Application Interfaces for the OSVault Archive Server

Overview

This document gives an overview of the application interfaces of the OSVault Archive server, both from a data access perspective and a management perspective. There are details on both user and programmatic interfaces, however the user interfaces are covered in depth in the *OSVault User Manual* and the *Advanced Web Administration Manual*. The programmatic interfaces, generally referred to as APIs are documented and examples are provided. Additionally, the source code for the server side functions is open-sourced under the *Artistic License*, so referring to the source for the particular function should be helpful when implementing a programmatic interface.

Data Storage Functions

Function	User Action	OSVault Action
Retrieve data from archive	Open file (no special API required, use CIFS, NFS or FTP)	If file is not resident, OSVault transparently retrieves from removable media. If removable media is not resident, notification is sent to syslog and error returned to application
Store data in archive	Open and write file (no special API required, use CIFS, NFS or FTP)	OSVault stores new files in magnetic disk cache and copies it later to removable media. When additional magnetic disk cache is required, OSVault purges files from magenetic disk cache.
Migrate data from disk cache to archive	Automatic, no user action required	OSVault copies files from disk cache to removable media according to migration rules setup by systems administrator
Query Standard File Metadata	Use standard O/S interfaces	OSVault returns standard file system information
Query extended File Metadata	Use nAPIs described in this document	The nAPIs are served thru a network interface and can be called from remote systems.

Management Interfaces

For management, the archive supports both user interfaces from command line and web-based graphical interfaces, as well as application program interfaces (APIs) to retrieve or set management data.

Graphical and Command-Line User Interfaces

Web-based administration

Basic interface is on port 80. This links to a simple interface on port 446 using secure HTTP (https). Advanced administration pages are on port 10000 (this is the WEBMIN package).

Command Line Interfaces

Please see the OSVault User Manual for details on the command line interface for OSVault. The following describes the method to access the CLI interface.

Secure Shell (ssh)

Secure Shell (ssh) is the preferred method to invoke command line access to the OSVault archive server. Secure shell encrypts all communications with the library, making it difficult for unauthorized use of the library, even across a public Internet. Ssh is included with all major operating systems, except Microsoft Windows. For Microsoft Windows use, we recommend the Putty program or the Cygwin environment. Putt is by far the easiest solution and can be downloaded directly at http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html.
Cygwin installs a LINUX environment on a Windows operating system and can be found at http://www.cygwin.com.

TELNET

TELNET is disabled for security reasons, but can be enabled by using the Secure Shell to edit /etc/xinetd.d/telnet

Management APIs

OSVault has an network accessible Application Program Interface (nAPI) for querying the status of files or media stored under its control. The nAPI is implemented via the HTTP protocol (port 80 by default) and can be invoked from any programming language that supports TCP/IP sockets or via any available web browser.

Server Implementation

The nAPI server is implemented as a CGI-BIN program under the web server of the system running the Opti-Vault software.

Client Implementation

The nAPI client is any program that invokes the nAPI over the network. Although the results of the nAPI calls are displayed by the web browser (see below), the output is structured for input to an application program.
There are two interfaces, filestat for retrieving the status of a file stored in the /cache directory (and possibly migrated to the DVD library) and mediastat to retrieve the current status of media stored (or previously stored) in the DVD library. Filestat requires an argument, the file name.

Web Browser Invocation

The status of a file in the OptiVault cache directory can be retrieved via a secure or non-secure HTTP request on port 80. You can test this interface from any web browser by invoking the following URL (web link):

To retrieve a file status, use the following link (URL) in your browser address field:
https://server/cgi-bin/filestat?fn=test
or
https://server/cgi-bin/filestat?fn=test

To retrieve the media status, use the following URL:
https://server/cgi-bin/mediastat
or
http://server/cgi-bin/mediastat

PERL Invocation

```
#!/usr/bin/perl
use LWP::UserAgent;
$FILENAME = ‘myfile’;
$URL = 'http://server/cgi-bin/filestat?fn=$FILENAME’;
$request = new HTTP::Request('GET', "$URL");
```

The LWP modules can be retrieved from the libwww modules at a CPAN of your choice.

“C” Programming Language

The “C” language allows for greater control of the interface to the nAPI server. In general the “C” program performs the following actions:

- Create a network socket (usually with the “socket” system call under Windows or UNIX/LINUX)
- Bind the socket to port 80 with the servers IP address with the bind routine
- Write the request to the socket with the string “GET CGI-BIN/filestat?fn=filename” or “GET CGI-BIN/mediastat”.

- Read the return results from the socket until the read command terminates with a 0 (or an error)
- If you desire to use Secure Sockets, see the reference manual for the particular system you are using.

For more information on related C subroutines, see the CGIC project at **ttp**://freshmeat.net/redir/cgic/1187/url_homepage/cgic

In particular see the HTTP protocol RFC,
available at http://www.ietf.org/rfc/rfc2068.txt.

There is a program that does HTTP protocol actions at
http://www.gnu.org/software/wget/wget.html

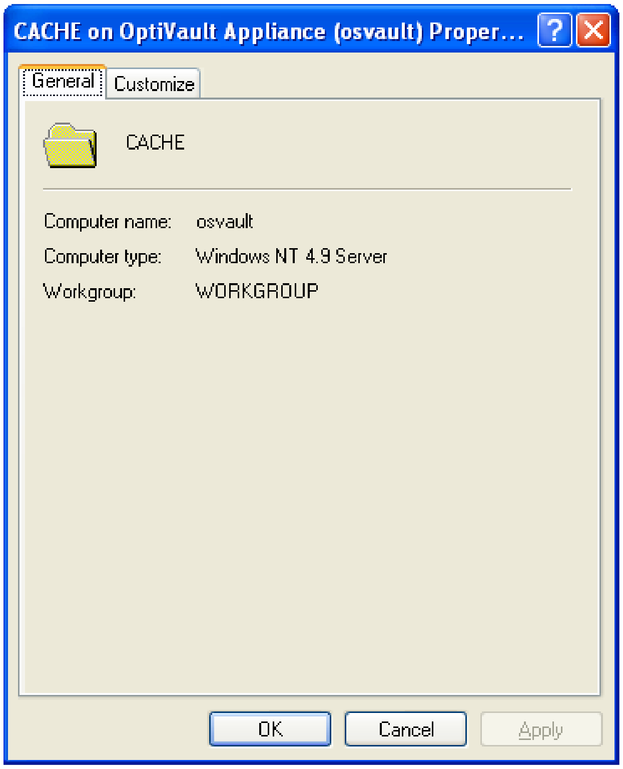
If you wish to use Microsoft Visual C++, you can use the WinHTTP library to access the nAPI. Please contact Open Source Storage Inc. (support@osstorage.com) if you wish to receive sample programs that use this library.

Network Storage Access Interfaces

The OSVault archive supports the following protocols in the manners described:

CIFS/Windows Share

From Windows, the server shares two directories: CACHE and MEDIA. From a Windows client, the file system looks like a standard Windows NT 4.9 Server, as shows in the screen shot below.



CACHE on OptiVault Appliance (osvault) Proper...

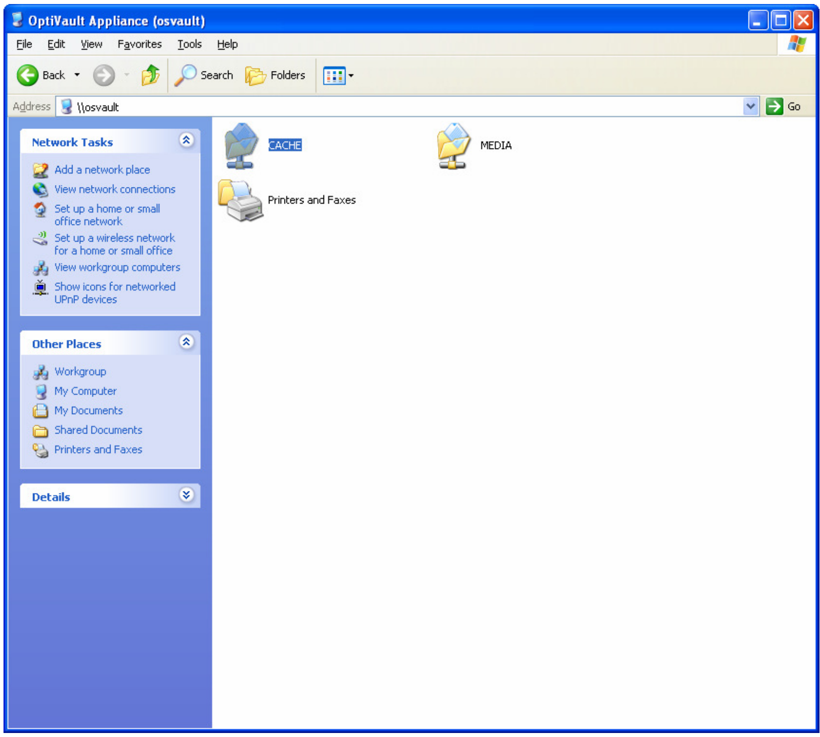
CACHE

Computer name: osvault

Computer type: Windows NT 4.9 Server

Workgroup: WORKGROUP

The shares on the OSVault server are seen as shown below:



You should, in nearly all cases, use CACHE for data storage and retrieval. CACHE is a hard drive file system using the XFS journalling file system.

The MEDIA share is direct access to files stored on removable optical media. Under MEDIA are a series of directories, each directory corresponding to the label on the optical media in question. Generally, the subdirectories under MEDIA do not appear in the CIFS browser window. To enable the subdirectories (media labels) to appear, the OSVault server must be reconfigured to enable "ghosting".

The CIFS shares are implemented with standard SAMBA 3.0.10. The redirection of file access to either hard drive or optical media occurs transparently to the Samba software at the operating system level using the XDSM standard. The only noticeable effect to the clients is a possible delay of up to 15 seconds while removable optical media is mounted.

Opportunistic locks on the client system can cause unwanted network file access problems. On the client system, oplocks should be disabled per Microsoft instructions. On the OSVault server, oplocks are disabled as much as possible.

NFS Version 3 and 4

This is standard LINUX NFS as implemented in the 2.6.16 kernel. If using NFS access, care should be taken on the client system to set the "timeo" mount parameter correctly, at approximately 1 to 10 seconds (rather than the default .01 seconds). This is particularly important with NFS version 3.

The OSVault server follows standard procedures to inform the network of file access delays due to removable optical media mount times.

FTP

FTP access on the server is supported by the WU-FTP package as described at http://www.wu-ftpd.org/ The package is not modifed from the standard distribution.