WebDAV: What It Is, What It Does, Why You Need It

Luis O. Hernández
Institutional Technology
Ringling School of Art & Design
Sarasota, FL 34234
941-359-7633

lhernand@ringling.edu

Mahmoud Pegah
Institutional Technology
Ringling School of Art & Design
Sarasota, FL 34234
941-359-7633

mpegah@ringling.edu

ABSTRACT

Legacy network file services such as NFS, SMB/CIFS, and AFP changed the way we worked. File services running on cross-platform networks allowed us to save our files on remote systems and changed the way we exchanged files with each other. Web technology changed our lives and the way we work again. In the fall of 2002, due to the demand for a web integrated network file service with powerful content management capabilities, we are offering WebDAV (Web-based Distributed Authoring and Versioning) service in our instructional computer laboratories at Ringling School of Art and Design. WebDAV is a set of extensions to HTTP (the protocol that drives the web), which allows users to collaborate with each other to edit and manage files on web servers across the network.

WebDAV allows our institution to host a shared file space for each course. Students in a course can access the shared space for collaboration with the instructor and other students in the same course. Access to the WebDAV share is seamless by incorporating LDAP (Lightweight Directory Access Protocol) into the system to maintain single sign-on.

In our implementation, we have integrated the Apache web server, Apache extensions for WebDAV and the LDAP based meta-directory model for user authentication information. Furthermore, we have implemented a Perl script, which runs every hour to monitor WebDAV storage usage and triggers alarms for any disk quota breach.

In this technical report, we will share our framework, tips and tricks we learned from our deployment of WebDAV with our colleagues. We will explain how legacy network file services can be strengthened by providing WebDAV service on both the Windows and Unix platforms. Furthermore, we will discuss the issues we had to address and the lessons we learned during and after the deployment process.

Categories and Subject Descriptors

D.4.3 [**Operating Systems**]: File Systems Management – *access methods, distributed file systems, file organization.*

General Terms

Documentation, Performance, Design, Reliability, Security

Keywords

Apache, Authoring, File Service, File Sharing, HTTP, Internet, LDAP, Server, Quota, Versioning, Web, WebDAV

1. INTRODUCTION

WebDAV (Web Distributed Authoring and Versioning) allows for file management through a web server by extending the HTTP protocol. WebDAV is documented in RFC 2518 and extended in RFC 3253. RFC 2518 specifies a set of methods, headers, and content-types ancillary to HTTP/1.1 for the management of resource properties, creation and management of resource collections, namespace manipulation, and resource locking (collision avoidance) [20]. RFC 3253 specifies a set of methods, headers, and resource types that define the WebDAV versioning extensions to the HTTP/1.1 protocol [21]. For common users, WebDAV allows web development teams and other workgroups to use a remote web server as easily as if it were a local file server [2].

Software vendors have adopted WebDAV with minor deviations from the RFCs, such as authoring extensions and versioning. These deviations from the RFCs are not critical for clients to access files from a server. Mac OS X, Apple's latest operating system, offers WebDAV support to the desktop level. Microsoft Windows has supported WebDAV since Windows 2000 and Office 2000 [9].

Since WebDAV support to the desktop level is available for two of the most popular operating systems, institutions can offer WebDAV shares instead of the compute-intensive SMB shares, or the incompatible AFP or NFS shares. For instance, every connection to a Samba share requires a separate running process that does not expire until the share is closed. If an institution already uses a web server such as the Apache web server, the institution can easily lower the use of Samba shares and free the load on the servers for other uses.

Apple's AFP has been available since 1984. Although the protocol has improved over time, support for the aging protocol is thinning in favor of NFS as of Mac OS X. Although NFS is the de facto standard for sharing files across multiplatform systems, Windows support is non-existent without elaborate and expensive support from third-party vendors. WebDAV offers a solution compatible with both systems that is easy to use and inexpensive to implement.

In this report, we will explain our approach to using WebDAV on our campus, installation, configuration, the lessons we learned from our framework installation, alternative clients available for WebDAV support in operating systems other than Windows and Mac OS X, and suggestions that would make WebDAV easier to administer and use.

2. DISTRIBUTIONS, INSTALLATION, AND CONFIGURATION

Enabling WebDAV support in the Apache web server is a straightforward process. A web server administrator needs to find the package distributions, compile the module into the Apache web server, and then add a few lines to the Apache configuration file to enable web shares in specific folders.

2.1 Distributions Used

An Apache module for WebDAV support within the Apache web server has been available since 2000 [15]. Since then, the WebDAV module has been updated regularly to include new functionality and support for later Apache distributions.

To maintain single sign-on, another module can be included during compilation. The LDAP Authorization Module has been available for the Apache web server since 1998 [11]. The module allows administrators to bypass *htpasswd* generated files for authentication within Apache and instead use an LDAP server to authenticate users with their existing accounts. A simple change within the standard WebDAV configuration in the Apache configuration file allows user authentication to WebDAV shares against an LDAP server. The installation and configuration that follows refers to the following distributions: Apache web server version 1.3.27, WebDAV module version 1.3.6, and LDAP module version 2.3.

2.2 Installation

Experienced Apache administrators will find the WebDAV and LDAP module installations straightforward [11][14][19]. During the configuration of the Apache source distribution, the administrator must specify that both the WebDAV and LDAP module should be marked for inclusion during compilation of the package:

```
# Apache web server configure command
./configure \
--prefix=/local/apache_1.3.27 \
--enable-module=proxy \
--activate-module=src/modules/dav/libdav.a \
--activate-
module=src/modules/ldap/mod_auth_ldap.c
```

Library modules should be compiled statically to improve performance during execution time and startup time [5], and to allow for binary portability across other systems with the same operating system as where the package is originally compiled. Another advantage of static compilation is that the Apache web server software is generally upgraded more often than the Apache modules. The ease of administration that Apache's Dynamically Shared Objects (DSO) provides diminishes when the administrator must install a newer version that includes security enhancements as DSO modules must be recompiled.
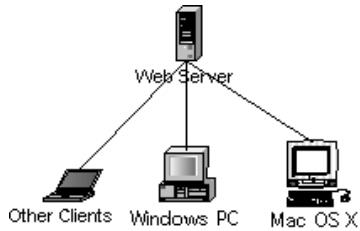
2.3 Configuration

Once the WebDAV and LDAP modules are compiled into the Apache web server, the web server configuration does not require any changes until WebDAV functionality is required. The WebDAV module can be enabled in the Apache configuration as follows [14]:

```
# Enable WebDAV support
Addmodule mod_dav.c
DAVLockDB /local/apache/var/DAVLock
```

To enable standard WebDAV support within specific web folders using *htpasswd* generated files, the administrator must include several WebDAV configuration parameters in the Apache configuration within the web folder chosen for WebDAV support. The configuration consists of activating WebDAV support for the specific folder, and limiting access to the following requests using the "Limit" directive: PUT, POST, DELETE, PROPFIND, PROPPATCH, MKCOL, COPY, MOVE, LOCK, and UNLOCK. Browsing should be disabled for WebDAV enabled folders that should not be accessible over the web. To do so, the attribute "GET" and "POST" should be added to the "Limit" directive. The tool *htpasswd* (created during the Apache web server compilation and generally located in the apache/bin folder) is used to store user accounts and passwords to a file. For security purposes, the password file must be saved to a folder not included in the web server's *DocumentRoot*; otherwise, the file will be accessible from the Internet. The file is referenced with the directive "AuthUserFile." The attribute "require user [username]" within the "Limit" directive allows the administrator to specify which users in the "AuthUserFile" have access to the web share.

A simple WebDAV setup consists of a web server and the clients that connect to the service as shown in Figure 1. The following is an example of a standard WebDAV configuration section within the Apache configuration file:

```
# WebDAV folder for a Project
<Directory "[DocumentRoot]/project">
        DAV On
#       RSADShareDAVOwnerUID [owneruid]
#       RSADShareDAVSize [Quota in MB]
        AllowOverride None
        AuthName "[Project Title]"
        AuthType Basic
        AuthUserFile [/.hiddenFolder/.hiddenFile]
        <Limit [PUT DELETE PROPFIND
PROPPATCH MKCOL COPY MOVE LOCK
UNLOCK]>
                Require user [user names]
                Order Deny,Allow
                Deny from all
                Allow from trusteddomain.com
        </Limit>
</Directory>
```
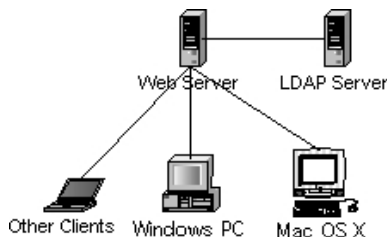
Figure 1. Simple WebDAV Configuration.

Administrators can maintain single sign-on by compiling the LDAP module with the web server. LDAP authentication to the WebDAV shares is made possible by removing the "AuthUserFile" directive and adding the following directives: LDAP_Server, LDAP_Port, Base_DN, and UID_Attr [1]. The "require user [username]" attribute can still be used to limit authentication to specific usernames; however, LDAP allows authentication based on username, group membership, or any other attribute found in the directory service.

A WebDAV setup with LDAP authentication consists of a web server, an LDAP server, and the clients that connect to the WebDAV service as shown in Figure 2. The configuration is similar to a basic WebDAV configuration. The following is an example of a WebDAV configuration with LDAP authentication enabled within the Apache configuration file:

```
# WebDAV folder for a Project
<Directory "[DocumentRoot]/project">
        DAV On
#       RSADShareDAVOwnerUID [owneruid]
#       RSADShareDAVSize [Quota in MB]
        AllowOverride None
        AuthName "[Project Title]"
        AuthType Basic
        <Limit [PUT DELETE PROPFIND
PROPPATCH MKCOL COPY MOVE LOCK
UNLOCK]>
                Require user [user names]
                Order Deny,Allow
                Deny from all
                Allow from trusteddomain.com
        </Limit>
        LDAP_Server ldap.yourdomain.com
        LDAP_Port 389
        Base_DN "[BaseDN]"
        UID_Attr uid
</Directory>
```


Figure 2. WebDAV and LDAP Configuration.

It is important to note that one user owns Apache web server processes. The same user must also own all web shares to which users will write files; otherwise, the web shares will exist as read-only.

The attributes "RSADShareDAVOwnerUID" and "RSADShareDAVSize" are not standard Apache or WebDAV configuration attributes. A monitoring tool described later in the report uses the attributes to monitor the share disk usage. As you can see, they are commented out so the web server does not use them.

2.4 Other Distributions

WebDAV support is available on several distributions other than for the Apache web server. Available alternatives exist as servers, clients, and as plug-ins.

2.4.1 Servers

Several other web servers besides the Apache web server are available with embedded WebDAV support in varying forms. Microsoft's Internet Information Server [10] and Mac OS X Server [3] support WebDAV out of the box. Although both platforms have added functionality in their implementations that are not compatible outside of their environment, any WebDAV client is able to connect to the servers without the need to use the added functionality of the individual implementation.

Another server on the open source arena is the Neon web server, which is in development stages (release 0.23.9). Neon is an HTTP and WebDAV client library, with a C interface which features integrated WebDAV support [12].

2.4.2 Clients

WebDAV support is embedded in the latest version of Microsoft Windows, Mac OS X, and *GNOME Nautilus* and *KDE* for Linux. Client alternatives exist for other systems: *Goliath* for Mac OS 9 and *Java DAV Explorer* for the JVM. A command-line interface is also available for use in UNIX terminals, called *Cadaver* [15].

2.4.3 Additional Distributions

Some application programs use plug-ins to enable WebDAV capabilities. Such plug-ins includes: *jEdit* WebDAV plugin for the *Jedit* text-editing application [16] and *Eldav* for *Emacs* [18].

Application programming interfaces (API) are also available for popular programming languages: *PerlDAV* [4] and *PythonDAV* [13].

A Linux file system driver, DAVfs, is available and allows mounts of web folders as a disk drive [17].

3. How to Connect to a Web Share

The process to connect to a web share depends on the operating system or the client used. Here are the steps required to connect from Windows or Mac OS X.

3.1 Connecting from Windows

The following steps allow a user to connect to web folders from a Windows machine:

- Open Internet Explorer and click on File->Open...

- Enter the full URL of the web share. For example, http://www.domain.com/WebShare.

- Check the box on "Open as Web Folder" and click "Ok."

3.2 Connecting from Mac OS X

The following steps allow a user to connect to web folders from Mac OS X.

- Within the Finder, click on Go->Connect to Server... or press "Command Key + K."

- Enter the full URL of the web share. For example, http://www.domain.com/WebShare.

- The "Connect" button will trigger a login screen.

- Upon successful login, a mounted drive will appear on the desktop. For security purposes, the drive should be unmounted when no longer in use.

When users connect to the web shares, they need to log in using their username and password. Upon successful authentication, the user is able to view the web folder as any other folder in the computer.

4. MONITORING TOOLS

Since the Apache web server should not run as root due to security concerns, the web server will not be able to *sudo* and act on behalf of other users. The user that owns the web server process must also own all WebDAV enabled folders and files within. This creates a situation in which users can write to a file system without a quota, allowing the file system to become full.

Perl scripts can monitor the usage of each WebDAV enabled folder in a system [7]. A *cronjob* can be set every hour to activate the Perl script. The script scans the Apache configuration file for each WebDAV share, for attributes that reveal the owner of each share, and for the space allotted to each share. The Perl script then calls the UNIX command *du* on each share to determine disk

usage. After comparing the allotted space and the disk usage, it informs the system administrator of any quota breaches, the owner of any delinquent share, and the allotted space for the share. The script can be web-enabled and can output the results of all shares to a web page for use by any IT staff member that requires access to the information.

The load of the WebDAV enabled web server is negligible since the added load of the web shares is similar to that of an NFS file system. *Webalizer* is a tool used for web server statistical data that creates snapshots of daily usage in web servers. Last semester we offered twenty-three WebDAV shares, several of which were used simultaneously by classes of more than twenty students. *Webalizer* reported that web server hits and visits barely increased while WebDAV shares were in use.

5. SECURITY

As with any web service, WebDAV comes with its share of security concerns, specifically authentication and web based attacks [22].

5.1 Authentication

Basic authentication sends the user's password in clear text format across the network, allowing the possibility of sniffers to steal passwords. WebDAV shares can be limited to accept incoming connections that generate from specific domains, allowing an institution to use basic authentication since the allowed domains should be trusted. Other solutions available for authentication include shared keys and encrypted connections.

5.2 Denial of Service

Web shares are prone to Denial of Service (DoS) attacks. Given the possibility of multiple users flooding the server with large files, the server would not be able to respond to other requests until the large files are processed. More server memory and an increase in the number of allowed Apache web server processes would diminish the probability of DoS attacks. The increase in number of allowed web server processes can be accomplished through the web server configuration file.

5.3 Module Signature

An attacker is able to determine exactly what a web server is running by visiting http://www.netcraft.net. The WebDAV module is identified as an addition to the web server. Unless the attacker is aware of the exact location of the web share, or unless anonymous shares exist, an attacker cannot compromise the system if proper authentication mechanisms are in place.

6. LESSONS LEARNED

We have learned how to work around issues inherent to the WebDAV module implementation: mainly no quota system, a single authoritative user, and how to replicate the installation in other systems. In the following sections are some suggestions on how to make the system stronger and more usable through modifications to the module and through third-party tools.

6.1 No Quota System

As explained previously, customized monitoring tools are necessary to monitor disk usage of the shares or users can abuse disk space limits. In our experience, technology savvy users generally do not abuse disk space. During setup, we ask the

faculty member how much space is required for the share and generally, the class stays within limits. We have only experienced few instances when we must call upon the faculty member to clean up the share to conserve space. Faculty members have been helpful and cooperative in the cleanup process and in following up with students who use the most space.

When we create a web share, we also create a folder for each user of the share to place his/her work. Separate student folders allow the faculty and IT to monitor disk space usage more efficiently.

6.2 Single Authoritative User

Even though users can overwrite each other's work since one user owns the web server processes, students have been respectful of each other's work while working in the shared space. Regardless, limiting the web server to run as only one user also limits the capability and potential applications that can be created with the web server. For example, if the web server were allowed to change running user identification, the web shares would not need to be owned by one user and disk quotas could potentially be applied to all users or groups who need access to the share. Another advantage of allowing the web server to change running user identification would be that users would not be able to overwrite each other's files.

It is unfortunate that root should not own the Apache web server process due to security concerns. Until such security issues are cleared, the Apache WebDAV module will be limited to read and write files as one user.

6.3 Replication

Our implementation of WebDAV was accomplished with the Apache web server version 1.3.27, the WebDAV module version 1.3.6, and LDAP module version 2.3 on a Sun Fire 280R machine using the Solaris 8 operating system.

Installation and basic configuration information is available at http://www.rsad.edu/~lhernand/webdav and also through the WebDAV community [16]. Experienced administrators should compile and configure web servers used in production environments. A simple oversight can render a server open to security breaches and intermittent service.

6.4 Helpful Suggestions

The WebDAV module is an addition to the Apache web server; hence, it lacks functionality that would make the system stronger and more usable. Here is a short list of suggestions that would make a WebDAV enabled Apache web server easier to administer:

- Allow quotas for individual folders regardless of ownership as opposed to user-based or group-based quotas (i.e. "quota.directory".) The addition would require an extension to the file systems quota service. Some software vendors already feature directory quotas with their file services while no open source solution is currently available.

- Implement new attributes in the WebDAV module to alert administrators or users of quota breaches in web shares. (i.e. DAVDiskLimit, DAVOwner, DAVAlertOn) Alerts could be accomplished by a brute-force method using the command-line application *du* or in a more elegant manner with a folder-based quota file system. Upon authentication to the web share, the module calls a check on either the quota or the

disk usage and sends an alert to DAVOwner when the user reaches the boundary DAVAlertOn in relation to DavDiskLimit.

- Allow the WebDAV module to change the ownership of the running process so that it can act on behalf of authenticated users and not the process owner of the web server. This change would take care of both recommendations above since users would not be able to write to the web shares once they reach the user-based quota limit. We still do not want to run web server processes as root, however. The Internet Draft, *Quota and Size Properties for DAV Collections*, explains properties and minor behaviors needed for clients to interoperate with quota implementations on web folders [8]. The draft describes how the client should provide common language warning about the status of the user's quota.

The latter recommendation involves changes to the module and the Apache web server only. The second recommendation can be implemented into the module; however, it is only a temporary fix to the problem. The second solution would create more processing and disk input/output load on the server. The first recommendation would involve changes to the file systems quota service, which is not as gracious.

Interestingly enough, while two of the most popular desktop operating systems support WebDAV out of the box, providers of Network Appliance Servers (NAS, self-contained storage systems) offer their appliances with support for the following protocols: CIFS, NFS, HTTP, FTP, NDMP. Even though HTTP is supported, WebDAV is not. In conversations with one of the providers, they have expressed interest in WebDAV technology.

Administration and use of NAS with WebDAV support would improve with a WebDAV "broker" as shown in Figure 3.
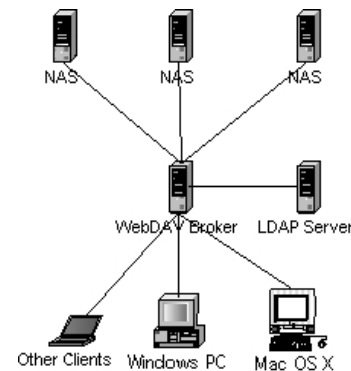

Figure 3. WebDAV Broker with WebDAV Enabled NAS.

The configuration above would require changes in both the WebDAV client and server implementations. The idea would be that a user would only need to remember one URL to access all of his/her shares. For example, a user would connect to a URL that points to the WebDAV broker, at that point the user authenticates to the broker against LDAP. Once the user authenticates successfully, the broker queries the NAS appliances to obtain a list of the shares available to the user and sends the list to the WebDAV client. Once the user obtains the list of WebDAV shares available to him/her, the client would connect directly to the NAS appliance as shown in Figure 4, which would consider the client authenticated by checking with the WebDAV broker. In

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.

fastcase
Smarter legal research.