Protect Apache Webfolder From Unauthorized Access

In this article, I cover how you can easily implement a secure, web-accessible file depot using Linux, Apache, PHP, and an LDAP Authentication Backend (in this case, Microsoft Active Directory).

Overview

The configuration in question employs a simple flat text file named .htaccess to force authentication for a particular web path.

This will be accomplished through a dedicated user object that serves to *glue (or bind)* the authentication session to the LDAP instance and a security group that further restricts access by membership.

The end result:
The only user accounts allowed access to the url in question are those adhering to these constraints:

- Only user objects contained under the defined LDAP path
- Only those user objects that are members of the MySpecialGroup security group

This is the environment from which I accomplished this:
Web server details:
OS: CentOS 6.x
Apache Version: 2.2.15
PHP Version: 5.3.3

LDAP:
Active Directory on Windows Server 2008 R2

In this article, we are assuming the following:

• The URL in question is *http://filedopot.contoso.com*

• The physical path to the web folder is */var/www/vhosts/filedopot.contoso.com*

• The Active Directory Domain is *contoso.com*

• The IP Address for the Domain Controller is 192.168.1.1

• The LDAP binding user account is named *MyServiceAccount*

• The Organizational Unit containing the binding account is located under contoso.com • Service Accounts:

ou=service accounts,dc=contoso,dc=com

* With the full path to the binding account user object being:

cn=myserviceaccount,ou=service accounts,dc=contoso,dc=com

• The Organizational Unit containing the user objects is located under contoso.com • MyOU • Users:

ou=users,ou=myou,dc=contoso,dc=com

• The LDAP security group is named *myspecialgroup*

• The Organizational Unit containing the security group objects is located under contoso.com • Groups:

ou=groups,dc=contoso,dc=com

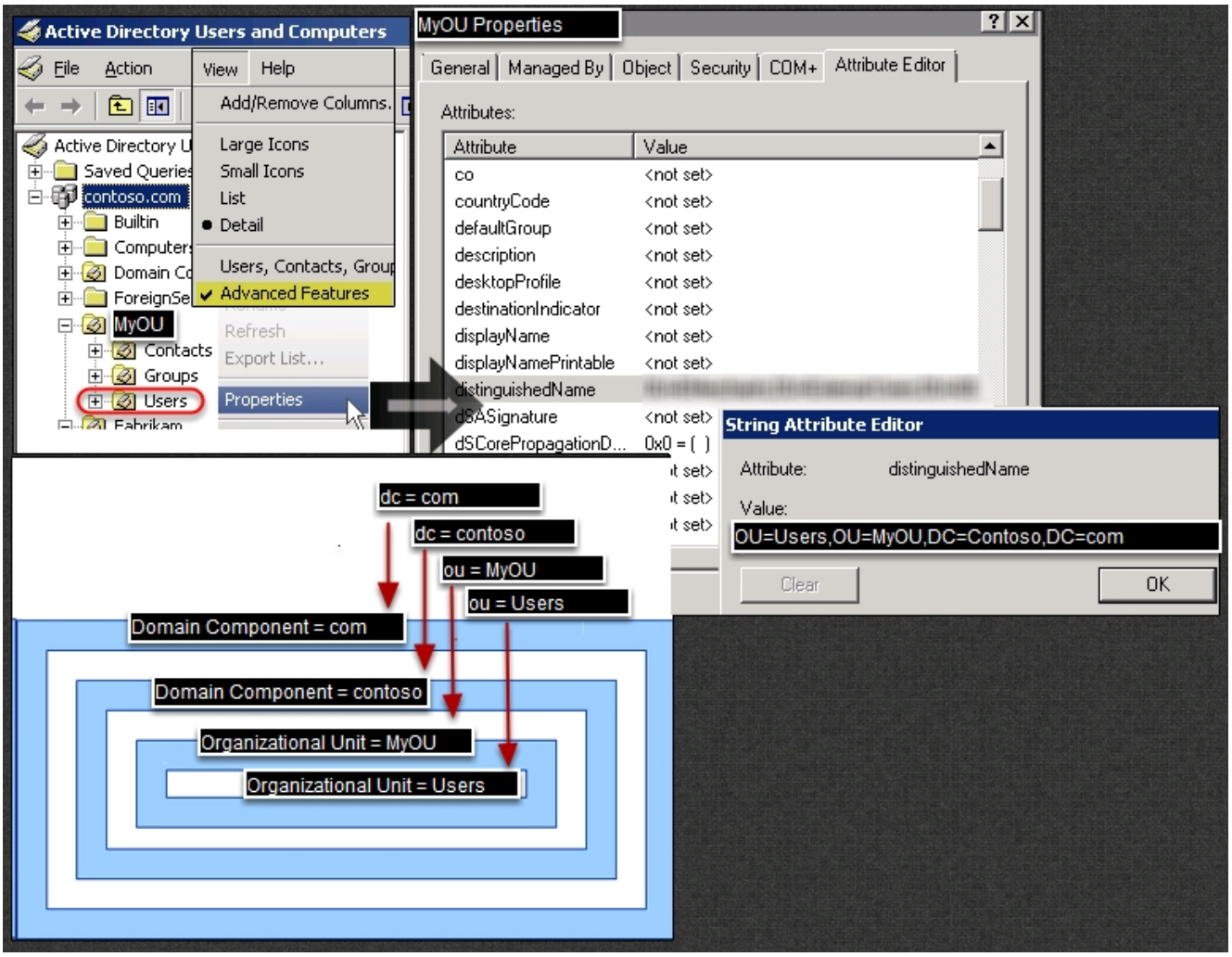
* With the full path to the security group object being:

cn=myspecialgroup,ou=groups,dc=contoso,dc=com

See:{Appendix for more information on Apache .htaccess files.

Let’s proceed with the general workflow, shall we?

Determine LDAP Path to User Objects



Apache needs to know what bucket holds the user objects that will be allowed to authenticate.

As illustrated in this example, the user objects are contained in the **Users** Organizational Unit whch is a child of the **MyOU** Organizational Unit within the **contoso.com** domain.

In like fashion, we can determine the paths to the binding user object and the security group.

Create The .htaccess File

With the LDAP information defined, we can now build a .htaccess file for our given web folder.

Login to the machine in question • navigate to your web root •

create the .htaccess file

cd */var/www/vhosts/filedopot.contoso.com*
vi ./.htaccess

According to our LDAP settings, the contents of this file should be:

```
AuthType Basic
AuthName "Network Credentials Required"
AuthBasicProvider ldap
AuthLDAPURL
"ldap://192.168.1.1:389/ou=users,ou=myou,dc=contoso,dc=com?sAM
AccountName?sub?(objectClass=*)"
AuthLDAPBindDN       "cn=myserviceaccount,ou=service
accounts,dc=contoso,dc=com"
AuthLDAPBindPassword "somepassword"
Require                                        ldap-group
cn=myspecialgroup,ou=groups,dc=contoso,dc=com
```

Test Access & Troubleshoot

1. Attempt navigation to the url in question: *http://filedopot.contoso.com*
2. Verify that you are prompted for credentials.
3. Try entering in a valid username and password combination.
4. If Problems, you can troubleshoot access by producing a live view of the site’s apache error log, e.g.:

```
tail -f /var/log/httpd/contoso.com-error.log
```

[divider]

Appendix

[divider]

Sources

URL		Description		Apache .htaccess files
		http://httpd.apache.org/docs/2.2/howto/htaccess.html		