SecureObject Vault - Installation & Configuration

Title, Copyright Information & Trademark Notices

The AsterionDB SecureObject Vault - Installation & Configuration

Primary Author: Steve Guilford

This document is not warranted to be error-free and may be updated and/or enhanced at any time. Current versions are available on-line.

Table of Contents

1. System Requirements

Operating System

The AsterionDB SecureObject Vault is certified to run upon:

- RedHat Enterprise Linux 7
- CentOS 7
- Oracle Linux 7

Database

The AsterionDB SecureObject Vault is certified to run upon the Oracle Database, versions 12 and 18c. Enterprise and Standard editions of the Oracle Database are supported. The SecureObject Vault supports both on premises and cloud based installations.

PHP

As of this writing the AsterionDB SecureObject Vault is certified to run on 7.x versions of PHP.

You must use a version of PHP that provides the Oracle OCI8 driver. We provide instructions on accessing an Oralce maintained repository with a compliant version of PHP.

Nginx

We recommend using Nginx but other web servers (e.g. Apache) will work. Instructions to install the EPEL based release of Nginx are provided.

2. Installing The SecureObject Vault

Go through these steps and observe these prerequisites to ensure that your server(s) are configured properly.

The SecureObject Vault relies upon the following AsterionDB technologies:

- The Database Plugin Server
- The DbObscura File System Gateway
- The DbStreamer Streaming Component

You must have the required AsterionDB technology components installed upon your system before beginning the installation of the Object Vault.

The SecureObject Vault can run on a single server or be spread about on multiple servers depending upon your requirements. Basically, you just edit your connect string in the API and the API point in the web application to determine how your installation scales from one server or more. There are four basic components:

1. The Oracle Database Server
2. The Nginx/PHP Server for the REST API
3. The Nginx Server for the web application.
4. The Nginx Proxy Server for DbStreamer

This documentation is written from the perspective of a single server with a dedicated 'asterion' user. You will see how you can easily adopt this process for multiple/scaled servers.

Configuring The Database Server

We assume you know how to connect to your database and perform basic operations.

Here are the steps that you will have to go through in order to properly setup your database server.

2.1 Install the EPEL Repo

Install the EPEL Repository (Extra Packages for Enterprise Linux)

```
yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

2.2 Configure SMTP Access

The SecureObject Vault utilizes SMTP capabilities within the database to send registration and password recovery emails. Therefore, you will need to setup an email account and SMTP server that the database can use. We utilize the services as exposed by the UTL_SMTP database package supplied by Oracle to send emails from the database.

The installation procedure described here relies upon a Gmail account to do the actual sending of emails. We use postfix as a Mail Transfer Agent (MTA) on our database server. This allows us to simplify our database ACL rules for external services by relying upon a local MTA. The database interacts with the local MTA which transfers the outgoing mail to our Gmail account. Our Gmail account has several authorized email aliases that it can use for outbound emails (e.g. signup-help@asteriondb.com, hostmaster@asteriondb.com etc.) We setup postfix on our local MTA to connect to Gmail using our Gmail account credentials.

Your system administrator or database administrator may have a preferred way to enable SMTP capabilities within the Database Server via UTL_SMTP. That is fine so long as we can send emails from the database. Use whatever process works in your environment.

Here's how we set things up. First off, install postfix and it's requirement from the standard repository. Here's the command we use:

```
yum install -y postfix cyrus-sasl-plain
```

You may want to remove sendmail as it is frequently installed by default. :

```
yum remove -y sendmail
```

Next, configure postfix itself. Configuration consists of modifying postfix's config file and creating an SASL password file. Here's what we add to the config file. You'll want to modify the settings for myhostname and possibly mynetworks.

```
# Added to the end of /etc/postfix/main.cf (note modification is required for myhostname)

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination

myhostname = your.server.com

alias_maps = hash:/etc/aliases

alias_database = hash:/etc/aliases

mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128

mailbox_size_limit = 0
```

```
recipient_delimiter = +

inet_protocols = all

relayhost = [smtp.gmail.com]:587

smtp_tls_security_level = secure

smtp_sasl_auth_enable = yes

smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwords

smtp_sasl_security_options = noanonymous, noplaintext

smtp_sasl_tls_security_options = noanonymous

inet_interfaces = loopback-only

myorigin = $mydomain

mydestination =

mynetworks_style = host

smtpd_client_restrictions = permit_mynetworks, reject

#eof
```

You'll configure Postfix to access your SMTP server in the sasl_passwords file. This file defines your SMTP server and an email account / password that can be used to rely emails. Create a file called 'sasl_passwords' in '/etc/postfix'. You will change the values of:

- [smtp.server.com] to be the web address for your SMTP server
- 'xxx' is the port number of your SMTP server
- 'email@company.com:email-password' is a valid email address and password for your SMTP server. Do not use one of your aliases but an actual user account.

Here's the contents:

```
[smtp.server.com]:xxx email@company.com:email-password
```

The 'sasl_passwords' file has to be processed by postmap and have the owner:group and privs set properly. Follow these steps:

```
postmap /etc/postfix/sasl_passwords

rm /etc/postfix/sasl_passwords
```

```
chown root:postfix /etc/postfix/sasl_passwords.db

chmod 640 /etc/postfix/sasl_passwords.db
```

Don't forget to start Postfix. If you are running an EL7 branch of Linux your commands will be:

```
systemctl enable postfix

systemctl start postfix
```

2.3 Install and Configure Nginx

Here is an example of a baseline Nginx installation that we use. Your system administrator may prefer a different orchestration. Use the following as a guideline in that case.

Install Nginx from the EPEL repository.

```
yum install -y nginx
```

Depending upon the limitations you wish to impose on the size of files uploaded through the WebApplication, edit the value of CLIENT_MAX_BODY_SIZE in the HTTP section of nginx.conf accordingly:

```
# /etc/nginx/nginx.conf

http

{

    ..

    client_max_body_size 1G;

    ..

}
```

You will need to make the Nginx user/owner a member of the group associated with the user that owns the SecureObject Vault installation files - 'asterion' in our examples. This is in order to allow Nginx to serve the JavaScript based WebApplication and the PHP based REST API.

Your web systems administrator may have a preferred way to serve content. The following is a guideline to properly serve content from the AsterionDB SecureObject Vault.

```
# /etc/nginx/conf.d/asterion.conf

# Replace %object_vault_user% with the database account that owns the Object Vault schema.

server
```

```
# This is not a complete server block.  You'll have other entries in here.  We're just showing you
# what you need as far as the SecureObject Vault is concerned.
{
    location /download/
    {
        proxy_pass http://127.0.0.1:6510/%object_vault_user%/object_vault_pkg/;
        proxy_set_header X-Forwarded-For $remote_addr;
    }

    location /streaming/
    {
        proxy_pass http://127.0.0.1:6510/%object_vault_user%/object_vault_pkg/;
        proxy_set_header X-Forwarded-For $remote_addr;
    }

    location ~ ^/objVaultAPI/(.*)$
    {
        alias /home/asterion/asterion/oracle/objVault/php/objVaultAPI/public;
        include fastcgi_params;
        fastcgi_pass unix:/var/php/run/php-fpm/php-fpm.sock;
        fastcgi_param SCRIPT_FILENAME $realpath_root/index.php;
        fastcgi_param DOCUMENT_ROOT $realpath_root;
        fastcgi_param REQUEST_URI $1;
    }

    location /
    {
        try_files $uri /index.html =404;
        root /home/asterion/asterion/oracle/objVault/javaScript/objVault-webApp/build;
    }
}
```

You will need to set the value of %object_vault_user% to be the owner of the SecureObject Vault schema. In addition, you will need to properly set the values for 'root' and 'alias' in the location blocks.

We recommend using the PHP FPM CGI processor. If you setup PHP as recommended below then you will not have to change any of the fastcgi parameters.

The location entries for download and streaming allow us to use Nginx as a proxy front-end to the DbStreamer component. This is also the mechanism used to enable HTTPS streaming of content - the Nginx server acts as a HTTPS proxy server to the requesting client.

2.4 SSL Configuration

Note that we do not provide any configuration details on serving HTTPS traffic from your Nginx server. Your web server systems administrator will have a policy in place for obtaining and serving SSL certificates. You will need to consult the Nginx documentation for further information on serving HTTPS traffic.

That being said, we use LetsEncrypt as our certificate authority and Certbot for certificate maintenance. Here's our configuration statements in the Nginx site specific configuration file:

```
# /etc/nginx/conf.d/asterion.conf

server

{

    listen 443 ssl http2;

    ssl_certificate /etc/letsencrypt/lib/host.domain.com/fullchain.pem

    ssl_certificate_key /etc/letsencrypt/live/host.domain.com/privkey.pem

    include /etc/letsencrypt/options-ssl-nginx.conf

    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem

    ..

}
```

2.5 Configure PHP

This configuration assumes that you will create a PHP user. We recommend this in order to properly isolate applications from intrusion. Here's how we setup our PHP user:

```
useradd -m -U -r -d /var/php -s /bin/false php

rm -Rf /var/php/.*

mkdir -p /var/php/run/php-fpm
mkdir /var/php/chunkedUploads

chown php:php -R /var/php/*

chmod 700 /var/php/chunkedUploads

chmod 770 -R /var/php/run
```

You will need to make the Nginx user a member of the PHP group.

Enroll the PHP user as a member of the oinstall and the 'asterion' group.

Oracle provides a PHP distribution that has been compiled and linked with the OCI8 driver. The instructions for installing and using this repository are located here:

```
https://yum.oracle.com/oracle-linux-php.html
```

After you have installed and enabled the Oracle Linux repository that contains the OCI8 enabled PHP RPM, install PHP and other requirements with the following line:

```
yum install php php-oci8-12c  php-mbstring php-json php-common php-fpm
```

The default configuration of the Oracle based PHP distribution needs to be tweaked a little.

The limits on uploaded files imposed by the default PHP configuration are inadequate. Modify the following settings in /etc/php.ini to set the value accordingly (note: setting the value to 0 disables the limits). You'll also need to set the location of the temporary upload directory.

```
; /etc/php.ini

upload_max_filesize = 1G

post_max_size = 1G

upload_tmp_dir  = /var/php
```

We prefer using the Linux SOCKS based CGI listener. You'll need to setup some environment variables that get passed into the executing PHP scripts. Here are the modifications to /etc/php-fpm.d/www.conf. Edit the path values accordingly:

```
; /etc/php-fpm.d/www.conf

user = php

group = php

listen = /var/php/run/php-fpm/php-fpm.sock

listen.owner = php
listen.group = php
security.limit_extensions = .php
env[HOSTNAME] = $HOSTNAME
env[LD_LIBRARY_PATH] = '/home/oracle/oracle12.2/server/lib'
env[ORACLE_HOME] = '/home/oracle/oracle12.2/server
```

In order to start the php-fpm daemon using systemd in Oracle Linux7 you have to create a configuration file in '/etc/systemd/system/php-fpm.service.d'. These entries set the environment that is used when PHP-FPM is started. Edit the path settings according to your installation details. Here's the contents:

```
# /etc/systemd/system/php-fpm.service.d/php-fpm-oracle.conf
[Service]
Environment=ORACLE_HOME=/home/oracle/oracle12.2/server
Environment=LD_LIBRARY_PATH=/home/oracle/oracle12.2/server/lib
```

2.6 Create A Dedicated Tablespace

It is highly recommended that you have at least one dedicated tablespace that will be used to store your unstructured data. The data is stored in BLOB/CLOB columns within tables using Oracle's SecureFiles technology. When the SecureObject Vault Administrator enables a specific file-type for storage, they can select the tablespace that will be used to store the data. We strongly advise against using your 'regular' tablespaces for this and suggest creating one or more dedicated tablespaces for the BLOB/CLOB data.

Here's an example of a 'CREATE TABLESPACE' command that we typically use. You should limit your changes to the location of the data file(s) and the size values. You will encounter performance issues if you modify the other parameters.

```
ALTER SYSTEM SET DB_32K_CACHE_SIZE = 32M;

CREATE TABLESPACE "LOB_DATA"
 DATAFILE '/u05/app/oracle/oradata/ORCL/PDB1/PDB1_lobData1.dbf'
 SIZE 50G AUTOEXTEND ON NEXT 5G
 BLOCKSIZE 32K
 LOGGING
 DEFAULT NOCOMPRESS NO INMEMORY
 ONLINE
 EXTENT MANAGEMENT LOCAL AUTOALLOCATE
```

```
SEGMENT SPACE MANAGEMENT AUTO;
```

2.7 SYSDBA Access

The installation script requires SYSDBA access in order to perform the following installation actions:

- Create a user to own the SecureObject Vault schema.
- Create synonyms and grant execute privileges.
- Create a database ACL entry allowing outbound emails.

The script handles the creation of a user to own the SecureObject Vault schema. The following privileges are granted to this user:

- create session
- create table
- create sequence
- create view
- create procedure
- create trigger
- create type

The user is granted an unlimited quotas on the default and LOB data tablespaces. If you want to restrict these quotas this must be done by-hand after the installation.

2.8 Unpack the SecureObject Vault Distribution Files

The AsterionDB SecureObject Vault distribution consists of three files:

- objVault-oracle-1.5.5.1.tgz (schema objects)
- objVault-api-oracle-1.2.7.1.tgz (REST API)
- objVault-webApp-1.4.2.tgz (JavaScript based Web App)

Note that the version numbers will probably differ and are provided as an example.

Unpack the distribution files into a directory of your choice. You can create a dedicated 'asterion' user or use an existing user account. These documentation examples assume you have created a dedicated 'asterion' user.

2.9 Install The Vault Schema

A simple installation script is used in conjunction with SQL*Plus to install all of the Object Vault's schema objects. Before proceeding, it is best to determine your answers to the prompts presented by the installation script.

You will need to determine the following:

- The username and password for a database user account that can connect as a SYSDBA
- The username and password for the database account that will own the SecureObject Vault schema (e.g. object_vault_user)
- The default tablespace. This is used to store the meta-data that is associated with your unstructured data.
- The dedicated tablespace for unstructured data.
- The username of the database account that owns the DbObscura schema.
- The username of the database account that owns the DbStreamer schema.
- The username to be associated with the SecureObject Vault Administrator.
- The administrator's first name, middle initial/name, last name and email address.
- The password to be used for the administrator.
- The fully qualified DNS name of the server that hosts the SecureObject Vault installation. This is the server that provides the SecureObject Vault's web application front-end. You will find this value referenced by the server_name entry in your site-specific nginx configuration file (e.g. /etc/nginx/conf.d/asterion.conf)
- The fully qualified DNS name of the server that hosts the DbStreamer installation. This is your CDN server. This server is frequently the same one that is hosting the SecureObject Vault's web application if you are hosting off of a single machine. You will find this value referenced by the server_name entry in your site-specific nginx configuration file. It is best to use a dedicated host name for the CDN server. This allows you to scale/split processing and network loads.
- The SecureObject Vault can optionally use the proxy services provided by Nginx. It is preferred to use this (or an equivalent) proxy serving capability.
- The SecureObject Vault can be SSL enabled - thus generating HTTPS compliant resource links. Local installations frequently run with out having SSL/HTTPS enabled. If you are hosting a public server or one that is available across your corporate intranet you most certainly will want to enable SSL/HTTPS capabilities.

- You can customize the proxy links used for streaming and downloading content from the Object Vault. In most cases you will accept the default settings unless you have changed the location entries for the proxy server in your /etc/nginx/conf.d/asterion.conf configuration file.
- You will need to know the host name or IP address of your SMTP server. In our example, we use Sendmail as a local Mail Transfer Agent - therefore the default value is 127.0.0.1.
- You will need to know the email address to use for outbound emails generated by the system. Once again, in our example, we use an MTA that has credentials allowing it to connect to a Gmail based server. The credentials used also enable an email alias. In this instance, we use this alias value - 'signup-help@company.com' - as our outboud email address.

With the above information gathered and determined in advance you are now ready to run the installation script. Change directories to ./asterion/oracle/objVault/dba and execute the following command:

```
sqlplus /nolog @install
```

This will run the installation script and prompt you for the values that you have gathered. If the script completes successfully you will receive the message:

```
The AsterionDB SecureObject Vault schema has been created.
```

You will also receive a confirmation email. You have a few more steps to complete before you can confirm your account!

A log file, install.log, will be created. If errors are encountered, the easiest thing to do after fixing any issues is to drop the Object Vault user and start the installation process over.

2.10 Configure the SecureObject Vault REST API

The SecureObject Vault REST API lives in the './asterion/oracle/objVault/php/objVaultAPI' subdirectory. In this directory you will find a hidden file '.env.example'. Rename this file to '.env' (still a hidden file) and edit it's contents. You will set the following variables:

- DB_HOST
- DB_USERNAME
- DB_PASSWORD

This is a sensitive file so be sure to set the file permissions (e.g. chmod 440 .env).

You can test your REST API by bringing up the following link:

```
https://your.objectVault.server/objVaultAPI/getApiVersion
```

2.11 Build The Web Application

The distribution file for the Vault's Web Application uses a sub-directory name that incorporates a version number. Upon unpacking the distribution file you will fine a sub-directory such as:

```
./asterion/oracle/objVault/javaScript/objVault-webApp-1.4.6
```

If you have followed the recommendations for configuring Nginx, this directory will get renamed to

```
./asterion/oracle/objVault/javaScript/objVault-webApp
```

after a process that configures and compiles the JavaScript application. Don't rename it yet!

Navigate to the 'public/assets' sub-directory within the objVault-webApp directory tree. There you will find a file called 'asteriondbConfig.example'. Copy this file as 'asteriondbConfig.js'. Edit this file. You will set the following variable:

```
var asterionRestApi = 'https://your.objectVault.server/objVaultAPI'
```

Navigating back to the objVault-webApp directory you will use NPM to install, compile and build the JavaScript application:

```
npm ci ; npm run build
```

With that done you can now rename your distribution subdirectory as described above.

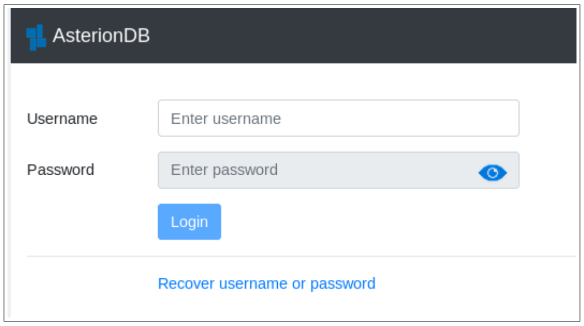
```
rm -Rf objVault-webApp  -- in case you have an existing installation that you are replacing...

mv objVault-webApp-1.4.6 objVault-webApp
```

You may find it useful to save a copy of asteriondbConfig.js. This will allow you to copy it into the './public/assets' sub-directory and save yourself the step of copying and editing the file every time you have to update the web-application.

Navigate to the home page of your SecureObject Vault web site:

```
https://your.objectVault.server/
```

If your installation and configuration was successful you will get a login screen:



2.12 Activate the Administrator's Account

Upon successful installation of the Vault's schema an account confirmation email will be sent to your SecureObject Vault Administrator's email address. Access this email and click on the link in order to complete the registration of your SecureObject Vault Administrator's account.

2.13 Enable File Types

The Object Vault, by default, only accepts for storage the specific file types that have been enabled. One of the first actions the system administrator will do is enable the desired file types for storage in the SecureObject Vault. Consult the documentation in the section Administrative Functions - File Types for further information.

2.14 Enable Account Creation (or not)

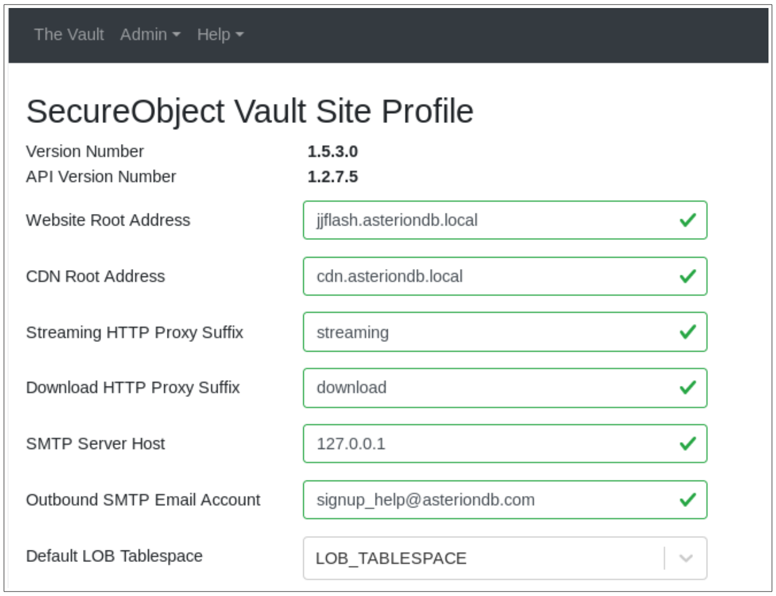
Depending upon your requirements, you may want enable your installation so that new user accounts can be created. Consult the documentation in the section Administrative Functions - Site Profile for further information.

3. Administrative Functions

The SecureObject Vault system administrator is created by the installation script. The system administrator has access to all of the functionality that a regular user does as well as additional capabilities that allow for the administration of the system.

Site Profile

The website profile page is where the system administrator adjusts the parameters that govern the operation of the Object Vault.



SecureObject Vault Site Profile

Version Number	**1.5.3.0**
API Version Number	**1.2.7.5**
Website Root Address	jjflash.asteriondb.local
CDN Root Address	cdn.asteriondb.local
Streaming HTTP Proxy Suffix	streaming
Download HTTP Proxy Suffix	download
SMTP Server Host	127.0.0.1
Outbound SMTP Email Account	signup_help@asteriondb.com
Default LOB Tablespace	LOB_TABLESPACE

Website Root Address

The website root address value is used when we have to create a web-link to the SecureObject Vault Web Application. This value will match the DNS entry for the server (or load balancer) that is hosting the SecureObject Vault Web Application.

CDN Root Address

The CDN root address value is used when we have to create a web-link to streamed or downloaded content from the SecureObject Vault. This value can be the same as the website root address. Using a different value (e.g. cdn.company.com) allows you to separate the DbStreamer component from the web server component. This can be useful in scaled and load balanced installations.

Streaming HTTP Proxy Suffix

The Streaming HTTP Proxy Suffix value is used when building links for streamed content. This value is referenced by the 'location' entries in your /etc/nginx/conf.d/asterion.conf configuration file. The configuration file provides an easy, secure way to transform external links which include the Streaming HTTP Proxy Suffix into links that can be processed by the DbStreamer component.

Here is an example showing how the Streaming HTTP Proxy Suffix is utilized in the asterion.conf configuration file:

```
location /streaming/

{
    proxy_pass http://127.0.0.1:6510/object_vault_user/object_vault_pkg/;
    proxy_set_header X-Forwarded-For $remote_addr;
}
```

This entry will transform an external link such as:

```
https://cdn.asteriondb.local/streaming/stream_object?IA3....
```

to

```
https://127.0.0.1:6510/object_vault_user/object_vault_pkg/stream_object?IA3...
```

and then pass the request on to the DbStreamer component running locally on port 6510. We are using technique to remove the schema owner (object_vault_user) and the package (object_vault_pkg) from the external link.

An additional benefit of this mechanism is that the built links are always valid regardless of any changes to underlying system topology or architecture.

Download HTTP Proxy Suffix

The Download HTTP Proxy Suffix value is used when building links for downloaded content. This value is referenced by the 'location' entries in your /etc/nginx/conf.d/asterion.conf configuration file. The configuration file provides an easy, secure way to transform external links which include the Download HTTP Proxy Suffix into links that can be processed by the DbStreamer component.

Here is an example showing how the Download HTTP Proxy Suffix is utilized in the asterion.conf configuration file:

```
location /download/
```

```
{
    proxy_pass http://127.0.0.1:6510/object_vault_user/object_vault_pkg/;
    proxy_set_header X-Forwarded-For $remote_addr;
}
```

This entry will transform an external link such as:

```
https://cdn.asteriondb.local/download/download_object?IA3....
```

to

```
https://127.0.0.1:6510/object_vault_user/object_vault_pkg/download_object?IA3...
```

and then pass the request on to the DbStreamer component running locally on port 6510. We are using technique to remove the schema owner (object_vault_user) and the package (object_vault_pkg) from the external link.

SMTP Server Host

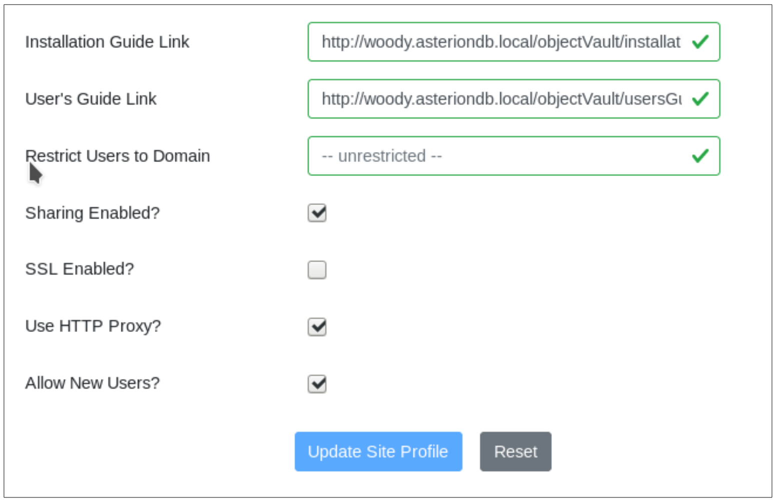
This is the name of the SMTP server (MTA) that will be used when sending outbound emails. Refer to the discussion on Configuring SMTP Access in the Installation and Configuration section for further information.

Outbound SMTP Email Account

This is the email account that will be used when sending outbound emails. Refer to the discussion on Configuring SMTP Access in the Installation and Configuration section for further information.

Default LOB Tablespace

The LOB data for the file-types that are enabled in the SecureObject Vault will be stored in this tablespace.



Installation Guide Link	http://woody.asteriondb.local/objectVault/installat ✓
User's Guide Link	http://woody.asteriondb.local/objectVault/usersGu ✓
Restrict Users to Domain	-- unrestricted -- ✓
Sharing Enabled?	☑
SSL Enabled?	□
Use HTTP Proxy?	☑
Allow New Users?	☑

Installation Guide Link / User's Guide Link

This provides the location of the SecureObject Vault Installation & Configuration Guide and the User's Guide. The default values point to the most current documentation on AsterionDB's server in the cloud. If you want to point to local copies you can enter the links here.

Restrict Users to Domain

If you provide a proper email domain (i.e. yourcompany.com) The Vault will require all users that sign-up to have email addresses within that domain.

Sharing Enabled

You can enable sharing for your installation by checking this box.

SSL Enabled

The value of SSL Enabled is taken into consideration when the system builds links for content or the web application. If you are running a local installation that is not exposed to the internet you will probably not have SSL enabled. For a local installation that is SSL enabled, you will have to use a self-signed certificate of some other mechanism that provides a valid SSL certificate.

If you are running a installation that is available on a corporate intranet or the public internet you will most certainly have SSL enabled.

Use HTTP Proxy

We recommend that you utilize Nginx as a proxy server to front-end the DbStreamer component. Doing so allows the system to construct links that prevent sensitive information from being revealed while also providing an easy mechanism to load-balance DbStreamer components.

Using Nginx as the proxy server is not a absolute requirement. Many other web servers are capable of acting as a proxy server in the same manner to which we employ Nginx.

Allow New Users

Allowing new users to sign up can be enabled or disabled by toggling this value.

File Types

The File Types page allows the administrator to enable specific file types for storage in the Object Vault. The SecureObject Vault will not accept a file type - as determined by it's file extension - unless it has been enabled for storage.

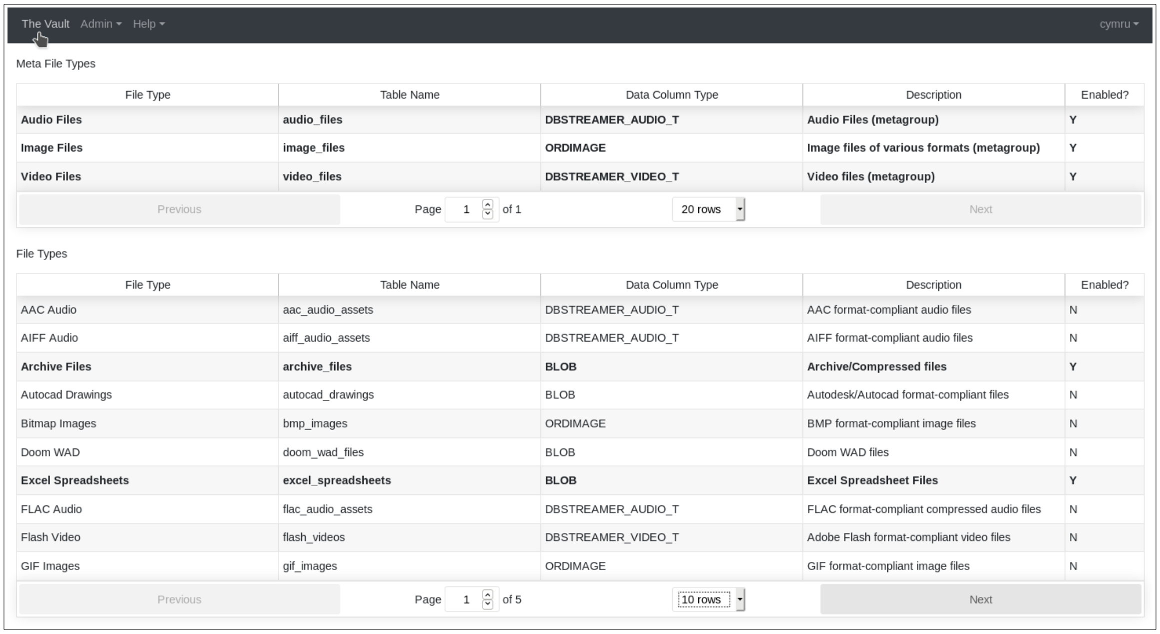
There are two classes of file types:

1. Meta-File Types - A collection of file types.
2. File Type - A singular, specific type of file.

What is significant here is the fact that each type of file will be stored in it's own table. Using meta-file types as opposed to strictly using singular file types is that it allows us to reduce the number of tables needed to store data.

Consider image file types; there are many different types of image files from PNG to JPEG, TIFF, BMP etc. If we were to enable each image type separately, we would have individual tables for PNG images, JPEG images and so forth. Using a meta-file type allows us to aggregate all image types within a single table. There may be operational reasons to separating out image types into individual tables. Using meta-file types and specific file-types gives us this flexibility.

Here is the File Types screen:



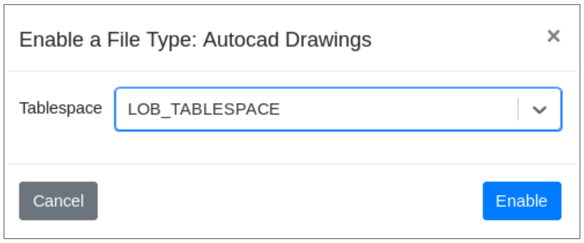
Meta File Types

File Type	Table Name	Data Column Type	Description	Enabled?
Audio Files	**audio_files**	**DBSTREAMER_AUDIO_T**	**Audio Files (metagroup)**	**Y**
Image Files	**image_files**	**ORDIMAGE**	**Image files of various formats (metagroup)**	**Y**
Video Files	**video_files**	**DBSTREAMER_VIDEO_T**	**Video files (metagroup)**	**Y**

File Types

File Type	Table Name	Data Column Type	Description	Enabled?
AAC Audio	aac_audio_assets	DBSTREAMER_AUDIO_T	AAC format-compliant audio files	N
AIFF Audio	aiff_audio_assets	DBSTREAMER_AUDIO_T	AIFF format-compliant audio files	N
Archive Files	**archive_files**	**BLOB**	**Archive/Compressed files**	**Y**
Autocad Drawings	autocad_drawings	BLOB	Autodesk/Autocad format-compliant files	N
Bitmap Images	bmp_images	ORDIMAGE	BMP format-compliant image files	N
Doom WAD	doom_wad_files	BLOB	Doom WAD files	N
Excel Spreadsheets	**excel_spreadsheets**	**BLOB**	**Excel Spreadsheet Files**	**Y**
FLAC Audio	flac_audio_assets	DBSTREAMER_AUDIO_T	FLAC format-compliant compressed audio files	N
Flash Video	flash_videos	DBSTREAMER_VIDEO_T	Adobe Flash format-compliant video files	N
GIF Images	gif_images	ORDIMAGE	GIF format-compliant image files	N

To enable a specific file type, double click on its row. A modal window will pop-up allowing you to select the tablespace that will be used to store the LOB data.

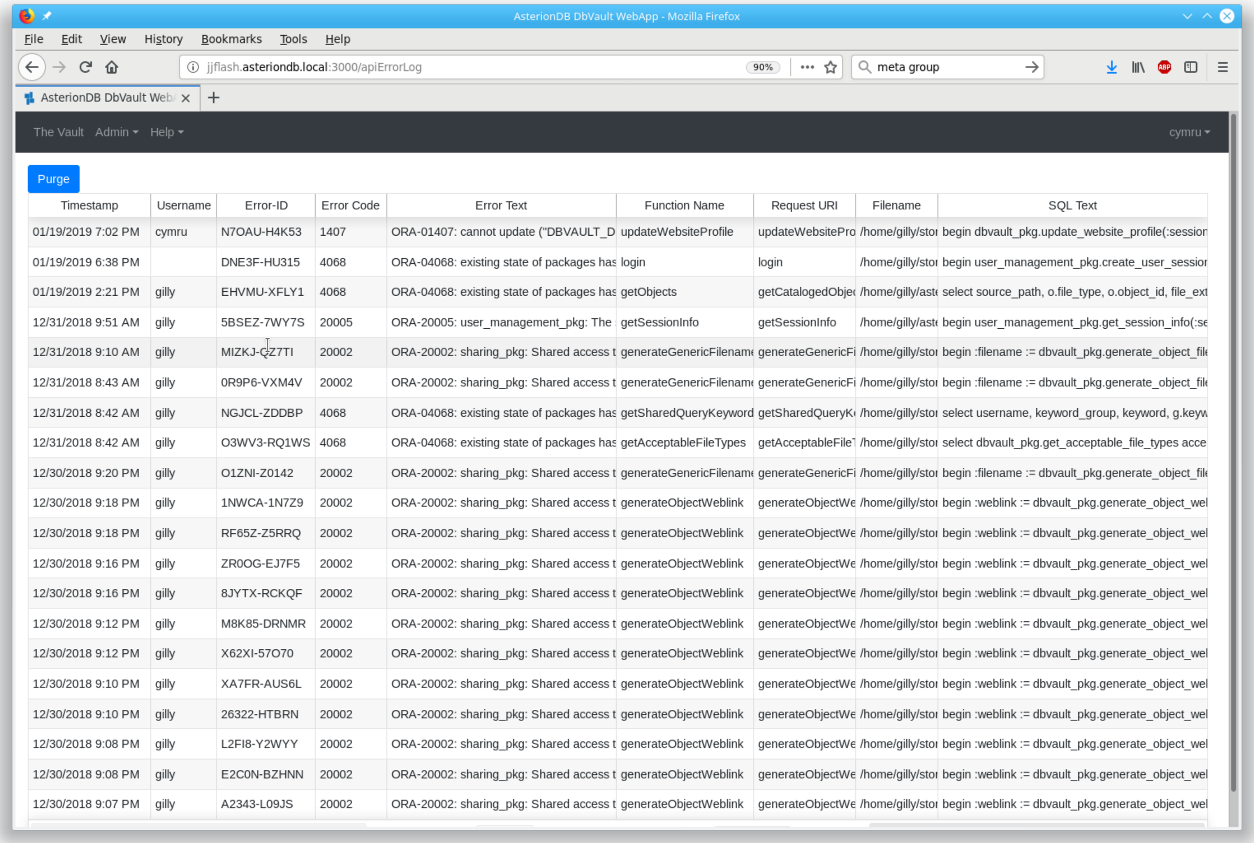


Enable a File Type: Autocad Drawings

Tablespace: LOB_TABLESPACE

API Error Log

The API Error Log allows the administrator to monitor errors.


AsterionDB DbVault WebApp - Mozilla Firefox
File Edit View History Bookmarks Tools Help
jjflash.asteriondb.local:3000/apiErrorLog
90%
meta group
AsterionDB DbVault Web
The Vault
Admin
Help
cymru
Purge

Timestamp	Username	Error-ID	Error Code	Error Text	Function Name	Request URI	Filename	SQL Text
01/19/2019 7:02 PM	cymru	N7OAU-H4K53	1407	ORA-01407: cannot update ("DBVAULT_D	updateWebsiteProfile	updateWebsitePro	/home/gilly/stor	begin dbvault_pkg.update_website_profile(:session
01/19/2019 6:38 PM		DNE3F-HU315	4068	ORA-04068: existing state of packages has	login	login	/home/gilly/stor	begin user_management_pkg.create_user_session
01/19/2019 2:21 PM	gilly	EHVMU-XFLY1	4068	ORA-04068: existing state of packages has	getObjects	getCatalogedObje	/home/gilly/aste	select source_path, o.file_type, o.object_id, file_ext
12/31/2018 9:51 AM	gilly	5BSEZ-7WY7S	20005	ORA-20005: user_management_pkg: The	getSessionInfo	getSessionInfo	/home/gilly/aste	begin user_management_pkg.get_session_info(:se
12/31/2018 9:10 AM	gilly	MIZKJ-CZ7TI	20002	ORA-20002: sharing_pkg: Shared access t	generateGenericFilename	generateGenericFi	/home/gilly/stor	begin :filename := dbvault_pkg.generate_object_file
12/31/2018 8:43 AM	gilly	0R9P6-VXM4V	20002	ORA-20002: sharing_pkg: Shared access t	generateGenericFilename	generateGenericFi	/home/gilly/stor	begin :filename := dbvault_pkg.generate_object_file
12/31/2018 8:42 AM	gilly	NGJCL-ZDDBP	4068	ORA-04068: existing state of packages has	getSharedQueryKeyword	getSharedQueryK	/home/gilly/stor	select username, keyword_group, keyword, g.keyw
12/31/2018 8:42 AM	gilly	O3WV3-RQ1WS	4068	ORA-04068: existing state of packages has	getAcceptableFileTypes	getAcceptableFileT	/home/gilly/stor	select dbvault_pkg.get_acceptable_file_types acce
12/30/2018 9:20 PM	gilly	O1ZNI-Z0142	20002	ORA-20002: sharing_pkg: Shared access t	generateGenericFilename	generateGenericFi	/home/gilly/stor	begin :filename := dbvault_pkg.generate_object_file
12/30/2018 9:18 PM	gilly	1NWCA-1N7Z9	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:18 PM	gilly	RF65Z-Z5RRQ	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:16 PM	gilly	ZR0OG-EJ7F5	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:16 PM	gilly	8JYTX-RCKQF	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:12 PM	gilly	M8K85-DRNMR	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:12 PM	gilly	X62XI-57O70	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:10 PM	gilly	XA7FR-AUS6L	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:10 PM	gilly	26322-HTBRN	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:08 PM	gilly	L2FI8-Y2WYY	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:08 PM	gilly	E2C0N-BZHNN	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel
12/30/2018 9:07 PM	gilly	A2343-L09JS	20002	ORA-20002: sharing_pkg: Shared access t	generateObjectWeblink	generateObjectWe	/home/gilly/stor	begin :weblink := dbvault_pkg.generate_object_wel

Alphabetical Index