



Mercurion Systems, Inc.

Information Technology Consulting and Support Services

Phone: 919.266.5957 Fax: 919.217.5769
<http://www.mercurionsystems.com>

USING APACHE TO REVERSE PROXY OUTLOOK WEB ACCESS

Products: Apache 2.0.49; Exchange Server 2003

Overview

Outlook Web Access (OWA) is the web-based interface for accessing e-mail and other resources handled by Microsoft Exchange Server. Built on IIS, OWA uses WebDAV to provide web-based access to e-mail messages, calendar appointments, and tasks. However, the integration between components—Windows Server 2003, IIS, and Exchange Server 2003—that offers this functionality also can make OWA more vulnerable to attack or exploit. In addition, perhaps due in part to their widespread deployment, both of these products has been the target of numerous worms and security exploits. As a result, many organizations seek to deploy OWA behind a reverse proxy that can help shield OWA from web-based attacks and exploits. Apache, the open source web server, also has the ability to function as a reverse proxy, and is well suited for this role.

This technical document describes how to Apache 2.0 as a reverse proxy for Outlook Web Access. In addition to proxying all web requests to and from the OWA server, this configuration also offloads the SSL processing (to provide SSL-encrypted access to OWA) to the Apache server, thus freeing up resources on the OWA server.

More Information

The configuration described in this document was built based on a couple of criteria. These criteria are described below.

1. The configuration had to support SSL encryption between the browser and the proxy. However, the connection between the proxy and the OWA server should be clear HTTP (this reduces SSL overhead on the OWA server).
2. The configuration has to support other web-based applications on other URLs using the same hostname. For example, the hostname “extranet.domain.com” is used in the configuration described in this document. Instead of proxying the entire URL space (/), this configuration needed to proxy only the OWA-specific URLs. This left other URLs open for other web-based applications or content.
3. The configuration had to support multiple name-based virtual hosts, so that other URLs could also be proxied through the same reverse proxy server.

Based on these criteria, Apache 2.0 was installed and configured on a system running Red Hat Linux 9.0. For additional security, iptables was configured to only allow the appropriate traffic to/from the Internet and the OWA server (specifically, HTTP and HTTPS).

A partial listing of the appropriate `httpd.conf` file for Apache is found below. Most of the configuration has been omitted for brevity; only the pertinent portions, such as the virtual host configuration and the proxy directives, are included. All IP addresses and fully qualified domain names (FQDNs) have been randomized and are not actual or valid entries.

```
NameVirtualHost 1.2.3.4:80
NameVirtualHost 1.2.3.4:443
ProxyRequests Off

<VirtualHost 1.2.3.4:443>
    ServerAdmin webmaster@domain.com
    ServerName extranet.domain.com
    DocumentRoot /var/www/extranet

    RequestHeader set Front-End-Https "On"

    ProxyRequests Off
    ProxyPreserveHost On

    SSLEngine On
    SSLCertificateFile conf/extranet-ssl-cert.pem

    <Location /exchange>
        ProxyPass http://mail.domain.com/exchange
        ProxyPassReverse http://mail.domain.com/exchange
        SSLRequireSSL
    </Location>

    <Location /exchweb>
        ProxyPass http://mail.domain.com/exchweb
        ProxyPassReverse http://mail.domain.com/exchweb
        SSLRequireSSL
    </Location>

    <Location /public>
        ProxyPass http://mail.domain.com/public
        ProxyPassReverse http://mail.domain.com/public
        SSLRequireSSL
    </Location>

</VirtualHost>
```

There are several key components to this configuration.

- **NameVirtualHost:** The `NameVirtualHost` directive enables Apache to use name-based virtual hosts on the specified IP addresses and ports. The parameter to the `NameVirtualHost` directive must match one of the `VirtualHost` definitions, as shown in the sample configuration, or else the content will be served from the default virtual host (the first virtual host listed in the configuration). Note that if the Apache reverse proxy will not be using name-based virtual hosts (using IP address-based virtual hosts or running only a single server instance), then this directive is not required.
- **RequestHeader:** This directive instructs Apache to add a header “Front-End-Https: On” to requests sent to the internal OWA server. This header is proprietary to OWA and forces OWA

to build URLs using “https://” references instead of ordinary “http://” references. This directive is critical to terminating the SSL tunnel at the reverse proxy and using clear-text HTTP between the reverse proxy and the internal OWA server. This directive requires `mod_headers`.

- *ProxyPreserveHost*: This directive configures Apache to pass the original Host header, supplied by the client, to the server to which the request is being proxied. (This is instead of the host name supplied in the ProxyPass directive.) Again, this facilitates the construction of URLs with the correct hostname when accessing resources inside OWA.
- *SSLCertificateFile*: Apache expects the web server’s SSL certificate to be in PEM format. If the certificate’s key is encrypted, Apache will prompt upon startup for the passphrase to the key (this prevents any form of automated startup). It is considered a security best practice to keep the key in a separate file (using the SSLCertificateKeyFile directive) in encrypted form and supply the password upon the startup of Apache.

Please note that the configuration listing above is only a partial listing of a complete `httpd.conf` file for Apache.

Other Notes

This configuration was only tested with Exchange Server 2003, although the configuration should also work with Exchange 2000 Server.

Related Articles/Resources

The technical document titled “Converting Certificate Formats with OpenSSL” provides additional information on converting certificates into PEM format for use by Apache.

Numerous Internet-based resources (too numerous to list here) are also available to assist in the configuration of Apache and related modules.

Legal Information

This document was created by Mercurion Systems, Inc., and may be freely distributed as long as it is distributed without modification and in its entirety. Mercurion Systems assumes no liability as a result of using the information contained in this document.