工具和应用程序指南

StorageGRID solutions and resources

NetApp
December 10, 2025

This PDF was generated from https://docs.netapp.com/zh-cn/storagegrid-enable/tools-apps-guides/use-cloudera-hadoop-s3a-connector.html on December 10, 2025. Always check docs.netapp.com for the latest.

目录

工具和应用程序指南 1
将Cloudera Hadoop S3A连接器与StorageGRID 结合使用 1
为什么要使用S3A执行Hadoop工作流? 1
将S3A连接器配置为使用StorageGRID 1
测试与StorageGRID 的S3A连接 4
使用S3cmd测试和演示StorageGRID 上的S3访问 7
安装和配置S3cmd 7
初始配置步骤 7
基本命令示例 8
使用NetApp StorageGRID 作为公共存储的Vertica Eon模式数据库 8
简介 8
NetApp StorageGRID 建议 10
在StorageGRID 上使用公用存储在内部安装Eon模式 11
从何处查找追加信息 22
版本历史记录 22
使用ELK堆栈进行StorageGRID 日志分析 22
要求 22
示例文件 22
假设 23
说明 23
其他资源 27
使用Prometheus和Grafana延长指标保留期限 28
简介 28
联合Prometheus 28
安装和配置Grafana 37
使用 F5 DNS 实现StorageGRID 的全局负载均衡 44
简介 44
F5 BIG-IP 多站点StorageGRID配置 44
结论 59
Datadog SNMP配置 59
配置Datadog 59
使用rclone在StorageGRID 上迁移、放置和删除对象 62
安装和配置rclone 62
基本命令示例 71
使用Veeam备份和复制进行部署的StorageGRID最佳实践 74
概述 74
Veeam配置 74
StorageGRID配置 75
实施要点 78

监控StorageGRID . . . 83
从何处查找追加信息 . . . 86
使用StorageGRID配置d不良 数据源 . . . 86
配置d不良 数据源 . . . 86
说明 . . . 86
NetApp StorageGRID与GitLab . . . 89
对象存储连接示例 . . . 89

工具和应用程序指南

将Cloudera Hadoop S3A连接器与StorageGRID 结合使用

作者：郑安杰

一段时间以来、Hadoop一直是数据科学家的最爱。通过Hadoop、可以使用简单的编程框架在多个计算机集群之间分布式处理大型数据集。Hadoop旨在从单个服务器扩展到数千台计算机、每台计算机都拥有本地计算和存储。

为什么要使用S3A执行Hadoop工作流?

随着数据量的不断增长、使用自己的计算和存储添加新计算机的方法变得效率低下。线性扩展为高效使用资源和管理基础架构带来了挑战。

为了应对这些挑战、Hadoop S3A客户端可为S3对象存储提供高性能I/O。使用S3A实施Hadoop工作流有助于将对象存储用作数据存储库、并将计算和存储分开、进而使您能够独立扩展计算和存储。通过分离计算和存储、您还可以将适当数量的资源专用于计算作业、并根据数据集的大小提供容量。因此、您可以降低Hadoop工作流的总体TCO。

将S3A连接器配置为使用StorageGRID

前提条件

- 用于Hadoop S3A连接测试的StorageGRID S3端点URL、租户S3访问密钥和机密密钥。
- Cloudera集群以及对集群中每个主机的root或sudo权限、用于安装Java软件包。

截至2022年4月、使用Cloudera 7.1.7的Java 11.0.14已针对StorageGRID 11.5和11.5进行了测试。但是、新安装时的Java版本号可能会有所不同。

安装Java软件包

1. 检查 "Cloudera支持表" 支持的JDK版本。
2. 下载 "Java 11.x软件包" 与Cloudera集群操作系统匹配。将此软件包复制到集群中的每个主机。在此示例中、rpm软件包用于CentOS。
3. 以root身份或使用具有sudo权限的帐户登录到每个主机。在每个主机上执行以下步骤：
   a. 安装软件包：

```
$ sudo rpm -Uvh jdk-11.0.14_linux-x64_bin.rpm
```

   b. 检查Java的安装位置。如果安装了多个版本、请将新安装的版本设置为默认值：

```
alternatives --config java

There are 2 programs which provide 'java'.

  Selection    Command
-----------------------------------------------
 +1           /usr/java/jre1.8.0_291-amd64/bin/java
  2           /usr/java/jdk-11.0.14/bin/java

Enter to keep the current selection[+], or type selection number: 2
```

c. 将此行添加到`/etc/profile`的末尾。路径应与上述选择的路径匹配：

```
export JAVA_HOME=/usr/java/jdk-11.0.14
```

d. 运行以下命令以使配置文件生效：

```
source /etc/profile
```

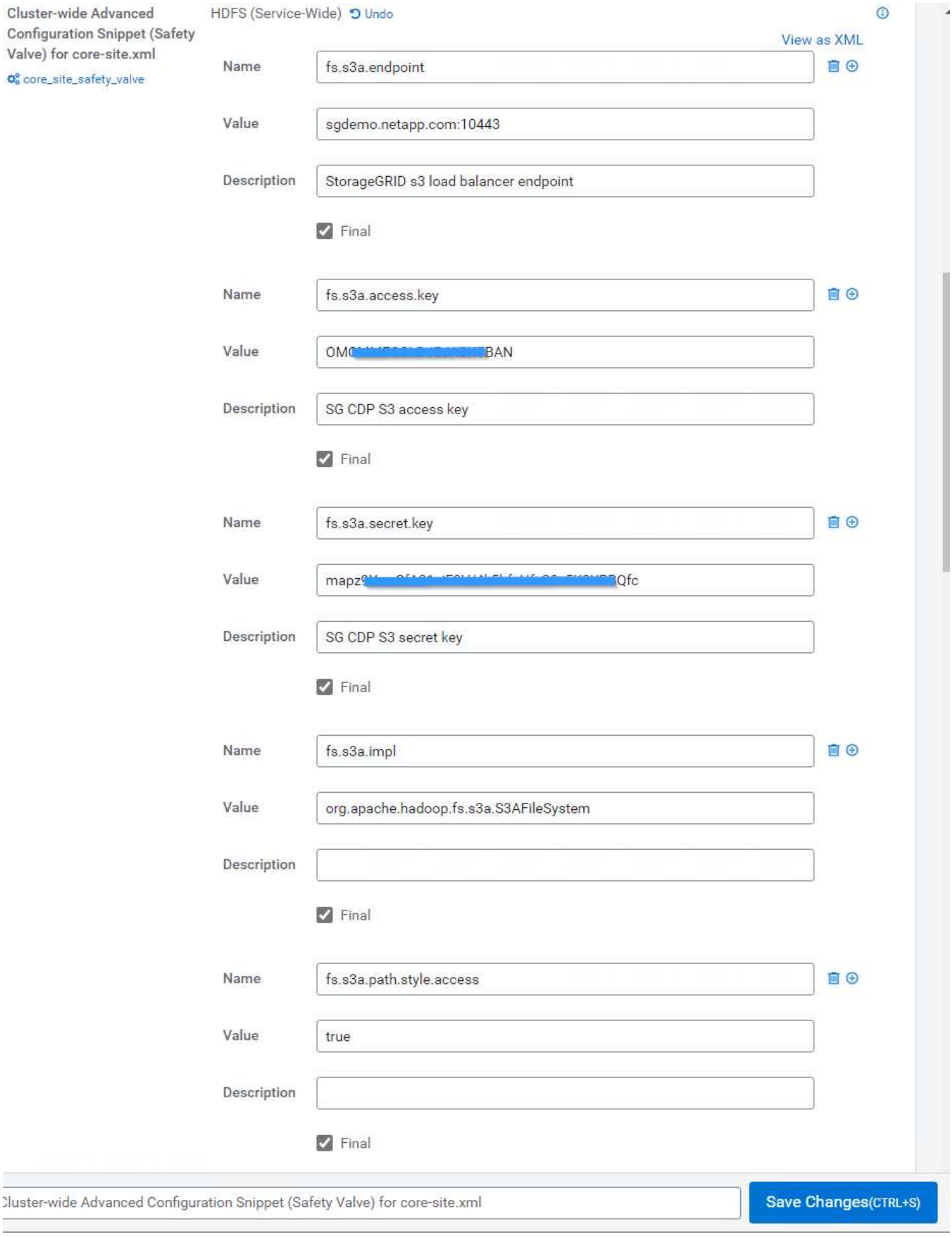
Cloudera HDFS S3A配置

- 步骤 *

1. 从Cloudera Manager GUI中、选择Clusters > HDFS、然后选择Configuration。
2. 在类别下、选择高级、然后向下滚动以找到`core-site.xml的集群范围高级配置片段(安全阀)`。
3. 单击(+)符号并添加以下值对。

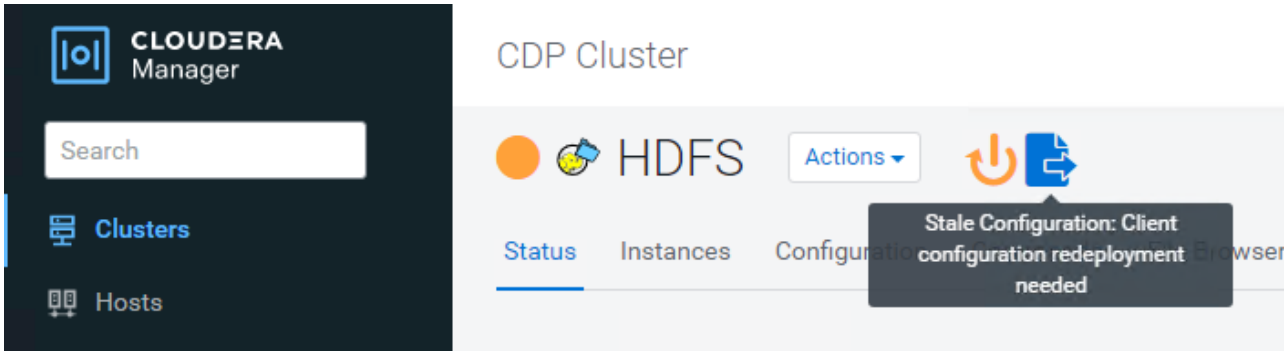
Name	价值
fs.s3a.access.key	< *StorageGRID* 中的租户*S3*访问密钥>
fs.s3a.secret.key	< *StorageGRID* 中的租户*S3*密钥>
fs.s3a.connection.ssl.enabled	true或false (如果缺少此条目、则默认为https)
fs.s3a.endpoint	StorageGRID S3端点：端口>_
fs.s3a.impl	org.apache.hadoop.fs.s3a.s3AFileSystem
fs.s3a.path.style.access	true或false (如果缺少此条目、则默认为虚拟主机模式)

屏幕截图示例



1. 单击保存更改按钮。从HDFS菜单栏中选择陈旧配置图标、在下一页上选择重新启动陈旧服务、然后选

择立即重新启动。



测试与StorageGRID 的S3A连接

执行基本连接测试

登录到Cloudera集群中的一个主机、然后输入`Hadoop FS -ls S3a：//<bucket-name>_/`。

以下示例将使用路径sysle和已有的HDFS-test分段以及一个测试对象。

```
[root@ce-n1 ~]# hadoop fs -ls s3a://hdfs-test/
22/02/15 18:24:37 WARN impl.MetricsConfig: Cannot locate configuration:
tried hadoop-metrics2-s3a-file-system.properties,hadoop-
metrics2.properties
22/02/15 18:24:37 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot
period at 10 second(s).
22/02/15 18:24:37 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system started
22/02/15 18:24:37 INFO Configuration.deprecation: No unit for
fs.s3a.connection.request.timeout(0) assuming SECONDS
Found 1 items
-rw-rw-rw-   1 root root       1679 2022-02-14 16:03 s3a://hdfs-test/test
22/02/15 18:24:38 INFO impl.MetricsSystemImpl: Stopping s3a-file-system
metrics system...
22/02/15 18:24:38 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system stopped.
22/02/15 18:24:38 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system shutdown complete.
```

故障排除

场景 **1**

使用HTTPS连接到StorageGRID 、并在15分钟超时后收到`shapore_failure`错误。

*原因：*旧版JRE/JDK使用过时或不受支持的TLS密码套件连接到StorageGRID。

错误消息示例

```
[root@ce-n1 ~]# hadoop fs -ls s3a://hdfs-test/
22/02/15 18:52:34 WARN impl.MetricsConfig: Cannot locate configuration:
tried hadoop-metrics2-s3a-file-system.properties,hadoop-
metrics2.properties
22/02/15 18:52:34 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot
period at 10 second(s).
22/02/15 18:52:34 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system started
22/02/15 18:52:35 INFO Configuration.deprecation: No unit for
fs.s3a.connection.request.timeout(0) assuming SECONDS
22/02/15 19:04:51 INFO impl.MetricsSystemImpl: Stopping s3a-file-system
metrics system...
22/02/15 19:04:51 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system stopped.
22/02/15 19:04:51 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system shutdown complete.
22/02/15 19:04:51 WARN fs.FileSystem: Failed to initialize fileystem
s3a://hdfs-test/: org.apache.hadoop.fs.s3a.AWSClientIOException:
doesBucketExistV2 on hdfs: com.amazonaws.SdkClientException: Unable to
execute HTTP request: Received fatal alert: handshake_failure: Unable to
execute HTTP request: Received fatal alert: handshake_failure
ls: doesBucketExistV2 on hdfs: com.amazonaws.SdkClientException: Unable to
execute HTTP request: Received fatal alert: handshake_failure: Unable to
execute HTTP request: Received fatal alert: handshake_failure
```

*解析：*确保已安装JDK 11.x或更高版本并将其设置为默认Java库。请参见 安装Java软件包 部分、了解更多信息。

场景2：

无法连接到StorageGRID 、并显示错误消息`无法找到所请求目标的有效证书路径`。

原因： StorageGRID S3端点服务器证书不受Java程序信任。

错误消息示例：

```
[root@hdp6 ~]# hadoop fs -ls s3a://hdfs-test/
22/03/11 20:58:12 WARN impl.MetricsConfig: Cannot locate configuration:
tried hadoop-metrics2-s3a-file-system.properties,hadoop-
metrics2.properties
22/03/11 20:58:13 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot
period at 10 second(s).
22/03/11 20:58:13 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system started
22/03/11 20:58:13 INFO Configuration.deprecation: No unit for
fs.s3a.connection.request.timeout(0) assuming SECONDS
22/03/11 21:12:25 INFO impl.MetricsSystemImpl: Stopping s3a-file-system
metrics system...
22/03/11 21:12:25 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system stopped.
22/03/11 21:12:25 INFO impl.MetricsSystemImpl: s3a-file-system metrics
system shutdown complete.
22/03/11 21:12:25 WARN fs.FileSystem: Failed to initialize fileystem
s3a://hdfs-test/: org.apache.hadoop.fs.s3a.AWSClientIOException:
doesBucketExistV2 on hdfs: com.amazonaws.SdkClientException: Unable to
execute HTTP request: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find
valid certification path to requested target: Unable to execute HTTP
request: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find
valid certification path to requested target
```

解决方法： NetApp建议使用由已知的公有 证书签名颁发机构颁发的服务器证书、以确保身份验证安全。或者、也可以向Java信任存储库添加自定义CA或服务器证书。

要将StorageGRID 自定义CA或服务器证书添加到Java信任存储、请完成以下步骤。

1. 备份现有的默认Java cacerts.

```
cp -ap $JAVA_HOME/lib/security/cacerts
$JAVA_HOME/lib/security/cacerts.orig
```

2. 将StorageGRID S3端点证书导入到Java信任存储。

```
keytool -import -trustcacerts -keystore $JAVA_HOME/lib/security/cacerts
-storepass changeit -noprompt -alias sg-lb -file <StorageGRID CA or
server cert in pem format>
```

故障排除提示

1. 提高Hadoop日志级别以进行调试。

   ```
   export Hadoop_root_logger = hadoop.root.logger = debug、console
   ```

2. 执行命令、并将日志消息定向到error.log。

   ```
   Hadoop FS -ls S3a：//<bucket-name>_/&>error.log
   ```

作者：郑安杰

使用S3cmd测试和演示StorageGRID 上的S3访问

作者：*Aron Klein*

S3cmd是一个用于S3操作的免费命令行工具和客户端。您可以使用s3cmd在StorageGRID 上测试和演示S3访问。

安装和配置S3cmd

要在工作站或服务器上安装S3cmd、请从下载它 "命令行S3客户端"。s3cmd会作为一种工具预先安装在每个StorageGRID 节点上、以协助进行故障排除。

初始配置步骤

1. s3cmd -configure
2. 请仅提供access_key和secret_key、其余请保留默认值。
3. 是否使用提供的凭据测试访问？[Y/n]：n (跳过测试、因为测试将失败)
4. 是否保存设置？[Y/N] y
   a. 配置已保存到"/root/.s3cfg"
5. 在.s3cfg中、使"="符号后面的字段host_base和host_bucket为空：
   a. host_base =
   b. host_bucket =

如果在步骤4中指定host_base和host_bucket、则无需在命令行界面中使用-host指定端点。示例

```
host_base = 192.168.1.91:8082
host_bucket = bucketX.192.168.1.91:8082
s3cmd ls s3://bucketX --no-check-certificate
```

基本命令示例

- 创建存储分段：

```
s3cmd MB S3：//s3cmdbucket -host=<endpoint>：<port>-no-check-certificate
```

- 列出所有分段：

```
s3ls命令-host=<endpoint>：<port>-no-check-certificate
```

- 列出所有分段及其内容：

```
s3cmd la -host=<endpoint>：<port>-no-check-certificate
```

- 列出特定分段中的对象：

```
s3cmd ls s3：//<bucket>-host=<endpoint>：<port>-no-check-certificate
```

- 删除分段：

```
s3RB cmd S3：//s3cmdbucket -host=<endpoint>：<port>-no-check-certificate
```

- 放置对象：

```
s3cmd PUT <file> s3：//<bucket>-host=<endpoint>：<port>-no-check-certificate
```

- 获取对象：

```
s3cmd get S3：//<bucket>/<object><file>-host=<endpoint>：<port>-no-check-
certificate
```

- 删除对象：

```
s3cmd del S3：//<bucket>/<object>-host=<endpoint>：<port>-no-check-certificate
```

使用NetApp StorageGRID 作为公共存储的Vertica Eon模式数据库

作者：郑安杰

本指南介绍在NetApp StorageGRID 上使用公共存储创建Vertica Eon模式数据库的操作步骤。

简介

Vertica是一款分析数据库管理软件。它是一个柱形存储平台、专为处理大量数据而设计、可在传统密集型情形下实现非常快速的查询性能。Vertica数据库以两种模式之一运行：Eon或Enterprise。您可以在内部或云中部署这两种模式。

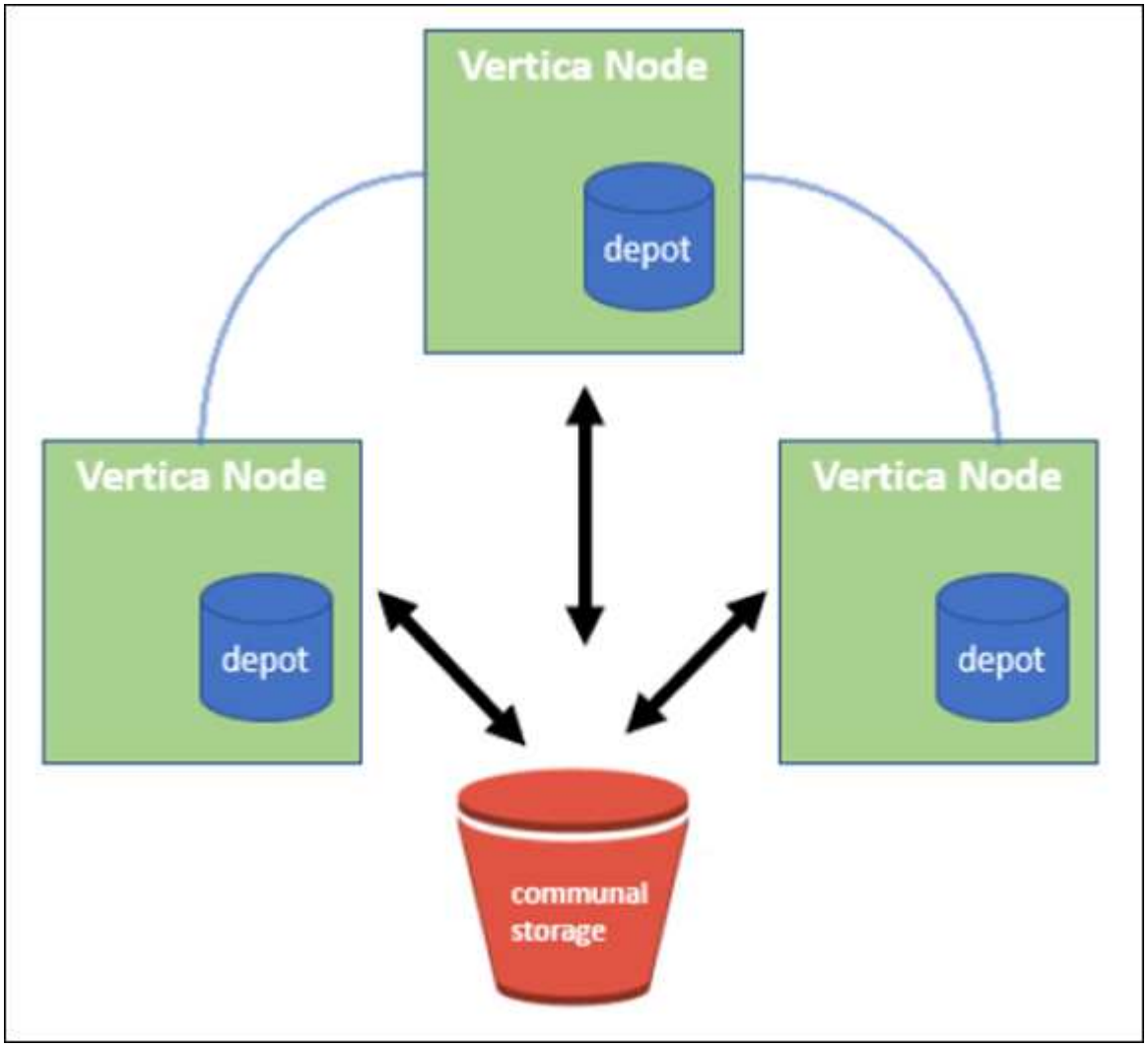
Eon和企业模式在数据存储位置方面主要不同：

- Eon模式数据库使用公共存储来存储其数据。这是Vertica的建议。
- 企业模式数据库将数据本地存储在构成数据库的节点的文件系统中。

Eon模式架构

Eon模式可将计算资源与数据库的公用存储层分离、从而使计算和存储可以单独进行扩展。Eon模式下的Vertica经过优化、可处理各种工作负载、并通过使用单独的计算和存储资源彼此隔离。

Eon模式将数据存储在名为公共存储的共享对象存储中、即S3存储分段、托管在内部或Amazon S3上。



公用存储

Eon模式不会在本地存储数据、而是对所有数据和目录(元数据)使用一个公共存储位置。公共存储是数据库的集中存储位置、在数据库节点之间共享。

公共存储具有以下属性：

- 与单个计算机上的磁盘存储相比、云或内部对象存储中的公共存储更具弹性、并且由于存储故障而不易受到数据丢失的影响。
- 任何使用相同路径的节点都可以读取任何数据。

- 容量不受节点上磁盘空间的限制。
- 由于数据是以社区方式存储的、因此您可以灵活地扩展集群以满足不断变化的需求。如果数据存储在节点本地、则添加或删除节点需要在节点之间移动大量数据、以便将其从要删除的节点或新创建的节点上移出。

仓库

公共存储的一个缺点是速度。从共享云位置访问数据比从本地磁盘读取数据要慢。此外、如果许多节点同时从公共存储读取数据、则与该存储的连接可能会成为瓶颈。为了提高数据访问速度、Eon模式数据库中的节点会维护一个名为存储库的本地数据磁盘缓存。执行查询时、节点会首先检查所需数据是否位于仓库中。如果是、则它将使用数据的本地副本完成查询。如果数据不在存储库中、则节点将从公共存储中提取数据、并在存储库中保存一份副本。

NetApp StorageGRID 建议

Vertica将数据库数据存储在对象存储中、即数千(或数百万)个压缩对象(观察到的大小为每个对象200到500 MB)。当用户运行数据库查询时、Vertica会使用byte-range get调用并行从这些压缩对象检索选定的数据范围。每个字节范围GET大约为8 KB。

在10 TB数据库仓库用户查询测试期间、每秒向网格发送4、000到10、000个GET (字节范围GET)请求。在使用SG6060设备运行此测试时、尽管每个设备节点的CPU利用率百分比较低(约为20%到30%)、但2/3的CPU时间正在等待I/O在SGF6024上观察到I/O等待的百分比非常小(0%到0.5%)。

由于对小型IOPS的需求较高且延迟要求非常低(平均值应小于0.01秒)、NetApp建议对对象存储服务使用SFG6024。如果非常大的数据库需要使用SG6060、则客户应与Vertica客户团队合作进行仓库规模估算、以支持主动查询的数据集。

对于管理节点和API网关节点、客户可以使用SG100或SG1000。选择此选项取决于并行用户查询请求的数量和数据库大小。如果客户希望使用第三方负载平衡器、NetApp建议为高性能需求工作负载配置一个专用的负载平衡器。有关StorageGRID 规模估算、请咨询NetApp客户团队。

其他StorageGRID 配置建议包括：

- 网格拓扑。请勿在同一网格站点上将SGF6024与其他存储设备型号混合使用。如果您希望使用SG6060进行长期归档保护、请在其自己的网格站点(物理或逻辑站点)中为活动数据库保留具有专用网格负载平衡器的SGF6024、以提高性能。在同一站点混用不同型号的设备会降低站点的整体性能。
- 数据保护。使用复制副本进行保护。请勿对活动数据库使用纠删编码。客户可以使用纠删编码对非活动数据库进行长期保护。
- 请勿启用网格压缩。Vertica会先压缩对象、然后再存储到对象存储。启用网格压缩不会进一步节省存储使用量、并且会显著降低字节范围GET性能。
- * HTTP与HTTPS S3端点连接*。在基准测试期间、我们观察到从Vertica集群到StorageGRID 负载平衡器端点使用HTTP S3连接时、性能提高了大约5%。此选项应根据客户的安全要求来选择。

Vertica配置的建议包括：

- 读取和写入操作已启用* Vertica数据库默认存储库设置(值= 1)*。NetApp强烈建议保持启用这些仓库设置以提高性能。
- 禁用流式传输限制。有关配置详细信息、请参见一节 禁用流限制。

在StorageGRID 上使用公用存储在内部安装Eon模式

以下各节介绍了在StorageGRID 上使用公共存储在内部安装Eon模式的顺序操作步骤。用于配置内部简单存储服务(S3)兼容对象存储的操作步骤 类似于Vertica指南中的操作步骤。 "在内部安装Eon模式数据库"。

以下设置用于功能测试：

- StorageGRID 11.4.0.4
- Vertica 10.1.0
- 使用三个虚拟机(VM)和CentOS 7.x操作系统为Vertica节点构建集群。此设置仅用于功能测试、不适用于Vertica生产数据库集群。

这三个节点都设置了安全Shell (SSH)密钥、以允许在集群中的节点之间使用SSH而不使用密码。

NetApp StorageGRID 需要提供的信息

要在StorageGRID 上使用公共存储在内部安装Eon模式、您必须具备以下前提条件信息。

- StorageGRID S3端点的IP地址或完全限定域名(FQDN)和端口号。如果您使用的是HTTPS、请使用在StorageGRID S3端点上实施的自定义证书颁发机构(CA)或自签名SSL证书。
- 存储分段名称。它必须已预先存在且为空。
- 访问密钥ID和密钥访问密钥、对存储分段具有读写访问权限。

创建授权文件以访问S3端点

在创建授权文件以访问S3端点时、需要满足以下前提条件：

- 已安装Vertica。
- 集群已设置、配置完毕、可用于创建数据库。

要创建授权文件以访问S3端点、请执行以下步骤：

1. 登录到要运行`admintools`的Vertica节点以创建Eon模式数据库。

   默认用户为`dbadmin`、在Vertica集群安装期间创建。

2. 使用文本编辑器在`/home/DBadmin`目录下创建文件。文件名可以是所需的任何内容、例如、`sg_auth.conf`。
3. 如果S3端点使用的是标准HTTP端口80或HTTPS端口443、请跳过端口号。要使用HTTPS、请设置以下值：

   - `awsenablehttps = 1`、否则将值设置为`0`。
   - `awsauth =<S3 access key ID>`：<机密访问密钥>
   - `awsendpoint =< StorageGRID S3 Endpoint>`：<端口>

     要对StorageGRID S3端点HTTPS连接使用自定义CA或自签名SSL证书、请指定证书的完整文件路径和文件名。此文件必须位于每个Vertica节点上的同一位置、并对所有用户具有读取权限。如果StorageGRID S3端点SSL证书由公共已知CA签名、请跳过此步骤。

-awscfilm =<文件路径/文件名>

例如、请参见以下示例文件：

```
awsauth = MNVU4OYFAY2xyz123:03vuO4M4KmdfwffT8nqnBmnMVTr78Gu9wANabcxyz
awsendpoint = s3.england.connectlab.io:10443
awsenablehttps = 1
awscafile = /etc/custom-cert/grid.pem
```

+

在生产环境中、客户应在StorageGRID S3负载平衡器端点上实施一个由公共已知CA签名的服务器证书。

在所有**Vertica**节点上选择存储库路径

在每个节点上为存储库存储路径选择或创建一个目录。为depot storage path参数提供的目录必须具有以下内容：

- 集群中所有节点上的相同路径(例如、`/home/DBadmin/depot`)
- 可由DBadmin用户读取和写入
- 存储充足

  默认情况下、Vertica会将包含目录的文件系统空间的60%用于存储库存储。您可以在`create_db`命令中使用`-storage-size`参数来限制存储库的大小。请参见 "估算Eon模式数据库的Vertica集群规模" 有关Vertica规模估算一般准则的文章、或者咨询您的Vertica客户经理。

  如果不存在存储库路径、`admintools create_db`工具会尝试为您创建一个路径。

创建**Eon**内部数据库

要创建Eon内部数据库、请执行以下步骤：

1. 要创建数据库、请使用`admintools create_db`工具。

   以下列表简要说明了本示例中使用的参数。有关所有必需参数和可选参数的详细说明、请参见Vertica文档。

   - -x <在中创建的授权文件的路径/文件名 "创建授权文件以访问S3端点" >。

     成功创建后、授权详细信息将存储在数据库中。您可以删除此文件、以避免公开S3密钥。

   - -communal-storage-location <S3：//storagegrid bucketname>
   - -s <用于此数据库的Vertica节点的逗号分隔列表>
   - -d <要创建的数据库名称>
   - -p <要为此新数据库设置的密码>。例如、请参见以下命令示例：

```
admintools -t create_db -x sg_auth.conf --communal-storage
-location=s3://vertica --depot-path=/home/dbadmin/depot --shard
-count=6 -s vertica-vm1,vertica-vm2,vertica-vm3 -d vmart -p
'<password>'
```

根据数据库的节点数、创建新数据库需要几分钟的持续时间。首次创建数据库时、系统将提示您接受许可协议。

例如、请参见以下授权文件示例和`create db`命令：

```
[dbadmin@vertica-vm1 ~]$ cat sg_auth.conf
awsauth = MNVU4OYFAY2CPKVXVxxxx:03vuO4M4KmdfwffT8nqnBmnMVTr78Gu9wAN+xxxx
awsendpoint = s3.england.connectlab.io:10445
awsenablehttps = 1

[dbadmin@vertica-vm1 ~]$ admintools -t create_db -x sg_auth.conf
--communal-storage-location=s3://vertica --depot-path=/home/dbadmin/depot
--shard-count=6 -s vertica-vm1,vertica-vm2,vertica-vm3 -d vmart -p
'xxxxxxxx'
Default depot size in use
Distributing changes to cluster.
    Creating database vmart
    Starting bootstrap node v_vmart_node0007 (10.45.74.19)
    Starting nodes:
        v_vmart_node0007 (10.45.74.19)
    Starting Vertica on all nodes. Please wait, databases with a large
catalog may take a while to initialize.
    Node Status: v_vmart_node0007: (DOWN)
    Node Status: v_vmart_node0007: (DOWN)
    Node Status: v_vmart_node0007: (DOWN)
    Node Status: v_vmart_node0007: (UP)
    Creating database nodes
    Creating node v_vmart_node0008 (host 10.45.74.29)
    Creating node v_vmart_node0009 (host 10.45.74.39)
    Generating new configuration information
    Stopping single node db before adding additional nodes.
    Database shutdown complete
    Starting all nodes
Start hosts = ['10.45.74.19', '10.45.74.29', '10.45.74.39']
    Starting nodes:
        v_vmart_node0007 (10.45.74.19)
        v_vmart_node0008 (10.45.74.29)
        v_vmart_node0009 (10.45.74.39)
    Starting Vertica on all nodes. Please wait, databases with a large
```

```
catalog may take a while to initialize.
    Node Status: v_vmart_node0007: (DOWN) v_vmart_node0008: (DOWN)
v_vmart_node0009: (DOWN)
    Node Status: v_vmart_node0007: (DOWN) v_vmart_node0008: (DOWN)
v_vmart_node0009: (DOWN)
    Node Status: v_vmart_node0007: (DOWN) v_vmart_node0008: (DOWN)
v_vmart_node0009: (DOWN)
    Node Status: v_vmart_node0007: (DOWN) v_vmart_node0008: (DOWN)
v_vmart_node0009: (DOWN)
    Node Status: v_vmart_node0007: (UP) v_vmart_node0008: (UP)
v_vmart_node0009: (UP)
Creating depot locations for 3 nodes
Communal storage detected: rebalancing shards

Waiting for rebalance shards. We will wait for at most 36000 seconds.
Installing AWS package
    Success: package AWS installed
Installing ComplexTypes package
    Success: package ComplexTypes installed
Installing MachineLearning package
    Success: package MachineLearning installed
Installing ParquetExport package
    Success: package ParquetExport installed
Installing VFunctions package
    Success: package VFunctions installed
Installing approximate package
    Success: package approximate installed
Installing flextable package
    Success: package flextable installed
Installing kafka package
    Success: package kafka installed
Installing logsearch package
    Success: package logsearch installed
Installing place package
    Success: package place installed
Installing txtindex package
    Success: package txtindex installed
Installing voltagesecure package
    Success: package voltagesecure installed
Syncing catalog on vmart with 2000 attempts.
Database creation SQL tasks completed successfully. Database vmart created
successfully.
```

对象大小(字节)	存储分段/对象密钥完整路径
61	s 3 : //Vertica/051/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a07/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a07_0_0_0.dfs
145	s 3 : //Vertica/2c4/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a3d/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a3d_0dfdfd0.dfdf
146	s 3 : //Vertica/33c/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a1d/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a1d_0_dfdfd.df
40	s 3 : //Vertica/382/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021a31/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a31_0_0.dfs
145	s 3 : //Vertica/42f/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a21/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a21a21a21_0_0.dfs
34	s 3 : //Vertica/472/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021a25/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a25_0_0.dfs
41	s 3 : //Vertica/476/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021a2d/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a2d_0_0.dfs
61	s 3 : //Vertica/52a/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a5d/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a5d_0_dfdfd.df
131	s 3 : //Vertica/5d2/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a19/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a19_0_0.dfs

对象大小(字节)	存储分段/对象密钥完整路径
91	s 3 : //Vertica/5F7/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a11/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a11_0_0.dfs
118	s 3 : //Vertica/82d/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a15/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a15_0_0.dfs
115	s 3 : //Vertica/9a2/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a61/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a61_0_0.dfs
33	s 3 : //Vertica/ACD/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021a29-026d63ae9d4a33237bf0e2c2cf2a794a00a000021a29_0_0.dfs
133	s 3 : //Vertica/b98/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a4d/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a4d_0_dfdfd.df
38	s 3 : //Vertica/db3/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a49-026d63ae9d4a33237bf0e2c2cf2a794a00a000021a49_0_0.dfs
38	s 3 : //Vertica/EBA/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a599/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a59_0_0.dfs
21521920	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a000000002152/026d63ae9d4a33237bf0e2c2cf2a794a00a00002152.tar

对象大小(字节)	存储分段/对象密钥完整路径
6865408	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021602/026d63ae9d4a33237bf0e2c2cf2a794a00a00002162.tar
204217344	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021610/026d63ae9d4a33237bf0e2c2cf2a794a00a000021610.tar
16109056	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000217e0/026d63ae9d4a33237bf0e2c2cf2a794a00a0000217e0.tar
12853248	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000021800/026d63ae9d4a33237bf0e2c2cf2a794a00a00002180.tar
8937984	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000002187a/026d63ae9d4a33237bf0e2c2cf2a794a00a00002187a.tar
56260608	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000218b2/026d63ae9d4a33237bf0e2c2cf2a794a00a0000218b2.tar
53947904	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a0000000219ba/026d63ae9d4a33237bf0e2c2cf2a794a00a0000219ba.tar

对象大小(字节)	存储分段/对象密钥完整路径
44932608	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a00000000219de/026d63ae9d4a33237bf0e2c2cf2a794a00a0000219de.tar
256306688	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a00000000021a6e/026d63ae9d4a33237bf0e2c2cf2a794a00a000021a6e.tar
8062464	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a00000000021e34-026d63ae9d4a33237bf0e2c2cf2a794a00a000021e34.tar
20024832	s 3 : //Vertica/metadata/VMart/Libraries/026d63ae9d4a33237bf0e2c2cf2a794a00a00000000021e70/026d63ae9d4a33237bf0e2c2cf2a794a00a000021e70.tar
10444	s 3 : //Vertica/metadata/VMart/cluster_config.json
823266	s 3 : //Vertica/metadata/VMart/nodes/v_vmart_node0016/Catalog/859703b06a3456d95d0be28575a673/checkpoints/c13_chkpt_1.cat.gz
254	s 3 : //Vertica/metadata/VMart/nodes/v_vmart_node0016/Catalog/859703b06a3456d95d0be28575a673/checkpoints/c13/已完成
2958	s 3 : //Vertica/metadata/VMart/nodes/v_vmart_node0016/Catalog/859703b06a3456d95d0be28575a673/checkpoints/c2_chkpt_1.cat.gz

对象大小(字节)	存储分段/对象密钥完整路径
231	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/checkpoints/c2_completed
822521	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/checkpoints/c4_chkpt_1.cat.gz
231	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/checkpoints/c4_4/completed
746513	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/Txnlogs/txn_14_g14.cat
2596	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/Txnlogs/txn_3_g3.cat.gz
821065	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/Txnlogs/txn_4_g4.cat.gz
6440	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/Txnlogs/txn_5_g5.cat
8518	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/Txnlogs/txn_8_g8.cat
0	s 3 : //Vertica/metadata/VMart/nodes/v_vmart node0016/Catalog/859703b06a3456d95d0be2 8575a673/tiered_catalog.cat

对象大小(字节)	存储分段/对象密钥完整路径
822922	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0017/Catalog/859703b06a3456d95d0be28575a673/checkpoint/C14_7/chkpt_1.cat.gz
232	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0017/Catalog/859703b06a3456d95d0be28575a673/checkpoint/C14_7/completed
822930	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0017/Catalog/859703b06a3456d95d0be28575a673/Txnlogs/txn_14_g7.cat.gz
755033	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0017/Catalog/859703b06a3456d95d0be28575a673/Txnlogs/txn_15_g8.cat
0	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0017/Catalog/859703b06a3456d95d0be28575a673/tiered_catalog.cat
822922	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0018/Catalog/859703b06a3456d95d0be28575a673/checkpoint/C14_7/chkpt_1.cat.gz
232	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0018/Catalog/859703b06a3456d95d0be28575a673/checkpoint/C14_7/completed
822930	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0018/Catalog/859703b06a3456d95d0be28575a673/Txnlogs/txn_14_g7.cat.gz
755033	s 3 : //Vertica/metadata/VMart/nodes/v_vmartnode0018/Catalog/859703b06a3456d95d0be28575a673/Txnlogs/txn_15_g8.cat

对象大小(字节)	存储分段/对象密钥完整路径
0	s 3 ：//Vertica/metadata/VMart/nodes/v_vmart node0018/Catalog/859703b06a3456d95d0be2 8575a673/tiered_catalog.cat

禁用流限制

此操作步骤 基于适用于其他内部对象存储的Vertica指南、应适用于StorageGRID。

1. 创建数据库后、通过将`AWSStreamingConnectionPercentage`配置参数设置为`0`来禁用该参数。对于使用公共存储的Eon模式内部安装、不需要此设置。此配置参数用于控制Vertica用于流式读取的对象存储连接数。在云环境中、此设置有助于避免对象存储中的流式数据占用所有可用的文件句柄。它会使某些文件句柄可用于其他对象存储操作。由于内部对象存储的延迟较低、因此没有必要使用此选项。
2. 使用`vsql`语句更新参数值。此密码是您在"创建Eon内部数据库"中设置的数据库密码。例如、请参见以下示例输出：

```
[dbadmin@vertica-vm1 ~]$ vsql
Password:
Welcome to vsql, the Vertica Analytic Database interactive terminal.
Type:   \h or \? for help with vsql commands
        \g or terminate with semicolon to execute query
        \q to quit
dbadmin=> ALTER DATABASE DEFAULT SET PARAMETER
AWSStreamingConnectionPercentage = 0; ALTER DATABASE
dbadmin=> \q
```

验证返厂设置

已为读写操作启用Vertica数据库默认存储库设置(值= 1)。NetApp强烈建议保持启用这些仓库设置以提高性能。

```
vsql -c 'show current all;' | grep -i UseDepot
DATABASE | UseDepotForReads | 1
DATABASE | UseDepotForWrites | 1
```

加载示例数据(可选)

如果此数据库用于测试并将被删除、您可以将样本数据加载到此数据库以进行测试。Vertica随附了示例数据集VMart、位于每个Vertica/node/opt/Vertica/Examples/VMart_Schema/``下。有关此示例数据集的详细信息、请参见 "此处"。

按照以下步骤加载示例数据：

1. 以DBadmin身份登录到Vertica节点之一：cd /opt/vertica/examples/VMart_Schemas/

2. 将示例数据加载到数据库中、并在子步骤c和d中出现提示时输入数据库密码：
   a. `cd /opt/vertica/examples/VMart_Schema`
   b. 。`/vmart根`
   c. `vsql< vmart定义架构.sql`
   d. `vsql < vmart load_data.sql`
3. 有多个预定义的SQL查询、您可以运行其中一些查询、以确认测试数据已成功加载到数据库中。例如：`vsql < vmart queries1.sql`

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- "NetApp StorageGRID 11.7产品文档"
- "StorageGRID 数据表"
- "Vertica 10.1产品文档"

版本历史记录

version	Date	文档版本历史记录
版本 1.0	2021年9月	初始版本。

作者：郑安杰

使用ELK堆栈进行StorageGRID 日志分析

作者：郑安杰

通过StorageGRID系统日志转发功能、您可以配置外部系统日志服务器来收集和分析StorageGRID日志消息。ELK (Elasticsearch、Logstash、Kibana)已成为最受欢迎的日志分析解决方案之一。观看以查看示例的 "使用ELK视频进行StorageGRID 日志分析" 模拟配置、以及如何使用它来识别失败的S3请求并对其进行故障排除。StorageGRID均衡器支持将负载平衡器端点访问日志导出到外部系统日志服务器。观看此视频 "YouTube视频"、了解有关此新功能的更多信息。本文提供了Logstash配置、Kibana查询、图表和信息板的示例文件、可帮助您快速开始StorageGRID 日志管理和分析。

要求

- StorageGRID 11.6.0.2或更高版本
- ELK (Elasticsearch、Logstash和Kibana)已安装并运行7.1x或更高版本

示例文件

- "下载Logstash 7.x示例文件包" +* MD5 checksum* 148c23d0021d9a4bb4a6c0287464deab +* SHA256 checksum* f5ec9e2e3f842d5a7861566b167a561b4373038b4e7bb3b3d522adf2d6

- "下载Logstash 8.x示例文件包" +* MD5 checksum* e11ba3a662f87c3ef363d0fe06835 +* SHA256 checksum* 5c670755742cfdf5aa723a596b087e0153a65bcaef3934afdb682f61cd278d
- "下载适用于StorageGRID 11.9的Logstash 8.x示例文件包"+* md5校验和* 41272857c4a54600f95995f6ed74800d +* SHA256校验和* 77048ee86610527 19990851e1ad960d4902fe537a6e135e8600177188da677c9

假设

读者熟悉StorageGRID 和ELK的术语和操作。

说明

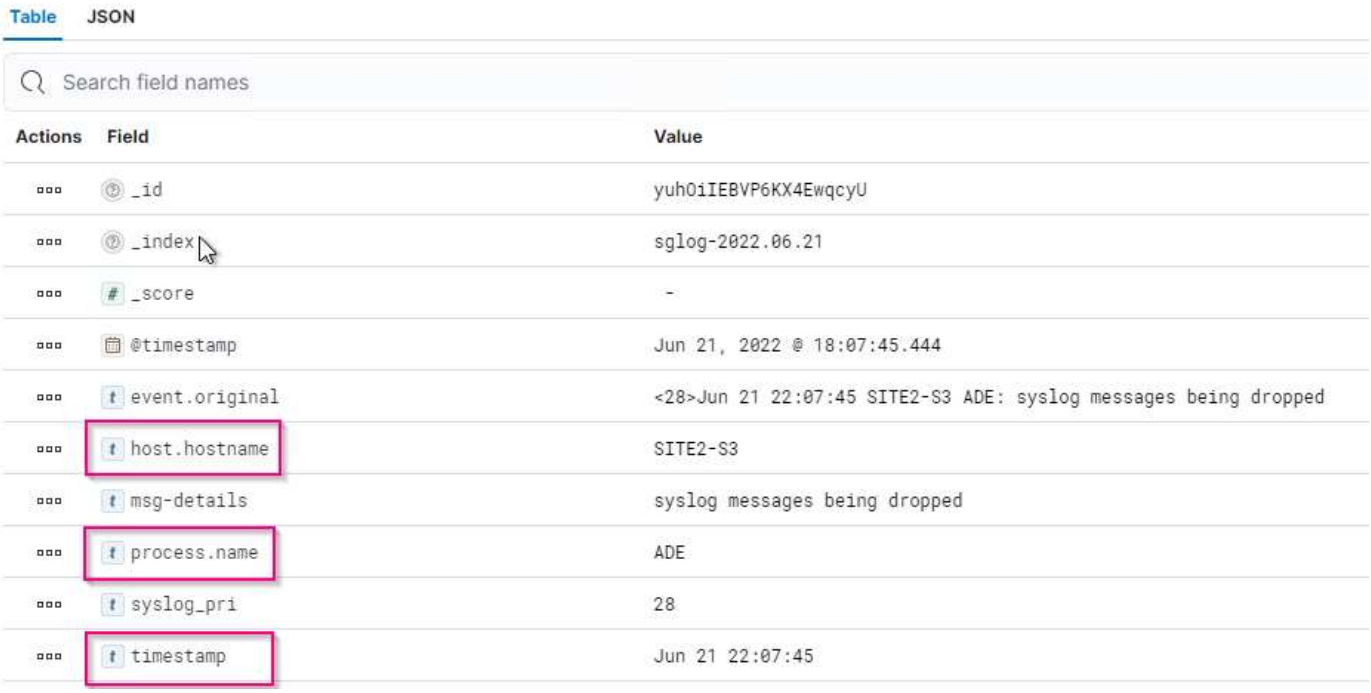
由于Grok模式定义的名称不同、因此提供了两个示例版本。+例如、Logstash配置文件中的SYSLOGBASE格式根据安装的Logstash版本定义不同的字段名称。

```
match => {"message" => '<%{POSINT:syslog_pri}>%{SYSLOGBASE}
%{GREEDYDATA:msg-details}'}
```

- Logstash 7.17示例*

Field	Value
t _id	7C1MaYEBRH8UbfKnIls8
t _index	sgrid2-2022.06.15
# _score	-
t _type	_doc
@timestamp	Jun 15, 2022 @ 17:36:46.038
t host	grid2-site2-s1
t logsource	SITE2-S1
t msg-details	Reloading syslog service
t pid	628
t program	update-sysl
t syslog_pri	37
t timestamp	Jun 15 21:36:46

- Logstash 8.23示例*



- 步骤 *

1. 根据您安装的ELK版本解压缩提供的示例。此示例文件夹包含两个Logstash配置示例：* sglog-2-file.conf：此配置文件会将**StorageGRID** 日志消息输出到**Logstash**上的文件中、而不会进行数据转换。您可以使用此命令来确认**Logstash**正在接收**StorageGRID** 消息、或者帮助了解**StorageGRID** 日志模式。+ sglog-2-es.conf：*此配置文件使用各种模式和筛选器转换StorageGRID 日志消息。其中包括示例drop语句、这些语句根据模式或筛选器丢弃消息。输出将发送到Elasticsearch以编制索引。+根据文件中的说明自定义选定的配置文件。
2. 测试自定义的配置文件：

```
/usr/share/logstash/bin/logstash --config.test_and_exit -f <config-file-path/file>
```

   如果返回的最后一行与以下行类似、则此配置文件没有语法错误：

```
[LogStash::Runner] runner - Using config.test_and_exit mode. Config Validation Result: OK. Exiting Logstash
```

3. 将自定义的conf文件复制到Logstash服务器的配置：/etc/logstash/conf.d+如果尚未在/etc/logstash/logstash.yml中启用config.reload.automatic、请重新启动Logstash服务。否则、请等待配置重新加载间隔过。

```
 grep reload /etc/logstash/logstash.yml
# Periodically check if the configuration has changed and reload the
pipeline
config.reload.automatic: true
config.reload.interval: 5s
```

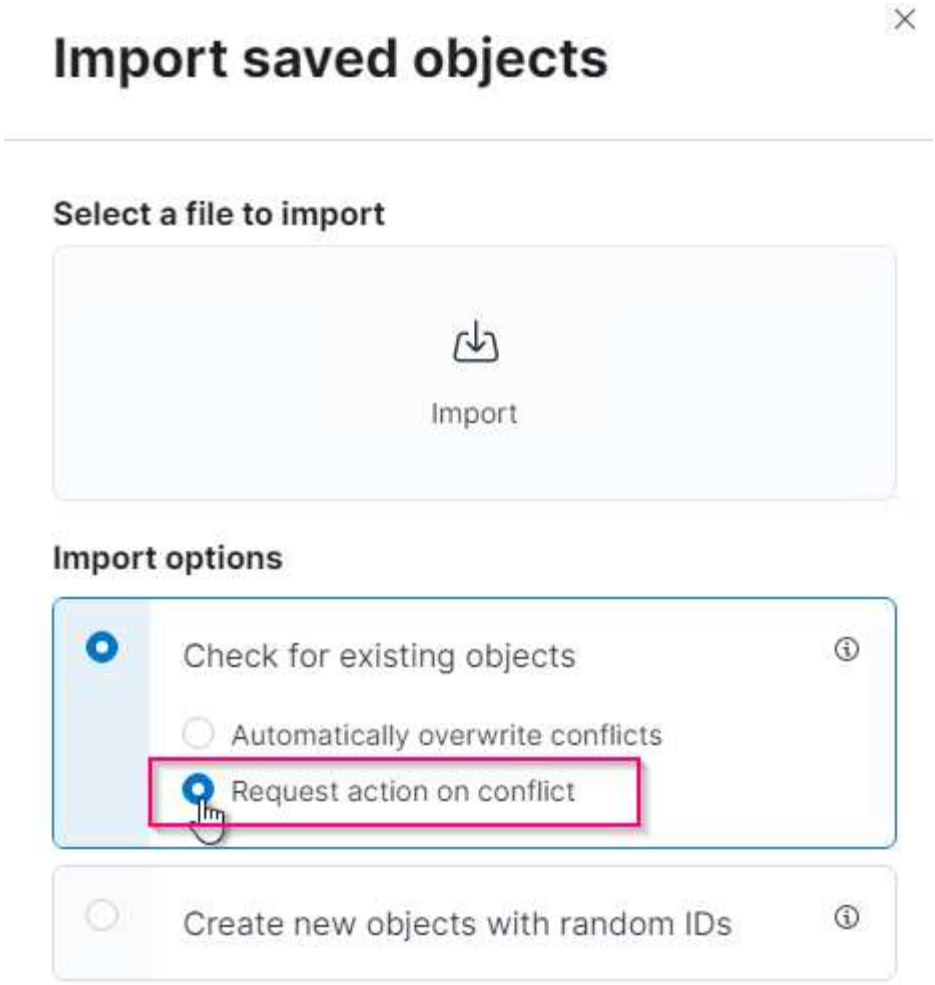
4. 检查/var/log/logstash/logstash-plain.log并确认在使用新配置文件启动Logstash时没有错误。
5. 确认TCP端口已启动并正在侦听。+在此示例中、使用TCP端口5000。

```
netstat -ntpa | grep 5000
tcp6       0      0 :::5000                 :::*
LISTEN      25744/java
```

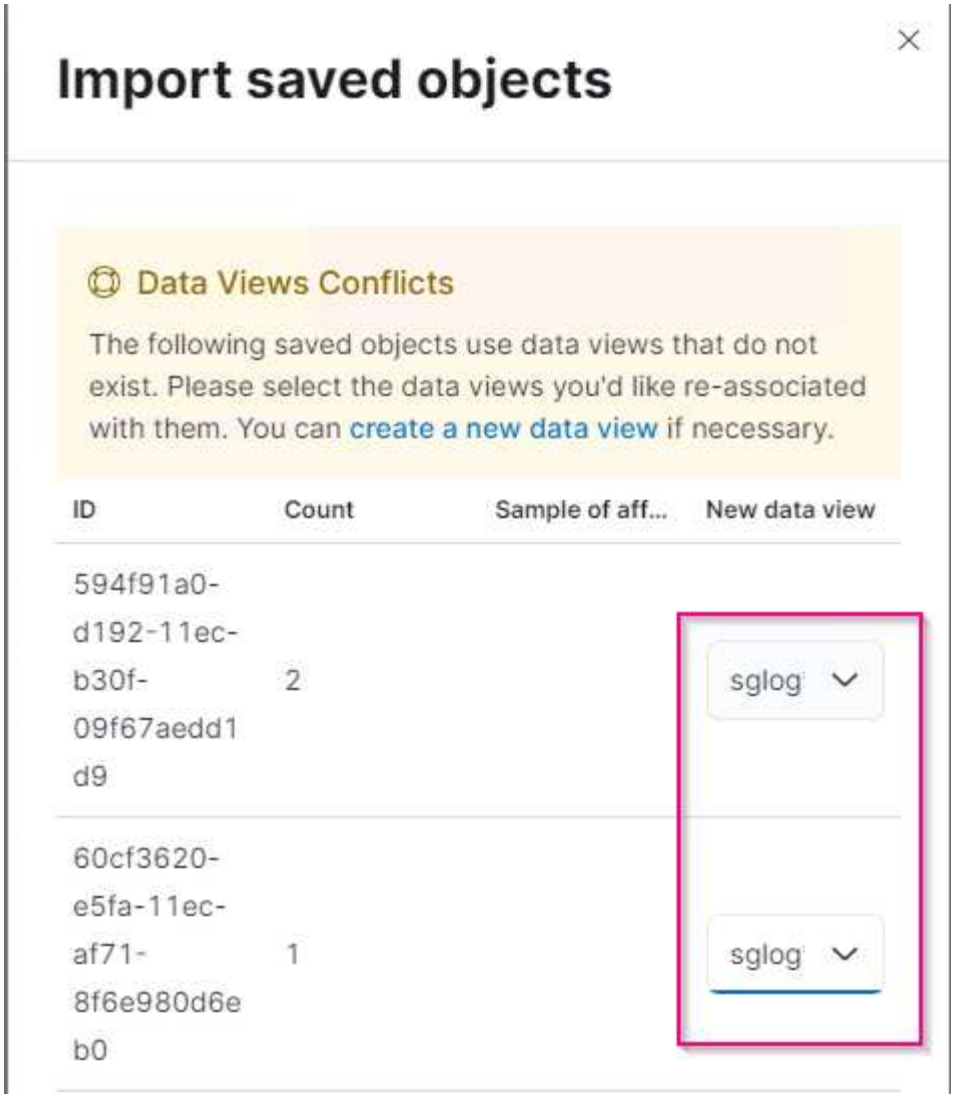
6. 在StorageGRID 管理器图形用户界面中、配置外部系统日志服务器以向Logstash发送日志消息。有关详细信息、请参见 "演示视频" 。
7. 您需要在Logstash服务器上配置或禁用防火墙、以允许StorageGRID 节点连接到定义的TCP端口。
8. 在Kibana GUI中、选择Management → Dev Tools。在控制台页面上、运行此get命令以确认已在Elasticsearch上创建新索引。

```
GET /_cat/indices/*?v=true&s=index
```

9. 在Kibana GUI中、创建索引模式(ELK 7.x)或数据视图(ELK 8.x)。
10. 在Kibana GUI的顶部中间的搜索框中输入"saved objects"。+在已保存对象页面上、选择导入。在导入选项下、选择"请求冲突操作"



导入ELK <version>-query-chart -sample.ndjson。+当系统提示您解决冲突时、请选择您在第8步中创建的索引模式或数据视图。



导入了以下基巴纳对象：* Query** audy-msg-s3rq-orln * bycast log S3相关消息* loglevel WARNING或 以上+失败的安全事件**+** NGINS-GW端点访问日志(仅在elk8-sSample—for-sg119.zip中提供)* Chart** S3请求计数(基于bycast.log * HTTP状态)按审核类型分类**)+ S3***信息板**+**审核请求的平均响应时间**)**。

现在、您可以使用Kibana执行StorageGRID 日志分析了。

其他资源

- "系统日志101"
- "什么是ELK堆栈"
- "Grok模式列表"
- "Logstash入门指南：Grok"
- "Logstash实用指南：系统日志深度剖析"
- "Kibana指南—浏览文档"
- "StorageGRID 审核日志消息参考"

使用Prometheus和Grafana延长指标保留期限

作者：*Aron Klein*

本技术报告提供了配置NetApp StorageGRID与外部 Prometheus 和 Grafana 服务的详细说明。

简介

StorageGRID 使用Prometheus存储指标、并通过内置的Grafana信息板对这些指标进行可视化。通过配置客户端访问证书并为指定客户端启用Prometheus访问、可以从StorageGRID 安全地访问Prometheus指标。目前、此指标数据的保留受管理节点存储容量的限制。为了获得更长的持续时间并能够创建这些指标的自定义可视化效果、我们将部署一个新的Prometheus和Grafana服务器、配置我们的新服务器以从StorageGRID实例中擦除这些指标、并使用对我们重要的指标构建一个信息板。您可以获取有关在中收集的Prometheus指标的详细信息 "StorageGRID 文档"。

联合**Prometheus**

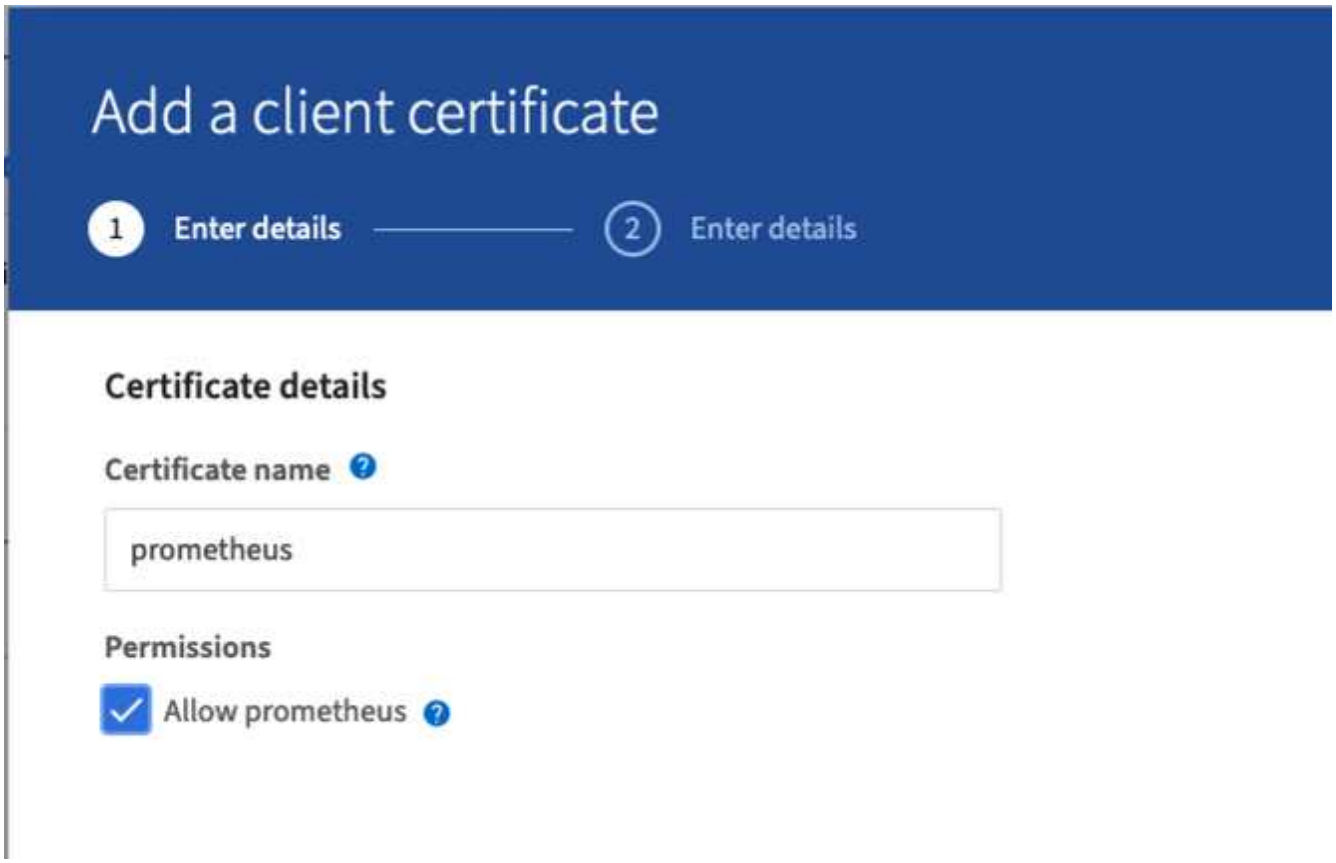
实验室详细信息

在本示例中、我将使用StorageGRID 11.6节点的所有虚拟机以及Debian 11服务器。StorageGRID 管理界面配置了一个公共信任的CA证书。本示例将不会介绍StorageGRID 系统或Debian Linux安装的安装和配置过程。您可以使用Prometheus和Grafana支持的任何Linux模式。Prometheus和Grafana都可以安装为Docker容器、从源代码构建或预编译的二进制文件。在此示例中、我将直接在同一Debian服务器上安装Prometheus和Grafana二进制文件。从下载并按照基本安装说明进行操作 https://prometheus.io 和 https://grafana.com/grafana/ 。

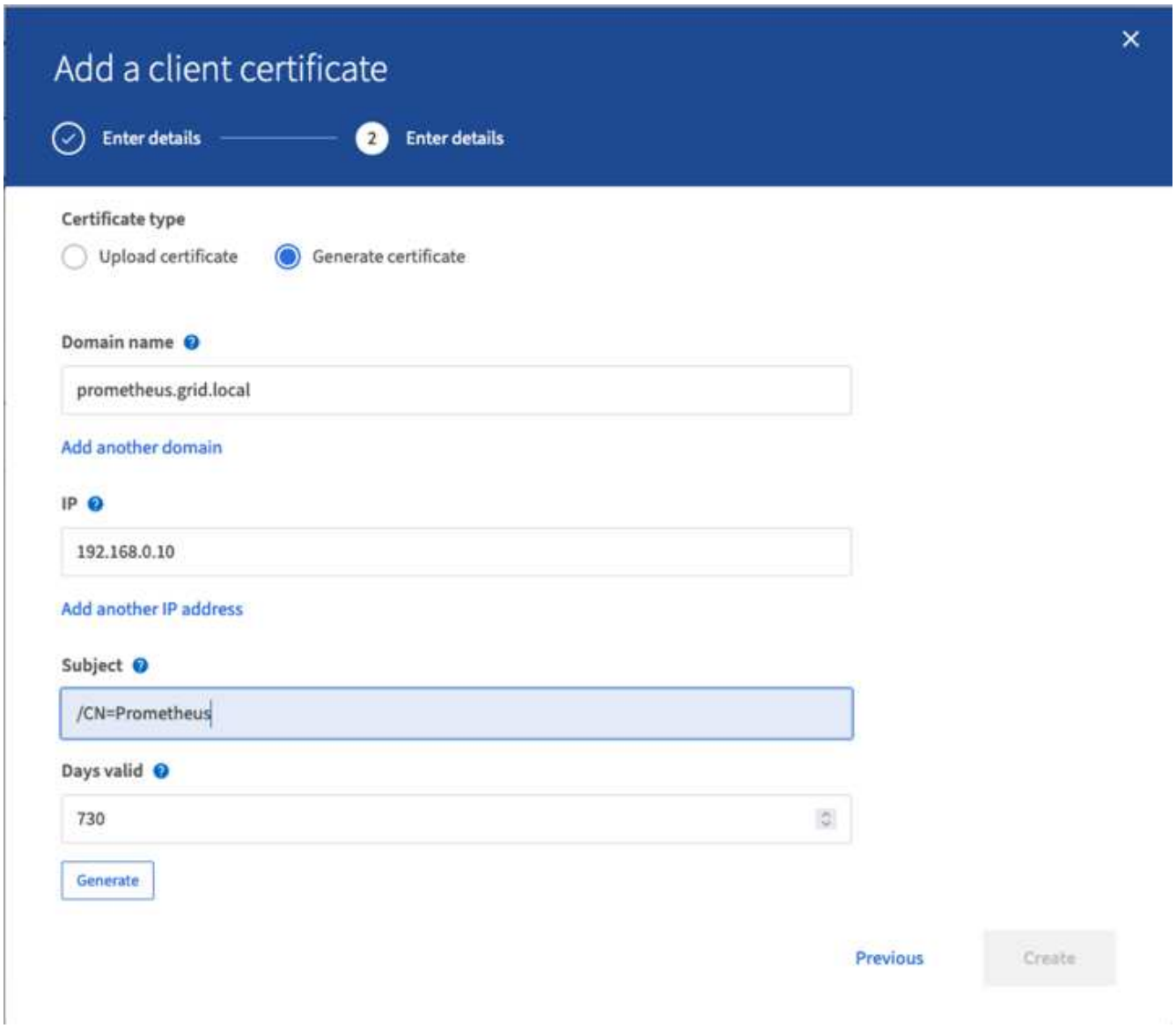
为**Prometheus**客户端访问配置**StorageGRID**

要访问StorageGRID Stored Prometheus指标、您必须生成或上传具有专用密钥的客户端证书、并为客户端启用权限。StorageGRID 管理接口必须具有SSL证书。此证书必须由Prometheus服务器信任、或者由可信CA信任、如果是自签名证书、则此证书必须手动受信任。要了解更多信息、请访问 "StorageGRID 文档"。

1. 在StorageGRID 管理界面中、选择左下方的"configuration"、然后在第二列的"Security"下单击Certificates。
2. 在"证书"页面上、选择"客户端"选项卡、然后单击"添加"按钮。
3. 提供要授予访问权限的客户端的名称并使用此证书。单击"允许用户"前面的"权限"下的框、然后单击"继续"按钮。

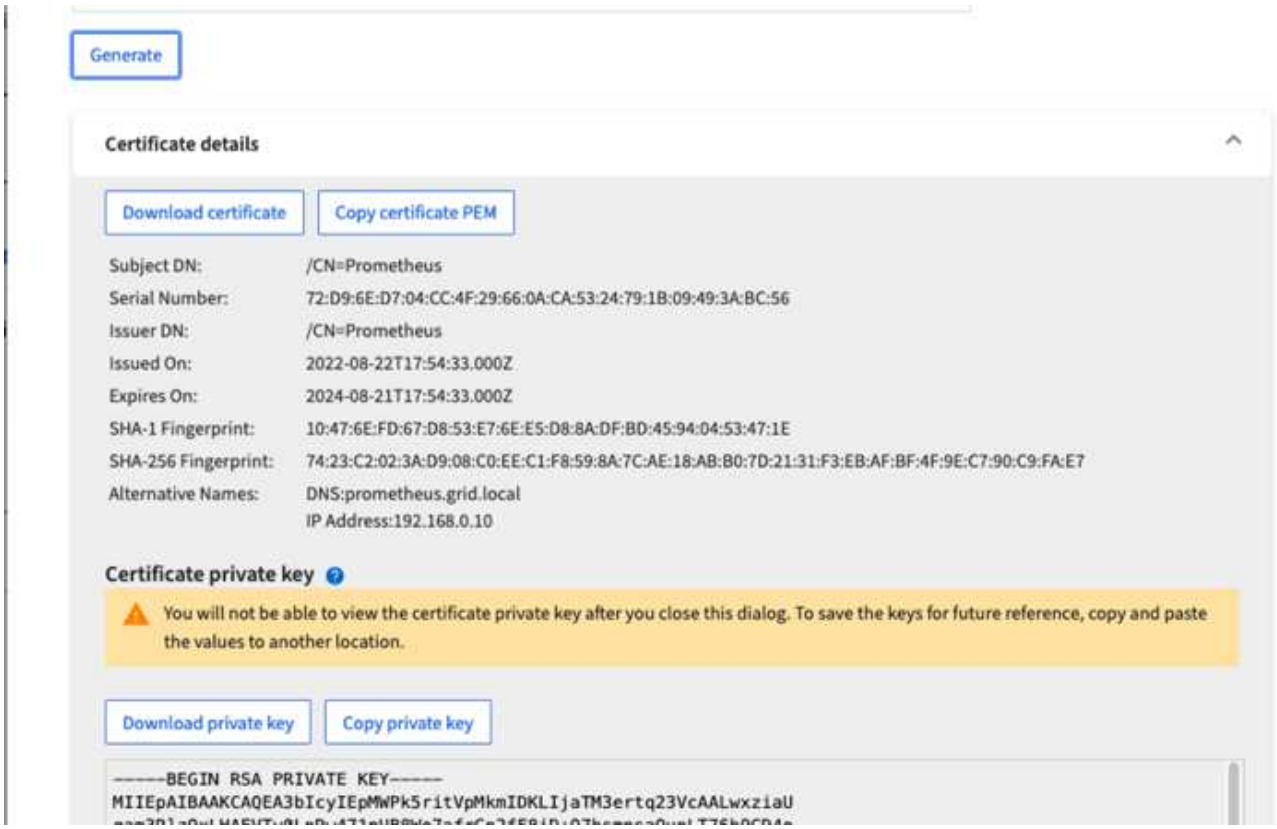


4. 如果您拥有CA签名的证书、则可以选择"上传证书"单选按钮、但在我们的情况下、我们将通过选择"生成证书"单选按钮让StorageGRID 生成客户端证书。此时将显示要填写的必填字段。输入客户端服务器的FQDN、服务器的IP、主题和有效天数。然后单击"生成"按钮。



> Be mindful of the certificate days valid entry as you will need to renew this certificate in both StorageGRID and the Prometheus server before it expires to maintain uninterrupted collection.

1. 下载证书对等文件和专用密钥对等文件。



> This is the only time you can download the private key, so make sure you do not skip this step.

准备Linux服务器以安装Prometheus

在安装Prometheus之前、我希望为我的环境做好准备、让Prometheus用户、目录结构做好准备、并为指标存储位置配置容量。

1. 创建Prometheus用户。

```
sudo useradd -M -r -s /bin/false Prometheus
```

2. 为Prometheus、客户端证书和指标数据创建目录。

```
sudo mkdir /etc/Prometheus /etc/Prometheus/cert /var/lib/Prometheus
```

3. 我使用ext4文件系统格式化了用于指标保留的磁盘。

```
mkfs -t ext4 /dev/sdb
```

4. 然后、我将文件系统挂载到Prometheus指标目录。

```
sudo mount -t auto /dev/sdb /var/lib/prometheus/
```

5. 获取用于指标数据的磁盘的UUID。

```
sudo ls -al /dev/disk/by-uuid/
   lrwxrwxrwx 1 root root   9 Aug 18 17:02 9af2c5a3-bfc2-4ec1-85d9-
ebab850bb4a1 -> ../../sdb
```

6. 在/etc/fstab/中添加一个条目、使挂载在重新启动后仍会使用/dev/sdb的uuid。

```
/etc/fstab
UUID=9af2c5a3-bfc2-4ec1-85d9-ebab850bb4a1 /var/lib/prometheus   ext4
defaults    0   0
```

安装和配置**Prometheus**

现在、服务器已准备就绪、我可以开始安装Prometheus并配置此服务。

1. 提取Prometheus安装包

```
tar xzf prometheus-2.38.0.linux-amd64.tar.gz
```

2. 将二进制文件复制到/usr/local/bin、并将所有权更改为先前创建的Prometheus用户

```
sudo cp prometheus-2.38.0.linux-amd64/{prometheus,promtool}
/usr/local/bin
sudo chown prometheus:prometheus /usr/local/bin/{prometheus,promtool}
```

3. 将控制台和库复制到/etc/Prometheus

```
sudo cp -r prometheus-2.38.0.linux-amd64/{consoles,console_libraries}
/etc/prometheus/
```

4. 将先前从StorageGRID 下载的客户端证书和专用密钥对等文件复制到/etc/Prometheus/Certs
5. 创建Prometheus配置YAML文件

```
sudo nano /etc/prometheus/prometheus.yml
```

6. 插入以下配置。作业名称可以是您所需的任何名称。将"-targets"更改为管理节点的FQDN、如果更改了证书名称和专用密钥文件名、请更新tls_config部分以使其匹配。然后保存文件。如果您的网格管理界面正在使用自签名证书、请下载此证书并将其与具有唯一名称的客户端证书一起放置、然后在tls_config部分中添加ca_file：/etc/Prometheus/Cert/UIcert.pem

   a. 在此示例中、我将收集以alertmanager、Cassandra、node和StorageGRID 开头的所有指标。您可以在中查看有关Prometheus指标的详细信息 "StorageGRID 文档"。

```
# my global config
global:
  scrape_interval: 60s # Set the scrape interval to every 15 seconds.
Default is every 1 minute.

scrape_configs:
  - job_name: 'StorageGRID'
    honor_labels: true
    scheme: https
    metrics_path: /federate
    scrape_interval: 60s
    scrape_timeout: 30s
    tls_config:
      cert_file: /etc/prometheus/cert/certificate.pem
      key_file: /etc/prometheus/cert/private_key.pem
    params:
      match[]:
        -
'{__name__=~"alertmanager_.*|cassandra_.*|node_.*|storagegrid_.*"}'
    static_configs:
    - targets: ['sgdemo-rtp.netapp.com:9091']
```

如果网格管理界面使用的是自签名证书、请下载此证书并将其与具有唯一名称的客户端证书一起放置。在tls_config部分中、将证书添加到客户端证书和专用密钥行上方

```
        ca_file: /etc/prometheus/cert/UIcert.pem
```

1. 将/etc/Prometheus和/var/lib/Prometheus中所有文件和目录的所有权更改为Prometheus用户

```
sudo chown -R prometheus:prometheus /etc/prometheus/
sudo chown -R prometheus:prometheus /var/lib/prometheus/
```

2. 在/etc/systemd/system中创建一个Prometheus服务文件

```
sudo nano /etc/systemd/system/prometheus.service
```

3. 插入以下行、请注意#-storage.tsdb.retention.time=1y#、它会将指标数据的保留期限设置为1年。或者、您也可以使用#-storage.tsdb.retention.size=300GiB#根据存储限制确定保留期限。这是设置指标保留的唯一位置。

```
[Unit]
Description=Prometheus Time Series Collection and Processing Server
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
        --config.file /etc/prometheus/prometheus.yml \
        --storage.tsdb.path /var/lib/prometheus/ \
        --storage.tsdb.retention.time=1y \
        --web.console.templates=/etc/prometheus/consoles \
        --web.console.libraries=/etc/prometheus/console_libraries

[Install]
WantedBy=multi-user.target
```

4. 重新加载systemd服务以注册新的Prometheus服务。然后启动并启用Prometheus服务。

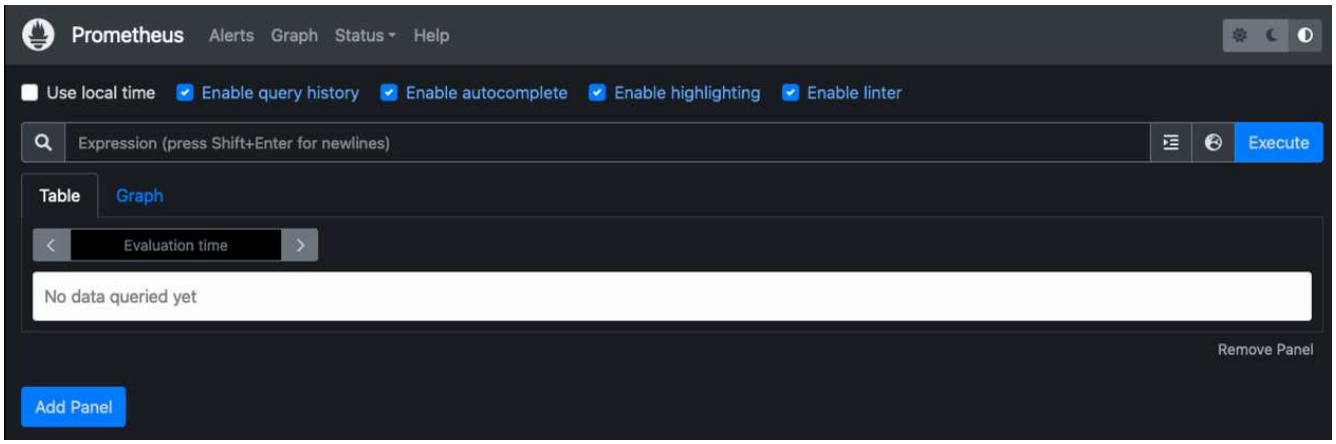
```
sudo systemctl daemon-reload
sudo systemctl start prometheus
sudo systemctl enable prometheus
```

5. 检查服务是否运行正常

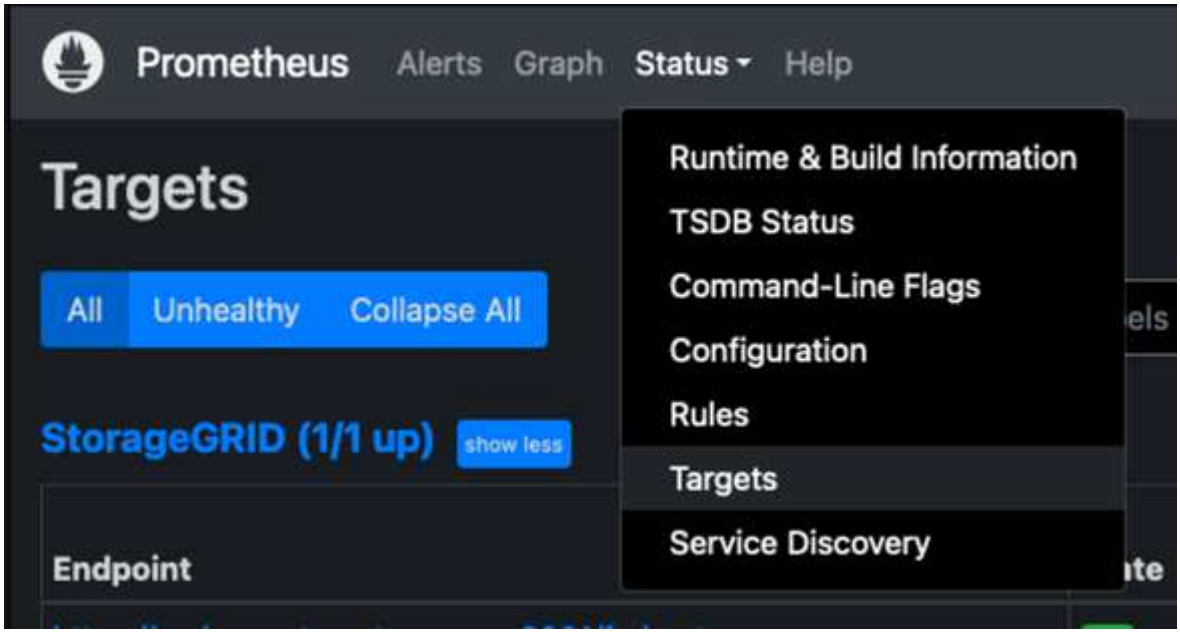
```
sudo systemctl status prometheus
```

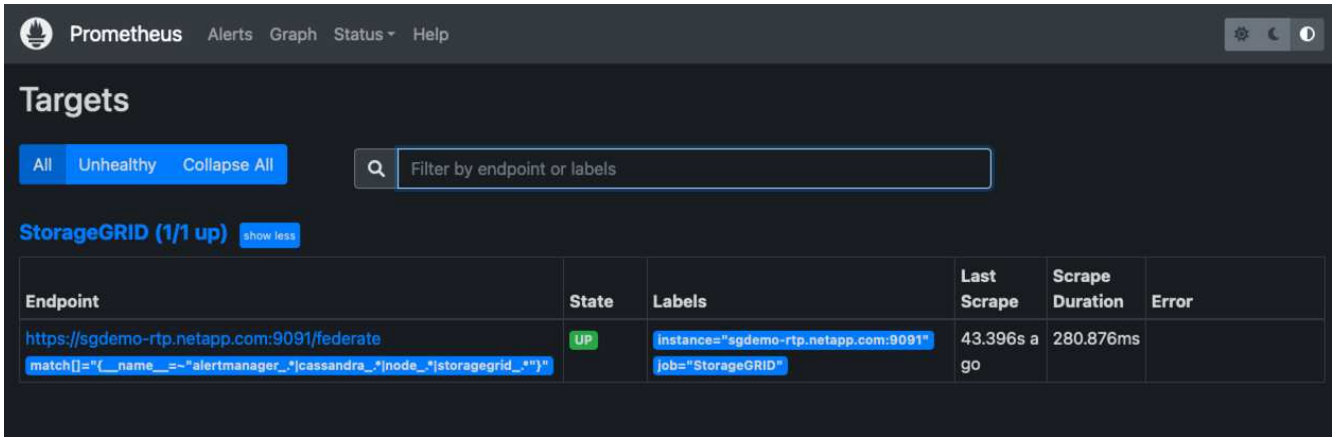
```
● prometheus.service - Prometheus Time Series Collection and Processing
Server
     Loaded: loaded (/etc/systemd/system/prometheus.service; enabled;
vendor preset: enabled)
     Active: active (running) since Mon 2022-08-22 15:14:24 EDT; 2s ago
   Main PID: 6498 (prometheus)
      Tasks: 13 (limit: 28818)
     Memory: 107.7M
        CPU: 1.143s
     CGroup: /system.slice/prometheus.service
             └─6498 /usr/local/bin/prometheus --config.file
/etc/prometheus/prometheus.yml --storage.tsdb.path /var/lib/prometheus/
--web.console.templates=/etc/prometheus/consoles --web.con>

Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.510Z caller=head.go:544 level=info component=tsdb
msg="Replaying WAL, this may take a while"
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.816Z caller=head.go:615 level=info component=tsdb msg="WAL
segment loaded" segment=0 maxSegment=1
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.816Z caller=head.go:615 level=info component=tsdb msg="WAL
segment loaded" segment=1 maxSegment=1
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.816Z caller=head.go:621 level=info component=tsdb msg="WAL
replay completed" checkpoint_replay_duration=55.57µs wal_rep>
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.831Z caller=main.go:997 level=info fs_type=EXT4_SUPER_MAGIC
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.831Z caller=main.go:1000 level=info msg="TSDB started"
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.831Z caller=main.go:1181 level=info msg="Loading
configuration file" filename=/etc/prometheus/prometheus.yml
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.832Z caller=main.go:1218 level=info msg="Completed loading
of configuration file" filename=/etc/prometheus/prometheus.y>
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.832Z caller=main.go:961 level=info msg="Server is ready to
receive web requests."
Aug 22 15:14:24 aj-deb-prom01 prometheus[6498]: ts=2022-08-
22T19:14:24.832Z caller=manager.go:941 level=info component="rule
manager" msg="Starting rule manager..."
```

6. 现在、您应该能够浏览到Prometheus服务器的UI http://Prometheus-server:9090 并查看UI

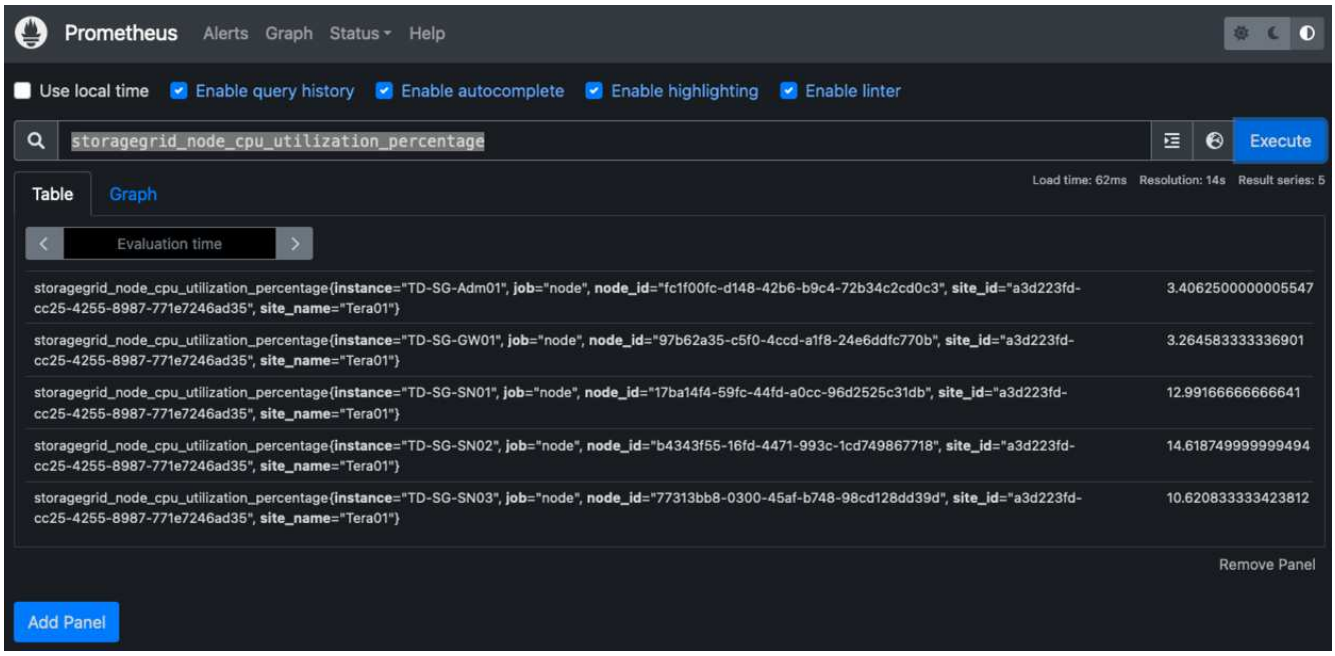


7. 在"Status" Targets下、您可以看到我们在Prometheus.yml中配置的StorageGRID 端点的状态





8. 在图形页面上、您可以执行测试查询并验证数据是否已成功擦除了。例如、在查询栏中输入"storagegRid_node_cpu_utilization_percentage "、然后单击执行按钮。



安装和配置Grafana

在Prometheus安装完毕并正常工作之后、我们可以继续安装Grafana并配置信息板

Grafana安装

1. 安装最新的企业版Grafana

```
sudo apt-get install -y apt-transport-https
sudo apt-get install -y software-properties-common wget
sudo wget -q -O /usr/share/keyrings/grafana.key
https://packages.grafana.com/gpg.key
```

2. 为稳定版本添加此存储库：

```
echo "deb [signed-by=/usr/share/keyrings/grafana.key]
https://packages.grafana.com/enterprise/deb stable main" | sudo tee -a
/etc/apt/sources.list.d/grafana.list
```

3. 添加存储库后。

```
sudo apt-get update
sudo apt-get install grafana-enterprise
```

4. 重新加载systemd服务以注册新的grafana服务。然后启动并启用Grafana服务。

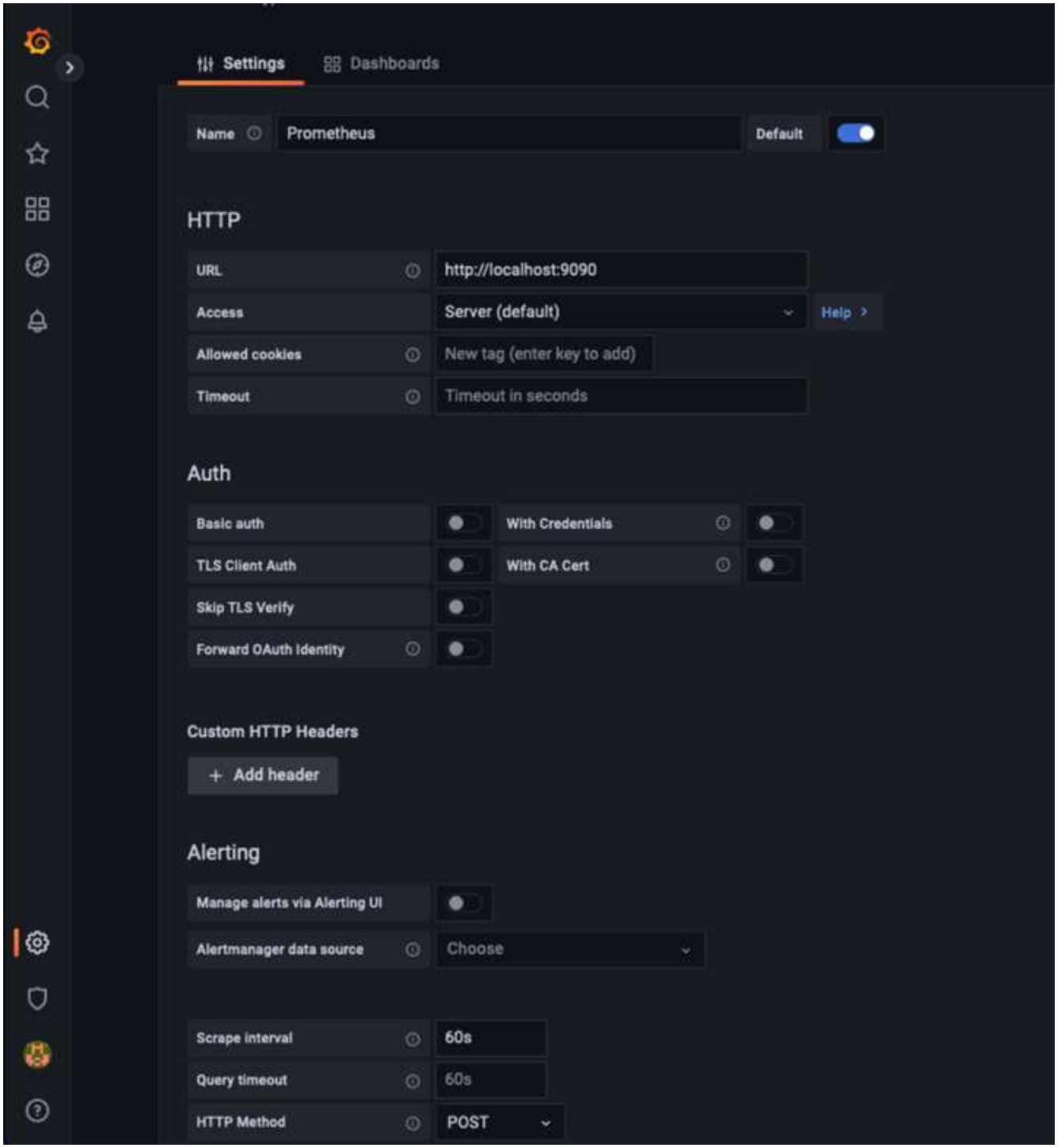
```
sudo systemctl daemon-reload
sudo systemctl start grafana-server
sudo systemctl enable grafana-server.service
```

5. 现在、Grafana已安装并正在运行。打开浏览器访问HTTP：//Prometheus-server：3000时、您将看到Grafana登录页面。
6. 默认登录凭据为admin/admin、您应根据提示设置新密码。

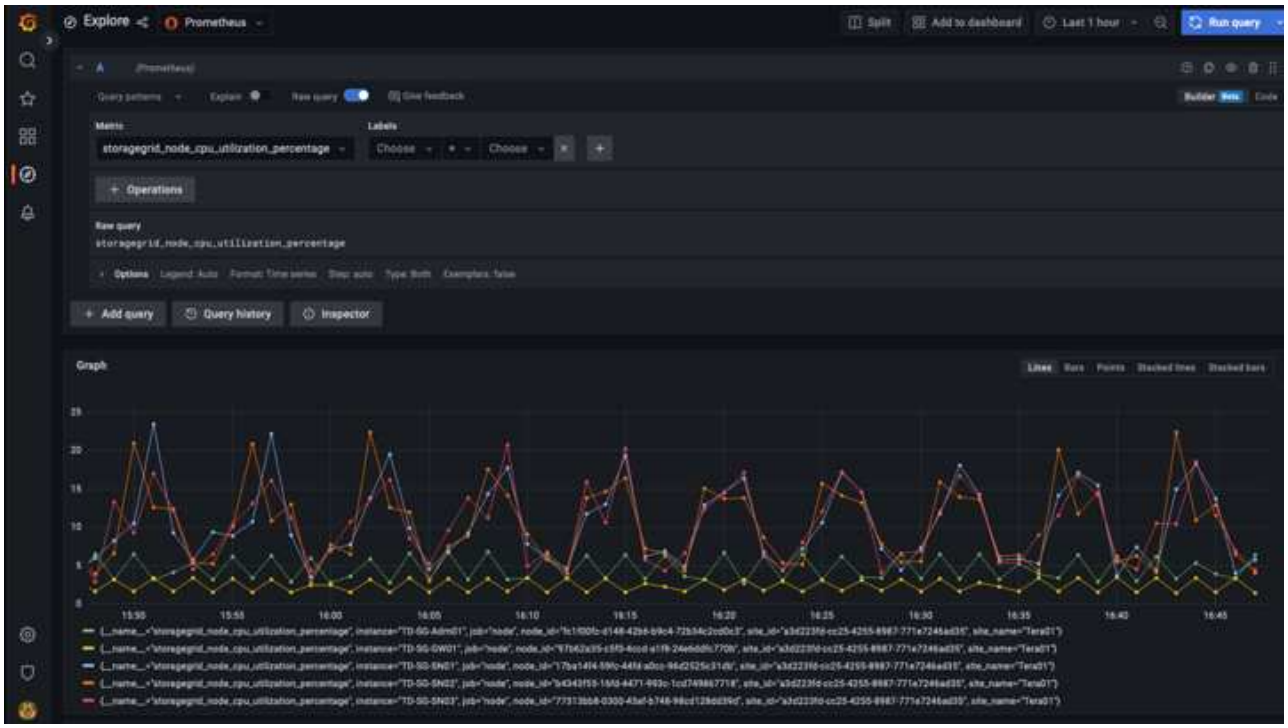
为**StorageGRID** 创建**Grafana**信息板

在Grafana和Prometheus安装并运行的情况下、现在是时候通过创建数据源和构建信息板来连接这两者了

1. 在左侧窗格中、展开"配置"并选择"数据源"、然后单击"添加数据源"按钮
2. Prometheus将是可供选择的顶级数据源之一。如果不是、请使用搜索栏找到"Prometheus"
3. 通过输入Prometheus实例的URL以及与Prometheus间隔匹配的擦除间隔来配置Prometheus源。我还禁用了警报部分、因为我未在Prometheus上配置警报管理器。



4. 输入所需设置后、向下滚动到底部、然后单击"Save & test"(保存并测试)
5. 配置测试成功后、单击Explore按钮。
   a. 在"浏览"窗口中、您可以使用我们使用"storagegrid node_cpu_utilization_percentage "测试的相同指标、然后单击"运行查询"按钮

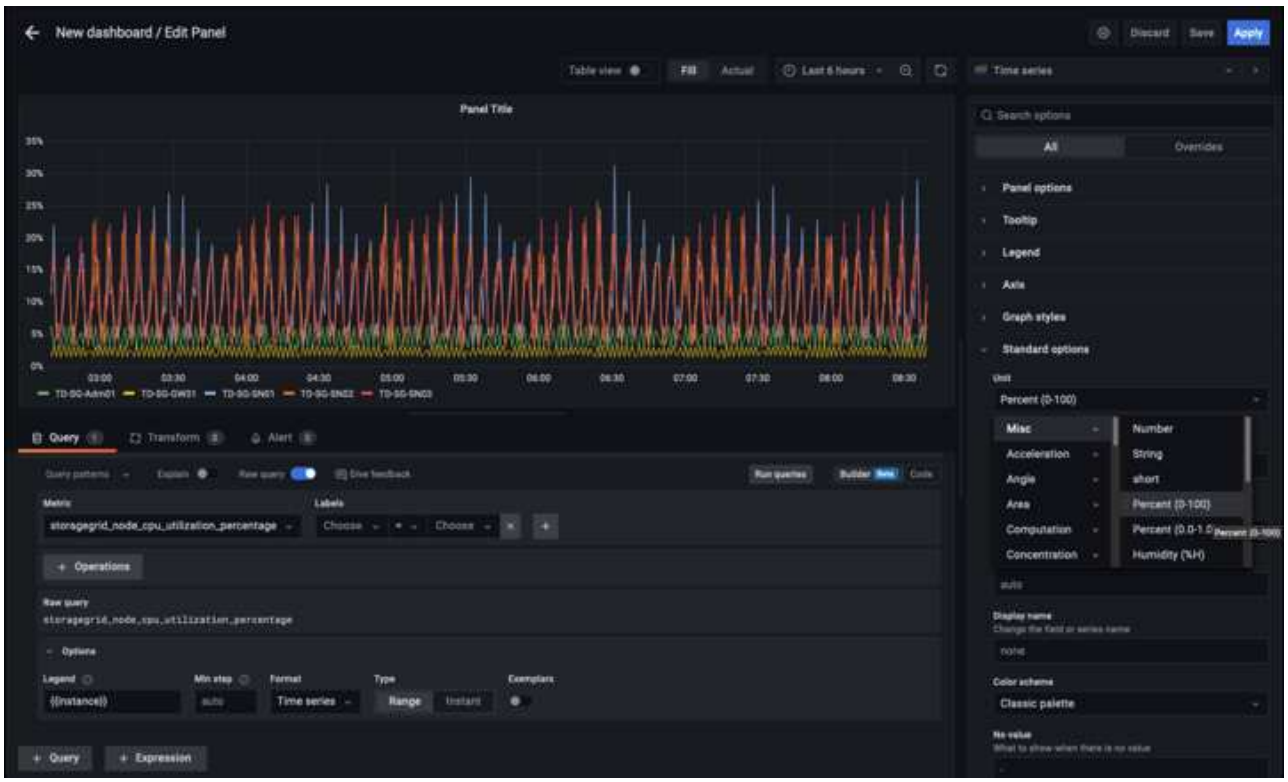


6. 现在、我们已配置数据源、可以创建一个信息板。

   a. 在左侧窗格中、展开Dashboards、然后选择"+" new Dashboard"

   b. 选择"添加新面板"

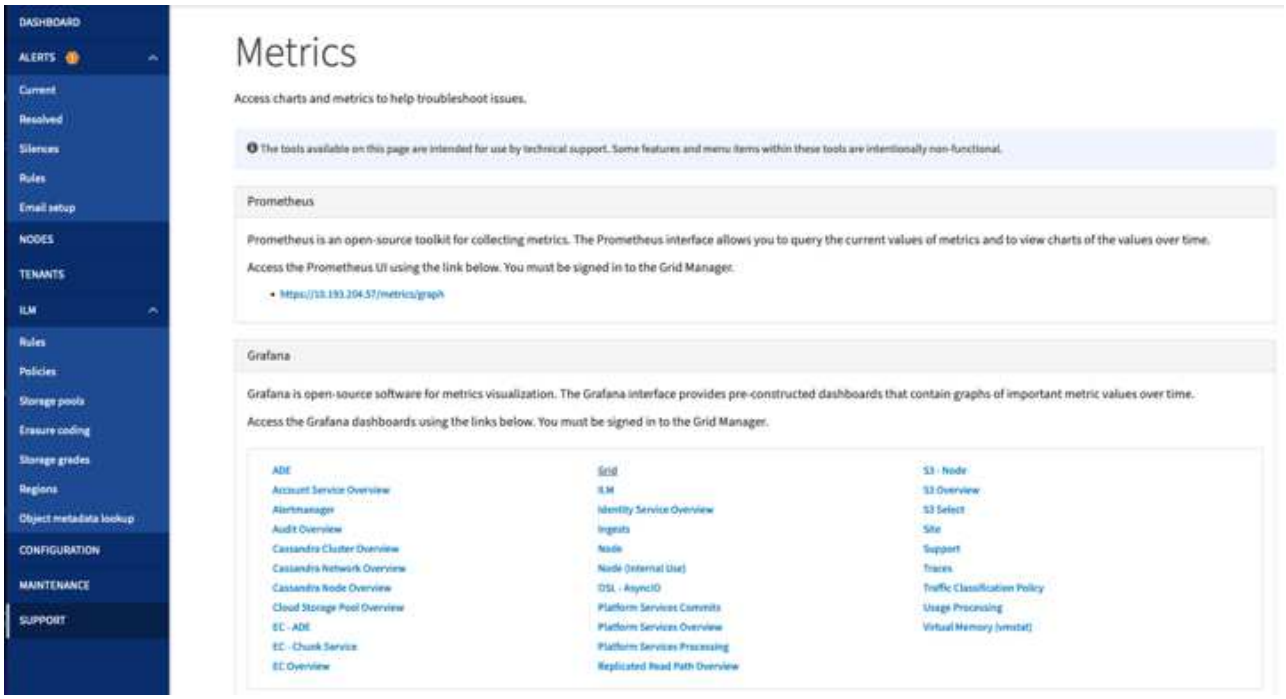
   c. 通过选择指标来配置新面板、我将再次使用"storagegrid node_cpu_utilization_percentage "、输入面板标题、展开底部的"选项"、并将图例更改为自定义、然后输入" ｛ {instance} ｝ "来定义节点名称、并在右侧窗格的"标准选项"下将"单元"设置为"Misc 100/percent (0%)"。然后单击"应用"将面板保存到信息板。



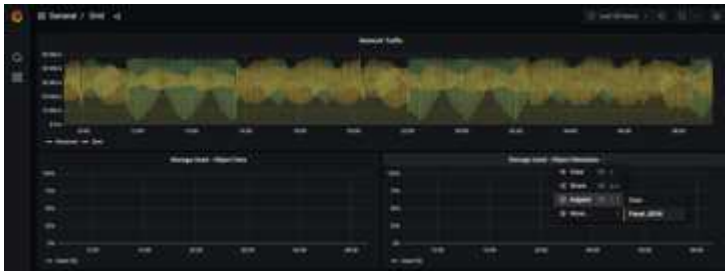
7. 我们可以继续为所需的每个指标构建这样的信息板、但幸运的是、StorageGRID 已经拥有包含面板的信息板、我们可以复制到自定义信息板中。

   a. 从StorageGRID 管理界面的左侧窗格中、选择"Support"、然后在"Tools"列的底部单击"Metrics "。

   b. 在指标中、我将选择中间列顶部的"网格"链接。

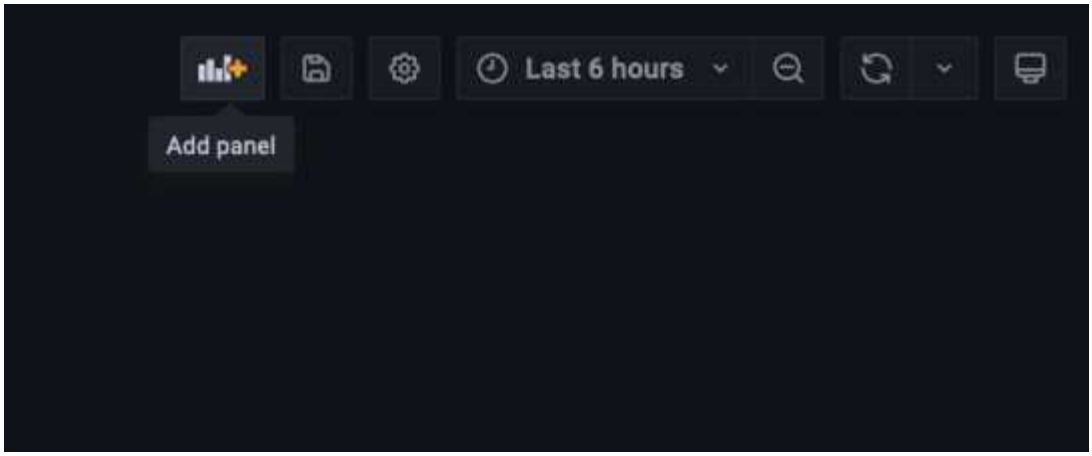


   c. 在网格信息板中、我们选择"已用存储-对象元数据"面板。单击小下箭头和面板标题的末尾以下拉菜单。从此菜单中选择"检查"和"面板JSON"。



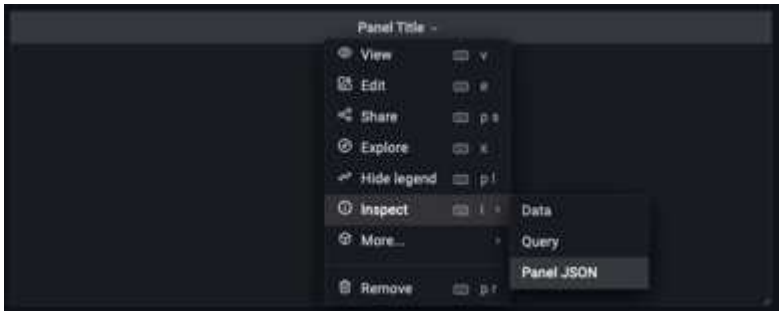
   d. 复制JSON代码并关闭窗口。

e. 在新信息板中、单击图标以添加新面板。

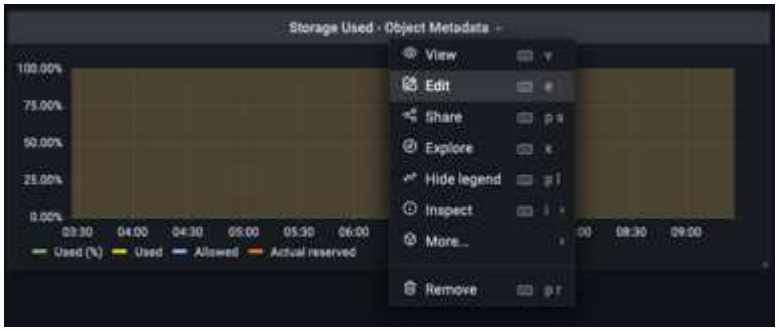


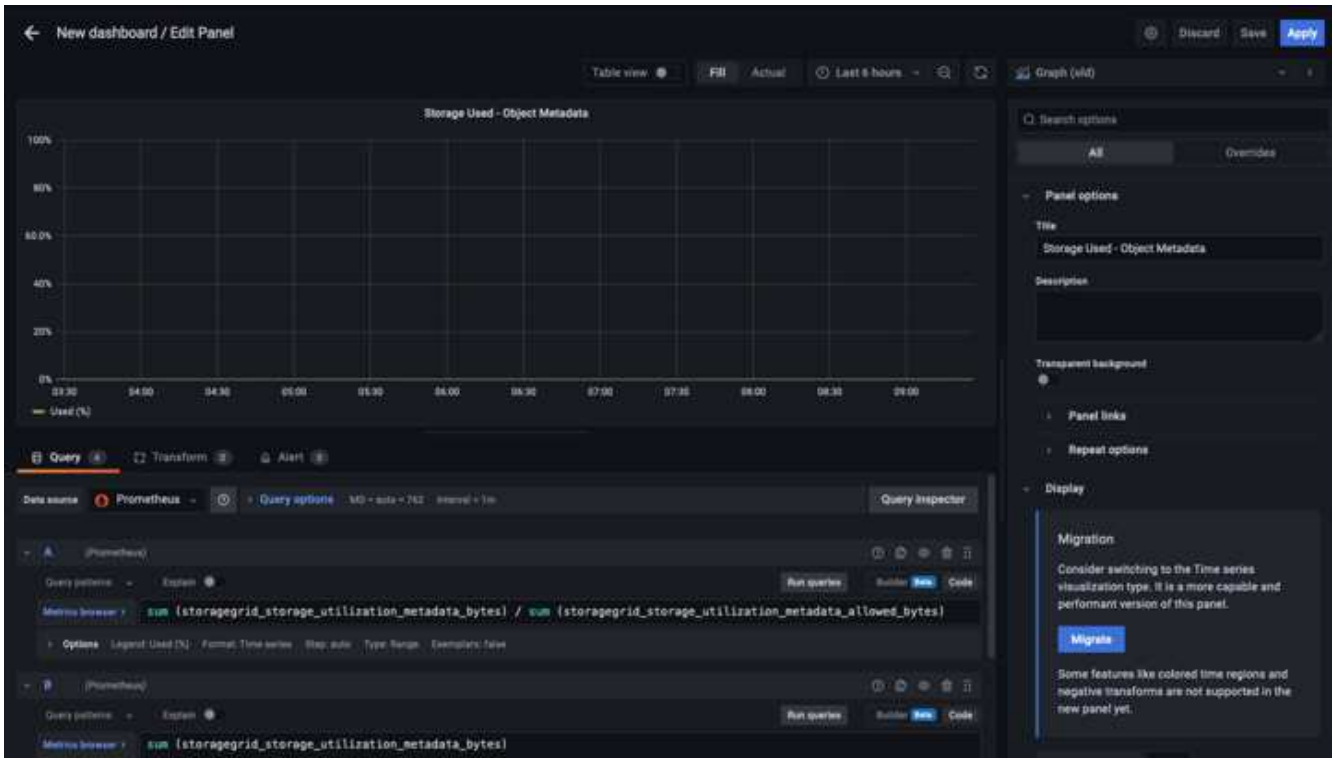
f. 应用新面板而不进行任何更改

g. 就像使用StorageGRID 面板一样、检查JSON。从StorageGRID 面板中删除所有JSON代码并将其替换为复制的代码。



h. 编辑新面板、在右侧、您将看到一条带有"迁移"按钮的迁移消息。单击按钮、然后单击"应用"按钮。





8. 将所有面板安装到位并根据需要进行配置后。单击右上角的磁盘图标以保存信息板、并为您的信息板指定一个名称。

结论

现在、我们推出了一款具有可自定义数据保留和存储容量的Prometheus服务器。这样、我们就可以继续构建自己的信息板、其中包含与我们的运营最相关的指标。您可以获取有关在中收集的Prometheus指标的详细信息 "StorageGRID 文档"。

使用 F5 DNS 实现StorageGRID 的全局负载均衡

作者：史蒂夫·戈尔曼 (F5)

本技术报告提供了详细的说明，指导如何配置NetApp StorageGRID与 F5 DNS 服务，以实现全局负载均衡，从而在网格分布于多个站点和/或 HA 组时，提供更好的数据可用性、更高的数据一致性并优化 S3 事务路由。

简介

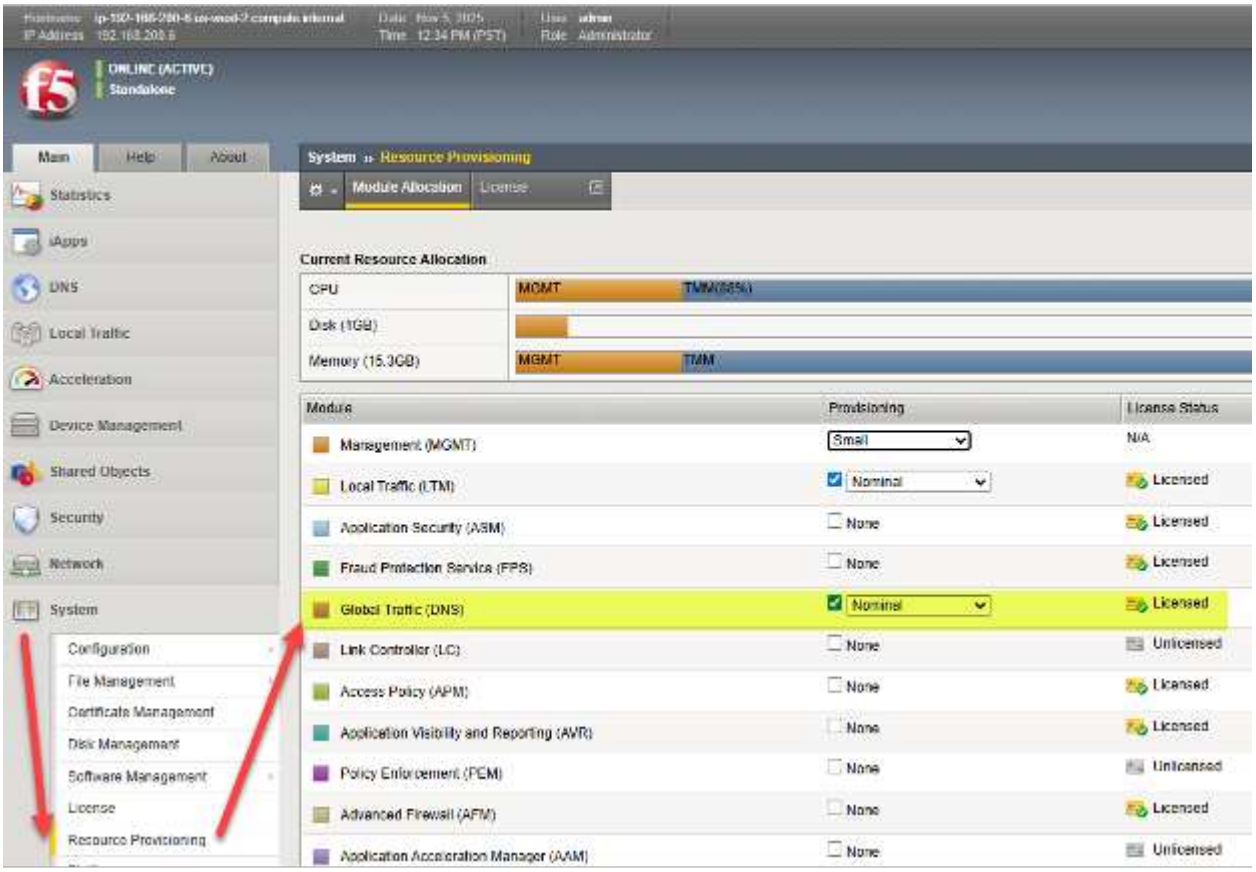
F5 BIG-IP DNS 解决方案以前称为 BIG-IP GTM（全球流量管理器），非正式名称为 GSLB（全球服务器负载均衡），它允许跨多个主动-主动 HA 组和主动-主动多站点StorageGRID解决方案实现无缝访问。

F5 BIG-IP 多站点StorageGRID配置

无论要支持多少个StorageGRID站点，至少需要两个 BIG-IP 设备（物理或虚拟）启用并设置 BIG-IP DNS 模块。DNS 设备越多，企业获得的冗余程度就越高。

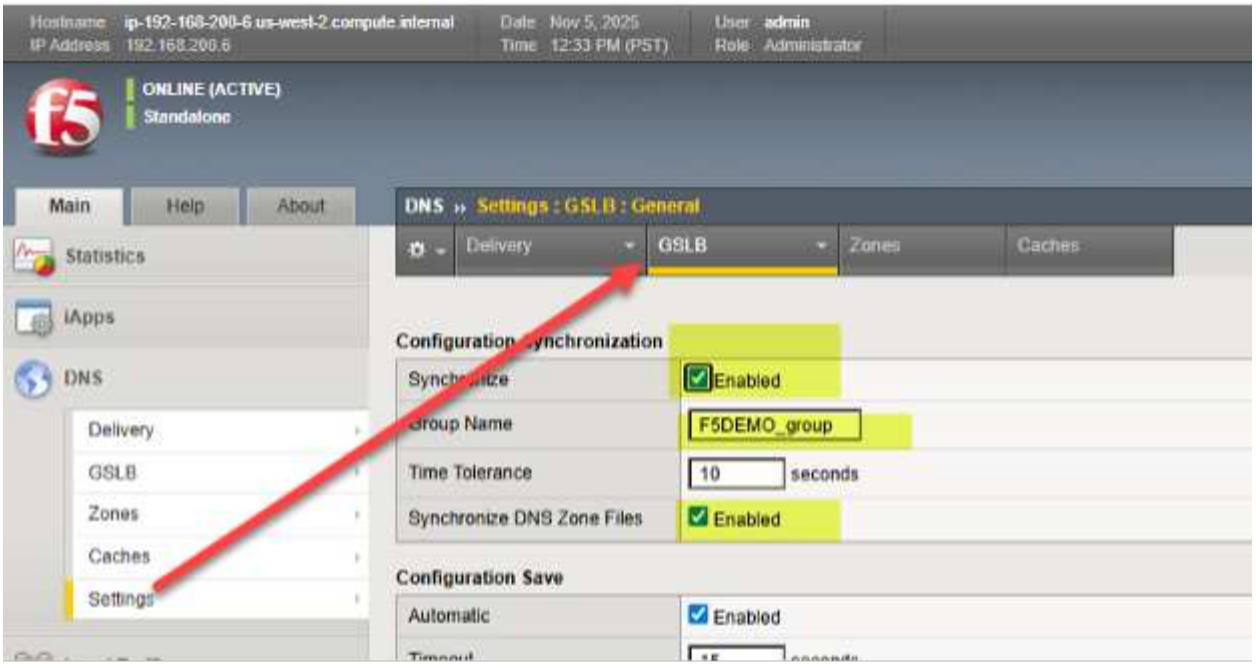
BIG-IP DNS - 初始设置入门指南

BIG-IP 设备完成至少初始配置后，使用 Web 浏览器登录到 TMUI（BIG-IP GUI）界面，然后选择“系统”→“资源配置”。如前所述，请确保“全球流量（DNS）”模块已勾选并显示已获得许可。请注意，如图所示，“本地流量(LTM)”通常可以配置在同一设备上。

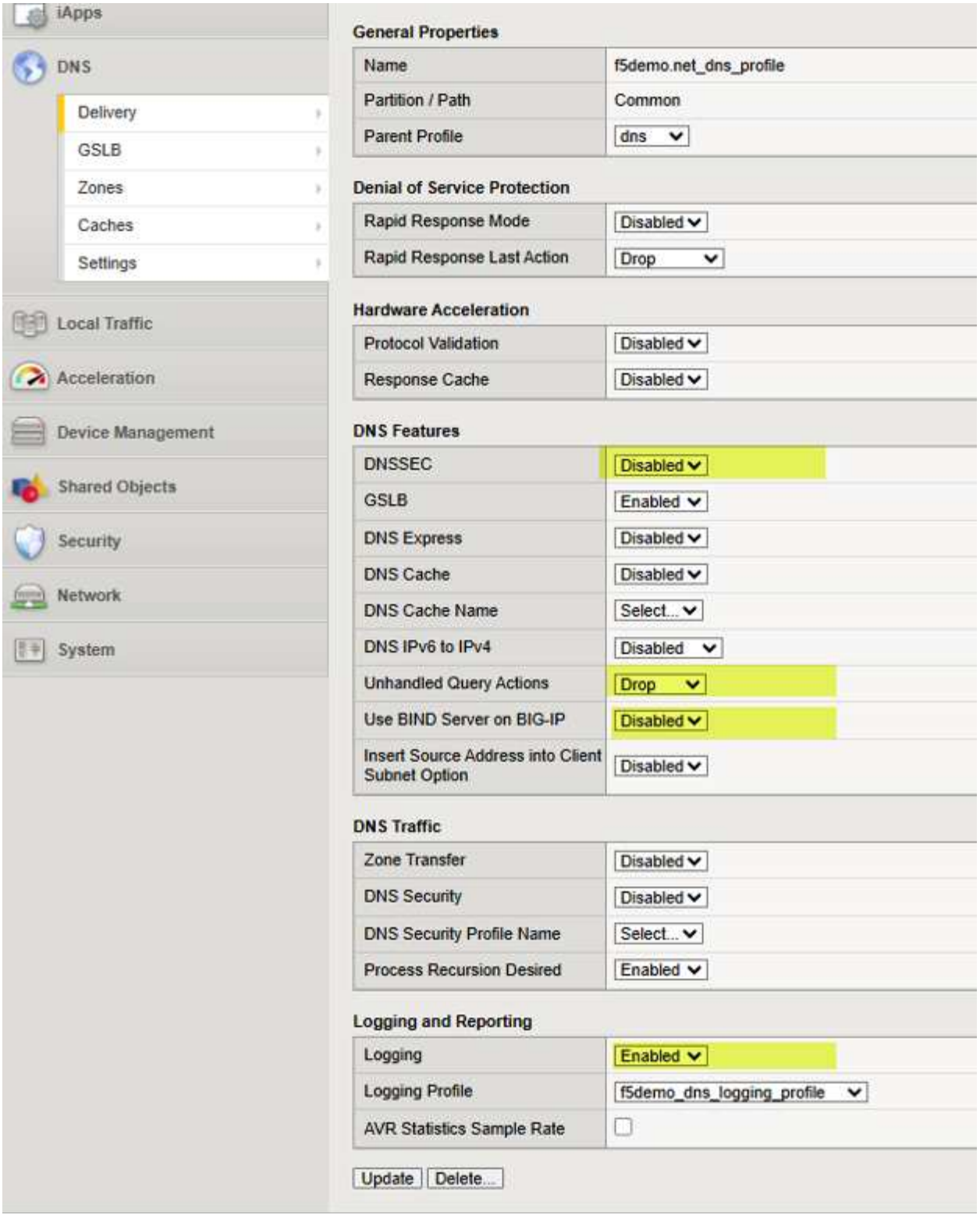


配置 **DNS** 协议基础元素

StorageGRID站点的全局流量管理的第一步是选择 DNS 选项卡，几乎所有全局流量控制都将在此配置，然后选择设置→ GLSB。启用这两个同步选项，并选择一个将在参与的 BIG-IP 设备之间共享的 DNS 组名称。



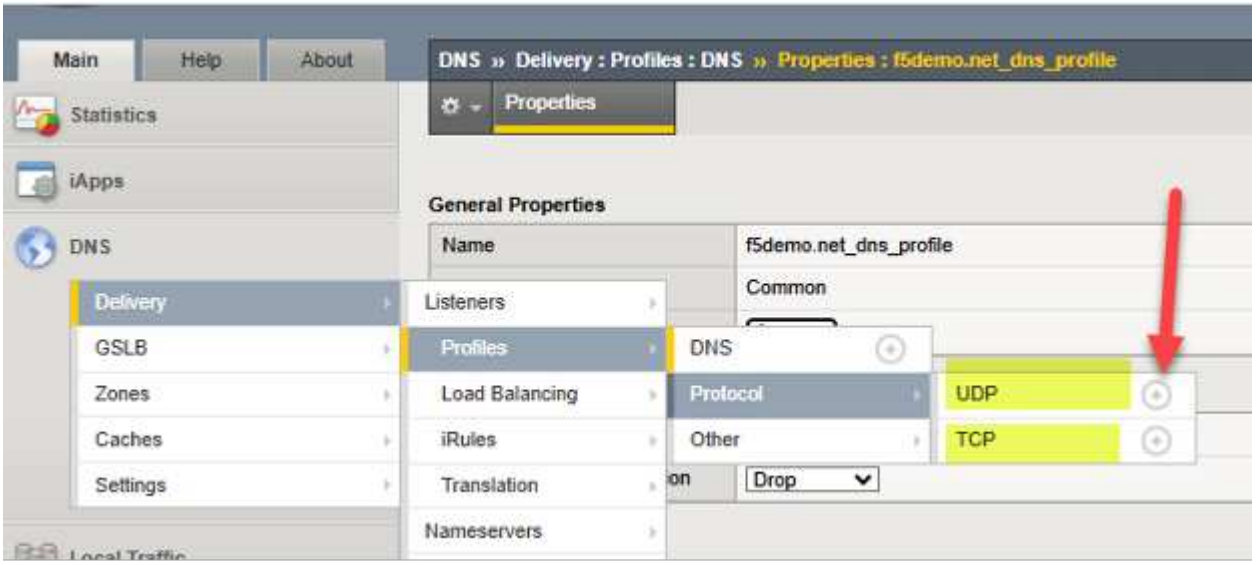
接下来，导航至 DNS > 交付 > 配置文件 > DNS：创建并创建一个配置文件，该配置文件将控制您希望启用或禁用的 DNS 功能。如果您对生成特定 DNS 日志感兴趣，请参阅前面的 DNS 课堂指南链接。以下是一个可用的 DNS 配置文件示例，请注意四个高亮部分，它们代表重要的设置值。为了便于理解，每种可能的设置都在以下 F5 知识库 (KB) 文章中进行了解释。 "此处"。



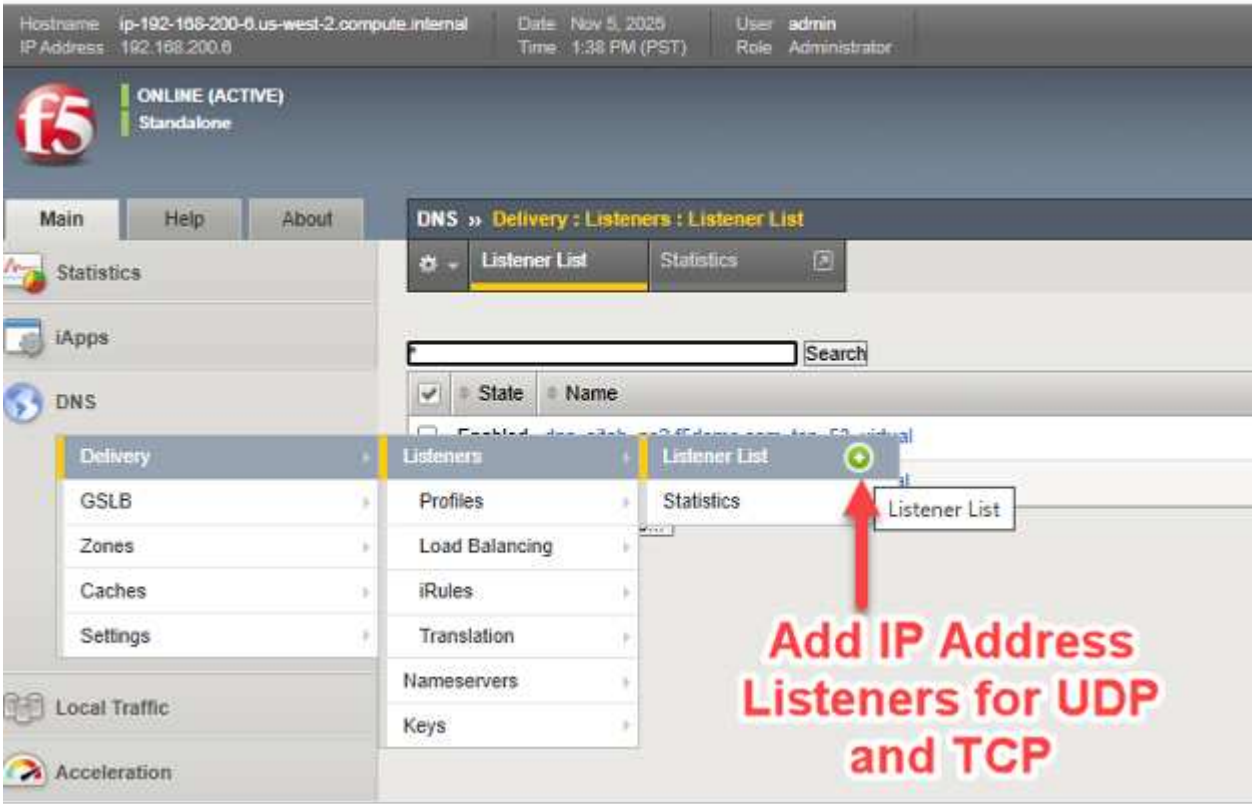
此时，我们可以通过创建的“配置文件”来调整 UDP 和 TCP 协议的特性，这两个协议都可以承载涉及 BIG-IP 的 DNS 流量。只需为 UDP 和 TCP 创建一个新的配置文件即可。假设 DNS 流量将跨越 WAN 链路，一个好的做法是直接继承在 WAN 环境中性能良好的 UDP 和 TCP 特性。要添加每个协议，只需单击每个协议旁边的“+”图标，并将父配置文件设置为以下内容：

UDP → 使用“父”配置文件“udp_gtm_dns”

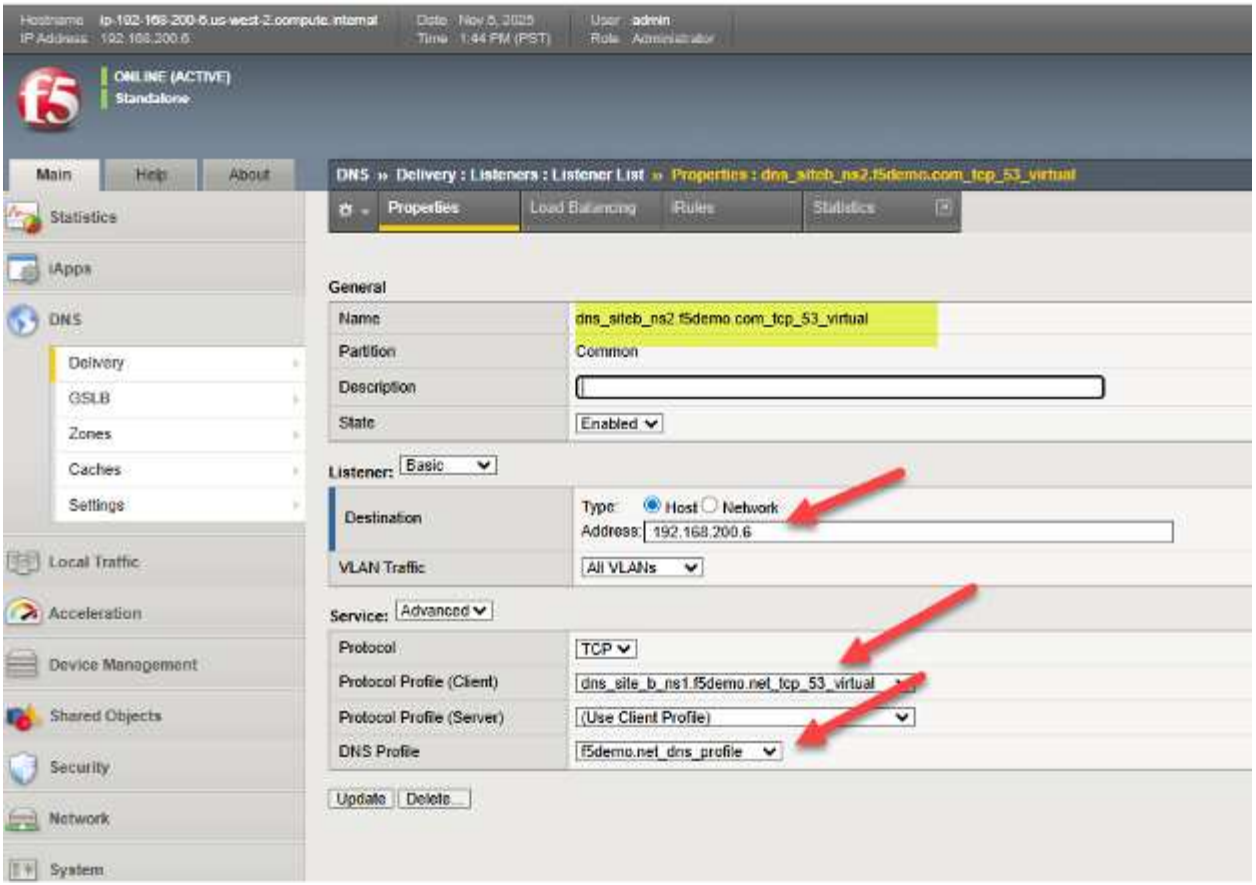
TCP → 使用“父”配置文件“f5-tcp-wan”



现在，我们只需要为涉及 BIG-IP DNS 的 UDP 和 TCP 流量分配一个 IP 地址即可。对于熟悉 BIG-IP LTM 的人来说，这本质上是创建 DNS 虚拟服务器，而虚拟服务器需要“监听器”IP 地址。如图所示，按照箭头指示创建 DNS/UDP 和 DNS/TCP 的监听器/虚拟服务器。



以下是来自实时 BIG-IP DNS 的一个示例，其中我们可以看到 TCP 虚拟服务器监听器设置，并可以看到它是如何将许多先前的步骤联系起来的。这包括引用 DNS 配置文件和协议 (TCP) 配置文件，以及配置 DNS 要使用的有效 IP 地址。与使用 BIG-IP 创建的所有对象一样，使用有意义的名称有助于识别对象是什么，例如示例名称中的 dns/siteb/TCP53。



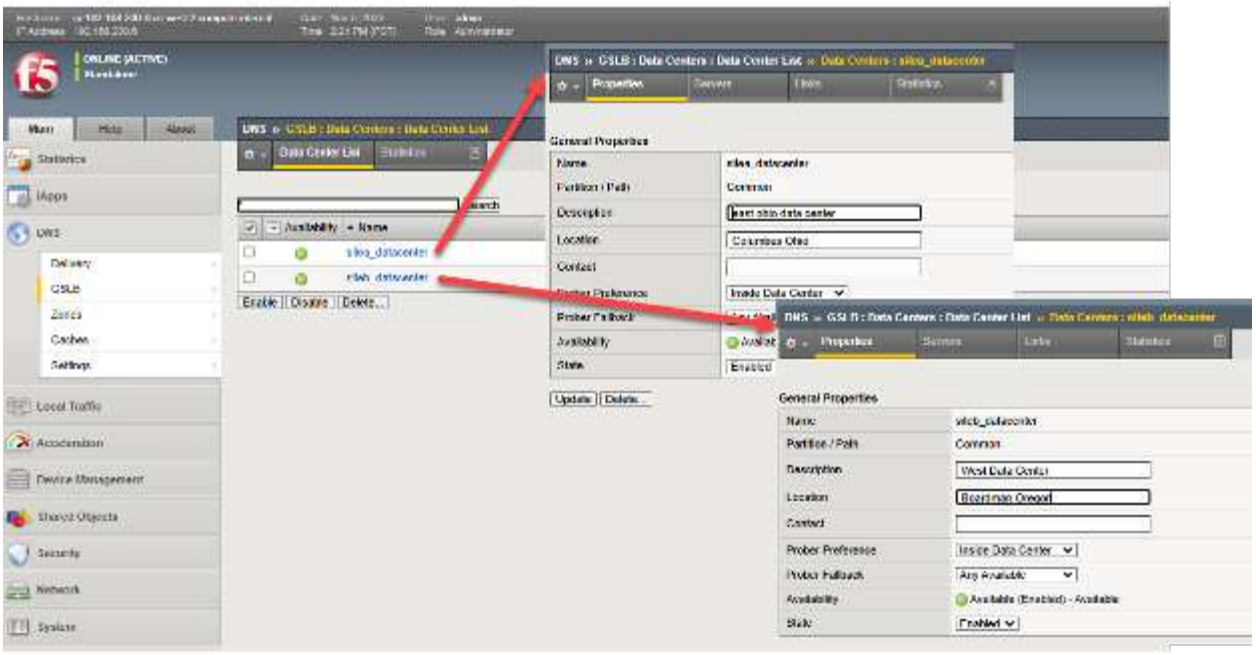
至此，启用 DNS 模块的 BIG-IP 设备的初步设置步骤（通常是“一次性”设置步骤）就完成了。目前，我们已准备好过渡到使用我们的设备建立全球流量管理解决方案的具体细节，这当然将与StorageGRID站点的特性相关联。

分四步完成数据中心站点搭建和**BIG-IP**间通信建立

第一步：创建数据中心

每个将容纳由 BIG-IP LTM 进行本地负载均衡的节点集群的站点都应该输入到 BIG-IP DNS 中。这只需要在一个 BIG-IP DNS 上完成，因为我们正在创建一个 DNS 同步组来支持流量管理，因此该配置将在组的 DNS 成员之间共享。

通过 TMUI GUI，选择 DNS > GSLB > 数据中心 > 数据中心列表，并为每个StorageGRID站点创建一个条目。如果使用与图 1 一致的网络设置，DNS 设备位于其他非StorageGRID站点，除了存储站点之外，还要为这些站点添加数据中心。在本例中，站点 a 和 b 分别创建在俄亥俄州和俄勒冈州，BIG-IP 是双 DNS 和 LTM 设备。



第二步：创建服务器（解决方案中所有 **BIG-IP** 设备列表）

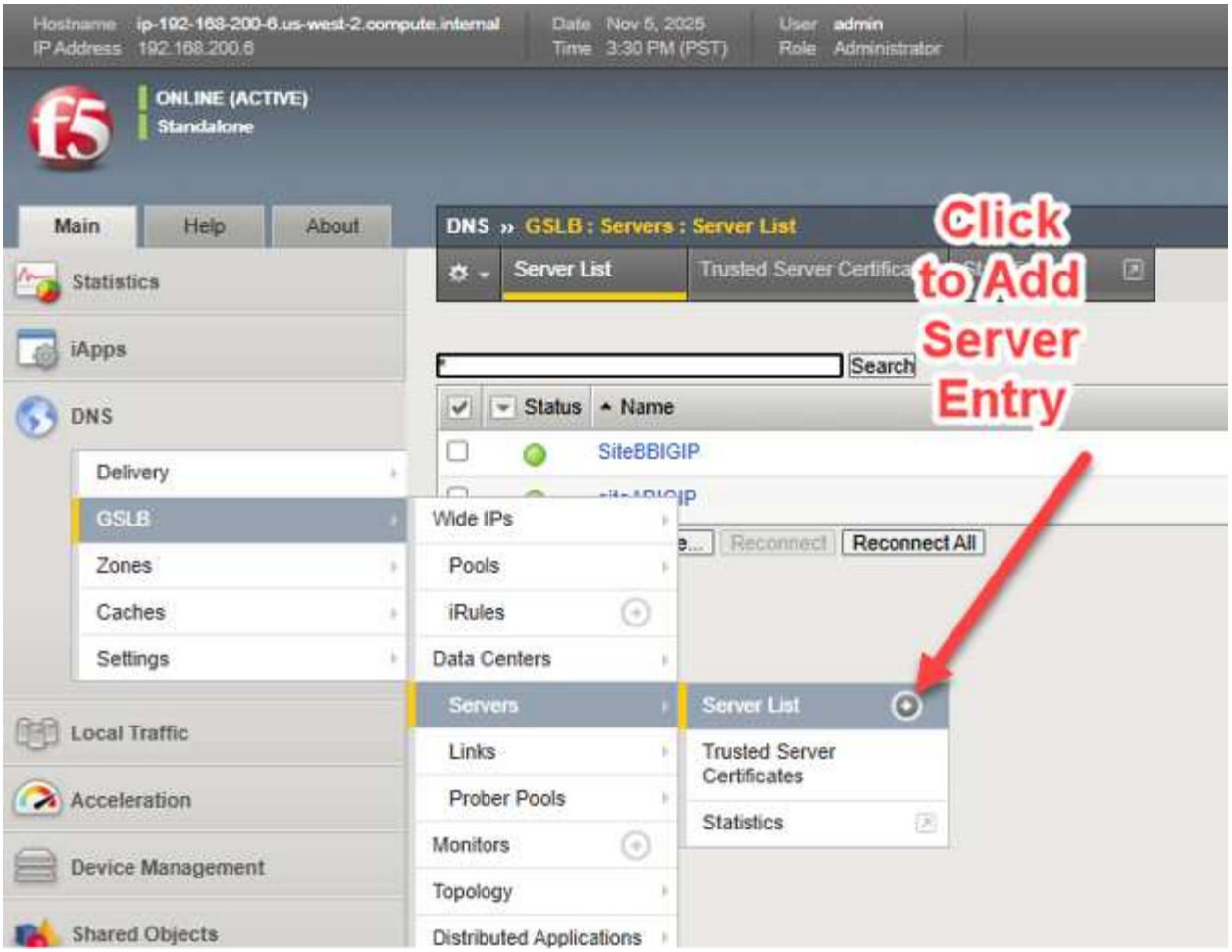
现在我们已准备好将各个StorageGRID站点集群连接到 BIG-IP DNS 设置。回想一下，每个站点的 BIG-IP 设备将通过配置虚拟服务器来实际进行 S3 流量的负载均衡，这些虚拟服务器将“前端”可达 IP 地址/端口绑定到一组后端存储节点设备的“池”，使用“后端”IP 地址/端口。

例如，如果池中的所有存储节点因管理原因（例如站点停用）或因实时健康检查失败而意外离线，则流量将通过更改 DNS 查询响应定向到其他站点。

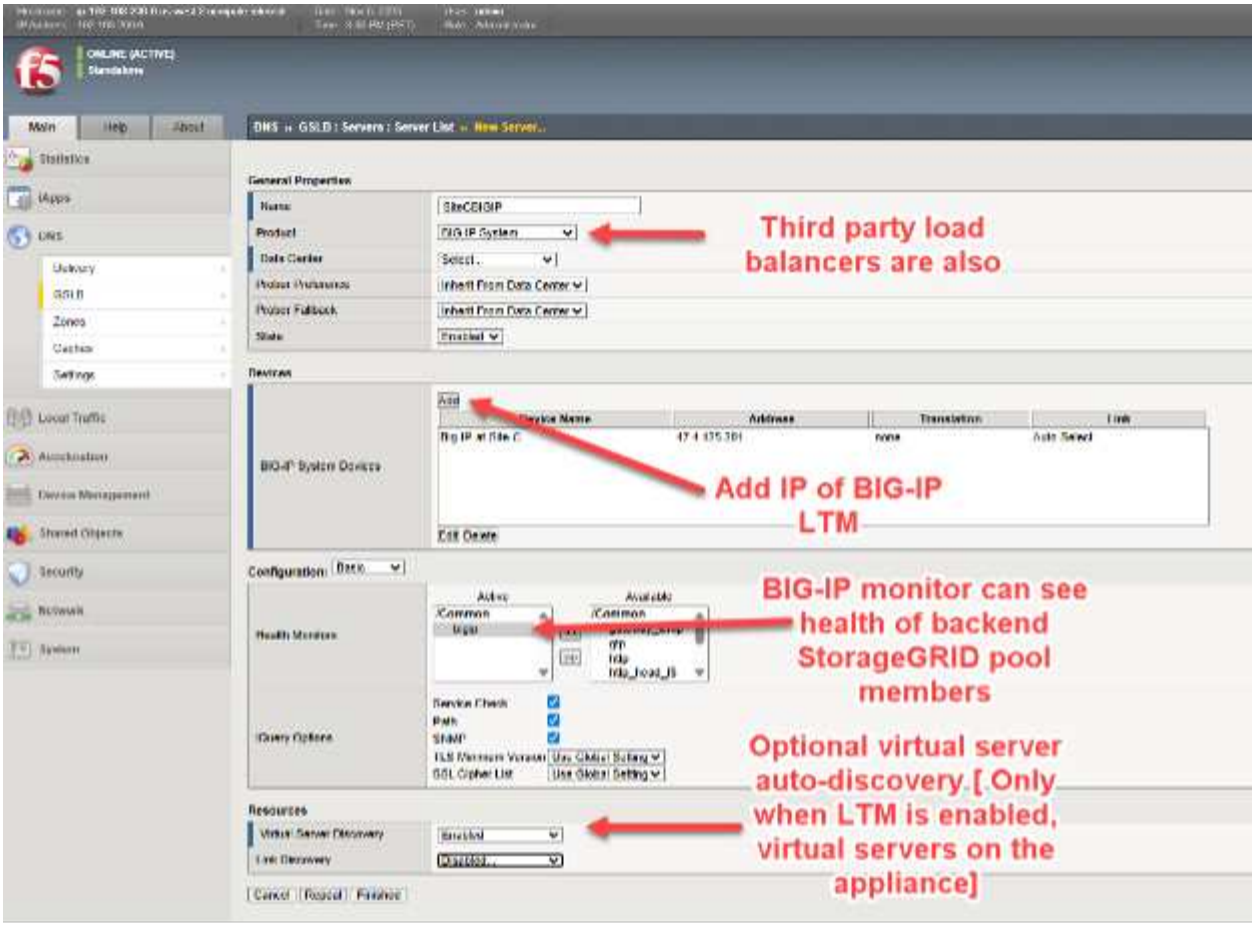
要将 StorageGrid 站点（特别是本地虚拟服务器）与每个设备上的 BIG-IP DNS 配置关联起来，只需设置一次即可。接下来，所有 BIG-IP DNS 设备都将进行同步设置。

简单来说，我们将创建一个列表，称为服务器列表，其中包含我们所有的 BIG-IP 设备，无论它们是否获得了 DNS、LTM 或 DNS 和 LTM 的许可。列表完成后，该主列表将与所有 BIG-IP DNS 设备同步。

在一台已获得 BIG-IP DNS 许可的设备上，选择 DNS > GSLB > 服务器 > 服务器列表，然后选择添加按钮 (+)。



添加每个 BIG-IP 时，四个关键要素包括：* 从产品下拉菜单中选择 BIG-IP，虽然也可以使用其他负载均衡器，但通常缺乏在每个站点后端节点健康状况恶化时的实时可见性和响应能力。* 添加 BIG-IP DNS 设备的 IP 地址。通常情况下，首次添加 BIG-IP DNS 设备时，地址将是当前可通过 GUI 访问的设备，以后添加的设备将是解决方案中的其他设备。* 选择健康监视器，当添加的负载均衡器是 BIG-IP 设备时，始终使用“BIG-IP”，以便考虑后端StorageGRID节点的健康状况。* 如果设备是双 DNS/LTM 设备，则可以选择请求虚拟服务器自动发现。

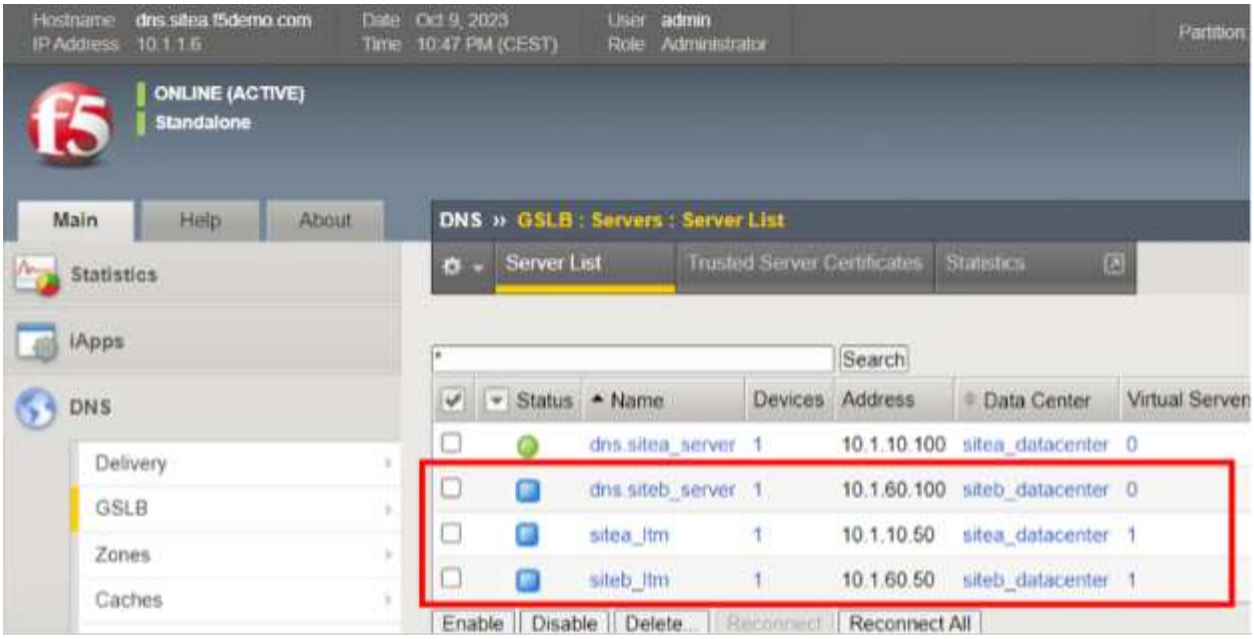


在某些情况下，例如瞬态网络问题或网络位置之间的防火墙 ACL 规则，在此阶段添加远程设备时，虚拟服务器发现可能不会显示已配置 LTM 的远程设备的条目。在这种情况下，添加新设备（“服务器”）后，可以按照以下指示手动添加虚拟服务器。如果添加的是仅用于 DNS 的 BIG-IP 设备，则不会发现或向该设备添加任何虚拟服务器。

我们需要在所有站点为解决方案中的每个设备添加这些服务器条目，包括 BIG-IP DNS 设备、BIG-IP LTM 设备以及任何同时承担 DNS 和 LTM 双重角色的设备。

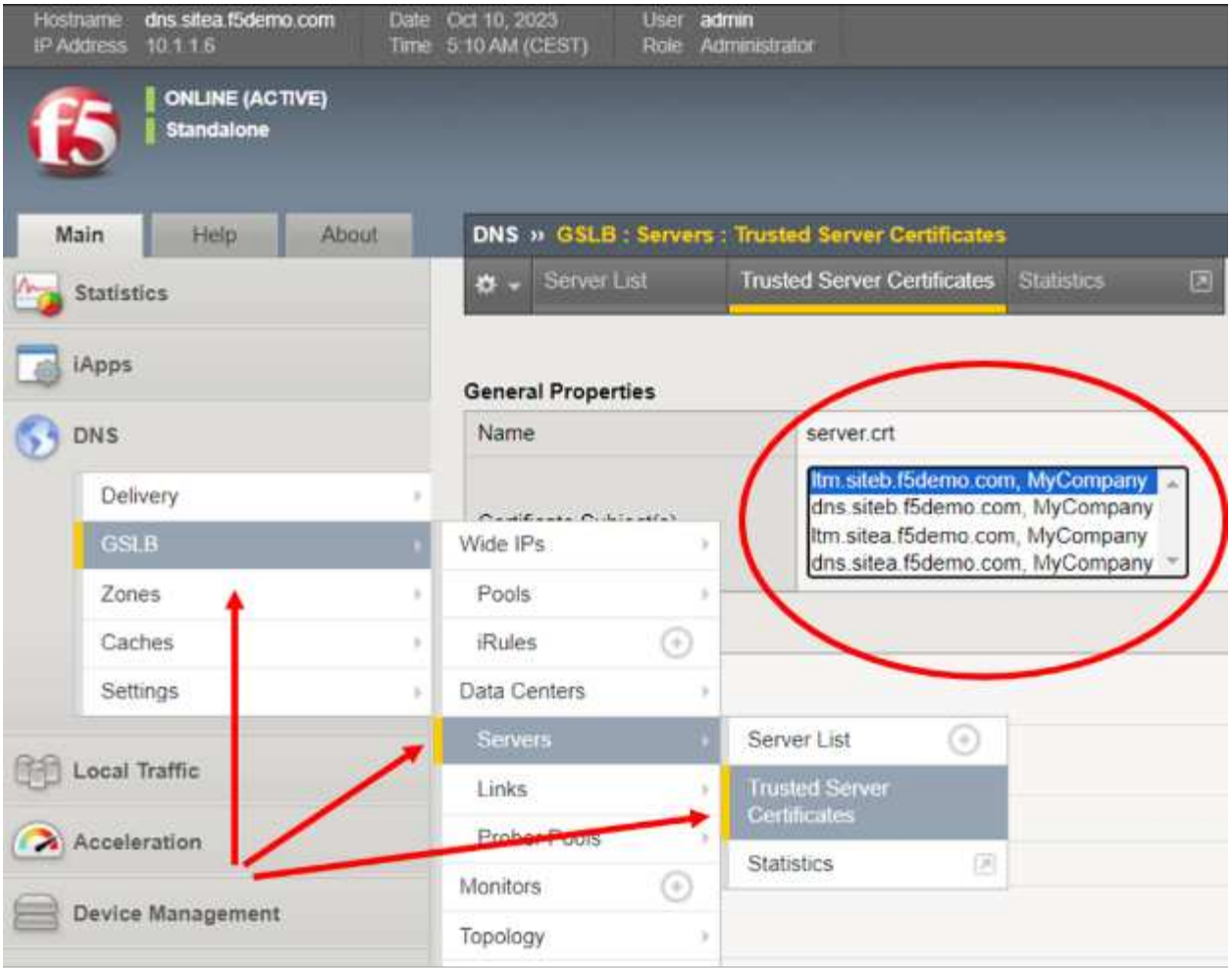
第三步：在所有**BIG-IP**设备之间建立信任

在以下示例中，添加了四台设备作为服务器，它们分布在两个站点。请注意，每个站点都有专用的 BIG-IP DNS 和 BIG-IP LTM。但是，除了当前登录的设备之外，所有其他设备的“状态”列中都显示蓝色图标。这意味着尚未与其他 BIG-IP 设备建立信任关系。



要添加信任，请使用 SSH 连接到刚刚通过 GUI 输入配置详细信息的 BIG-IP，并使用“root”帐户访问 BIG-IP 命令行界面。在提示符下发出以下单个命令：*bigip_add*

“bigip_add”命令从目标BIGIP设备拉取管理证书，用于在集群中的GSLB服务器之间建立加密的“iQuery”通道。默认情况下，iQuery使用TCP端口4353运行，它是一种心跳机制，使BOG-IP DNS成员保持同步状态。它在加密通道中使用了 xml 和 gzip。当不带任何选项运行“bigip_add”时，该命令将使用当前用户名连接到端点，对 GSLB 服务器列表中的所有 BIGIP 设备运行。为了快速检查是否成功，只需返回 BIG-IP GUI 并确认所有服务器的证书现在都列在显示的下拉菜单中即可。



第四步：将所有 **BIG-IP DNS** 设备同步到 **DNS** 组

最后一步将允许仅使用单个设备的 TMUI GUI 即可完全配置所有 BIG-IP DNS 设备。例如，如果有两个StorageGRID站点，这意味着现在需要使用 SSH 访问另一个站点的 BIG-IP DNS 的命令行。以 root 用户身份连接后，并确保防火墙策略/ACL 允许两个 BIG-IP DNS 设备通过 TCP 端口 22 (SSH)、443 (HTTPS) 和 4354 (F5 iQuery 协议) 进行通信，然后在提示符下执行以下命令：*gtm_add* <之前完成所有 *GUI* 步骤的第一个站点 *BIG-IP DNS* 的 *IP* 地址>

此时，所有进一步的 DNS 配置工作都可以在已添加到该组的任何 BIG-IP DNS 设备上执行。对于仅支持 LTM 的设备成员，无需应用上述命令 gtm_add。只有支持 DNS 的设备才需要此命令才能加入同步 DNS 组。

建立数据中心站点和建立**BIG-IP**间通信

至此，创建底层、健康的 BIG-IP DNS 设备组的所有步骤均已完成。现在我们可以继续创建指向我们在每个StorageGRID数据中心公开的分布式 Web/S3 服务的名称、FQDN。

这些名称被称为“广域 IP 地址”，简称 WIP，它们是具有 DNS A 资源记录的普通 DNS FQDN。然而，它们并不像传统的 A 资源记录那样指向服务器，而是在内部指向 BIG-IP 虚拟服务器池。每个池都可以单独由一个或多个虚拟服务器组成。请求 IP 地址进行名称解析的 S3 客户端将收到策略选择的最佳StorageGRID站点上的 S3 虚拟服务器的地址。

广域**IP**、**IP**池和虚拟服务器简述

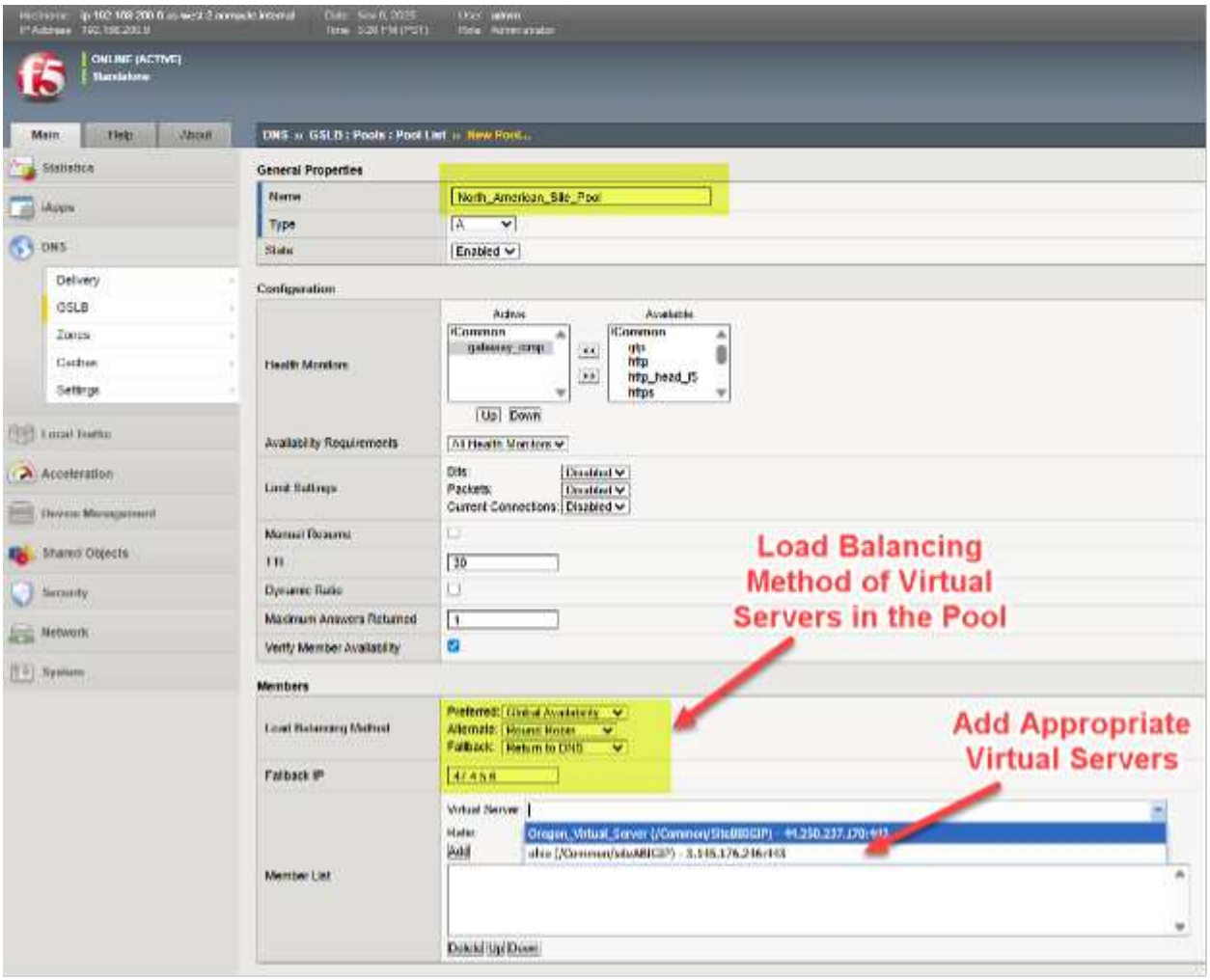
举一个简单的虚构例子，名称 **storage.quantumvault.com** 的 WIP 可能会将 BIG-IP DNS 解决方案与两个潜在

的虚拟服务器池连接起来。第一个池子可能由北美的 4 个地点组成；第二个池子可能由欧洲的 3 个地点组成。

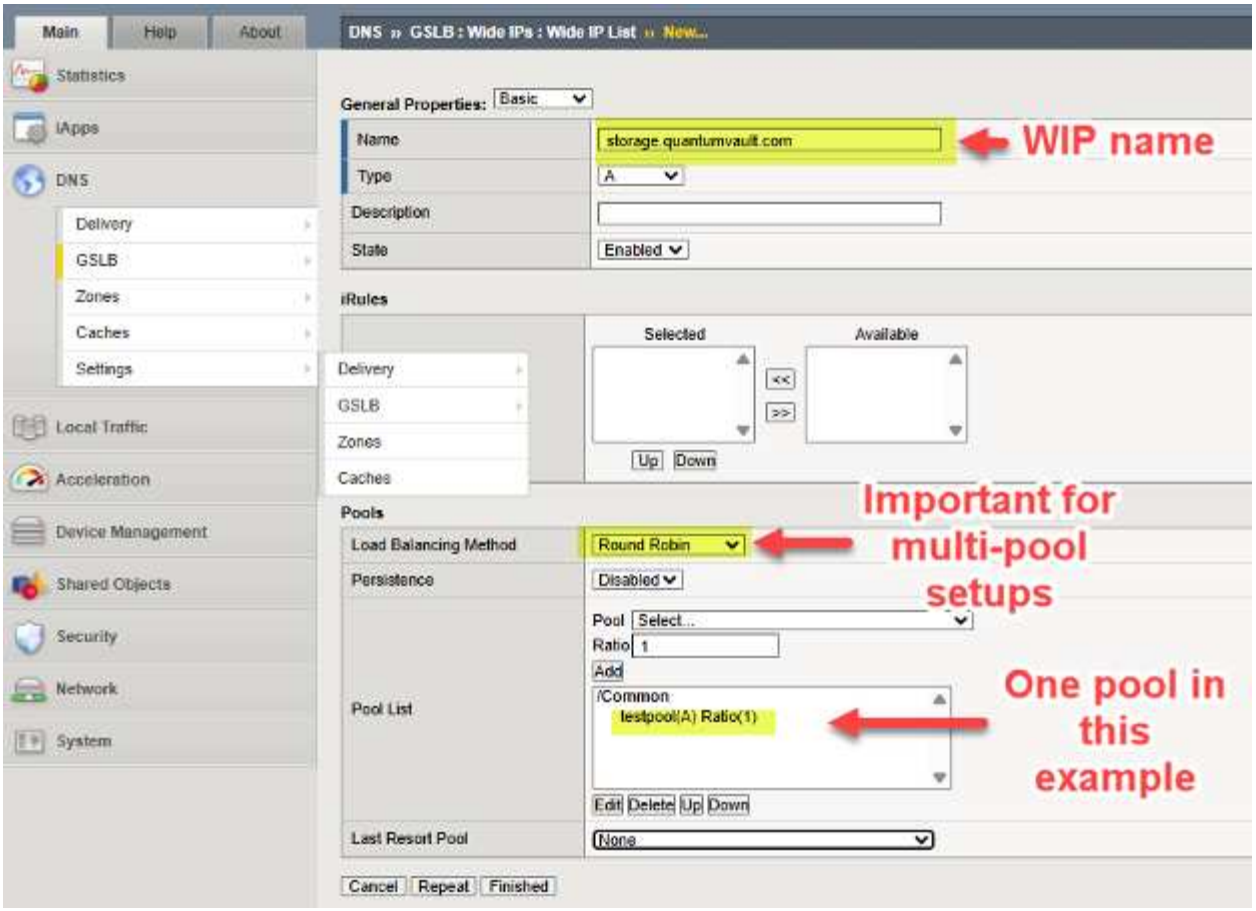
选定的池子可能是通过一系列政策决定得出的，也许可以使用 5:1 的简单比例，将大部分流量导向北美StorageGRID站点。更有可能的是，基于拓扑结构的选择，例如，将所有来自欧洲的 S3 流量定向到欧洲站点，而将世界其余的 S3 流量定向到北美数据中心。

一旦 BIG-IP DNS 确定了池，假设选择了北美池，则返回用于解析 storage.quantumvault.com 的实际 DNS A 资源记录可以是 BIG-IP LTM 在北美 4 个站点中支持的 4 个虚拟服务器中的任何一个。同样，具体选择取决于策略，存在简单的“静态”方法，例如轮询；而更高级的“动态”选择，例如使用性能探针来测量每个站点从本地 DNS 解析器的延迟，则被保留并用作站点选择的标准。

要在 BIG-IP DNS 上设置虚拟服务器池，请按照菜单路径 **DNS > GSLB > Pools > Pool List > Add (+)** 进行操作。在这个例子中，我们可以看到各种北美虚拟服务器被添加到池中，并且选择此池时，首选的负载均衡方法是分层选择的。



我们将 WIP（广域 IP），即我们的服务名称（将通过 DNS 解析），按照 DNS > GSLB > 广域 IP > 广域 IP 列表 > 创建 (+) 的步骤添加到部署中。在下面的示例中，我们提供了一个支持 S3 的存储服务的示例 WIP。



调整 **DNS** 以支持全球流量管理

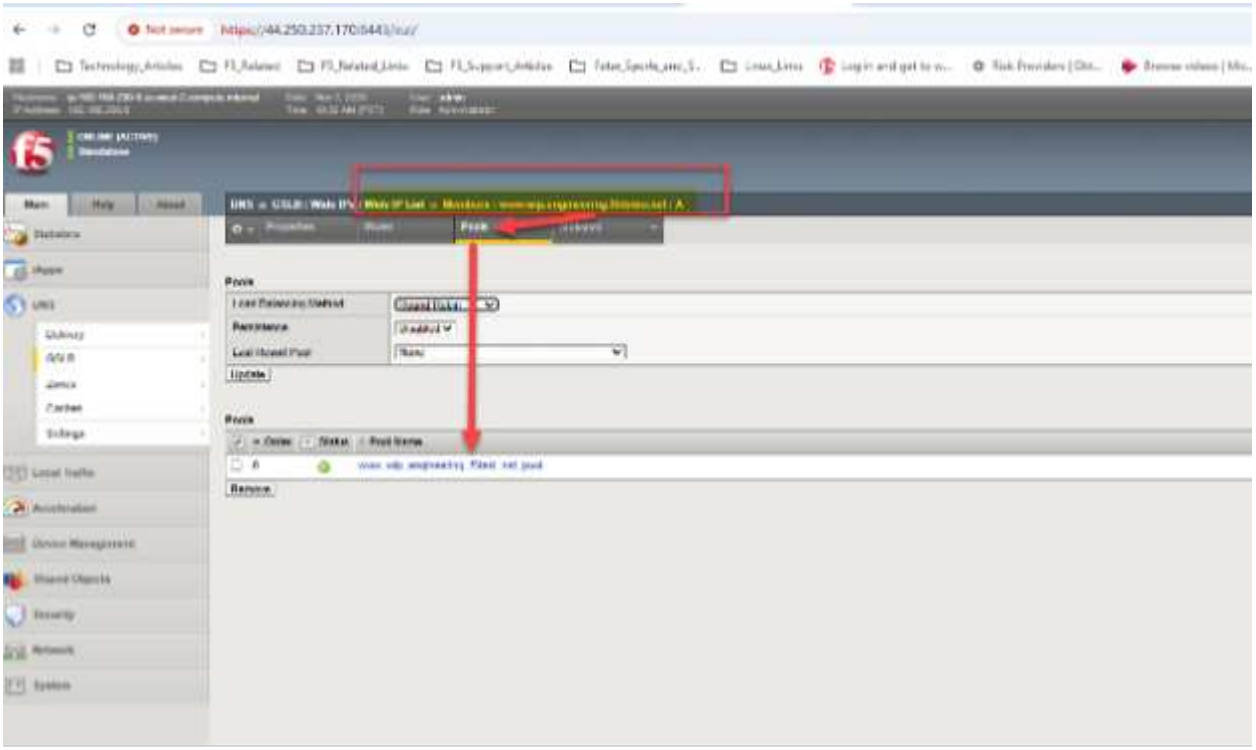
目前，我们所有的底层 BIG-IP 设备都已准备好执行 GSLB（全局服务器负载均衡）。我们只需要调整并分配用于 S3 流量的名称，即可利用该解决方案。一般做法是将企业现有 DNS 域的一部分委托给 BIG-IP DNS 控制。也就是说，要“划分”出一部分名称空间，即一个子域，并将该子域的控制权委托给 BIG-IP DNS 设备。从技术上讲，这是通过确保 BIG-IP DNS 设备在企业 DNS 中具有 A DNS 资源记录 (RR)，然后将这些名称/地址设置为委派域的名称服务器 (NS) DNS 资源记录来实现的。

如今企业维护 DNS 的方式有很多种，其中一种方法是完全托管的解决方案。例如，通过 Windows Server 2025 运行和管理 DNS 就是这种情况。企业还可以采用另一种方法，即利用 AWS Route53 或 Squarespace 等云 DNS 提供商。

以下示例纯属虚构，仅供参考。我们有StorageGRID支持通过 S3 协议对对象进行读取和写入，现有域由 AWS Route53 管理，现有示例域为 f5demo.net。

我们希望将子域名 engineering.f5demo.net 分配给 BIG-IP DNS 设备，以实现全球流量管理。为此，我们为 engineering.f5demo.net 创建一个新的 NS（名称服务器）资源记录，并将其指向 BIG-IP DNS 设备名称列表。在我们的示例中，我们有两个 BIG-IP DNS 设备，因此我们为它们创建了两条 A 资源记录。

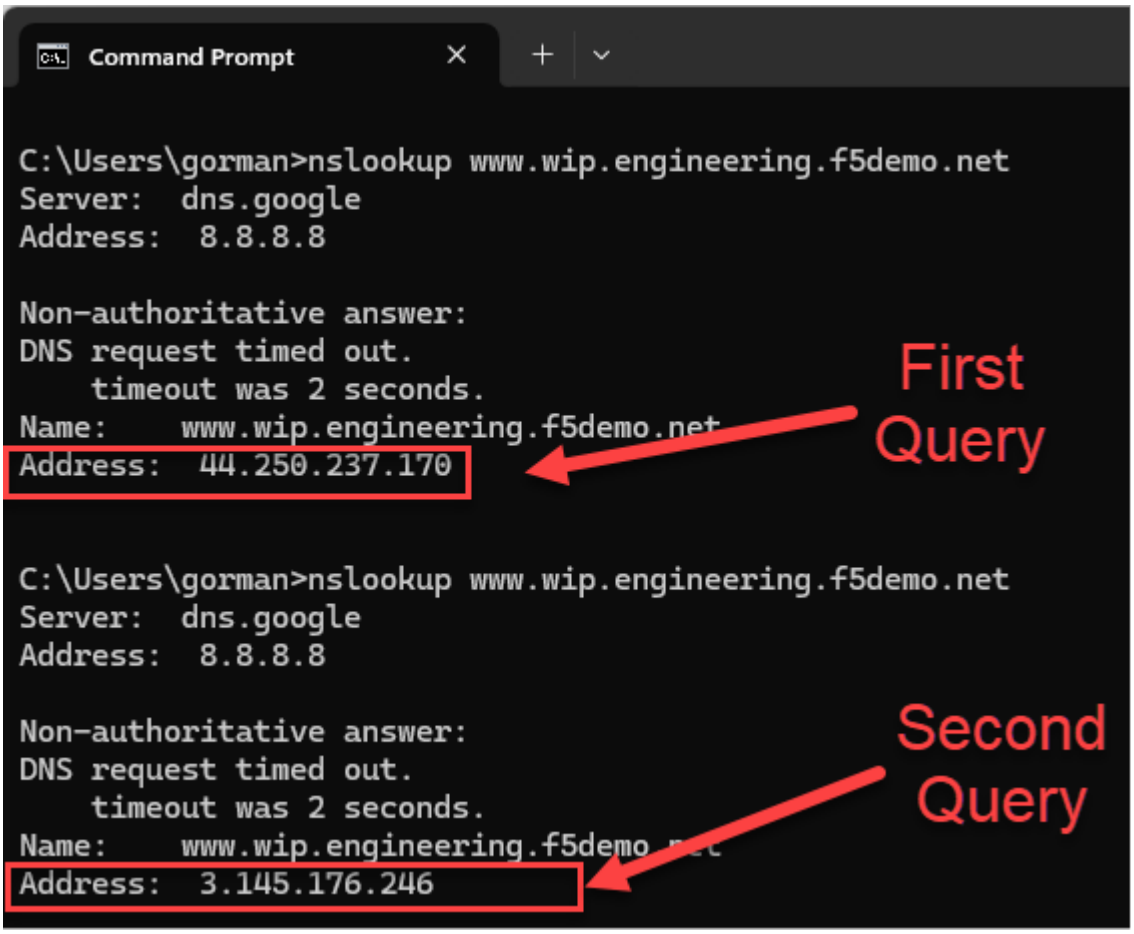
现在，我们以 BIG-IP DNS 为例，设置一个广域 IP (WIP)。由于 DNS 使用组同步，我们只需要使用一台设备的 GUI 进行调整。在 BIG-IP DNS GUI 中，转到 **DNS > GSLB >** 广域 **IP >** 广域 **IP** 列表 **(+)**。回想一下，在传统的 DNS FQDN 设置中，需要输入一个或多个 IPv4 地址，而在我们的例子中，我们只需指向一个或多个StorageGRID虚拟服务器池。



在我们的示例中，我们在俄亥俄州和俄勒冈州站点都设有通用的 Web HTTPS 服务器。通过简单的“轮询”方法，我们应该能够看到全局 DNS 响应对 *www.wip.engineering.f5demo.net* 的 A 资源记录映射的查询，其中包含两个虚拟服务器 IP。

可以使用网页浏览器进行简单的测试，或者，对于使用StorageGRID 的S3，可以使用 S3Browser 等图形工具进行测试。由于我们在池内选择了轮询调度，因此每次 DNS 查询都会将池中的下一个数据中心站点用作后续流量的目标。

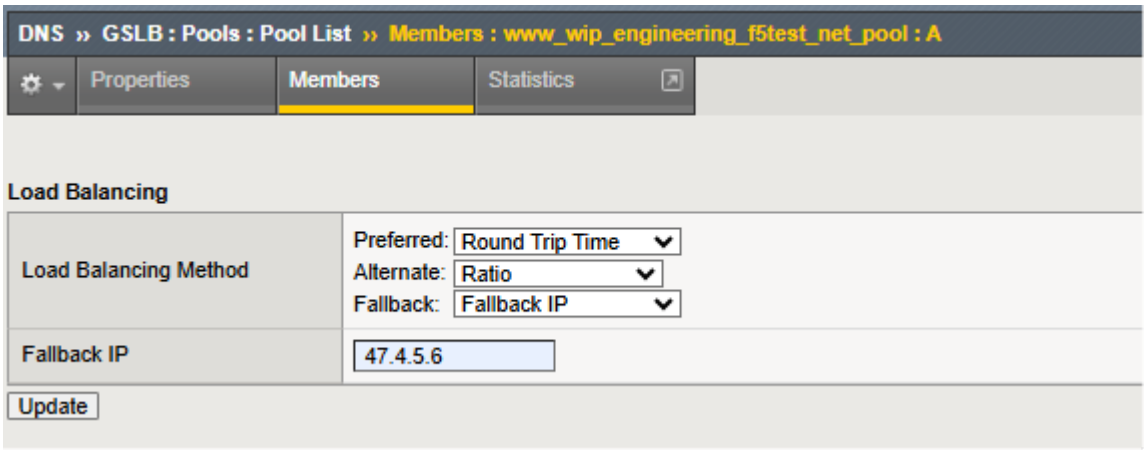
在我们的示例设置中，我们可以使用 dig 或 nslookup 快速生成一系列两个 DNS 查询，以确保 BIG-IP DNS 确实在进行轮询负载均衡，从而使两个站点都能随着时间的推移接收流量。



建议探索更高级的技术

众多可行方法之一，是使用“全局可用性”模式，而不是上面给出的简单“轮询”示例。通过全局可用性，可以将流量定向到池的顺序，或者单个池中的虚拟服务器。这样一来，所有 S3 流量默认都可以定向到例如纽约市的站点。

如果健康检查表明此站点的StorageGRID节点可用性存在问题，则流量届时可能会被定向到圣路易斯。如果圣路易斯出现健康问题，法兰克福的站点可以转而开始接收 S3 读取或写入交易。因此，全球可用性是提高 S3 StorageGRID整体解决方案弹性的一种方法。另一种方法是混合搭配负载均衡方法，采用分层方法。



在这个例子中，“动态”选项是已配置池中站点的首选负载均衡方式。在所示的示例中，持续测量方法采用主动探测本地 DNS 解析器性能的方法，这是站点选择的催化剂。如果这种方法不可行，则可以根据分配给每个站点的比例来选择各个站点。按比例计算，规模更大、带宽更高的StorageGRID站点可以接收比规模较小的站点更多的 S3 事务。最后，作为一种灾难恢复方案，如果池中的所有站点都出现故障，则指定的备用 IP 将用作最后的备选站点。BIG-IP DNS 中比较有趣的负载均衡方法之一是“拓扑”，它观察 DNS 查询的传入源（即 S3 用户的本地 DNS 解析器），并利用互联网拓扑信息从池中选择看似“最近”的站点。

最后，如果站点遍布全球，则值得考虑使用 F5 BIG-IP DNS 手册中详细讨论的动态“探测”技术。通过探测，可以监控频繁的 DNS 查询来源，例如，流量通常使用同一本地 DNS 解析器的企业对企业合作伙伴。可以从全球每个站点的 BIG-IP LTM 启动 BIG-IP DNS 探测，以大致确定哪个潜在站点可能为 S3 交易提供最低延迟。因此，来自亚洲的流量可能由亚洲的StorageGRID站点提供比位于北美或欧洲的站点更好的服务。

结论

F5 BIG-IP 与NetApp StorageGRID的集成解决了与跨多个站点的数据可用性和一致性相关的技术挑战，并优化了 S3 事务路由。部署此解决方案可增强存储弹性、性能和可靠性，使其成为寻求强大、可扩展和灵活的存储基础架构的企业的理想选择。

如需了解更多信息，请访问此处查看 F5 官方的 BIG-IP DNS 文档。 "链接"。还可以找到一份课堂风格的指导手册，其中包含示例设置的分步说明。 "此处"。

Datadog SNMP配置

作者：*Aron Klein*

配置Datadog以收集StorageGRID SNMP指标和陷阱。

配置Datadog

Datadog是一种监控解决方案 、可提供指标、可视化和警报功能。以下配置是在StorageGRID 系统本地部署的Ubuntu 22.04.1主机上使用Linux代理版本7.43.1实施的。

从**StorageGRID MIB**文件生成的数据日志配置文件和陷阱文件

Datadog提供了一种将产品MIB文件转换为映射SNMP消息所需的数据日志参考文件的方法。

按照找到的说明生成用于数据日志陷阱解析映射的StorageGRID YAML文件 "此处"。+将此文件放在/etc/datadog-agent/conf.d/snmp.d/traps_db/+中

- "下载陷阱YAML文件" +
  - * MD5校验和* 42e27e4210719945a46172b98c379517 +
  - * SHA256校验和* d0fe5c8e6ca3c902d054f854b70a85f928cb8b7c76391d356f05d2cf73b6887 +

此StorageGRID 配置文件YAML文件用于数据日志指标映射、此文件是按照找到的说明生成的 "此处"。+将此文件放置在/etc/datadog-agent/conf.d/snmp.d/profiles/+中

- "下载配置文件YAML文件" +
  - * MD5校验和* 72bb7784f4801adda4e0c3ea77df19aa +
  - * SHA256校验和* b6b7fadd330 63422a8bb8e39b3ead8ab38349ee0229926eadc8585f0087b8cee +

用于衡量指标的**SNMP Datadog**配置

可以通过两种方式管理为指标配置SNMP。您可以通过提供包含StorageGRID 系统的网络地址范围来配置自动发现、也可以定义各个设备的IP。根据所做的决定、配置位置会有所不同。自动发现在数据日志代理YAML文件中定义。在SNMP配置YAML文件中配置显式设备定义。以下是同一StorageGRID 系统中的每个示例。

自动发现

配置位于/etc/datadog-agent/datadog.yaml中

```
listeners:
  - name: snmp
snmp_listener:
  workers: 100  # number of workers used to discover devices concurrently
  discovery_interval: 3600  # interval between each autodiscovery in
seconds
  loader: core  # use core check implementation of SNMP integration.
recommended
  use_device_id_as_hostname: true  # recommended
  configs:
    - network_address: 10.0.0.0/24  # CIDR subnet
      snmp_version: 2
      port: 161
      community_string: 'st0r@gegrid'  # enclose with single quote
      profile: netapp-storagegrid
```

单个设备

/etc/datadog-agent/conf.d/snmp.d/conf.yaml

```
init_config:
  loader: core  # use core check implementation of SNMP integration.
recommended
  use_device_id_as_hostname: true  # recommended
instances:
- ip_address: '10.0.0.1'
  profile: netapp-storagegrid
  community_string: 'st0r@gegrid'  # enclose with single quote
- ip_address: '10.0.0.2'
  profile: netapp-storagegrid
  community_string: 'st0r@gegrid'
- ip_address: '10.0.0.3'
  profile: netapp-storagegrid
 community_string: 'st0r@gegrid'
- ip_address: '10.0.0.4'
  profile: netapp-storagegrid
  community_string: 'st0r@gegrid'
```

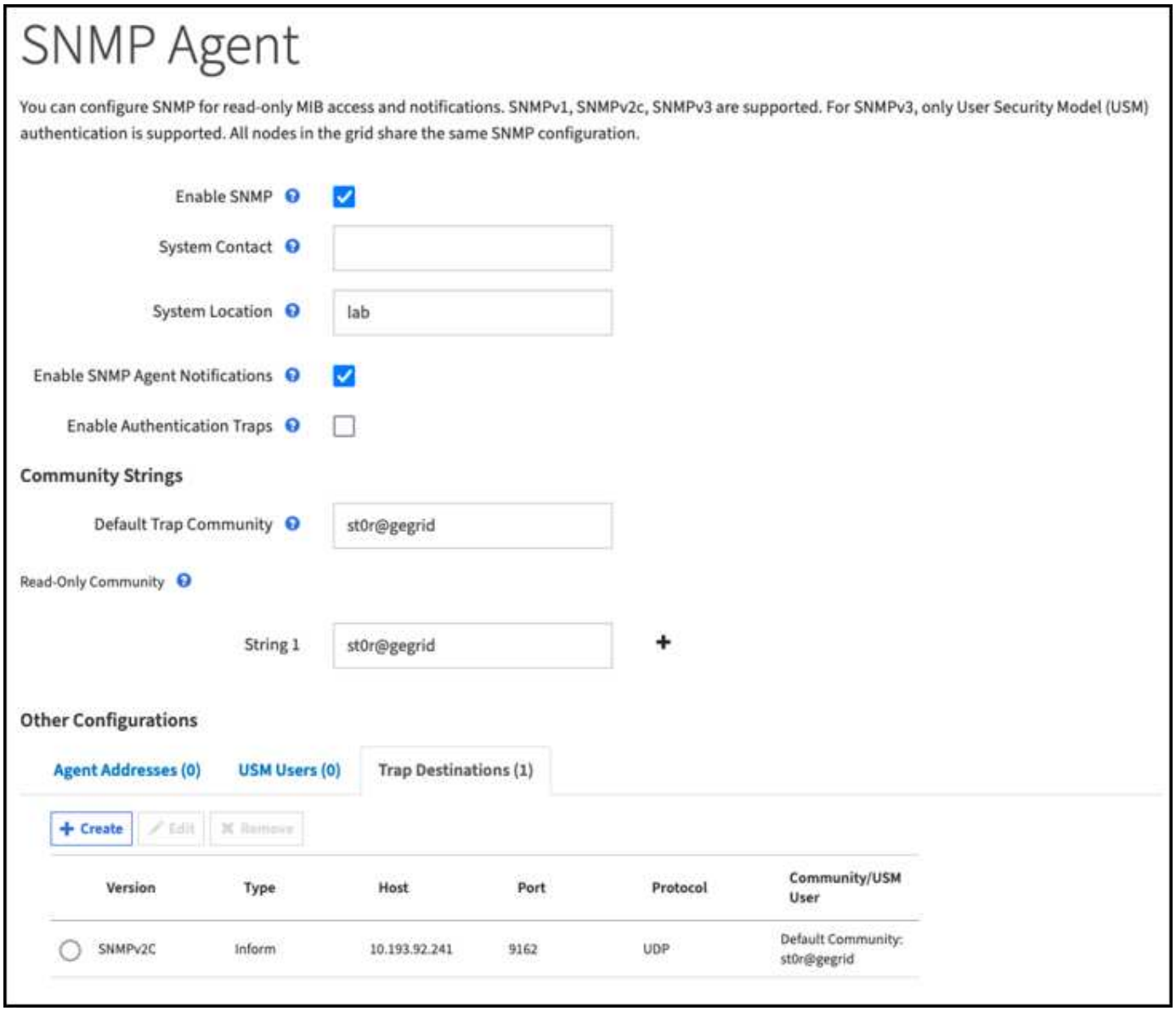
陷阱的**SNMP**配置

SNMP陷阱的配置在datadog配置yaml文件/etc/datadog-agent/datadog.yaml中定义

```
network_devices:
  namespace:  # optional, defaults to “default”.
  snmp_traps:
    enabled: true
    port: 9162 # on which ports to listen for traps
    community_strings: # which community strings to allow for v2 traps
      - st0r@gegrid
```

StorageGRID SNMP配置示例

StorageGRID 系统中的SNMP代理位于配置选项卡的监控列下。启用SNMP并输入所需信息。如果要配置陷阱、请选择"陷阱目标"并为包含陷阱配置的Datadog代理主机创建一个目标。



使用rclone在StorageGRID 上迁移、放置和删除对象

作者：Siegfried Hepp和Aron Klein_

rclone是一种用于S3操作的免费命令行工具和客户端。您可以使用rclone迁移、复制和删除StorageGRID 上的对象数据。rclone可以删除存储分段、即使不是空存储分段也可以使用"清除"功能、如以下示例所示。

安装和配置rclone

要在工作站或服务器上安装rclone、请从下载它 "rclone.org"。

初始配置步骤

1. 通过运行配置脚本或手动创建文件来创建rclone配置文件。
2. 在此示例中、我将使用sgdemo作为rclone配置中远程StorageGRID S3端点的名称。
   a. 创建配置文件~/.config/rclone/rclone.conf

```
[sgdemo]
type = s3
provider = Other
access_key_id = ABCDEFGH123456789JKL
secret_access_key = 123456789ABCDEFGHIJKLMN0123456789PQRST+V
endpoint = sgdemo.netapp.com
```

b. 运行rclone config

rclone配置#

```
2023/04/13 14:22:45 NOTICE: Config file
"/root/.config/rclone/rclone.conf" not found - using defaults
No remotes found - make a new one
n) New remote
s) Set configuration password
q) Quit config
n/s/q> n
name> sgdemo
```

```
Option Storage.
Type of storage to configure.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value.
 1 / 1Fichier
   \ "fichier"
 2 / Alias for an existing remote
   \ "alias"
 3 / Amazon Drive
   \ "amazon cloud drive"
 4 / Amazon S3 Compliant Storage Providers including AWS,
Alibaba, Ceph, Digital Ocean, Dreamhost, IBM COS, Minio,
SeaweedFS, and Tencent COS
   \ "s3"
 5 / Backblaze B2
   \ "b2"
 6 / Better checksums for other remotes
   \ "hasher"
 7 / Box
   \ "box"
 8 / Cache a remote
   \ "cache"
 9 / Citrix Sharefile
   \ "sharefile"
10 / Compress a remote
   \ "compress"
11 / Dropbox
   \ "dropbox"
12 / Encrypt/Decrypt a remote
   \ "crypt"
13 / Enterprise File Fabric
   \ "filefabric"
14 / FTP Connection
```

```
   \ "ftp"
15 / Google Cloud Storage (this is not Google Drive)
   \ "google cloud storage"
16 / Google Drive
   \ "drive"
17 / Google Photos
   \ "google photos"
18 / Hadoop distributed file system
   \ "hdfs"
19 / Hubic
   \ "hubic"
20 / In memory object storage system.
   \ "memory"
21 / Jottacloud
   \ "jottacloud"
22 / Koofr
   \ "koofr"
23 / Local Disk
   \ "local"
24 / Mail.ru Cloud
   \ "mailru"
25 / Mega
   \ "mega"
26 / Microsoft Azure Blob Storage
   \ "azureblob"
27 / Microsoft OneDrive
   \ "onedrive"
28 / OpenDrive
   \ "opendrive"
29 / OpenStack Swift (Rackspace Cloud Files, Memset Memstore,
OVH)
   \ "swift"
30 / Pcloud
   \ "pcloud"
31 / Put.io
   \ "putio"
32 / QingCloud Object Storage
   \ "qingstor"
33 / SSH/SFTP Connection
   \ "sftp"
34 / Sia Decentralized Cloud
   \ "sia"
35 / Sugarsync
   \ "sugarsync"
36 / Tardigrade Decentralized Cloud Storage
   \ "tardigrade"
```

```
37 / Transparently chunk/split large files
   \ "chunker"
38 / Union merges the contents of several upstream fs
   \ "union"
39 / Uptobox
   \ "uptobox"
40 / Webdav
   \ "webdav"
41 / Yandex Disk
   \ "yandex"
42 / Zoho
   \ "zoho"
43 / http Connection
   \ "http"
44 / premiumize.me
   \ "premiumizeme"
45 / seafile
   \ "seafile"
```

```
Storage> 4
```

```
Option provider.
Choose your S3 provider.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value.
 1 / Amazon Web Services (AWS) S3
   \ "AWS"
 2 / Alibaba Cloud Object Storage System (OSS) formerly Aliyun
   \ "Alibaba"
 3 / Ceph Object Storage
   \ "Ceph"
 4 / Digital Ocean Spaces
   \ "DigitalOcean"
 5 / Dreamhost DreamObjects
   \ "Dreamhost"
 6 / IBM COS S3
   \ "IBMCOS"
 7 / Minio Object Storage
   \ "Minio"
 8 / Netease Object Storage (NOS)
   \ "Netease"
 9 / Scaleway Object Storage
   \ "Scaleway"
10 / SeaweedFS S3
   \ "SeaweedFS"
11 / StackPath Object Storage
   \ "StackPath"
12 / Tencent Cloud Object Storage (COS)
   \ "TencentCOS"
13 / Wasabi Object Storage
   \ "Wasabi"
14 / Any other S3 compatible provider
   \ "Other"
provider> 14
```

```
Option env_auth.
Get AWS credentials from runtime (environment variables or
EC2/ECS meta data if no env vars).
Only applies if access_key_id and secret_access_key is blank.
Enter a boolean value (true or false). Press Enter for the
default ("false").
Choose a number from below, or type in your own value.
 1 / Enter AWS credentials in the next step.
   \ "false"
 2 / Get AWS credentials from the environment (env vars or IAM).
   \ "true"
env_auth> 1
```

```
Option access_key_id.
AWS Access Key ID.
Leave blank for anonymous access or runtime credentials.
Enter a string value. Press Enter for the default ("").
access_key_id> ABCDEFGH123456789JKL
```

```
Option secret_access_key.
AWS Secret Access Key (password).
Leave blank for anonymous access or runtime credentials.
Enter a string value. Press Enter for the default ("").
secret_access_key> 123456789ABCDEFGHIJKLMN0123456789PQRST+V
```

```
Option region.
Region to connect to.
Leave blank if you are using an S3 clone and you don't have a
region.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value.
   / Use this if unsure.
 1 | Will use v4 signatures and an empty region.
   \ ""
   / Use this only if v4 signatures don't work.
 2 | E.g. pre Jewel/v10 CEPH.
   \ "other-v2-signature"
region> 1
```

```
Option endpoint.
Endpoint for S3 API.
Required when using an S3 clone.
Enter a string value. Press Enter for the default ("").
endpoint> sgdemo.netapp.com
```

```
Option location_constraint.
Location constraint - must be set to match the Region.
Leave blank if not sure. Used when creating buckets only.
Enter a string value. Press Enter for the default ("").
location_constraint>
```

```
Option acl.
Canned ACL used when creating buckets and storing or copying
objects.
This ACL is used for creating objects and if bucket_acl isn't
set, for creating buckets too.
For more info visit
https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-
overview.html#canned-acl
Note that this ACL is applied when server-side copying objects as
S3
doesn't copy the ACL from the source but rather writes a fresh
one.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value.
   / Owner gets FULL_CONTROL.
 1 | No one else has access rights (default).
   \ "private"
   / Owner gets FULL_CONTROL.
 2 | The AllUsers group gets READ access.
   \ "public-read"
   / Owner gets FULL_CONTROL.
 3 | The AllUsers group gets READ and WRITE access.
   | Granting this on a bucket is generally not recommended.
   \ "public-read-write"
   / Owner gets FULL_CONTROL.
 4 | The AuthenticatedUsers group gets READ access.
   \ "authenticated-read"
   / Object owner gets FULL_CONTROL.
 5 | Bucket owner gets READ access.
   | If you specify this canned ACL when creating a bucket,
Amazon S3 ignores it.
   \ "bucket-owner-read"
   / Both the object owner and the bucket owner get FULL_CONTROL
over the object.
 6 | If you specify this canned ACL when creating a bucket,
Amazon S3 ignores it.
   \ "bucket-owner-full-control"
acl>
```

```
Edit advanced config?
y) Yes
n) No (default)
y/n> n
```

```
--------------------
[sgdemo]
type = s3
provider = Other
access_key_id = ABCDEFGH123456789JKL
secret_access_key = 123456789ABCDEFGHIJKLMN0123456789PQRST+V
endpoint = sgdemo.netapp.com:443
--------------------
y) Yes this is OK (default)
e) Edit this remote
d) Delete this remote
y/e/d>
```

```
Current remotes:
```

```
Name                 Type
====                 ====
sgdemo               s3
```

```
e) Edit existing remote
n) New remote
d) Delete remote
r) Rename remote
c) Copy remote
s) Set configuration password
q) Quit config
e/n/d/r/c/s/q> q
```

基本命令示例

- 创建存储分段：

  ```
  rclone mkdir remote:bucket
  ```

  ```
  # rclone mkdir sgdemo：test01
  ```

  如果需要忽略SSL证书、请使用-no-check-certificate。

- 列出所有分段：

```
rclone lsd remote:
```

rclone LSD sgdemo数：

- 列出特定分段中的对象：

```
rclone ls remote:bucket
```

rclone ls sgdemo：test01

```
    65536 TestObject.0
    65536 TestObject.1
    65536 TestObject.10
    65536 TestObject.12
    65536 TestObject.13
    65536 TestObject.14
    65536 TestObject.15
    65536 TestObject.16
    65536 TestObject.17
    65536 TestObject.18
    65536 TestObject.2
    65536 TestObject.3
    65536 TestObject.5
    65536 TestObject.6
    65536 TestObject.7
    65536 TestObject.8
    65536 TestObject.9
  33554432 bigobj
      102 key.json
       47 locked01.txt
4294967296 sequential-read.0.0
       15 test.txt
      116 version.txt
```

- 删除分段：

```
rclone rmdir remote:bucket
```

rclone rmdir sgdemo：test02

- 放置对象：

```
rclone copy filename remote:bucket
```

rclone copy ~/test/testfile.txt sgdemo：test01

- 获取对象：

```
rclone copy remote:bucket/objectname filename
```

rclone copy sgdemo：test01/testfile.txt ~/test/testfileS3.txt

- 删除对象：

```
rclone delete remote:bucket/objectname
```

rclone delete sgdemo：test01/testfile.txt

- 迁移存储分段中的对象

```
rclone sync source:bucket destination:bucket --progress

rclone sync source_directory destination:bucket --progress
```

rclone sync sgdemo：test01 sgdemo：clone01—-progress

```
Transferred:        4.032 GiB / 4.032 GiB, 100%, 95.484 KiB/s, ETA
0s
Transferred:           22 / 22, 100%
Elapsed time:       1m4.2s
```

使用-progress或-P显示任务的进度。否则、不会显示任何输出。

- 删除分段和所有对象内容

```
rclone purge remote:bucket --progress
```

rclone purge sgdemo：test01 -progress

```
Transferred:              0 B / 0 B, -, 0 B/s, ETA -
Checks:                46 / 46, 100%
Deleted:               23 (files), 1 (dirs)
Elapsed time:         10.2s
```

rclone ls sgdemo：test01

```
2023/04/14 09:40:51 Failed to ls: directory not found
```

使用Veeam备份和复制进行部署的StorageGRID最佳实践

作者：*Oliver Haensel*和*Aron Klein*

本指南重点介绍NetApp StorageGRID以及部分Veeam备份和复制的配置。本白皮书面向熟悉Linux系统并负责维护或实施NetApp StorageGRID系统与Veeam备份和复制的存储和网络管理员。

概述

存储管理员希望通过满足可用性、快速恢复目标、可扩展以满足其需求以及自动执行长期数据保留策略的解决方案来管理数据增长。这些解决方案还应提供保护、防止丢失或恶意攻击。Veeam和NetApp合作创建了一个将Veeam备份和恢复与NetApp StorageGRID相结合的数据保护解决方案、用于内部对象存储。

Veeam和NetApp StorageGRID提供了一个易于使用的解决方案、可协同工作、帮助满足全球数据快速增长和法规不断增长的需求。基于云的对象存储因其弹性、扩展能力、运营效率和成本效益而闻名、这使其成为备份目标的自然选择。本文档将为Veeam Backup解决方案和StorageGRID系统的配置提供指导和建议。

Veeam的对象工作负载会为小型对象创建大量并发放置、删除和列表操作。启用不可迁移性将增加对对象存储的请求数量、以设置保留和列出版本。备份作业的过程包括为每日更改写入对象、新写入完成后、该作业将根据备份的保留策略删除任何对象。备份作业的计划几乎总是重叠的。这种重叠将导致备份窗口的很大一部分在对象存储上包含50/50的放置/删除工作负载。在Veeam中调整任务插槽设置的并发操作数、通过增加备份作业块大小来增加对象大小、减少多对象删除请求中的对象数量、 选择完成作业的最长时间窗口将优化解决方案的性能和成本。

请务必阅读产品文档 "Veeam备份和复制"和 "StorageGRID"在你开始之前。 Veeam 提供了计算器，用于了解 Veeam 基础架构的大小和容量要求，这些计算器应在确定StorageGRID解决方案的大小之前使用。请始终在 Veeam Ready Program 网站上查看 Veeam- NetApp验证的配置，以了解 "Veeam Ready对象、对象不可变性和存储库"。

Veeam配置

建议版本

建议始终保持最新、并为Veeam Backup & Replication 12或12.1系统应用最新的修补程序。目前、我们建议至

少安装Veeam 12修补程序P20230718。

S3存储库配置

横向扩展备份存储库(SOBR)是S3对象存储的容量层。容量层是主存储库的扩展、可提供更长的数据保留期限和更低的存储解决方案成本。Veeam能够通过S3对象锁定API提供不可变功能。Veeam 12可以在一个横向扩展存储库中使用多个分段。StorageGRID对单个存储分段中的对象数量或容量没有限制。使用多个分段可以提高备份非常大的数据集时的性能、在这些数据集中、对象中的备份数据可能会达到PB级。

可能需要限制并发任务、具体取决于特定解决方案的规模估算和要求。默认设置为每个CPU核心指定一个存储库任务插槽、并发任务插槽限制为64。例如，如果服务器有2个CPU核心，则对象存储总共将使用128个并发线程。这包括Put、GET和批删除。建议先为任务时隙选择一个保守限制、并在Veeam备份达到新备份和即将过期备份数据的稳定状态后调整此值。请与您的NetApp客户团队合作、适当估算StorageGRID系统的规模、以满足所需的时间窗口和性能要求。要提供最佳解决方案、可能需要调整任务插槽数量和每个插槽的任务限制。

备份作业配置

Veeam备份作业可以使用不同的块大小选项进行配置、应仔细考虑这些选项。默认块大小为1 MB、而Veeam具有数据压缩和重复数据删除功能、可为初始完整备份创建大约500 KB的对象大小、为增量作业创建100-200 KB的对象大小。通过选择更大的备份块大小、我们可以大幅提高性能并降低对象存储的要求。尽管较大的块大小可以显著提高对象存储的性能、但由于存储效率性能降低、可能会增加主存储容量需求。建议为备份作业配置4 MB的块大小、以便为完整备份创建大约2 MB的对象、并为增量备份创建大约700 kB-1 MB的对象大小。客户甚至可以考虑使用8 MB块大小配置备份作业、此功能可在Veeam支持人员的协助下启用。

实施不可配置备份时会使用对象存储上的S3对象锁定。不可更改性选项会生成更多的对象存储请求、以更新对象的列表和保留。

随着备份保留过期、备份作业将处理对象删除。Veeam以多对象删除请求的形式将删除请求发送到对象存储、每个请求1000个对象。对于小型解决方案、可能需要对此进行调整、以减少每个请求的对象数量。降低此值还可以更均匀地在StorageGRID系统中的节点之间分布删除请求。建议使用下表中的值作为配置多对象删除限制的起点。将表中的值乘以所选设备类型的节点数、即可获得Veeam中设置的值。如果此值等于或大于1000、则无需调整默认值。如果需要调整此值、请与Veeam支持部门合作进行更改。

设备型号	每个节点的**S3MultiObjectDeleteLimit**
SG5712	34.
SG5760	75
SG6060	200

请与您的NetApp客户团队合作、根据您的特定需求确定建议的配置。Veeam配置设置建议包括：

- 备份作业块大小= 4 MB
- SOBR任务插槽限制为2-16
- 多对象删除限制= 34 - 1000

StorageGRID配置

建议版本

对于 Veeam 部署，建议使用带有最新修补程序的NetApp StorageGRID 11.9 或 12.0 版本。始终建议保持最新

状态并为StorageGRID系统应用最新的修补程序。

负载平衡器和S3端点配置

Veeam仅要求通过HTTPS连接端点。Veeam不支持非加密连接。SSL证书可以是自签名证书、专用可信证书颁发机构或公共可信证书颁发机构。为了确保持续访问S3存储库、建议在HA配置中至少使用两个负载平衡器。负载平衡器可以是位于每个管理节点和网关节点上的StorageGRID提供的集成负载平衡器服务、也可以是位于F5、Kemp、HAProxy、Loadbalancer.org等第三方解决方案上的集成负载平衡器服务 通过使用StorageGRID负载平衡器、可以设置流量划分器(QoS规则)、以便确定Veeam工作负载的优先级、或者限制Veeam、使其不会影响StorageGRID系统上优先级较高的工作负载。

S3 存储分段

StorageGRID是一个安全的多租户存储系统。建议为 Veeam 工作负载创建专用租户。可以选择分配存储配额。作为最佳实践，启用“使用自己的身份源”。使用适当的密码保护租户根管理用户。 Veeam Backup 12 要求 S3 存储桶具有强一致性。 StorageGRID提供在存储桶级别配置的多个一致性选项。对于 Veeam 从多个位置访问数据的多站点部署，请选择“strong-global”。如果 Veeam 备份和恢复仅在单个站点进行，则一致性级别应设置为“strong-site”。有关存储桶一致性级别的更多信息，请查看 "文档。"。要使用StorageGRID进行 Veeam 不变性备份，必须在全局启用 S3 对象锁，并在存储桶创建期间在存储桶上进行配置。

生命周期管理

StorageGRID支持在StorageGRID节点和站点之间进行复制和纠删编码、以实现对象级保护。纠删编码至少需要200 KB的对象大小。Veeam的默认块大小为1 MB、所产生的对象大小通常会低于Veeam存储效率后建议的200 KB最小大小。为了提高解决方案的性能、建议不要使用跨越多个站点的纠删编码配置文件、除非这些站点之间的连接足以避免增加延迟或限制StorageGRID系统的带宽。在多站点StorageGRID系统中、可以将ILM规则配置为在每个站点存储一个副本。为了获得最佳持久性、可以配置一条规则、以便在每个站点上存储一个经过删除的编码副本。对于此工作负载、最建议使用Veeam备份服务器本地的两个副本。

删除性能

Veeam 提供删除请求率调整和备份删除过程的调度。为了进一步调整删除性能，您可以禁用同步删除并让 ILM 扫描器管理对象的最终删除。

禁用同步删除的步骤

1. 打开StorageGRID网格管理器。
2. 在右上角，选择问号，然后选择 API 文档。
3. 在右上角，单击私有 API 文档页面链接。
4. 扩展 ilm-advanced。
5. 选择 GET ilm-advanced。
6. 选择“试用”，然后“执行”。
7. 查看响应结果。
    a. 如果值为空，则表示正在使用默认的 ilm-advanced 值。
    b. 如果值不为空，则表示自定义 ILM 高级值正在使用中。复制“数据”之后的所有输出：，从 { 开始直到倒数第二个 }。
        i. 将其保存在某些文本编辑器中。

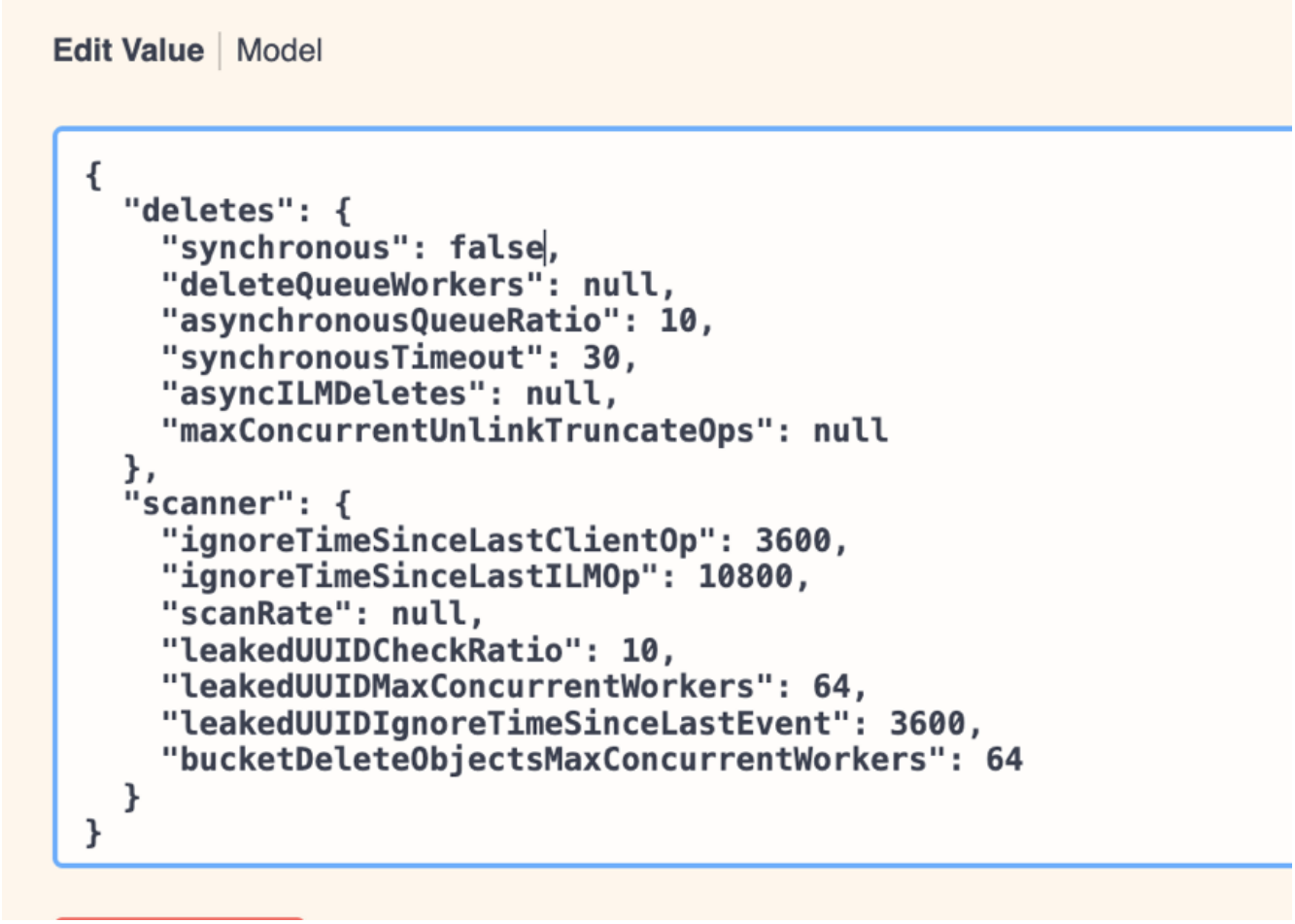
响应示例：

Response body


```
{
  "responseTime": "2025-09-19T15:01:28.142Z",
  "status": "success",
  "apiVersion": "4.2",
  "data": {
    "deletes": {
      "synchronous": null,
      "deleteQueueWorkers": null,
      "asynchronousQueueRatio": null,
      "synchronousTimeout": null,
      "asyncILMDeletes": null,
      "maxConcurrentUnlinkTruncateOps": null
    },
    "scanner": {
      "ignoreTimeSinceLastClientOp": null,
      "ignoreTimeSinceLastILMOp": null,
      "scanRate": null,
      "leakedUUIDCheckRatio": null,
      "leakedUUIDMaxConcurrentWorkers": null,
      "leakedUUIDIgnoreTimeSinceLastEvent": null,
      "bucketDeleteObjectsMaxConcurrentWorkers": null
    }
  }
}
```


8. 选择 PUT ilm-advanced。
9. 选择“试用”开始编辑 API 主体。
    a. 默认情况下，API 主体将包含默认值，而不包含任何先前配置的自定义值。这就是执行步骤 5-7 非常重要的原因。
10. 如果在步骤 5-7 中发现非默认值，则用步骤 7 中保存的输出替换 API 主体。。否则，如果步骤 5-7 中的值为空，则保留 API 主体不变。
11. 在API主体框中调整以下参数：
    a. 将同步值设置为false。

    API 正文示例：

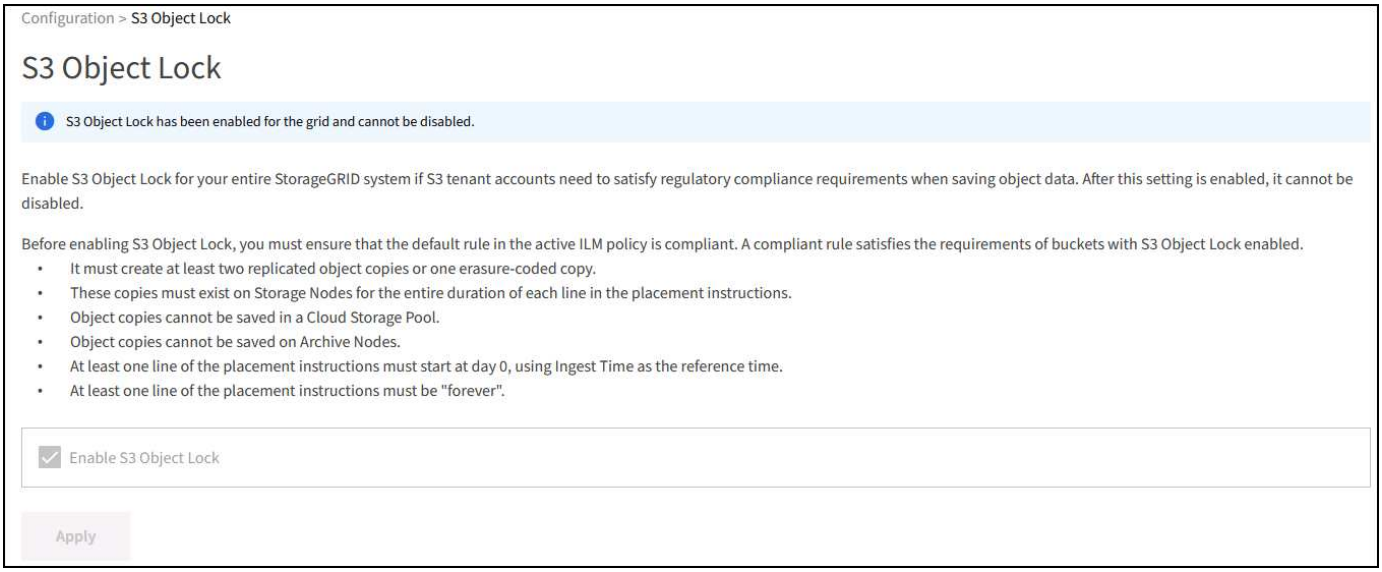


12. 完成后，选择执行

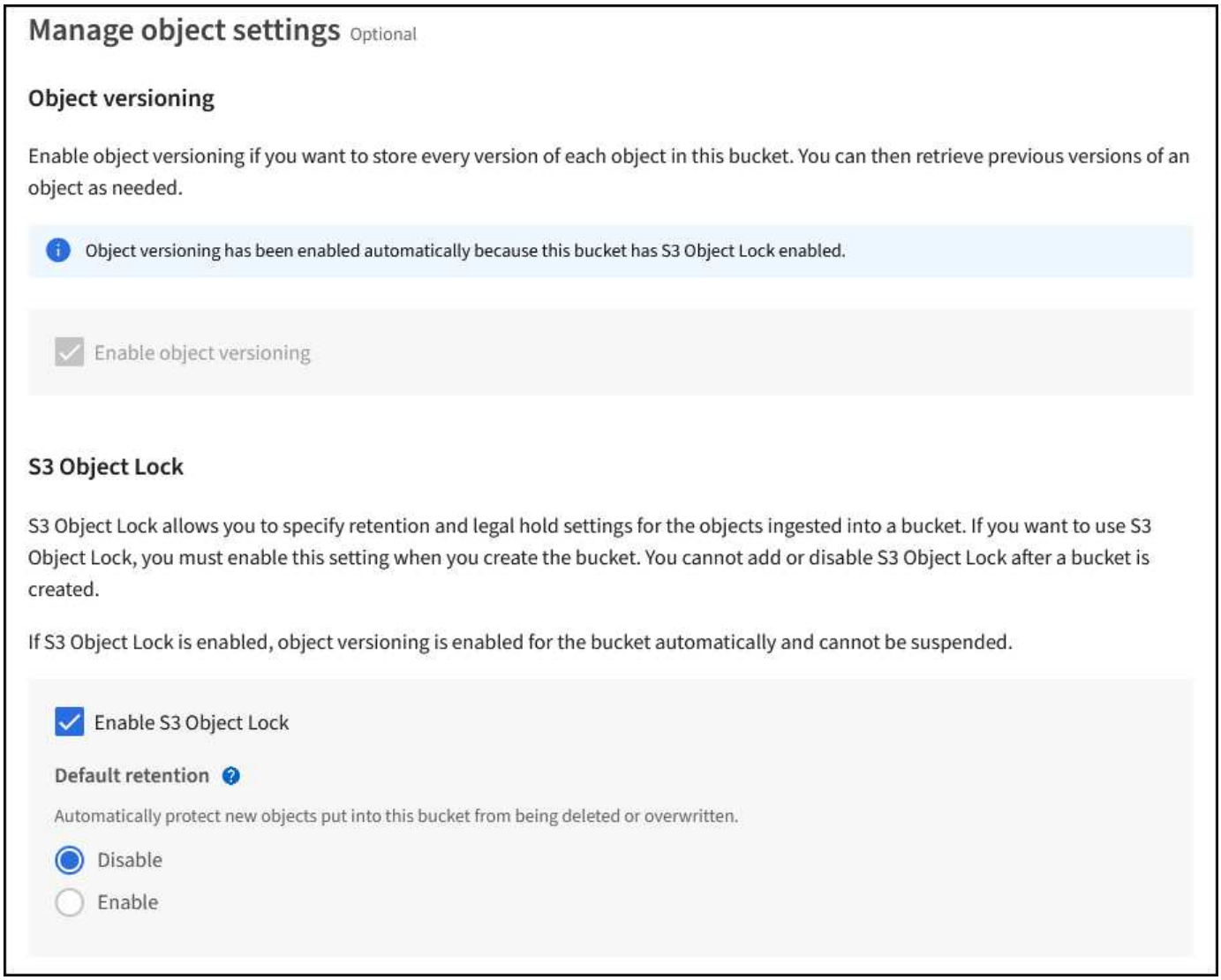
实施要点

StorageGRID

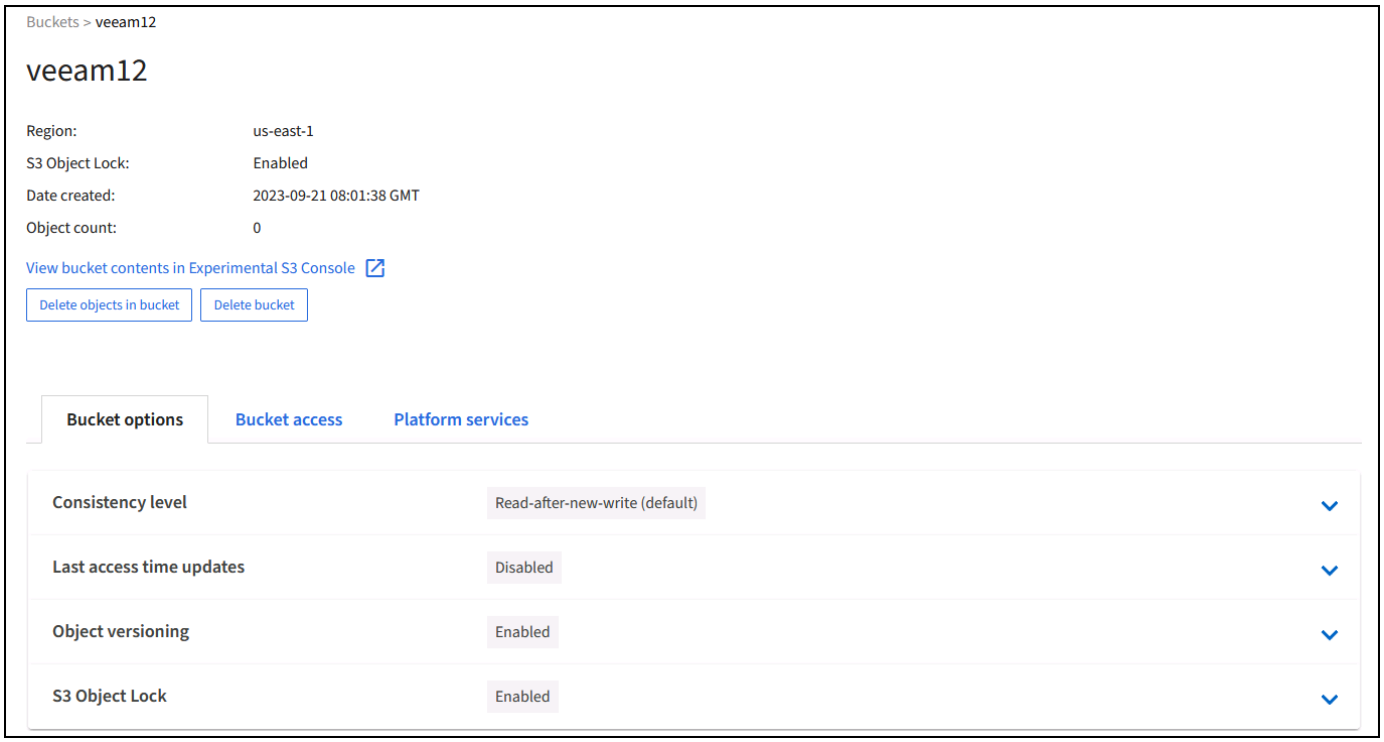
如果需要不可破坏性、请确保在StorageGRID系统上启用对象锁定。在管理UI中的"Configuration/S3 Object Lock"(配置/S3对象锁定)下找到相应选项。



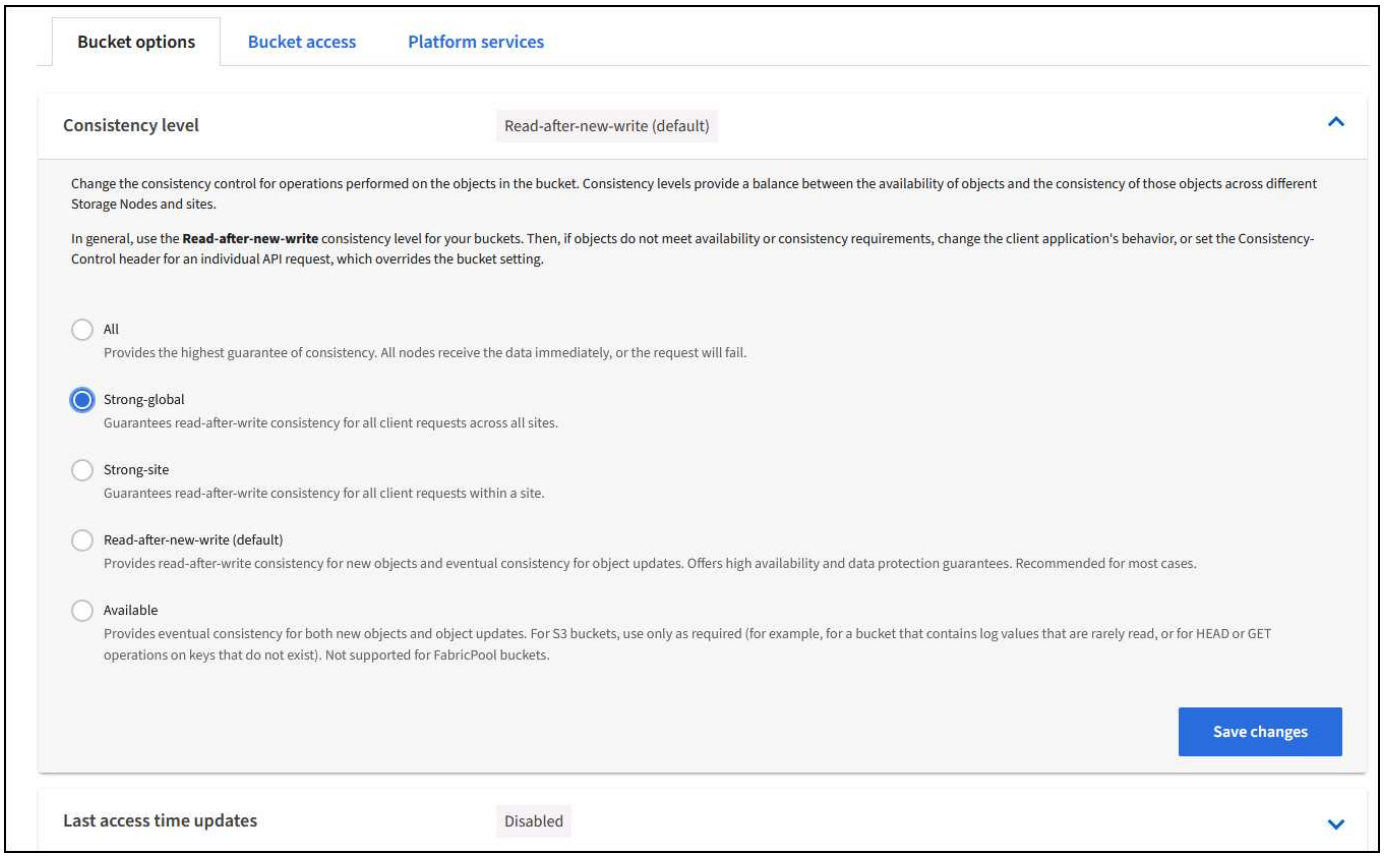
创建存储分段时、如果要将此存储分段用于不可移动备份、请选择"Enable S3 Object Lock"(启用S3对象锁定)。这将自动启用存储分段版本控制。保持禁用默认保留、因为Veeam将明确设置对象保留。如果Veeam不创建不可变备份、则不应选择版本控制和S3对象锁定。



创建存储分段后、转到所创建存储分段的详细信息页面。选择一致性级别。



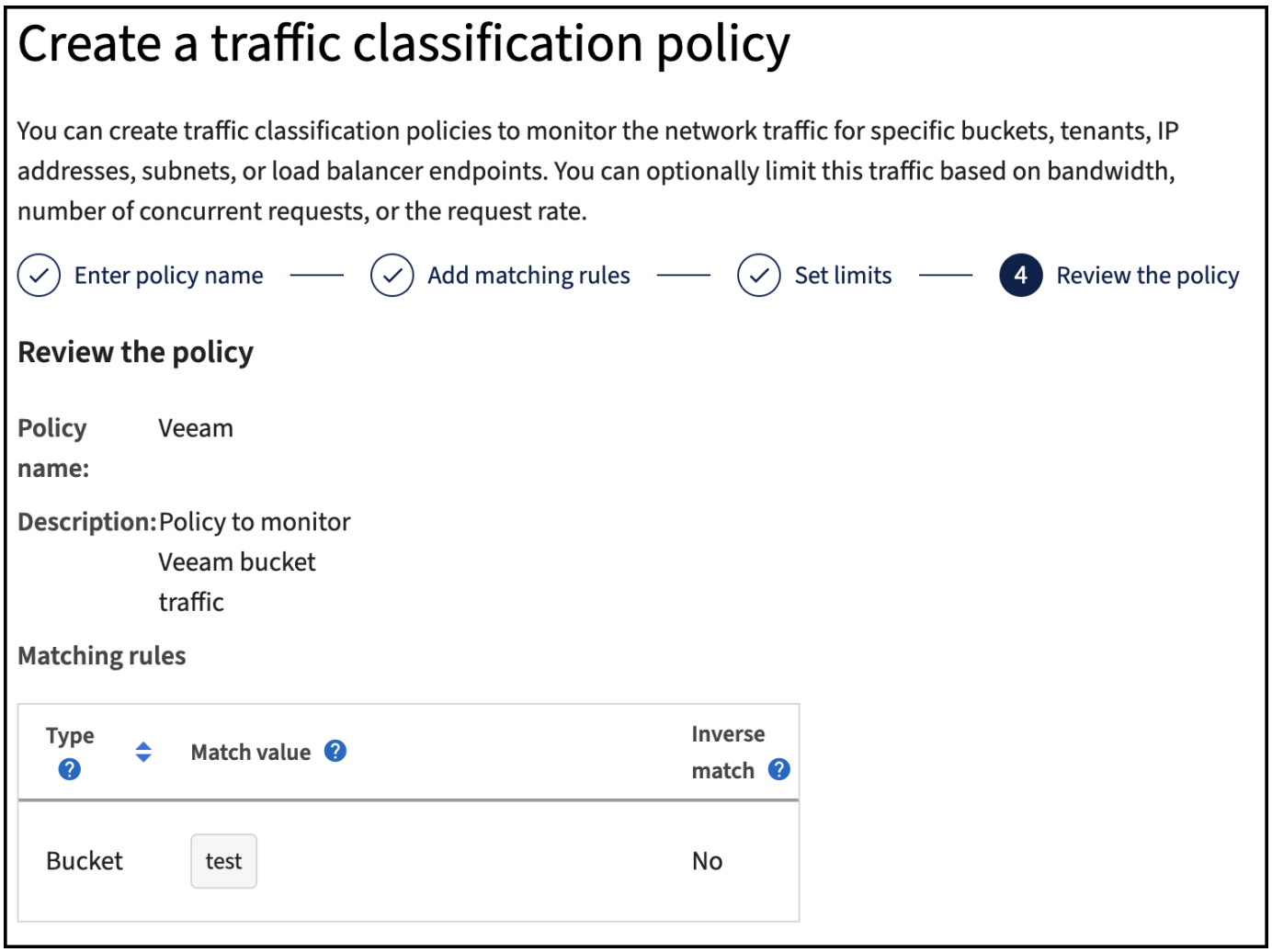
Veeam要求S3存储分段具有强大的一致性。因此、对于Veeam从多个位置访问数据的多站点部署、请选择"strong-globation"。如果Veeam备份和还原仅在单个站点上进行、则一致性级别应设置为"strong-site"。保存更改。



StorageGRID在每个管理节点和专用网关节点上提供集成负载平衡器服务。使用此负载平衡器的众多优势之一是能够配置流量分类策略(QoS)。虽然这些指标主要用于限制应用程序对其他客户端工作负载的影响或将工作负载

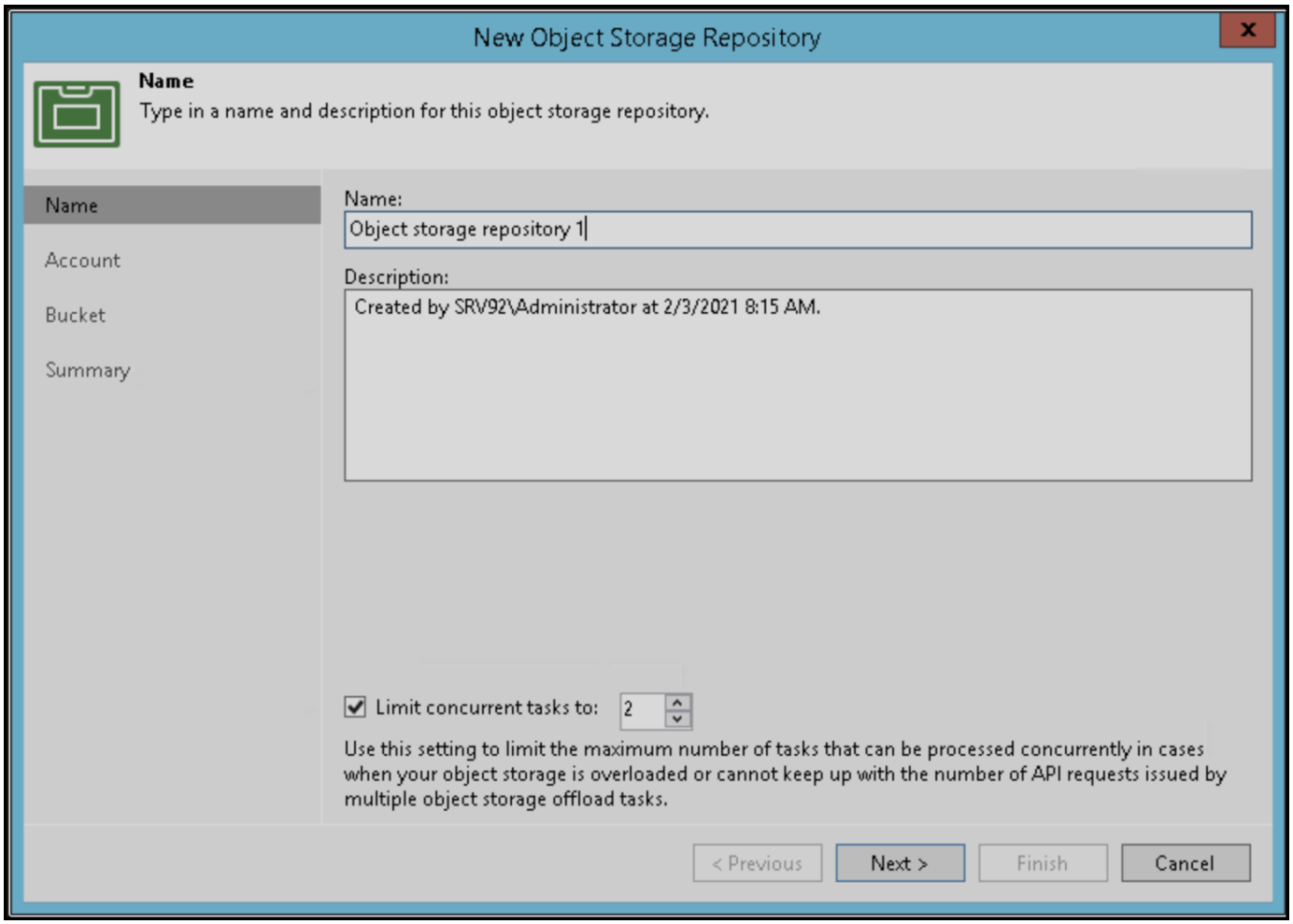
划分为优先级、但它们还提供了额外的指标收集功能、以协助监控。

在配置选项卡中、选择"Traffic Classification"(流量分类)并创建新策略。命名规则并选择存储分段或租户作为类型。输入存储分段或租户的名称。如果需要QoS、请设置限制、但对于大多数实施、我们只希望添加此功能提供的监控优势、因此不要设置限制。

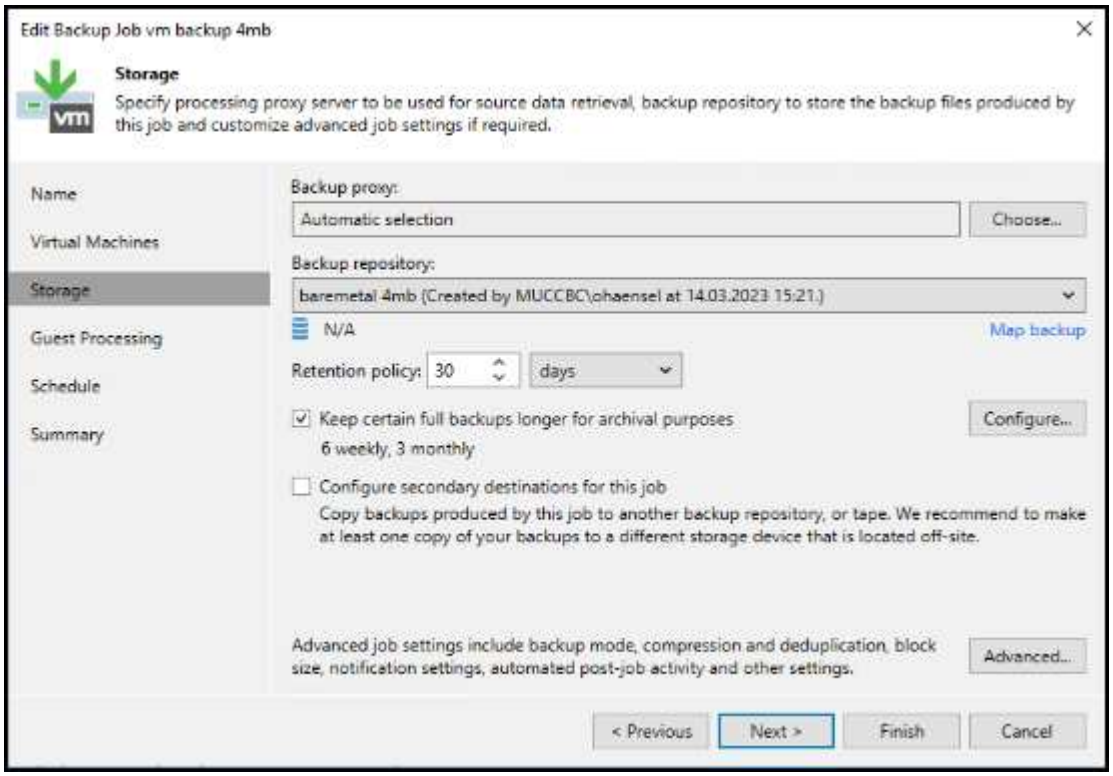


Veeam

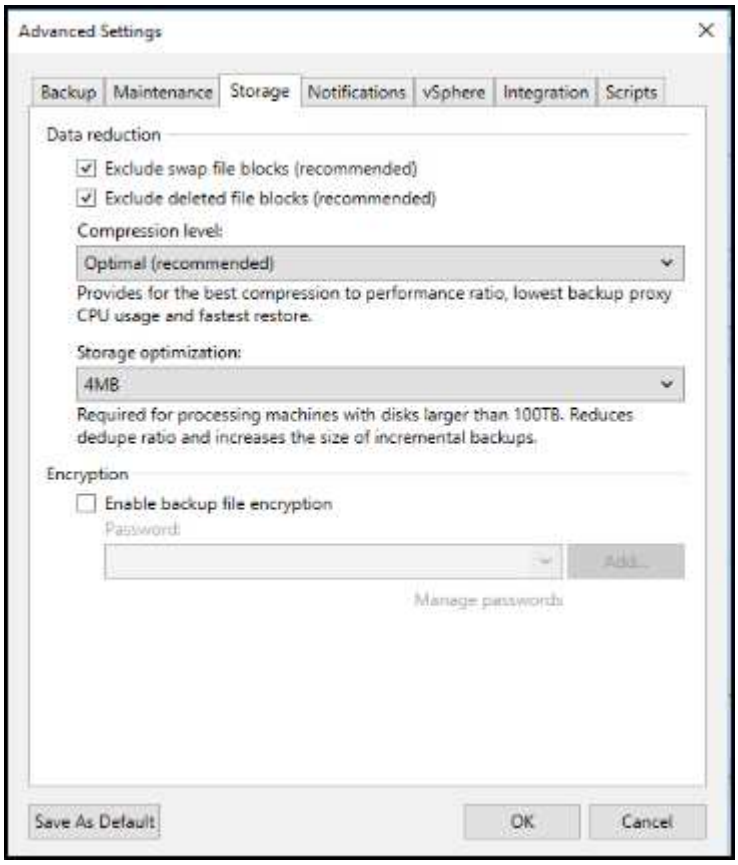
根据StorageGRID设备的型号和数量、可能需要选择并配置对存储分段上的并发操作数的限制。



按照Veeam控制台中有关备份作业配置的Veeam文档启动向导。添加VM后、选择SOBR存储库。



单击高级设置并将存储优化设置更改为4 MB或更大。应启用数据压缩和重复数据删除。根据需要更改子系统设置并配置备份作业计划。



监控StorageGRID

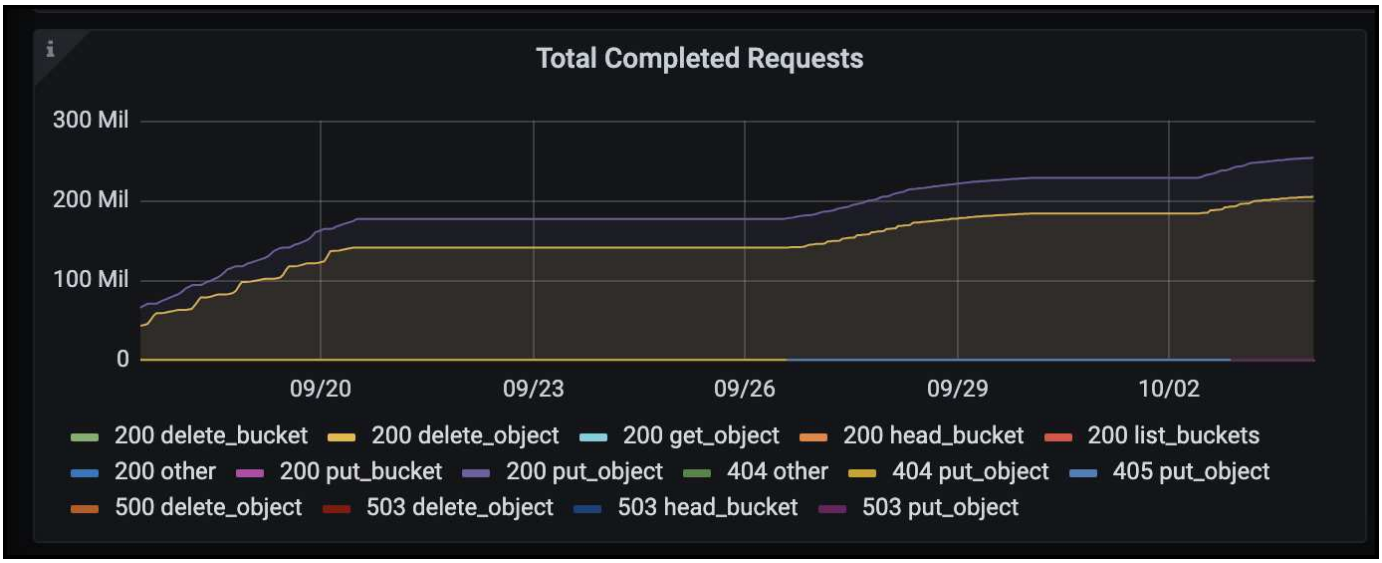
要全面了解Veeam和StorageGRID的协同运行情况、您需要等待第一个备份的保留时间到期。到目前为止、Veeam工作负载主要由Put操作组成、尚未执行任何删除操作。备份数据过期并进行清理后、您现在可以在对象存储中看到完全一致的使用情况、并根据需要调整Veeam中的设置。

StorageGRID在"Support"(支持)选项卡"Metrics (指标)"页面中提供了方便的图表来监控系统的运行。要查看的主要信息板是S3概述、ILM和流量分类策略(如果已创建策略)。在"S3概述"信息板中、您可以找到有关S3操作速率、延期和请求响应的信息。

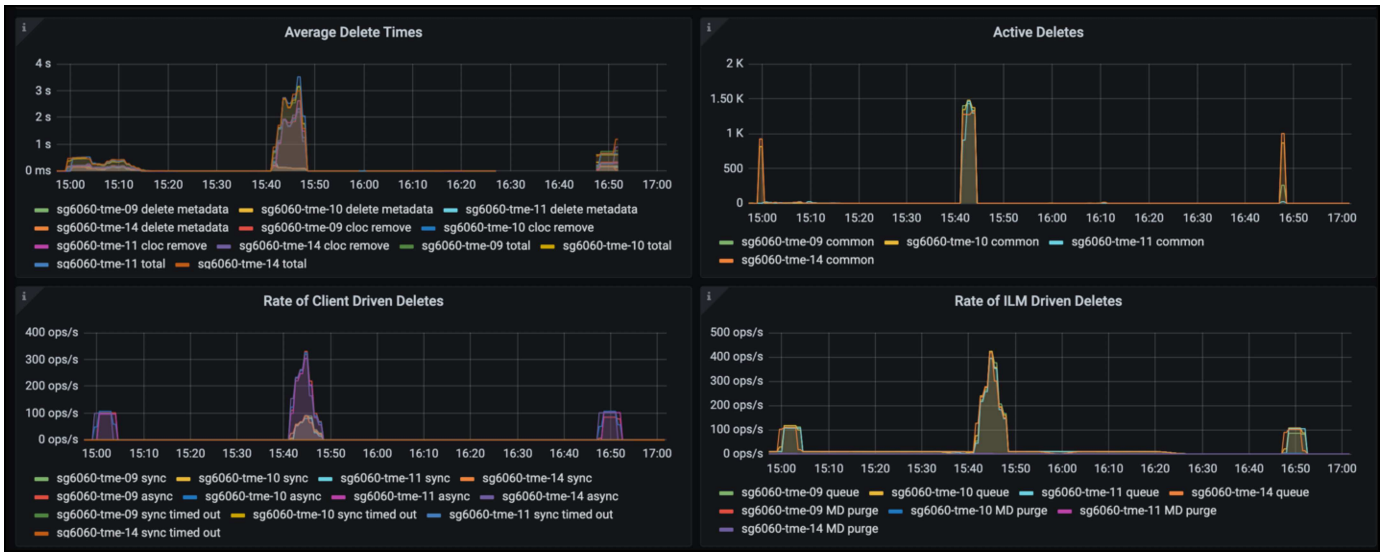
通过查看S3速率和活动请求、您可以按类型查看每个节点正在处理的负载以及请求总数。

"平均持续时间"图表显示每个节点针对每种请求类型花费的平均时间。这是请求的平均延迟、可能很好地指示可能需要进行额外调整、或者StorageGRID系统有承担更多负载的空间。

在“已完成请求总数”图表中，您可以按类型和响应代码查看请求。如果您看到的响应不是200 (OK)、则可能表示问题描述(如StorageGRID系统)负载过重、正在发送503 (减慢)响应、可能需要进行一些额外调整、或者现在是扩展系统以应对增加的负载的时候了。



在ILM信息板中、您可以监控StorageGRID系统的删除性能。StorageGRID会在每个节点上同时执行同步和异步删除、以尝试优化所有请求的整体性能。



通过流量分类策略、我们可以查看有关负载平衡器请求吞吐量、速率、持续时间以及Veeam正在发送和接收的对象大小的指标。

从何处查找追加信息

要了解有关本文档中所述信息的更多信息，请查看以下文档和 / 或网站：

- "NetApp StorageGRID产品文档"
- "Veeam备份和复制"

使用StorageGRID配置d不良 数据源

作者：郑安杰

多米奥支持多种数据源、包括基于云的或内部对象存储。 您可以将d不良 配置为使用StorageGRID作为对象存储数据源。

配置d不良 数据源

前提条件

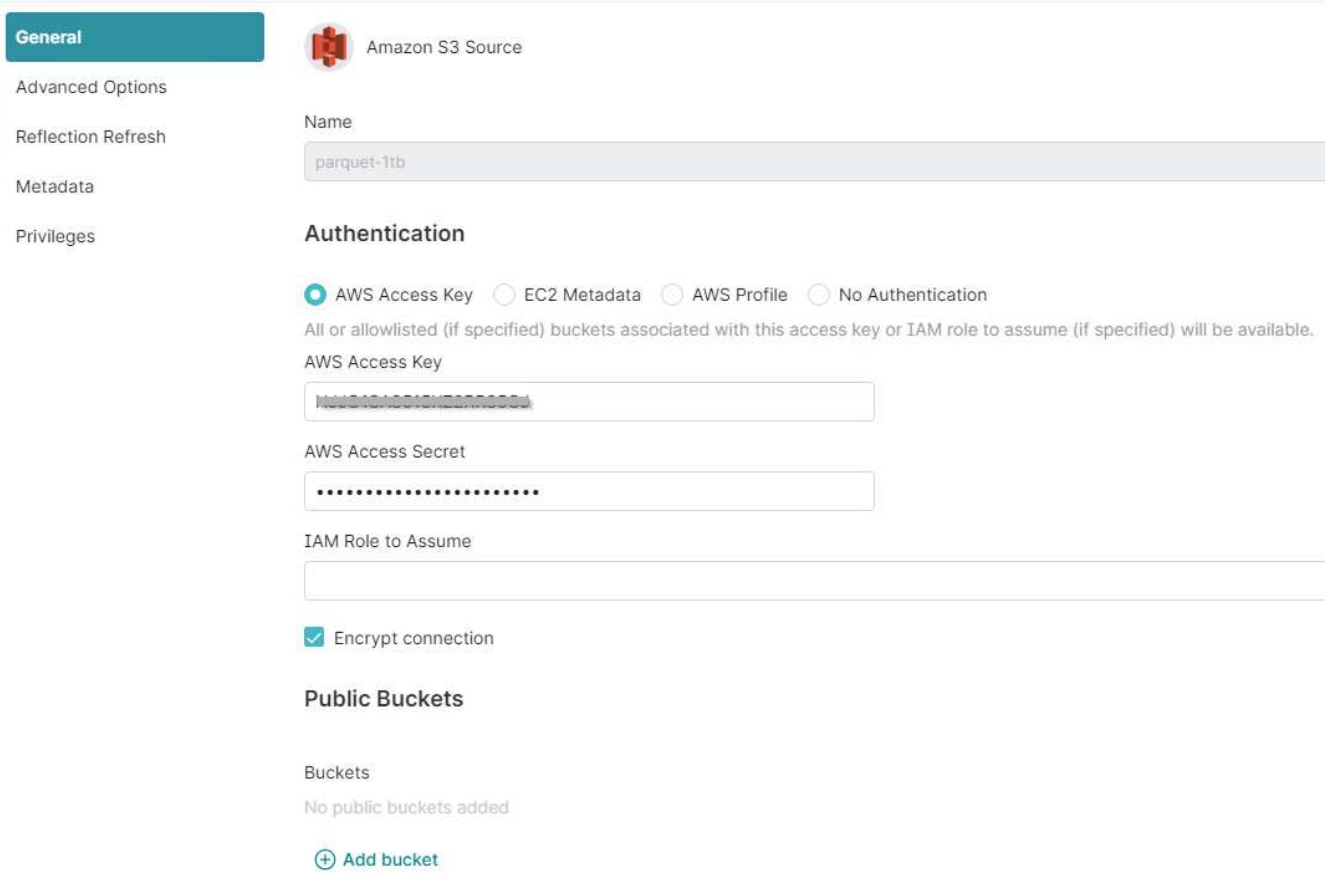
- StorageGRID S3端点URL、租户S3访问密钥ID和机密访问密钥。
- StorageGRID配置建议：禁用数据压缩(默认情况下处于禁用状态)。
  DERMIO使用字节范围GET在查询期间同时从同一对象中提取不同的字节范围。 字节范围请求的典型大小为1 MB。经过压缩的对象会降低字节范围GET性能。

d不良 指南

"连接到Amazon S3 -配置S3兼容存储"。

说明

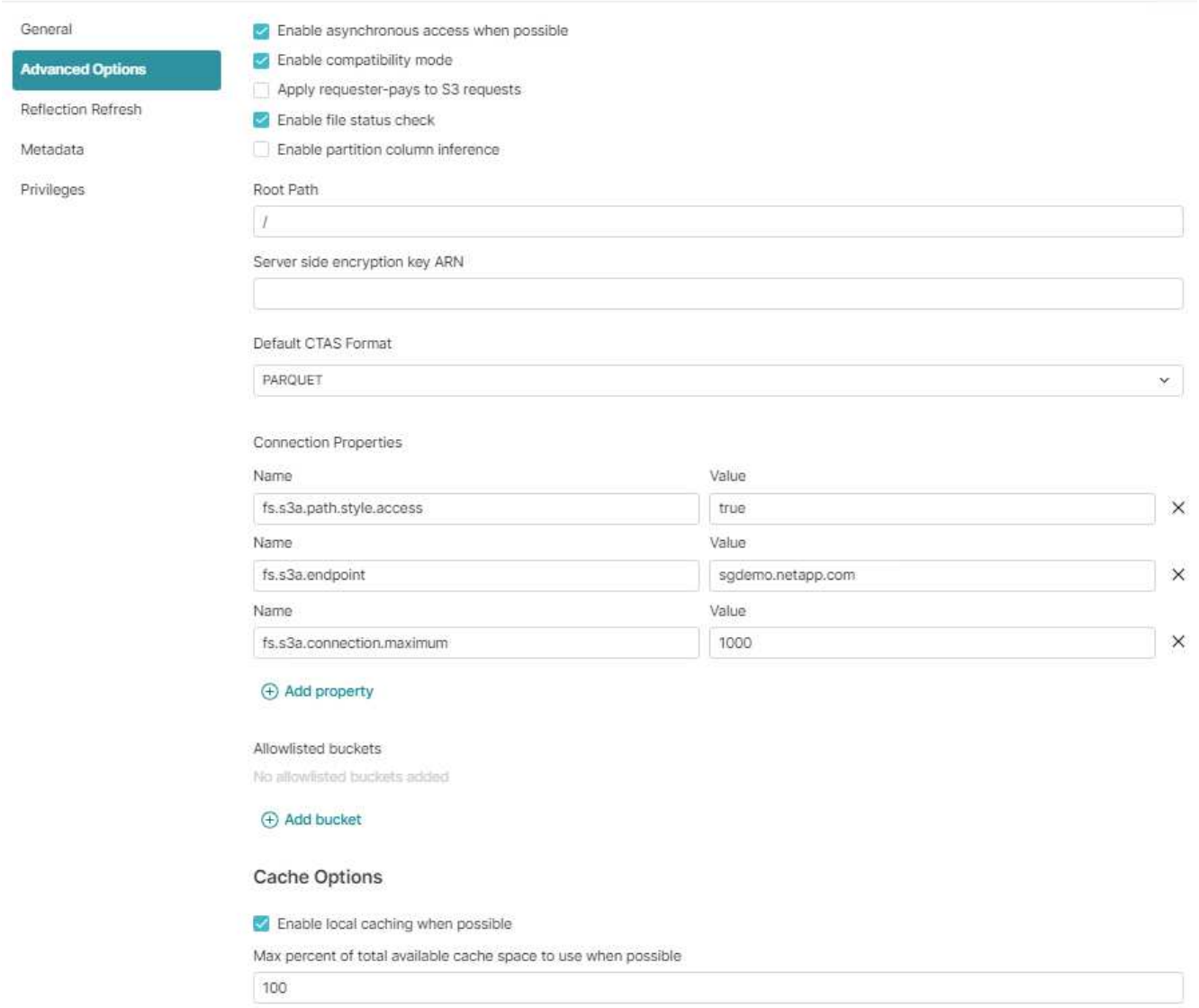
1. 在"Desmio数据集"页面上、单击+符号以添加源、然后选择"Amazon S3"。
2. 输入此新数据源的名称、StorageGRID S3租户访问密钥ID和机密访问密钥。
3. 如果使用https连接到StorageGRID S3端点、请选中"加密连接"复选框。+
   如果对此S3端点使用自签名CA证书、请按照dremio指南说明将此CA证书添加到<JAVA_HOME>服务器的dre/jre/lib/security +中
   屏幕截图示例



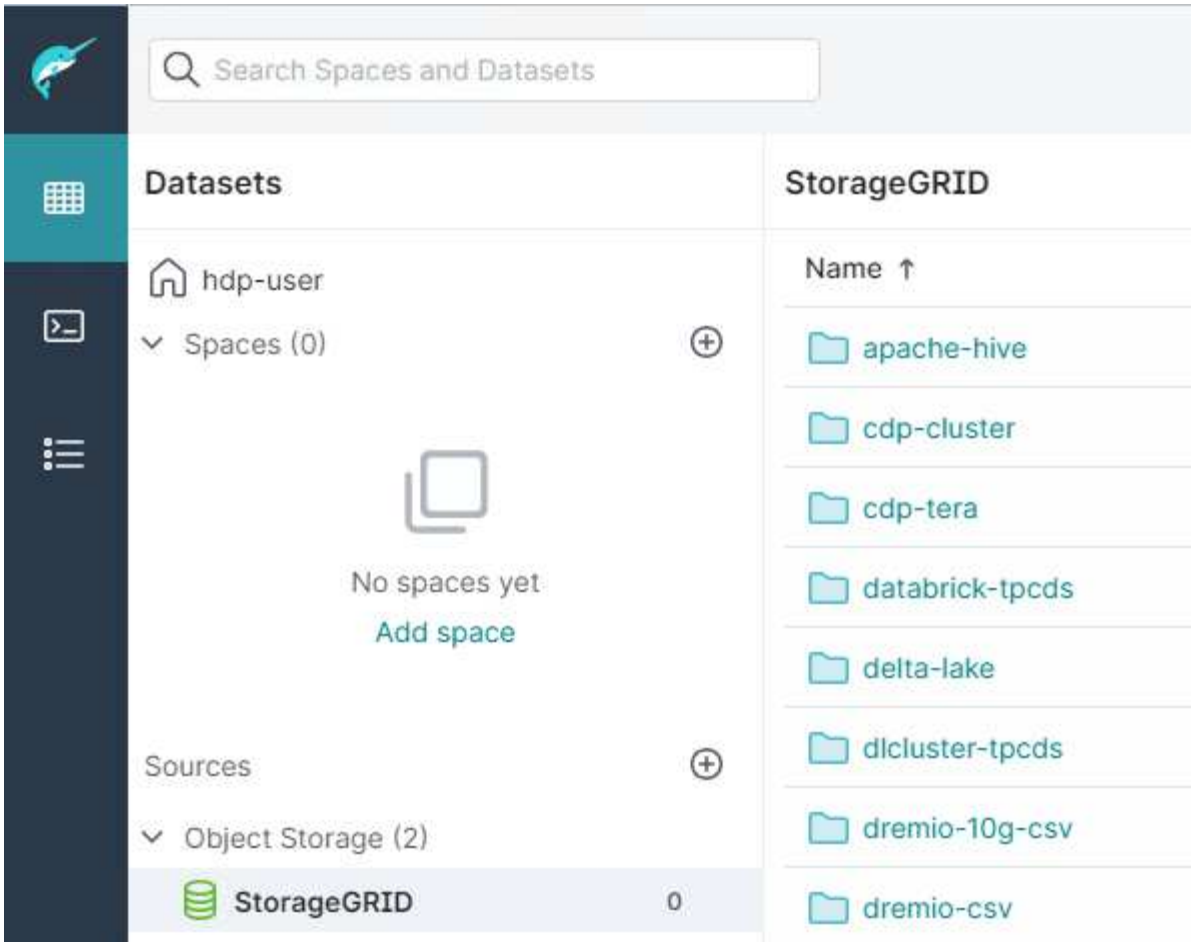
4. 单击"高级选项"、选中"启用兼容模式"
5. 在连接属性下、单击+添加属性并添加这些S3A属性。
6. fs.s3a.connection．maximum默认值为100。 如果S3数据集包含具有100个或更多列的大型镶木地板文件、则必须输入一个大于100的值。 有关此设置的信息、请参见《drefio指南》。

Name	价值
fs.s3a.endpoint	StorageGRID S3端点：端口>_
fs.s3a.path.style.access	true
FS.S3A.CONNECTION。最大值	_<大于100>_的值

屏幕截图示例



7. 根据您的组织或应用程序要求配置其他的多米奥选项。
8. 单击保存按钮以创建此新数据源。
9. 成功添加StorageGRID数据源后、左侧面板将显示存储分段列表。+
屏幕截图示例



NetApp StorageGRID与GitLab

作者：郑安杰

NetApp已使用GitLab对StorageGRID进行了测试。请参见下面的GitLab配置示例。 请参见 "GitLab对象存储配置指南" 了解详细信息。

对象存储连接示例

对于Linux软件包安装、这是的一个示例 `connection` 在合并表单中设置。编辑 `/etc/gitlab/gitlab.rb` 并添加以下行、替换所需的值：

```
# Consolidated object storage configuration
gitlab_rails['object_store']['enabled'] = true
gitlab_rails['object_store']['proxy_download'] = true
gitlab_rails['object_store']['connection'] = {
  'provider' => 'AWS',
  'region' => 'us-east-1',
  'endpoint' => 'https://<storagegrid-s3-endpoint:port>',
  'path_stype' => 'true',
  'aws_access_key_id' => '<AWS_ACCESS_KEY_ID>',
  'aws_secret_access_key' => '<AWS_SECRET_ACCESS_KEY>'
}
# OPTIONAL: The following lines are only needed if server side encryption
is required
gitlab_rails['object_store']['storage_options'] = {
  'server_side_encryption' => 'AES256'
}
gitlab_rails['object_store']['objects']['artifacts']['bucket'] = 'gitlab-
artifacts'
gitlab_rails['object_store']['objects']['external_diffs']['bucket'] =
'gitlab-mr-diffs'
gitlab_rails['object_store']['objects']['lfs']['bucket'] = 'gitlab-lfs'
gitlab_rails['object_store']['objects']['uploads']['bucket'] = 'gitlab-
uploads'
gitlab_rails['object_store']['objects']['packages']['bucket'] = 'gitlab-
packages'
gitlab_rails['object_store']['objects']['dependency_proxy']['bucket'] =
'gitlab-dependency-proxy'
gitlab_rails['object_store']['objects']['terraform_state']['bucket'] =
'gitlab-terraform-state'
gitlab_rails['object_store']['objects']['pages']['bucket'] = 'gitlab-
pages'
```

版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。 美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 http://www.netapp.com/TM 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。