog. It acts as a persistent data store for OpenSearch nodes to ensure redundancy and high availability of data.

MongoDB's scalability and fault tolerance make it ideal for large-scale log storage, providing reliable access to archived data when needed. It also supports sharding, which enables horizontal scaling of data storage across multiple servers.

It requires minimal system resources and storage capacity. In case MongoDB experiences downtime, Graylog can continue functioning. You can decide whether or not to incorporate it into a high-availability design. MongoDB suggests using three instances for optimal high availability. This is necessary because if one MongoDB instance fails, MongoDB must designate a primary, and without at least two other instances, confusion may arise between the first two.

Note:

- MongoDB and OpenSearch are databases to which Graylog sends API requests. MongoDB and OpenSearch do not send API requests to Graylog or communicate with each other.
- Graylog stores information about its run state and any data it ingests within the OpenSearch and MongoDB databases. The Graylog service itself requires storage only for its buffers and journal, which temporarily hold log messages while these are processed.

- Graylog suggests that one instance of MongoDB service may share a server with one instance of the Graylog service. This is commonly referred to as a Graylog node.
- Graylog suggests one instance of OpenSearch only and should not share a server with either MongoDB or Graylog. This is commonly referred to as an OpenSearch node.
- Graylog suggests that the minimum setup therefore consists of one Graylog Node and one OpenSearch Node.

Network Architecture

Graylog Nodes

For optimal performance, it is advised to group all Graylog nodes, including the Graylog leader node, within the same load balancer group. This setup ensures the even distribution of incoming log and UI traffic among the nodes. However, a load balancer is unnecessary between Graylog and MongoDB or OpenSearch since these platforms come with built-in mechanisms for distributing workloads among their nodes.

The Graylog API functions on port 9000, which must be reachable on all Graylog nodes. Communication on port 9000 is vital for collecting status metrics from other Graylog nodes within the cluster.

Each Graylog node should connect to OpenSearch nodes on ports 9200 and 9300-9310, and communicate with the MongoDB API on port 27017 for nodes with MongoDB.

Below is a list of the ports we recommend having open by default if port restrictions exist on your Graylog nodes. This includes the default ports used by each input type:

Core Functions

- :443
- :9000
- :9515

Inputs

- :514
- :1514
- :4739
- :5044
- :5555
- :9515

MongoDB

- :27017

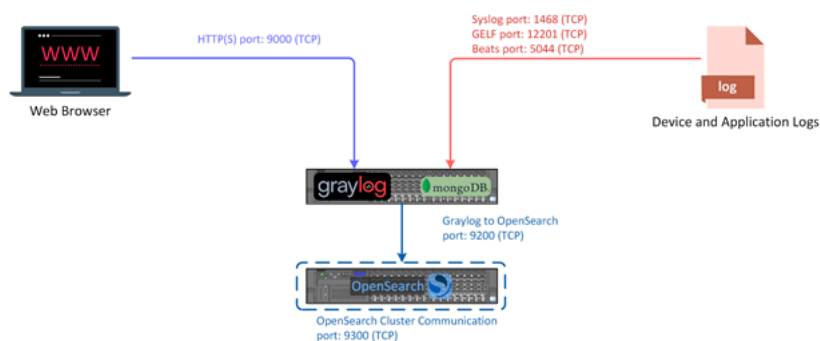
Single-Node Cluster

A single-node Graylog cluster is the simplest setup as it requires minimal configuration and hardware resources, making it a popular choice for small deployments. This type of environment can be used for testing and development purposes but can also be used as a production system if you have small volumes of log data to process. The architecture comprises a Graylog setup with one dedicated node running the entire Graylog stack.

The main drawback of this setup is that it offers no redundancy and will become unavailable if the node goes down. Hence, it should not be used in production environments as downtime can cause serious consequences. If your environment grows and requires more nodes, you can easily scale up to multi-node clusters.

1-10 GB Log Ingestion

⚠ The ports below are default ports. Each setup has unique requirements and ports may be changed during configuration. ⚠



| OpenSearch Servers | |
|----------------------|-------|
| Number of Servers | 1 |
| CPU Cores per server | 8 |
| RAM per server | 24 GB |

| Graylog Servers | |
|----------------------|-------|
| Number of Servers | 1 |
| CPU Cores per server | 8 |
| RAM per server | 16 GB |

Warning: We strongly recommend against installing Graylog and OpenSearch onto the same server. Doing this, could result in competition for system resources, which will in turn dramatically impede high-utilization performance of both services.

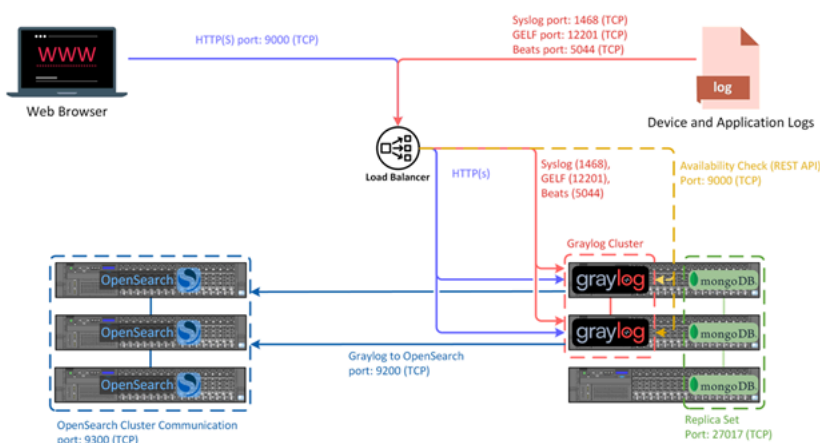
Multi-Node Cluster

The Graylog multi-node cluster architecture provides high availability and scalability for large production environments. Clustering is made possible using a leader-follower mechanism that is user-defined. It allows Graylog nodes, called leaders, to be distributed across multiple physical or virtual machines. These leader nodes are the primary points of contact between administrators and users whose requests are routed through them to their followers. Followers then process incoming messages using the appropriate processor.

Graylog, MongoDB, and OpenSearch instances can each be added to clusters; as a result, you can scale up the size of a Graylog cluster to meet higher ingest needs. Here is an example architecture of a multi-node Graylog cluster.

50-100 GB Log Ingestion

⚠ The ports below are default ports. Each setup has unique requirements and ports may be changed during configuration. ⚠



| OpenSearch Servers | |
|----------------------|-------|
| Number of Servers | 3 |
| CPU Cores per server | 16 |
| RAM per server | 32 GB |

| Graylog Servers | |
|----------------------|-------|
| Number of Servers | 2 |
| CPU Cores per server | 16 |
| RAM per server | 16 GB |

Some considerations when planning a multi-node cluster:

- MongoDB utilizes a quorum voting system, necessitating an odd number of participants in clusters (1, 3, 5, 7, etc.).

- OpenSearch clusters can have data nodes at smaller scales and should include leader and data nodes at larger scales.
- Graylog nodes should be behind a round-robin load balancer.
- Load balancing for OpenSearch and MongoDB is managed at the application layer, with no need for a load balancer between Graylog and OpenSearch or Graylog and MongoDB.

Environment Planning

While the Graylog service can be run on almost any Linux distribution, it is only tested and officially supported on [a limited subset of operating systems](#).

Graylog offers official DEB and RPM package repositories for the following supported operating systems:

- [Debian 10, 11, 12](#)
- [Ubuntu 20.04, 22.04](#)
- [Red Hat 7-9](#)
- [SUSE 12,15](#)

The repositories can be set up by installing a single package. Once that is done, the Graylog packages can be installed via `apt-get` or `yum`. The packages can also be downloaded [with a web browser](#).

Capacity Planning

Sizing

When considering the sizing of your Graylog deployment, a key factor is the volume of log messages your environment generates as this will directly influence the required hardware and storage capacities.

For effective capacity planning, start by estimating the average size of a log message within your environment. This can vary, but a reasonable approximation is between 200 to 300 bytes per message. Multiplying this by the number of messages per second gives you an estimate of the incoming data rate. Below is a sizing table to help guide you in determining the necessary hardware and storage capacity for your Graylog deployment.

| Daily Ingest Volume (GB/day) | Graylog Servers Qty | Graylog CPU Cores | Graylog RAM (GB) | OpenSearch Servers Qty | OpenSearch CPU Cores | OpenSearch RAM (GB) |
|-----------------------------------|---------------------|-------------------|------------------|------------------------|----------------------|---------------------|
| 1-10 | 1 | 8 | 16 | 1 | 8 | 24 |
| 10-20 | 1 | 8 | 16 | 2 | 8 | 32 |
| 20-50 (w/ Mongo Replica Set) | 2 | 16 | 16 | 2 | 16 | 32 |
| 20-50 (without Mongo Replica Set) | 2 | 16 | 16 | 2 | 16 | 32 |
| 50-100 | 2 | 16 | 16 | 3 | 16 | 32 |
| 100-300 | 3 | 16 | 16 | 4 | 16 | 32 |
| 300-500 | 4 | 16 | 16 | 6 | 16 | 32 |

Storage

When setting up Graylog, it is crucial to consider storage capacity for optimal performance.

Graylog server nodes should have the following (minimum) allocated for disk space:

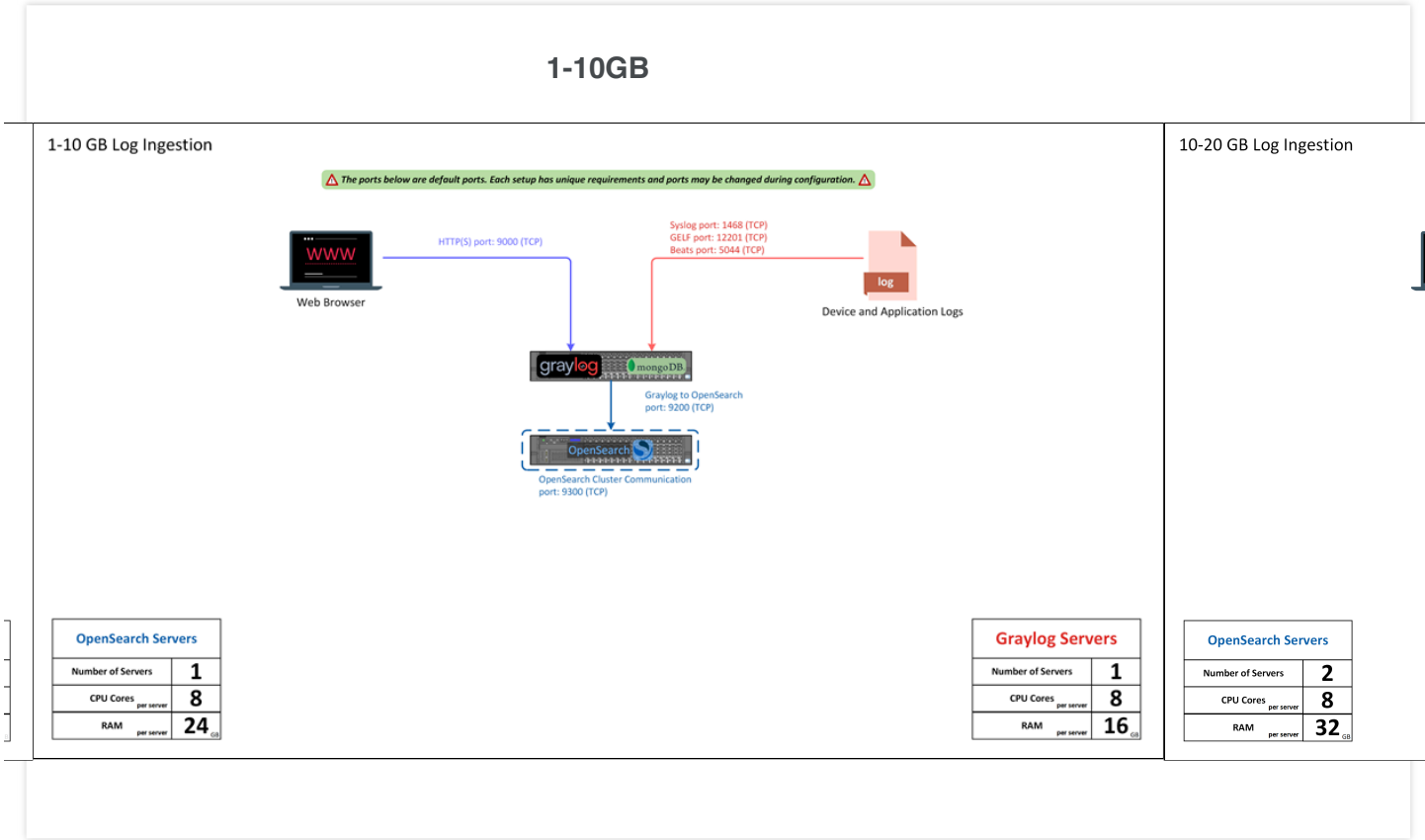
- 80-100GB for MongoDB
- 30GB for the Operating System
- The Graylog journal defaults to 5GB, but this can be configured in [the Graylog configuration file](#) via `message_journal_max_size`. The Graylog service needs fast storage only for its Journal (a buffer of unprocessed messages), from observations the basic SSD 3000 IOPS is sufficient.

Graylog and OpenSearch nodes need high IOPS SSD storage to handle raw logs, message indices, and configurations effectively. OpenSearch recommends a minimum of 50,000 IOPS, with 300,000 IOPS as the preferred option. Lower IOPS values can work with burst mechanisms like AWS Burst Credits.

IOPS play a significant role in the performance of OpenSearch nodes, especially during high utilization periods, given that OpenSearch performs data indexing and searching tasks. These nodes will require substantial memory and processing power, especially for environments with high log volumes or complex search queries. Observations indicate that maintaining IOPS above 8,000 is crucial for optimal cluster performance.

RAM/Memory

The slideshow below provides reference architecture for various deployment sizes. For sizes covered by the reference architecture, use the recommendation as shown. For all but 10-20GB/day, server RAM recommendation is 32 GB.



Note:

- OpenSearch should be configured to consume half (1/2) of the server RAM, and should not exceed 31 GB. This [article](#) provides details about why not to exceed 31 GB of OpenSearch heap.
- All nodes in a cluster need to be running the same OpenSearch version.

Shard

A [shard](#) is essentially a partition of data in a database or search engine. Shards plays a crucial role in enhancing the performance and scalability of your Graylog system by distributing data across multiple nodes. Sharding is the process of partitioning indices into shards, which are then

distributed across the OpenSearch cluster. Each shard can be replicated to provide fault tolerance and high availability. The key to effective [sharding](#) lies in balancing the number of shards and their sizes, as this can significantly impact query performance and system stability.

When [sharding](#), consider the total data volume and the search performance requirements of your environment. A common practice is to start with a conservative number of shards and adjust based on the observed performance and growth trends. Remember, having too many small shards can be as detrimental to performance as having too few large shards, as both scenarios can strain cluster resources.

Shard sizes should never exceed 40GB before being rotated. Going above this size will make searches run very slowly. If you are ingesting 20GB of traffic per day to one given index, your shard rotation period for that index should not be larger than 2 days. While optimal values are unique to each deployment, 20-25GB is a reasonable initial value which can be updated over time. The number of shards per GB of heap space should be less than 20, and for every 20 active shards you will need roughly 1GB of heap memory in your cluster. This can vary in practice (an index that receives almost no traffic will have less demanding shards).

Notes:

- Increasing the number of Shards per index increases the speed of indexing/searches.
- The ideal number of shards in an index should align with the quantity of nodes within your OpenSearch cluster. For instance, if you have 3 nodes, it is recommended to have a minimum of 3 shards.
- The number of active shards in an index should be calculated by the sum of the following for all your indices: (Data searchable period days/Rotation Period days) x Shards per Index = GB Heap recommended.

Daily Ingest Volumes

Daily Ingest Volumes represent a volume or rate at which data acting as input enters a Graylog Environment. This rate can be broken down into units of measurement more commonly used when describing bandwidth of various system resources. The bandwidth can then be compared to the available hardware to identify resources that can satisfy the minimum required amount of bandwidth required by the Daily Ingest Volume.

Understanding these factors and aligning storage resources with the required bandwidth ensures smooth data ingestion and system functionality. Refer to the table below for detailed breakdowns to effectively evaluate resource needs.

| GB / day | MB / day | MB / hr | MB / min | MB / sec | Mb / sec |
|----------|-------------|-----------|----------|----------|----------|
| 1 | 1024.000 | 42.667 | 0.711 | 0.012 | 0.095 |
| 5 | 5120.000 | 213.333 | 3.556 | 0.059 | 0.474 |
| 10 | 10240.000 | 426.667 | 7.111 | 0.119 | 0.948 |
| 20 | 20480.000 | 853.333 | 14.222 | 0.237 | 1.896 |
| 50 | 51200.000 | 2133.333 | 35.556 | 0.593 | 4.741 |
| 100 | 102400.000 | 4266.667 | 71.111 | 1.185 | 9.481 |
| 150 | 153600.000 | 6400.000 | 106.667 | 1.778 | 14.222 |
| 200 | 204800.000 | 8533.333 | 142.222 | 2.370 | 18.963 |
| 500 | 512000.000 | 21333.333 | 355.556 | 5.926 | 47.407 |
| 1024 | 1024000.000 | 42666.667 | 711.111 | 11.852 | 94.815 |

Hint: The Graylog index rotation strategy is used for [index sets](#).

Both Graylog and OpenSearch nodes contain databases and should be provisioned with high IOPS SSD storage. Graylog Superwarm Cloud instances are provisioned with NVMe SSD storage (850k/360k read/write IOPS according to Amazon).

The performance of OpenSearch nodes is sensitive to the IOPS of the storage medium due to the high volume of read/writes.

IOPS usage in the cluster is not constant. Rather, it typically has a low baseline, which can spike into multiples 10x or even 100x higher during moments of high utilization. Observing a week of activity on Graylog Cloud, once-daily spikes of 60k Read & 20k Write are not uncommon.

While 50,000 looks best, observations suggest that 25,000 IOPS would still be an acceptable value for a high performance cluster without burst. At this level of IOPS, storage speed will impact performance only during moments of high utilization, such as Index Rotation and Index Optimization, or running expensive reports.

Our observations also suggest that IOPS below 8000 IOPS will be a regular choke-point on performance.

Daily Ingest Volume Breakdown

Furthermore, you can then calculate a minimum baseline amount of IOPs required by the bandwidth by using this formula: $IOPs = (MB/sec \times 106) / Block\text{-}size\text{ in bytes}$

A block size of 4 Kilobytes (KB) is the most common size used in practice. In certain cases, a different value may be desired, but for this instance we will calculate a minimum baseline IOPs with a block size of 4KB.

Calculate IOPs

| GB / day | MB / day | MB / sec | IOPs |
|----------|-------------|----------|----------|
| 1 | 1024.000 | 0.012 | 2.894 |
| 5 | 5120.000 | 0.059 | 14.468 |
| 10 | 10240.000 | 0.119 | 28.935 |
| 20 | 20480.000 | 0.237 | 57.870 |
| 50 | 51200.000 | 0.593 | 144.676 |
| 100 | 102400.000 | 1.185 | 289.352 |
| 150 | 153600.000 | 1.778 | 434.028 |
| 200 | 204800.000 | 2.370 | 578.704 |
| 500 | 512000.000 | 5.926 | 1446.759 |
| 1024 | 1024000.000 | 11.852 | 2893.519 |

When reading these values, it must be kept in mind that these are the minimum amount of continuous transfer rates necessary to deliver the volume of data over the given period of time.

In reality (or application), data ingest is often the opposite of continuous; various resources that generate logging typically do so in a burst-like or sinusoidal wave fashion. This is extremely important to consider when deciding the appropriate amount of system resources to make available to the Graylog environment.

As most hardware bandwidth capabilities are often advertised in a rate of Megabits per second instead of Megabytes, the following table shows the relation between Daily Ingest Volume Tiers and their minimum IOPs and minimum throughput/bandwidth requirements.

Minimum Required IOPs and Bandwidth

| Daily Ingest Volume Tiers GB / day | MB / day | MB / sec | IOPs |
|--|-------------|----------|----------|
| 1 | 1024.000 | 0.012 | 2.894 |
| 5 | 5120.000 | 0.059 | 14.468 |
| 10 | 10240.000 | 0.119 | 28.935 |
| 20 | 20480.000 | 0.237 | 57.870 |
| 50 | 51200.000 | 0.593 | 144.676 |
| 100 | 102400.000 | 1.185 | 289.352 |
| 150 | 153600.000 | 1.778 | 434.028 |
| 200 | 204800.000 | 2.370 | 578.704 |
| 500 | 512000.000 | 5.926 | 1446.759 |
| 1024 | 1024000.000 | 11.852 | 2893.519 |

Beyond the Daily Ingest Volume

As previously mentioned, the values being derived here are minimums. Moreover, they only account for the volume or rate of I/O anticipated to be ingested. Therefore, it would not be wise to only use these values as filters when selecting adequate hardware since they represent only the bare minimum needed to ingest a certain volume of data or ingest data at a certain rate. To calculate the number of disk journal I/O operations performed for a given event, we use a ratio of 3 IOPS per event. So, for 2000 eps, we would recommend 6000 IOPS. This recommendation is consistent with observations of existing Graylog customers who are currently running a 100+GB cluster with their OpenSearch using 500 IOP storage.

Storage should use XFS file system, which will give better performance than Ext4.

Data Retention

Next, consider the retention period for your log messages. This is how long you intend to keep the logs before they are archived or deleted. The retention period will significantly influence the total storage requirements for your deployment. For instance, retaining logs for 30 days will require substantially less storage compared to a 365-day retention policy. Data retention policies often dictate how long data should be stored, for what purpose, and in what format. It is essential for organizations to understand and implement retention policies for their data infrastructure.

Graylog offers data retention features such as [Index Time Size Optimization](#) and [Archiving](#), which aids in optimizing resource utilization. It facilitates the secure storage of data for extended durations and empowers users to define data retention periods and select data for archiving. Moreover, it presents diverse archiving solutions and functionality to address data access queries. With Graylog's architecture, users can tailor data retention based on user-specific policies.

Archiving

[Archiving](#) in Graylog is exclusively handled by the leader node. During this process, the leader node retrieves data from OpenSearch, locally compresses it, and then transfers it to the assigned storage volume. Access to the storage for saving archives is crucial for the leader node and high resource usage on this node during archiving tasks is expected.

While there are various archiving standards available, the default choice is [Gzip](#). However, we recommend opting for [Zstandard](#) due to its superior compression speed and rate. Typically, archived indices reduce to about 30% of their original size.

Ideally, archiving indices should be completed in under 8 hours daily. When archiving to an S3 bucket, it is advisable to conduct a speed test for uploads from the Graylog leader node. Estimate the time needed daily for uploads by multiplying the daily ingest GB by 0.3. This evaluation is to identify any upload speed bottlenecks, not the archiving speed of the Graylog cluster.

All Graylog nodes must have access to the archiving storage volume. Ensure there are no speed bottlenecks between Graylog and the storage volume used for archiving. This evaluation should also encompass internet upload speeds, especially if the Graylog Master Node is chosen to store archives to Amazon S3.

If archiving to local or network storage, a similar assessment should be conducted from the Graylog Master to identify any transfer speed bottlenecks in the network or storage.

It is critical not to use the default directory, currently located within [/tmp](#). This directory is a temporary storage space that the OS periodically purges of content.



[Graylog Home](#)

[Resources](#)

[Community](#)

[Blog](#)

[Videos](#)

[Webinars](#)

[Events](#)

[Graylog Academy](#)

[Release Notes](#)

[Customer Support](#)

© 2024 Graylog, Inc. | All rights reserved

[Privacy Policy](#) | [Legal](#)