Object Drive 1.0
Software Installation Procedures Guide (SIPG)

Last Updated on September 13, 2019

Basic Steps to Deploy Object Drive Service

1. Acquire artifacts for installation
2. Setup S3 Bucket for storing content streams
3. Setup RDS Instance for storing metadata
4. Setup EC2 to Run Object Drive
5. Install RPM
    1. For a new installation
    2. Update existing installation
6. Configure Environment Variables
    1. Reference example env.sh
7. Apply Database Schema
    1. Config
    2. Initial Schema Creation
    3. Migration of Existing Schema
    4. Alternate config for Database Tool
8. Start Service
9. Verify Proper Running
10. Setup Gatekeeper Cluster Information

Deploying Object Drive Service

Acquire Artifacts for Installation

RPM for current release is available for download at

https://nexus.di2e.net/nexus3/repository/Private_DIME_YUM/object-drive-server/object-drive-1.0-1.0.23b4-3075.20190913.x86_64.rpm

Setup S3 Bucket for storing content streams

This step needs to be performed one time before running the Object Drive service.

You can either have an S3 bucket dedicated to the Object Drive service, or you can have a bucket used for other purposes that is also used by Object Drive. If you plan to use an EC2 that leverages IAM Roles already authorized to access a bucket, then this step is effectively done already.

When creating a bucket via the AWS console, the credentials used for accessing the bucket *must* be able to read and write data from and to the bucket. If you aren't using IAM Roles, then you'll need to have the access key id and secret access key which will need to be assigned later in the environment variables `OD_AWS_ACCESS_KEY_ID` and `OD_AWS_SECRET_ACCESS_KEY` respectively.

The credentials *must* also have the policy *IAMFullAccess* access to IAM.

> AWS Console -> S3 -> Create Bucket

Provide a name and select the region and click create.

Establish policies to either grant access to read/write the bucket from specific credentials or make use of IAM roles

> AWS Console -> Identity and Access Mgmt -> Users -> Create New Users

Make note of the name of the S3 bucket that is being used. This value will need to be assigned later in `OD_AWS_S3_BUCKET`.

If you want to use the bucket for more than this instance of Object Drive, then it is recommended to also specify a value for the environment variable OD_CACHE_PARTITION, which is used as a prefix for the object key in a folder like structure.

The following example uses a bucket named *bucket-object-drive* prefixing objects under *test/object-drive* allowing the same bucket to be used for other purposes

```
OD_AWS_S3_BUCKET=bucket-object-drive
OD_CACHE_PARTITION=test/object-drive
```

When the Object Drive service runs, it will store uploaded files in encrypted format, in randomly generated file names in a further subkey that is based upon the hostname and randomly generated instance identifier as tracked in the database. For example, an instance named 0093eef13fa5-4e51fe12 combined with the above, would potentially result in files stored as follows:

Example object keys resulting from this configuration

- /test/object-drive/**0093eef13fa5-4e51fe12**/315165892a1ba6b55e491ced48052
- /test/object-drive/**0093eef13fa5-4e51fe12**/72084fed177cc939546d3827de6fb1

If OD_CACHE_PARTITION was not specified, then objects would be created in a subfolder named for the instance off the root of the bucket like this

- /**0093eef13fa5-4e51fe12**/315165892a1ba6b55e491ced48052
- /**0093eef13fa5-4e51fe12**/72084fed177cc939546d3827de6fb1

The instance identifier is determined from the database automatically when the schema is first populated.

Setup RDS Instance for storing metadata

This step needs to be performed one time before running the Object Drive service and is the most time consuming. This assumes that you want to setup the Object Drive database on its own RDS instances apart from other databases. Alternatively, you can use an existing RDS instance shared with other applications and simply use a different schema and login credentials.

Object Drive 1.0 supports MySQL (5.6 and 5.7), MariaDB (10.0), and Aurora. Depending on which AWS instance and region you deploy to, you may not have all these options available.

AWS Console -> RDS -> Launch a DB Instance -> MySQL -> Select -> MySQL -> Next Step

On the DB Details screen

- Instance Specifications
    - DB Engine Version:
        - If MySQL 5.6, use **5.6.34** or higher
        - If MySQL 5.7, use **5.7.16** or higher
        - If MariaDB, use newest version
    - DB Instance Class: The size doesn't matter so much at the outset. You can upsize it later by snapshotting and creating a new instance to restore into.
        - **For development/testing, a db.t2.micro is sufficient.**
        - **For production, recommend starting with a db.r4.medium**
        - **Memory Optimized (R*) is preferable over General Purpose (M*)**
    - Allocated Storage: You can upsize later by snapshotting and creating a new instance to restore into.
        - **For development/testing, 10GB of general purpose storage is generally more than sufficient.**
        - **For production environments, start with 100GB. You should also use provisioned IOPS.**
        - **For sizing and growth expectations, space consumption is approximately 25GB per 1 million files.**

- Settings
    - DB Instance Identifier. The name used in assorted places is `metadatadb`, but this is fully configurable, and should be populated later in variable `OD_DB_SCHEMA`
    - Master Username: The login used for accessing the database. This value will be populated later in variable `OD_DB_USERNAME`
    - Master Password: The password associated with the login credentials. This value will be populated later in variable `OD_DB_PASSWORD`

Proceed through steps on screen until the database is created. If the database needs to be constrained to a specific VPC, be sure to select this as part of the security group settings.

The amount of time required to first instantiate the database depend greatly on the amount of storage allocated as this storage needs to be partitioned, formatted and mounted. Generally less than 1 minute per GB. Once this has been completed, a few more adjustments are required before the database can be used by Object Drive.

Create a custom parameter group

AWS Console -> RDS -> Parameter Groups -> Create Parameter Group

- Parameter Group Family: `mysql5.6`, or whichever DB Engine Version was chosen when creating the RDS above
- Group Name: Recommend setting this to `mysql56odrive` to delineate it specifically for Object Drive.
- Description: Provide the value `Custom Parameter Group for Object Drive`

Click Create to continue. This initializes with the default parameters.

Using the filter box at the top, type in the name of each parameter and set the desired settings.

Parameter	Value	Notes
log_bin_trust_function_creators	1	Required when log_bin is enabled
show_compatibility_56	1	Required for database tool to work for migrations when using MySQL 5.7

Save the settings for the parameter group.

Return to the list of RDS instances.

Expand the database instance created for Object Drive, and from the Instance Actions drop down, choose Modify.

Under Database Options, choose the recently created parameter group in the drop down labeled DB Parameter Group. Scroll down to the bottom of the page and check the box labeled Apply Immediately before clicking Continue. Expand the instance to view the details. From the Instance Actions drop down, choose Reboot

Make note of the value of Writer Endpoint. This value will be set later in environment variables for OD_DB_HOST.

Make note of the value of port. By default it should be 3306. This value will be set later in environment variables for OD_DB_PORT

Setup EC2 to Run Object Drive

This step needs to be performed one time before running the Object Drive service.

> AWS Console -> EC2 -> Launch Instance -> My AMIs

Choose the appropriate AMI to create a new EC2 instance from such that the security group sets the EC2 to be in the same VPC as that for which the RDS instance is in.

For the instance type

- a development/test instance can use a t2.micro.
- production instances should likely go with more processing power and memory.

For the storage needs

> Beyond the base os install, the Object Drive binary and configuration files itself doesn't consume much space. Block storage allocated to the EC2 used by Object Drive is predominantly for local cache support to improve performance of handling file streams both inbound and outbound. The larger the storage space, the more files will eventually be cached locally within the instance. This cache also acts as an upper bound on maximum file size because files are first received to the local cache before transmitted to S3 for permanent storage. It is recommended to allocate no less than 10GB specifically for cache, however, an overly large cache, particularly when content is smaller files can lead to performance degradation when the cache is being checked. Recommend limiting the total size of the cache to no more than 10 times the maximum file size you anticipate supporting. For example, if you plan to support files up to 5GB, a cache size of 50GB is reasonable.

Install RPM

This step needs to be performed whenever there is a new release of Object Drive.

Log into the EC2, sudo, and transfer the RPM to the EC2.

For a new installation

Install the RPM

```
yum install object-drive-1.0-1.0.23b4-3075.20190913.x86_64.rpm
```

This will also create a file at `/opt/services/object-drive-1.0/env.sh` which you will reference later.

Proceed to sections below for applying the database schema and configuring environment variables before starting the service.

Update existing installation

Upgrade the existing package

```
yum upgrade object-drive-1.0-1.0.23b4-3075.20190913.x86_64.rpm
```

 If the following message is reported, the upgrade was not successful.

```
Object-drive-1.0-1.0.23b4-3075.20190913.x86_64.rpm: does not update installed package
No Packages marked for Update
```

This is caused by version scheme changes in package metadata.

You will need to use the RPM tool to force the upgrade via:

```
rpm --force --upgrade object-drive-1.0-1.0.23b4-3075.20190913.x86_64.rpm
```

If there are database patch scripts, separate guidance will be given for how to update the instance, but the scripts will be found in the same location as the database schema. You can use the database tool to do a migration of existing schema. Proceed to sections below for updating the schema.

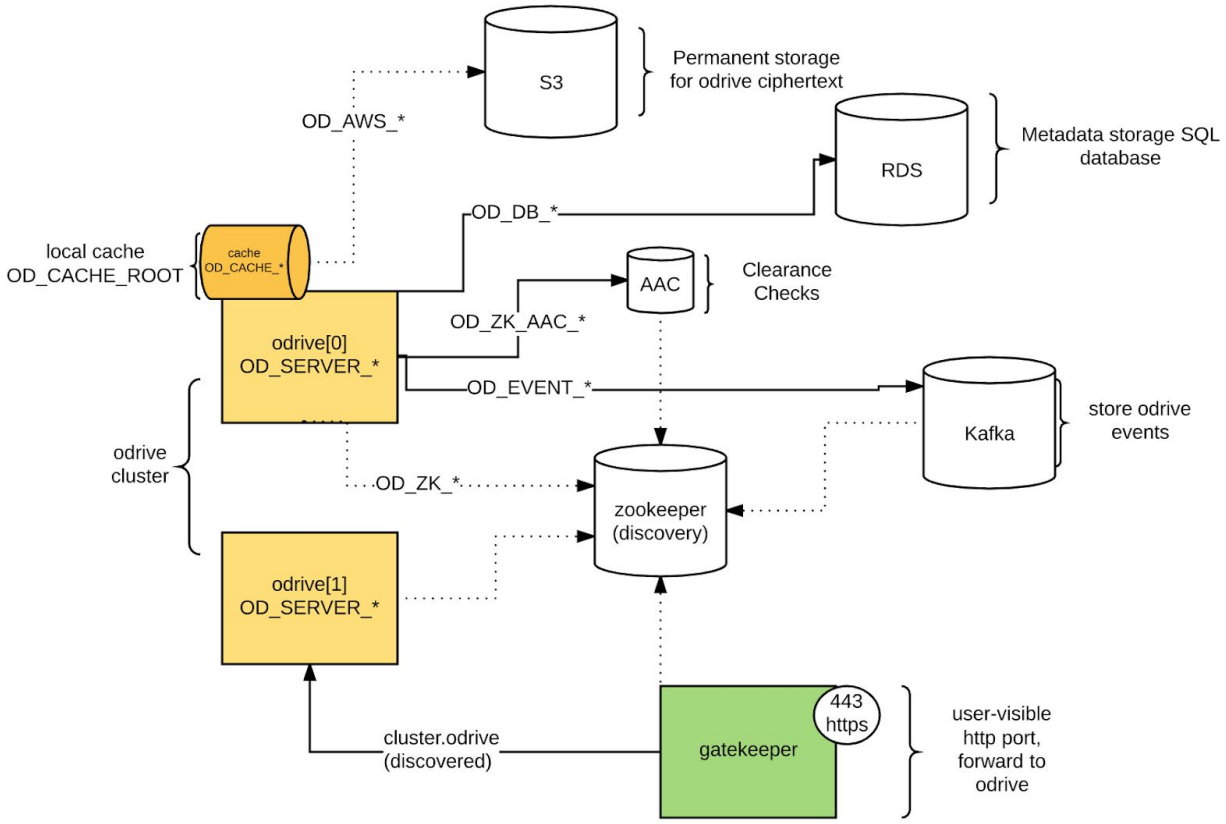
Tail the logs to verify everything restarted properly

```
tail -f /opt/services/object-drive-1.0/log/object-drive.log
```

More details in section below named Verify Proper Running

Configure Environment Variables

The environment variables have a naming pattern that corresponds to the components being wired together. The most important components are the relational database that we write metadata to, and the S3 store that populates the local cache.



This step needs to be performed one time before running the Object Drive service. Subsequent updates may introduce new variables needing configured but such guidance will be given independently.

When the Object Drive service was installed from the RPM, a file named env.sh should be created in the /opt/services/object-drive-1.0 folder. Edit this file, and reference the environment variables page for guidance on what the values should be.

Reference example env.sh

A reference example is depicted below

```
#!/bin/bash
export OD_CERTPATH=/opt/services/object-drive-1.0/certs
export OD_LOG_LOCATION=/opt/services/object-drive-1.0/log/object-drive.log

export OD_AAC_CA=$OD_CERTPATH/client-aac/trust/client.trust.pem
export OD_AAC_CERT=$OD_CERTPATH/client-aac/id/client.cert.pem

export OD_AAC_CN=twl-server-generic2
export OD_AAC_KEY=$OD_CERTPATH/client-aac/id/client.key.pem
export OD_AAC_CN=twl-server-generic2
```

```
export OD_AAC_ZK_ADDRS=zk:2181

export OD_AWS_ACCESS_KEY_ID=the-aws-access-key-id-with-access-to-the-s3-bucket
export OD_AWS_REGION=us-east-1
export OD_AWS_S3_BUCKET=the-s3-bucket-name
export OD_AWS_S3_ENDPOINT=s3.amazonaws.com
export OD_AWS_SECRET_ACCESS_KEY=the-aws-secret-access-key-with-access-to-the-s3-bucket
export OD_AWS_S3_FETCH_MB=16

export OD_AWS_ASG_EC2=odrive1
export OD_AWS_ASG_ENDPOINT=autoscaling.us-east-1.amazonaws.com
export OD_AWS_ASG_NAME=odriveScalingGroup
export OD_AWS_CLOUDWATCH_ENDPOINT=monitoring.amazonaws.com
export OD_AWS_CLOUDWATCH_INTERVAL=300
export OD_AWS_CLOUDWATCH_NAME=$OD_ZK_ANNOUNCE
export OD_AWS_SQS_BATCHSIZE=10
export OD_AWS_SQS_ENDPOINT=sqs.us-east-1.amazonaws.com
export OD_AWS_SQS_INTERVAL=60
export OD_AWS_SQS_NAME=odriveLifecycleQueue

export OD_CACHE_EVICTAGE=300
export OD_CACHE_FILELIMIT=20000
export OD_CACHE_FILESLEEP=1
export OD_CACHE_HIGHTHRESHOLDPERCENT=75
export OD_CACHE_LOWTHRESHOLDPERCENT=50
export OD_CACHE_ROOT=${OD_BASEPATH}/cache
export OD_CACHE_PARTITION=.
export OD_CACHE_WALKSLEEP=30

export OD_DB_CA=$OD_CERTPATH/aws/rds-combined-ca-bundle.pem
export OD_DB_CN=
export OD_DB_CERT=
export OD_DB_CONN_PARAMS="parseTime=true&collation=utf8_unicode_ci&readTimeout=30s"
export OD_DB_HOST=database-instance-name.cjb9y0rlmcrl.us-east-1.rds.amazonaws.com
export OD_DB_KEY=
export OD_DB_MAXIDLECONNS=20
export OD_DB_MAXOPENCONNS=20
export OD_DB_PASSWORD=the-database-password
export OD_DB_PORT=3306
export OD_DB_SCHEMA=metadatadb
export OD_DB_USERNAME=the-database-username
```

```
export OD_ENCRYPT_ENABLED=true
export OD_ENCRYPT_MASTERKEY=this-value-should-be-made-up-of-random-characters-when-first-installed-and-left-alone

export OD_EVENT_ZK_ADDRS=zk:2181
export OD_EVENT_PUBLISH_FAILURE_ACTIONS=*
export OD_EVENT_PUBLISH_SUCCESS_ACTIONS=create,delete,undelete,update

export OD_PEER_ENABLED=true
export OD_PEER_CN=twl-server-generic2
export OD_PEER_SIGNIFIER=P2P

export OD_SERVER_CA=$OD_CERTPATH/server-web/trust
export OD_SERVER_CERT=$OD_CERTPATH/server/server.cert.pem
export OD_SERVER_CIPERS=TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA
export OD_SERVER_KEY=$OD_CERTPATH/server/server.key.pem
export OD_SERVER_PORT=4430

export OD_TOKENJAR_LOCATION=/opt/services/object-drive-1.0/token.jar
export OD_TOKENJAR_PASSWORD=

export OD_USERAOCACHE_LRU_TIME=600
export OD_USERAOCACHE_TIMEOUT=40

export OD_ZK_AAC=/services/aac/1.2/thrift
export OD_ZK_ANNOUNCE=/services/object-drive-1.0
export OD_ZK_MYIP=
export OD_ZK_MYPORT=
export OD_ZK_TIMEOUT=5
export OD_ZK_URL=10.2.11.36:2181,10.2.11.37:2181,10.2.11.38:2181
```

⚠ Notes Regarding Common Names for AAC and PEER

Note! OD_AAC_CN and OD_PEER_CN must be set now, as we are actually fully checking the certificates for AAC and peer odrive instances now. OD_AAC_CN value will be whatever CN you get in the AAC server certificate when you make a TLS connection to it. OD_PEER_CN must also match the actual CN of that odrive server's cert. If you cannot make this work (which is a security problem if you cannot, particularly because zookeeper is open and writeable by

anyone), then you can use OD_AAC_INSECURE_SKIP_VERIFY and OD_PEER_INSECURE_SKIP_VERIFY to return the the previous setup that is not checking certificates. We assume that a cluster of AAC servers are sharing the same certificate, as is a cluster of odrive servers. If it is in fact NOT setup like this, where all odrive and aac instances actually have certificates with CN that matches hostname, then use "OD_PEER_CN=" and "OD_AAC_CN=" in that case to tell odrive to presume that the hostname it connects with must match the CN in the server's presented certificate. The Go runtime security model is not allowing simple hostname check disabling, requiring us to just set this up securely so that if the hostname either matches CN of what we connected with, or what we told the connection the CN would be.

⚠ Notes Regarding Encryption Masterkey

OD_ENCRYPT_MASTERKEY is the same randomly generated value for all Object Drive instances within a cluster (same database, zookeeper, S3 bucket and partition in the bucket). This value is used to decrypt the encrypted keys stored in the relational database (from the Object Drive process), which is then used to decrypt the cipher texts coming from S3. *If this setup has only one Object Drive (i.e.: no physical redundancy), then ensure that this key is backed up to a secure offline location to prevent a loss of all object drive data for this cluster.*

⚠ Notes Regarding Encryption at Rest Flag

OD_ENCRYPT_ENABLED is a flag introduced in 1.0.19 that permits operation of the service without encryption at rest when the value provided is false. If this environment variable is not defined, it will default to true, ensuring data is encrypted at rest. As a warning, if the value of this variable is set to false, then a header will be present in all API responses appearing as follows:

Content-Encrypted-At-Rest:

```
FALSE. The service is running without encrypting data at rest.
Files are encrypted in transit only.
```

Furthermore, at service startup, a warning banner will be depicted

```
============================================================
============================================================
============================================================
                    W  A  R  N  I  N  G

This service is running without encryption at rest enabled.
This means that data that is uploaded to the service will be
stored in the local cache in an unencrypted, plain text form.

Any data stored in S3 buckets will also be in plain text and
anyone with read access to that bucket directly, or via IAM
roles, will be able to see the raw content without being
limited by authorization checks on the metadata.

Extreme caution should be taken in use of this system.

There is no way to convert the system back to encrypted mode
without re-uploading files.

Responses to all API calls will indicate that data is being
stored in an unencrypted format. This is to provide similar
warning to those users who would otherwise expect the data
to be encrypted based upon past experience using the service.
============================================================
============================================================
============================================================
```

Environment Variable Table

The following environment variables can be set in the environment for usage by the object drive services

New in 1.0.23

OD_DB_ACMGRANTEE_LRU_TIME *since v1.0.23*	The time in seconds that an acmgrantee will be cached in memory unless necessary to evict per least-recently-used caching constraints Default: 600
OD_SERVER_MAXPAGESIZE *since v1.0.23*	The maximum number of results per page allowed for list/search operations. Default: 100

AAC Integration

AAC Integration is used for authorization requests. At the time of this writing it is tightly coupled for CRUD type operations and uses snippets for listing/querying sets of objects.

Name	Description
OD_AAC_CA *since v1.0*	The path to the certificate authority folder or file containing public certificate(s) to trust as the server when connecting to AAC.
OD_AAC_CERT *since v1.0*	The path to the public certificate for the user credentials connecting to AAC.
OC_AAC_CN *since v1.0.1.12*	The CN that we expect all AAC servers to have. We use this when we enforce certificate verification Default: twl-server-generic2
OD_AAC_HEALTHCHECK *since v1.0.12*	An acm expected to validate against the AAC service. Default: {"version":"2.1.0","classif":"U"}
OD_AAC_HOST *since v1.0*	The host of the AAC server to perform a direct connect instead of discovery.
OD_AAC_INSECURE_SKIP_VERIFY *since v1.0.1.22*	This turns off certificate verification for connecting to the AAC service. Normally this should not be enabled. Default: false
OD_AAC_KEY *since v1.0*	The path to the private key for the user credentials connecting to AAC.

OD_AAC_PORT *since v1.0*	The port of the AAC server to perform a direct connect instead of discovery.
OD_AAC_RECHECK_TIME *since v1.0.14*	The interval seconds between AAC health status checks (1-600). Default: 30
OD_AAC_WARMUP_TIME *since v1.0.14*	The number of seconds to wait for ZooKeeper before checking health of AAC (1-60) Default: 10
OD_AAC_ZK_ADDRS *since v1.0.1.7*	Comma-separated list of host:port pairs to connect to a Zookeeper cluster specific to AAC discovery. If this value is not set, AAC will be discovered using list of host:port pairs in OD_ZK_URL

AWS S3

Amazon Web Services environment variables contain credentials for AWS used for S3 when configuring permanent storage.

Name	Description
OD_AWS_ACCESS_KEY_ID *since v1.0*	The AWS Access Key. Available here: https://console.aws.amazon.com/iam/home If leverage IAM Roles, either do not set this variable, or assign it as an empty string
OD_AWS_ENDPOINT *deprecated in v1.0.1.7 (Nov 2016)* *removed in v1.0.19 (Feb 2019)*	The AWS S3 URL endpoint to use. Documented at: http://docs.aws.amazon.com/general/latest/gr/rande.html
OD_AWS_REGION *since v1.0*	The AWS region to use. (i.e. us-east-1, us-west-2).
OD_AWS_S3_BUCKET *since v1.0*	The S3 Bucket name to use. The credentials used defined in OD_AWS_SECRET_ACCESS_KEY and OD_AWS_ACCESS_KEY_ID must have READ and WRITE privileges to the bucket.
OD_AWS_S3_ENDPOINT *since v1.0*	The AWS S3 URL endpoint to use. Some environments will need to override this value. OD_AWS_ENDPOINT is a deprecated duplicate of this variable Default: s3.amazonaws.com
OD_AWS_S3_FETCH_MB *since v1.0.1.3*	The size (in MB) of chunks to pull from S3 in cases where odrive is re-caching from S3. This is a compromise between response time vs billing caused by S3 billing per request. Default: 16

OD_AWS_SECRET_ACCESS_KEY *since v1.0*	AWS secret key. Access and secret key variables override credentials stored in credential and config files. If leverage IAM Roles, do not set this variable. Values wrapped in ENC{...} are decrypted using token.jar

AWS AutoScaling

CloudWatch, SQS, and AutoScale with alarms (installed in AWS) interact to produce autoscaling behavior.

Name	Description
OD_AWS_ASG_EC2 *since v1.0.1.5*	This is the name assigned to the AMI instance that got launched, like a host name in the autoscaling group. Set it to the AWS EC2 InstanceId if SQS and ASG are enabled.
OD_AWS_ASG_ENDPOINT *since v1.0.1.5*	This is the location of the autoscaling endpoint. Default: autoscaling.amazonaws.com
OD_AWS_ASG_NAME *since v1.0.1.5*	This is the name of the autoscaling group. If not using autoscaling, do not set this variable.
OD_AWS_CLOUDWATCH_ENDPOINT *since v1.0.1.5*	The location of cloudwatch monitoring Default: monitoring.us-east-1.amazonaws.com
OD_AWS_CLOUDWATCH_INTERVAL *since v1.0.1.5*	The frequency in seconds for how often stats are computed and sent to cloudwatch Default: 300
OD_AWS_CLOUDWATCH_NAME *since v1.0.1.5*	When reporting to cloud watch, we must report into a namespace. In production, it's the same as the zk url. If not using cloudwatch reports, either do not set this variable, or assign it as an empty string
OD_AWS_SQS_BATCHSIZE *since v1.0.1.12*	The number of messages (1-10) to request from lifecycle queue per polling interval to examine for shutdown Default: 10
OD_AWS_SQS_ENDPOINT *since v1.0.1.5*	The location of the SQS endpoint. Default: sqs.us-east-1.amazonaws.com
OD_AWS_SQS_INTERVAL *since v1.0.1.5*	Poll interval for the lifecycle queue in seconds Default: 60
OD_AWS_SQS_NAME *since v1.0.1.5*	This is the name of the lifecycle queue. If not using autoscale for termination, do not set this variable.

Cache For Files

Storage cache on disk as an intermediary for upload/download to and from S3

Name	Description
OD_CACHE_EVICTAGE *since v1.0*	Denotes the minimum age, in seconds, a file in cache before it is eligible for eviction (purge) from the cache to free up space. Default: 30
OD_CACHE_FILELIMIT *since v1.0.20*	Denotes the maximum number of files that will be retained in cache. This is useful to prevent exceedingly long service restarts when there are many small size files. A value of 0 indicates unlimited files can be cached, the same behavior prior to 1.0.20 Default: 0
OD_CACHE_FILESLEEP *since v1.0.20*	Denotes the duration in milliseconds to put the cache purge thread to sleep between reviewing each file in the cache. A value greater than zero can help prevent CPU exhaustion during cache purge operations Default: 0
OD_CACHE_HIGHWATERMARK *since v1.0* *removed in v1.0.20 (Apr 2019)*	Denotes a percentage of the file storage on the local mount point as the high size such that when the total space used exceeds the allocated percentage, a file in the cache will be purged if its age last used exceeds the eviction age time. Default: 0.75
OD_CACHE_HIGHTHRESHOLDPERCENT *since v1.0.20*	Replaces OD_CACHE_HIGHWATERMARK Denotes a percentage of the file storage on the local mount point as the high size such that when the total space used exceeds the allocated percentage, a file in the cache will be purged if its age last used exceeds the eviction age time. Default: 75
OD_CACHE_LOWWATERMARK *since v1.0* *removed in v1.0.20 (Apr 2019)*	Denotes a percentage of the file storage on the local mount point as the low size where total consumption must be at least that specified for files to be considered for purging. Default: 0.50
OD_CACHE_LOWTHRESHOLDPERCENT *since v1.0.20*	Replaces OD_CACHE_LOWWATERMARK Denotes a percentage of the file storage on the local mount point as the low size where total consumption must be at least that specified for files to be considered for purging. Default: 50

OD_CACHE_PARTITION *since v1.0*	An optional path for prefixing folders as part of the key in S3 prior to the cache folder. Intended for delineating different environments.
OD_CACHE_ROOT *since v1.0*	An optional absolute or relative path to set the root of the local cache settings to override the default which beings in the same folder as working directory from which the object drive instance was started. Default: .
OD_CACHE_WALKSLEEP *since v1.0*	Denotes the frequency, in seconds, for which all files in the cache are examined to determine if they should be purged. Default: 30

Cache Peer To Peer

When multiple instances of object-drive need to contact each other to collaborate on ciphertext

Name	Description
OD_PEER_CN *since v1.0.1.7*	The name associated with the certificate. This may need to change when certificates are changed, but if it works at default, leave it. Default: twl-server-generic2
OD_PEER_ENABLED *since v1.0.20*	Indicates whether this instance is able to retrieve and service ciphertext requests from its peers. Disabling this feature can be useful if the desire is to force each instance to use only their own file cache. Default: true
OD_PEER_INSECURE_SKIP_VERIFY *since v1.0.1.7*	This can turn off certificate verification for connecting to peer instances in the cluster. Normally this should not be enabled. The trust, certificate, and key used by peer connections are those defined in the OD_SERVER_CA, OD_SERVER_CERT, and OD_SERVER_KEY values. Default: false
OD_PEER_SIGNIFIER *since v1.0.1.7*	This is a pseudonym used to signify a P2P client, which is set because it prevents users from accessing via nginx. This generally doesn't need to be changed. Default: P2P

Cache For User AO

Configuration settings for managing the internal user authorization object cache that contributes to improved performance in ACM associations.

Name	Description

OD_USERAOCACHE_LRU_TIME *since v1.0.20*	The time in seconds that a user AO will be cached in memory unless necessary to evict per least-recently-used caching constraints Default: 600
OD_USERAOCACHE_TIMEOUT *since v1.0.20*	The permitted time in seconds to allow a User AO Cache rebuild to happen asynchronously before it will be assumed to have failed and permit another thread to attempt a rebuild. Default: 40

Database For Metadata

The database is used to store metadata about objects and supports querying for matching objects to drive list operations and filter for user access.

Name	Description
OD_DB_ACMGRANTEE_LRU_TIME *since v1.0.23*	The time in seconds that an acmgrantee will be cached in memory unless necessary to evict per least-recently-used caching constraints Default: 600
OD_DB_CA *since v1.0*	The path to the certificate authority folder or file containing public certificate(s) to trust as the server when connecting to the database over TLS.
OD_DB_CERT *since v1.0*	The path to the public certificate for the user credentials connecting to the database.
OD_DB_CN *since v1.0.1.12*	The common name (cn) of the x509 certificate to the database.
OD_DB_CONN_PARAMS *since v1.0*	Custom parameters to include for the database connection. For MySQL/MariaDB, must specify **parseTime=true&collation=utf8_unicode_ci&readTimeout=30s** This is not a default. However, if readTimeout is not specified, it will be defaulted to 30s
OD_DB_CONNMAXLIFETIME *since v1.0.17*	The maximum amount of time, in seconds, that a database connection may be reused. 0 indicates indefinitely. Default: 30
OD_DB_DEADLOCK_RETRYCOUNTER *since v1.0.19*	Indicates the number of times a create or update operation should be retried if the transaction fails due to a database deadlock. Default: 30

OD_DB_DEADLOCK_RETRYDELAYMS *since v1.0.19*	The duration in milliseconds between retry attempts for a create or update operation when a transaction fails due to a deadlock in the database. Default: 55
OD_DB_DRIVER *since v1.0.19*	The database driver to use. Recognized values are `mysql`. Default: mysql
OD_DB_HOST *since v1.0*	The name or IP address of the MySQL / MariaDB / Aurora conforming database. Default: metadatadb
OD_DB_KEY *since v1.0*	The path to the private key for the user credentials connecting to the database.
OD_DB_MAXIDLECONNS *since v1.0*	The maximum number of database connections to keep idle. Overrides language default of 2. Default: 10
OD_DB_MAXOPENCONNS *since v1.0*	The maximum number of database connections to keep open. Overrides language default of unlimited. Default: 10
OD_DB_PASSWORD *since v1.0*	The password portion of credentials when connecting to the database. Note that if a token.jar is installed onto the system, we can use the encrypted indicator like `ENC{...}
OD_DB_PORT *since v1.0*	The port that the MySQL / MariaDB / Aurora instance is listening on. Usually this will be port 3306. Default: 3306
OD_DB_PROTOCOL *since v1.0.19*	The protocol to use when communicating with the database. Recognized values are `tcp`. Default: tcp
OD_DB_RECHECK_TIME *since v1.0.20*	The interval seconds between database health status checks. Values less than 1 will disable the health check. Default: 30
OD_DB_SCHEMA *since v1.0*	The schema to connect to after logging into the database.
OD_DB_SKIP_VERIFY *since v1.0.19*	Indicates whether the verification of the hostname of the database server vs what it identifies in its certificate is skipped. Default: false
OD_DB_USE_TLS *since v1.0.19*	Indicates whether the database connection should use TLS. Default: true
OD_DB_USERNAME *since v1.0*	The username portion of credentials when connecting to database.

Event Publishing

Object Drive publishes a single event stream for client applications.

Name	Description
OD_EVENT_KAFKA_ADDRS *since v1.0*	A comma-separated list of host:port pairs. These are Kafka brokers. If discovering Kafka through ZooKeeper, either do not set this variable, or assign it as an empty string. If this value is set, OD_EVENT_ZK_ADDRS will be ignored. If connection is made by setting this list of addresses, then there is no watch for member changes, and durability and message delivery is not guaranteed. Configurations are strongly advised to use OD_EVENT_ZK_ADDRS instead.
OD_EVENT_PUBLISH_FAILURE_ACTIONS *since v1.0.1.14*	A comma delimited list of event action types that should be published to kafka if request failed. Recognized values are `access`, `authenticate`, `create`, `delete`, `list`, `undelete`, `unknown`, `update`, `zip`, `*`. To disable failure actions from being published, use the value `disabled` The value * denotes all failures will be published Recommended: * Default: *
OD_EVENT_PUBLISH_SUCCESS_ACTIONS *since v1.0.1.14*	A comma delimited list of event action types that should be published to kafka when the request is successful. Recognized values are `access`, `authenticate`, `create`, `delete`, `list`, `undelete`, `unknown`, `update`, `zip`, `*` To disable success actions from being published, use the value `disabled` The value * denotes all success will be published Recommended: create,delete,undelete,update Default: *
OD_EVENT_TOPIC *since v1.0.10*	An override value for the kafka topic to emit events to. Default: odrive-event
OD_EVENT_ZK_ADDRS *since v1.0.1.8*	Discovery of the kafka nodes may be supported through the use of a ZooKeeper cluster. A comma-separated list of host:port pairs. This may be set to the same value as OD_ZK_URL. Discovering kafka nodes via ZooKeeper cluster allows for

	reconnection as members change as the path is watched.

NOTE: If neither OD_EVENT_ZK_ADDRS nor OD_EVENT_KAFKA_ADDRS are set, or they are both empty strings, then object drive will not publish events. Events are required to support Auditing

Headers

Some request and response headers may be disabled or given a different name other than the default.

Name	Description
OD_HEADER_BANNER_ENABLED *since v1.0.21*	Indicates whether a response header representing the banner field of the object ACM should be provided in the response for content streams. Default: true
OD_HEADER_BANNER_NAME *since v1.0.21*	The name of the response header representing the banner field of the object ACM. Default: Classification-Banner
OD_HEADER_SERVER_ENABLED *since v1.0.21*	Indicates whether a response header denoting the server version should be set Default: true
OD_HEADER_SERVER_NAME *since v1.0.21*	The name of the response header denoting the server version. Default: odrive-server
OD_HEADER_SESSIONID_ENABLED *since v1.0.21*	Indicates whether a response header denoting the session identifier should be set, as well as picked up in requests for session correlation Default: true
OD_HEADER_SESSIONID_NAME *since v1.0.21*	The name of the header used for the session identifier Default: Session-Id

Logging

Object Drive itself just logs to stdout. But when the service script launches it from /etc/init.d, it reads an env.sh of environment variables. One of the things that this environment variable does is to set a default log location and will take an override in env.sh itself.

Name	Description
OD_LOG_LEVEL *since v1.0*	Controls the verbosity and which logs get written. Only log statements where the log level is equal to or greater then the log level are written. If left at the default (0), then Info, Warn, Error and

	Fatal messages will be written to the logs, but not Debug. Recognized values are `-1` (debug), `0` (info), `1` (warn), `2` (error), `3` (fatal), `debug`, `info`, `warn`, `error`, `fatal` Default: 0 (indicates only log 'info' and above)
OD_LOG_LOCATION *since v1.0*	The absolute pathname to use for the object drive service when overriding the default. Default: /opt/services/object-drive-1.0/log/object-drive.log
OD_LOG_MODE *since v1.0.17*	Denotes whether logging is in development or production mode. When in development mode, stack traces will be output for WARN level messages and above. For production mode, stack traces are only output in ERROR level. Permissible values are production, development Default: production

Server

Remaining server settings are noted here

Name	Description
~~OD_DEADLOCK_RETRYCOUNTER~~ *since v1.0.1.26* *renamed v1.0.19 -> OD_DB_DEADLOCK_RETRYCOUNTER*	~~Indicates the number of times a create or update operation should be retried if the transaction fails due to a database deadlock.~~ ~~Default: 30~~
~~OD_DEADLOCK_RETRYDELAYMS~~ *since v1.0.1.26* *renamed v1.0.19 -> OD_DB_DEADLOCK_RETRYDELAYMS*	~~The duration in milliseconds between retry attempts for a create or update operation when a transaction fails due to a deadlock in the database.~~ ~~Default: 55~~
OD_ENCRYPT_ENABLED *since v1.0.19*	Indicates whether file content should be encrypted at rest in local cache and permanent storage. This variable should be set the same across all instances within a cluster, and should not be changed once set at initialization. A value of false indicates that content will not be encrypted, and instead be stored in plain text in permanent storage (e.g. S3) which may facilitate use of other analytical tools outside of accessing the file streams through the API. Default: true
OD_ENCRYPT_MASTERKEY *since v1.0*	The secret master key used as part of the encryption key for all files stored in the system. If this value is changed, all file keys must be adjusted at the same time. If you don't set this, the service will shut down. Note that if a token.jar is installed onto the system, we can use the encrypted indicator format like `ENC{...}

~~OD_OPTION_409~~ *since v1.0.1.19 (April 2017)* *removed v1.0.1.25 (June 2017)*	~~An option flag for enabling recent performance improvements. This should be set to true.~~
OD_SERVER_ACL_WHITELIST *since v1.0.11*	One or more environment variable prefixes to denote distinguished name assigned to the access control whitelist that controls whether a connector can impersonate as another identity
OD_SERVER_BASEPATH *since v1.0* *removed in v1.1 (Mar 2019)*	The base URL root. Used in debug UIs. Default: /services/object-drive/1.0
OD_SERVER_BINDADDRESS *since v1.0.19*	The default interface address to bind the listener to. For all interfaces, use 0.0.0.0. Default: 0.0.0.0
OD_SERVER_CA *since v1.0*	The path to the certificate authority folder or file containing public certificate(s) to trust as the server.
OD_SERVER_CERT *since v1.0*	The path to the public certificate for the server credentials. **Note:** If the certificate you are setting is not twl-server-generic2, then you should also set OD_PEER_CN to the common name of that certificate to properly support peer to peer exchange of encrypted data in a load balanced environment to efficiently leverage each peer's cache space.
OD_SERVER_CIPHERS *since v1.0.11*	A comma delimited list of ciphers to be allowed for connections. Recognized values are TLS_RSA_WITH_RC4_128_SHA, TLS_RSA_WITH_3DES_EDE_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_256_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_RC4_128_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_RC4_128_SHA, TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305,

	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305 Recommend using ● TLS_RSA_WITH_AES_128_CBC_SHA ● TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
OD_SERVER_KEY *since v1.0*	The path to the server's private key.
OD_SERVER_MAXPAGESIZE *since v1.0.23*	The maximum number of results per page allowed for list/search operations. Default: 100
OD_SERVER_PORT *since v1.0*	The port for which this object-drive instance will listen on. Binding to ports below 1024 require setting additional security settings on the system. Default: 4430
OD_SERVER_STATIC_ROOT *since v1.0.19*	The location on disk where static assets are stored given as a path relative to where the service binary is run from.
OD_SERVER_TEMPLATE_ROOT *since v1.0.19*	The location on disk where Go templates are stored given as a path relative to where the service binary is run from.
OD_SERVER_TIMEOUT_IDLE *since v1.0.17*	This is the maximum amount of time to wait for the next request when keep-alives are enabled, in seconds Default: 60
OD_SERVER_TIMEOUT_READ *since v1.0.17*	This is the maximum duration for reading the entire request, including the body. In most cases you want to set OD_SERVER_TIMEOUT_READHEADER instead
OD_SERVER_TIMEOUT_READHEADER *since v1.0.17*	This is the maximum amount of time allowed to read request headers, in seconds Default: 5
OD_SERVER_TIMEOUT_WRITE *since v1.0.17*	This is the maximum amount of time before timing out writes of the response, in seconds Default: 3600
OD_TOKENJAR_LOCATION *since v1.0.1.11*	If a token.jar is placed on the filesystem to support Bedrock secret encryption format, then this is the full location of that jar file. That jar is presumed to have used OD_TOKENJAR_PASSWORD in its generation Default: /opt/services/object-drive-1.0/token.jar
OD_TOKENJAR_PASSWORD *since v1.0.1.11*	This is the password that is embedded into code that is authorized to decrypt secrets that we cannot avoid writing down on the system. The security of the system does not lie in this password, but in the fact that each token.jar should be using a fresh sample.dat that has a fresh key per cluster. This value generally does not need an

	override, but it is here in case it does get changed without recompiling the code.

Zookeeper

Zookeeper is used to announce the availability of this instance of the object drive services as well as discover dependencies. At the edge, gatekeeper and nginx rely upon this information to publish availability and facilitate routing requests to the service.

Name	Description
OD_ZK_AAC *since v1.0*	The announce point for AAC nodes. Matches gatekeeper config cluster.aac.zk-location Default: /cte/service/aac/1.0/thrift
OD_ZK_ANNOUNCE *since v1.0*	The mount point for announcements where our zookeeper https node is placed. The point of this variable is to match the gatekeeper cluster.odrive.zk-location without the https part Default: /services/object-drive/1.0
OD_ZK_MYIP *since v1.0*	The IP address of the Object-Drive server as reported to Zookeeper. If this environment variable is defined it will override the value detected as the server's IP address on startup.
OD_ZK_MYPORT *since v1.0*	The Port of the Object-Drive server as reported to Zookeeper. If this environment variable is defined it will override the value detected as the server's listening port on startup. Default: 4430
OD_ZK_RECHECK_TIME *since v1.0.17*	The interval seconds between ZK health status checks (1-600) Default: 30
OD_ZK_RETRYDELAY *since v1.0.14*	The interval seconds between retry attempts when connecting to ZooKeeper (1-10) Default: 3
OD_ZK_TIMEOUT *since v1.0*	Timeout in seconds for zookeeper sessions Default: 5
OD_ZK_URL *since v1.0*	A comma delimited list of zookeeper instances to announce to. The structure of this value should be server1:port1,server2:port2,serverN:portN. Default: zk:2181

Apply Database Schema

This section describes new db schema setups as well as migrations for existing schemas with the database binary tool. If you installed Object Drive via RPM, the database tool should be installed at /opt/services/object-drive-1.0/database.

Config

The database tool can be configured with the same environment variables as the server. Source the **env.sh** and print the current configuration like this:

```
source /opt/services/object-drive-1.0/env.sh
/opt/services/object-drive-1.0/database debug
```

Initial Schema Creation

If your connection parameters look correct, initialize a brand new schema like this. Bundled migrations will also be applied.

```
/opt/services/object-drive-1.0/database init
```

Migration of Existing Schema

Upgrading a schema is easy. Simply apply pending migrations like this:

```
source /opt/services/object-drive-1.0/env.sh
/opt/services/object-drive-1.0/database status
/opt/services/object-drive-1.0/database migrate up #down will also work
```

Alternate config for Database Tool

Configuration can also be supplied via yaml. This is useful if you want to run schema creation on a different database, or if you require a high-privilege database user for schema write access.

Template an empty yaml file with the template command, or pipe the output of debug to a yaml file. The name of the file does not matter. It will be parsed as yaml.

```
/opt/services/object-drive-1.0/database debug > db.yml
```

To use the yaml config instead of environment variables, pass the conf flag.

```
/opt/services/object-drive-1.0/database status --conf db.yml
```

Get help for the Database Tool with

```
/opt/services/object-drive-1.0/database help
```

Start Service

Finally start the service by issuing the following command

```
service object-drive-1.0 start
```

Verify Proper Running

Tail the logs

```
tail -f /opt/services/object-drive-1.0/log/object-drive.log
```

An reference example is depicted here based upon docker log output. Yellow highlighted lines are descriptors of what follows

```
Starting, now with version information reported
2019-04-03T02:05:01.856311849Z  INFO  Starting Object Drive  {"version": "1.0.20b4 build SNAPSHOT
(80147f68)"}
Where the configuration settings come from
2019-04-03T02:05:01.856469328Z  INFO  configuration-settings  {"--conf": "./odrive.yml", "--staticRoot":
"/go/src/bitbucket.di2e.net/dime/object-drive-server/server/static", "--templateDir":
"/go/src/bitbucket.di2e.net/dime/object-drive-server/server/static/templates", "--tlsMinimumVersion": "1.2"}
What DNs will be allowed to impersonate other users. Configured via yaml file or environment variables
2019-04-03T02:05:01.85652232Z  INFO  permitted to impersonate  {"whitelisted dn":
"cn=twl-server-generic2,ou=dae,ou=dia,ou=twl-server-generic2,o=u.s. government,c=us"}
Status of the runtime whether built with Golang that leverages boring-crypto and its version
2019-04-03T02:05:01.856678828Z  INFO  boring-crypto  {"update": "4", "runtime.Version": "go1.11.5b4"}
Zookeeper discovery
2019-04-03T02:05:01.856731978Z  INFO  waiting for zookeeper to come online
2019/04/03 02:05:01 ZK try: [zk:2181]
2019-04-03T02:05:01.886261152Z  INFO  zookeeper cluster found  {"addrs": "zk:2181"}
Indication of whether encryption at rest is enabled. If disabled, a large banner will be presented warning of such.
2019-04-03T02:05:01.899398641Z  INFO  encryption of data at rest enabled
Database setup and wait for migration to expected schema state
2019-04-03T02:05:01.899677682Z  INFO  preparing certificate pool  {"filepath":
"/go/src/bitbucket.di2e.net/dime/object-drive-server/defaultcerts/client-mysql/trust", "pool": "for client"}
2019-04-03T02:05:01.899840196Z  INFO  adding pem file  {"pem":
"/go/src/bitbucket.di2e.net/dime/object-drive-server/defaultcerts/client-mysql/trust/ca.pem"}
2019-04-03T02:05:01.90135885Z  INFO  using this connection string  {"dbdsn":
"{username}:{password}@tcp(metadatadb:3306)/metadatadb?tls=custom&parseTime=true&collation=utf8_unicode_ci&readTime
out=30s"}
2019-04-03T02:05:01.909939908Z  INFO  database online with schema at version 20161230 but expecting one of
20170726,20190225. rechecking in 1 seconds for pending migration
2019-04-03T02:05:02.911138955Z  INFO  database online with schema at version 20161230 but expecting one of
20170726,20190225. rechecking in 1 seconds for pending migration
2019-04-03T02:05:03.912384458Z  INFO  database online with schema at version 20161230 but expecting one of
20170726,20190225. rechecking in 1 seconds for pending migration
2019-04-03T02:05:04.91365355Z  INFO  database online with schema at version 20161230 but expecting one of
20170726,20190225. rechecking in 1 seconds for pending migration
2019-04-03T02:05:05.914974432Z  INFO  database online with schema at version 20161230 but expecting one of
20170726,20190225. rechecking in 1 seconds for pending migration
2019-04-03T02:05:06.916291939Z  INFO  database online at schema version 20190225
Setting up local cache, s3 permenant storage, canary check, and background cache cleanup
2019-04-03T02:05:06.917475266Z  INFO  aws.credentials  {"session": "CiphertextCache", "provider": "environment
variables", "purpose": "S3 ciphertextcache"}
2019-04-03T02:05:06.918216527Z  INFO  creating cache  {"session": "CiphertextCache", "filename":
"/cacheroot/lucasmoten/44db10074421-e2164197"}
2019-04-03T02:05:06.918307132Z  INFO  ciphertextcache created  {"session": "CiphertextCache", "mount":
"/cacheroot", "location": "lucasmoten/44db10074421-e2164197"}
2019-04-03T02:05:06.918445268Z  INFO  recache from PermanentStorage  {"session": "CiphertextCache", "key":
```

"canary"}
Kafka Event configuration
2019-04-03T02:05:07.869465115Z INFO kafka config {"conf": {"KafkaAddrs":null,"ZKAddrs":null,"PublishSuccessActions":["*"],"PublishFailureActions":["*"],"Topic":"odrive-event"}}
2019-04-03T02:05:07.869756318Z INFO using fakeasyncproducer
Checking zookeeper for our path prefix
2019-04-03T02:05:07.869844471Z INFO zk connect attempt {"uri": "/services/object-drive/1.0", "address": "zk:2181", "timeout": 5}
2019-04-03T02:05:07.869895604Z INFO cache purge start {"session": "CiphertextCache"}
2019-04-03T02:05:07.885940118Z INFO zk create {"pathtype": "part 1", "newpath": "/services", "appendpath": "services"}
2019-04-03T02:05:07.889269855Z INFO zk create {"pathtype": "part 2", "newpath": "/services/object-drive", "appendpath": "object-drive"}
2019-04-03T02:05:07.897346302Z INFO zk create {"pathtype": "part 3", "newpath": "/services/object-drive/1.0", "appendpath": "1.0"}
Setting up listener for service
2019-04-03T02:05:07.903067821Z INFO preparing certificate pool {"filepath": "/go/src/bitbucket.di2e.net/dime/object-drive-server/defaultcerts/server/trust.pem", "pool": "for server"}
2019-04-03T02:05:07.903469771Z INFO adding pem file {"pem": "/go/src/bitbucket.di2e.net/dime/object-drive-server/defaultcerts/server/trust.pem"}
2019-04-03T02:05:07.910964262Z INFO enabling cipher suite {"suite": "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256"}
2019-04-03T02:05:07.911112028Z INFO enabling cipher suite {"suite": "TLS_RSA_WITH_AES_128_CBC_SHA"}
2019-04-03T02:05:07.911226403Z INFO tls minversion set {"ver": "1.2"}
2019-04-03T02:05:07.914115455Z INFO setting up announcer to check for peers
2019-04-03T02:05:07.914916466Z INFO zk mount check {"zk watch": "/cte/service/aac/1.2/thrift"}
2019-04-03T02:05:07.91567293Z INFO zk announcement check {"zk watch": "/cte/service/aac/1.2/thrift"}
2019-04-03T02:05:07.918664998Z INFO zk receive announcement {"zk watch": "/cte/service/aac/1.2/thrift", "child": "/cte/service/aac/1.2/thrift/member_0000000000"}
2019-04-03T02:05:07.918762616Z INFO zk membership change {"zk watch": "/cte/service/aac/1.2/thrift", "announcements": {"/cte/service/aac/1.2/thrift/member_0000000000":{"serviceEndpoint":{"host":"8b44ac8ea568","port":9000},"status":"ALIVE"}}}
2019-04-03T02:05:07.918994807Z INFO aac thrift client is nil and wont be able to service requests. attempting to reconnect
2019-04-03T02:05:07.925046223Z INFO zk mount check {"zk watch": "/services/object-drive/1.0/https"}
2019-04-03T02:05:07.926555685Z INFO zk mount check again {"zk watch": "/services/object-drive/1.0/https"}
bind address and port
2019-04-03T02:05:07.914553032Z INFO starting server {"addr": "0.0.0.0:4430"}
Cloudwatch state
2019-04-03T02:05:07.926799335Z INFO metrics reporting to cloudwatch disabled as OD_AWS_CLOUDWATCH_INTERVAL set to <= 0{"session": "cloudwatch"}
2019-04-03T02:05:07.927038143Z INFO aws.credentials {"provider": "environment variables", "purpose": "autoscaler SQS"}
2019-04-03T02:05:07.927632939Z INFO aws.credentials {"provider": "environment variables", "purpose": "autoscaler ASG"}
Background check for AAC availability
2019-04-03T02:05:07.928169494Z INFO waiting for aac to be created
2019-04-03T02:05:07.928360091Z INFO sqs queue is configured to be turned off
2019-04-03T02:05:09.451572848Z INFO aac chosen {"announcement": {"serviceEndpoint":{"host":"8b44ac8ea568","port":9000},"status":"ALIVE"}}
Ready to announce our availability now that dependencies have been met
2019-04-03T02:05:09.452301865Z INFO zk create {"pathtype": "protocols", "newpath": "/services/object-drive/1.0/https", "appendpath": "https"}
2019-04-03T02:05:09.455047652Z INFO zk mount check {"zk watch": "/services/object-drive/1.0/https"}
2019-04-03T02:05:09.455385468Z INFO zk create {"pathtype": "announcement", "newpath": "/services/object-drive/1.0/https/e6543fe0", "appendpath": "e6543fe0"}
2019-04-03T02:05:09.45560554Z INFO zk announcement check {"zk watch": "/services/object-drive/1.0/https"}
2019-04-03T02:05:09.457667798Z INFO zk our address {"ip": "172.18.0.9", "port": 4430}
2019-04-03T02:05:09.457760201Z INFO registering odrive AppServer with ZK {"ip": "172.18.0.9", "port": "4430", "announcementPoint": "/services/object-drive/1.0", "address": "zk:2181"}
This example has only one node, but this shows what happens when we discover any peers
2019-04-03T02:05:09.458174898Z INFO zk receive announcement {"zk watch": "/services/object-drive/1.0/https", "child": "/services/object-drive/1.0/https/e6543fe0"}
2019-04-03T02:05:09.458352954Z INFO zk membership change {"zk watch": "/services/object-drive/1.0/https",

```
"announcements":
{"/services/object-drive/1.0/https/e6543fe0":{"serviceEndpoint":{"host":"172.18.0.9","port":4430},"status":"ALIVE"}}}
```

When a log with msg "registering odrive AppServer with ZK" appears, the Object Drive server is ready to start servicing requests.

Other actions to try

- Check that Object Drive Service is shown as Up in Grey Matter Dashboard https://hostname/services
- Access API Documentation https://hostname/services/object-drive/1.0
- Retrieve Objects https://hostname/services/object-drive/1.0/objects
- Retrieve Statistics https://hostname/services/object-drive/1.0/stats
- Use the Drive App (Drive UI), https://hostname/apps/drive

The above are all run through Gatekeeper which requires Cluster Information to be defined

Setup Gatekeeper Cluster Information

The Object Drive service, like other services will be fronted by a Gatekeeper/NGINX server that routes requests in a round robin format to 1 or more Object Drive instances.

Full guidance on setting up NGINX and Gatekeeper is outside the scope of this guide, but this is the relevant portions for the mustache files

Add the value odrive to the clusters property if it does not already exist.

```
# Upstreams to look for
clusters=activity-stream,address-book,assignment-plan,...,odrive,...
```

Add the cluster definition for Object Drive, positioned alphabetically, as follows

```
### odrive ###
cluster.odrive.zk-location=/service/object-drive/1.0/https
cluster.odrive.context=/services/object-drive/1.0/
cluster.odrive.protocol=https
cluster.odrive.category=Utils
cluster.odrive.description=Object Drive 1.0 Service
cluster.odrive.local-directory=
```

```
cluster.odrive.directives=proxy_request_buffering:off,proxy_buffering:off,client_max_
body_size:100000m
```

The cluster.odrive.zk-location value is the Zookeeper Announcement point for Object Drive. The above is an example only. This value should coincide with the environment variable set below for OD_ZK_ANNOUNCE. If multiple instances of odrive are launched, this configuration is how load balancing gets done. Whenever a new odrive comes up, it registers a node containing its ip and port under cluster.odrive.protocol so that gatekeeper can load balance between odrive instances.

Log Rotation and SELinux Policy

If you have installed the object drive service on a system with SELinux enabled and enforcing its policies, then you will need to grant access to the log directory to the appropriate context. This can be done using the following commands.

```
semanage fcontext -a -t var_log_t /opt/services/object-drive-1.0/log
restorecon -v /opt/services/object-drive-1.0/log
```

An explanation of SELinux, and steps for installing the packages supporting the commands above is outside of the scope of this guide. Errors regarding permissions may be sent to the root mailbox which can be viewed via

```
cat /var/spool/mail/root
```