By Nick Owen
Published: 2008-12-09 11:58

How To Configure Apache To Use Radius For Two-Factor Authentication On Ubuntu

This document describes how to add **WiKID two-factor authentication** to Apache 2.x using mod_auth_radius on Ubuntu 8.1. A previous article described how to **add two factor authentication to apache on Fedora**. Interestingly, a patch has been created to update mod_auth_radius to work with Apache 2.2+, however, it has only been updated for Debian and Ubuntu. For Fedora and other RedHat flavors of Linux, it is recommended that you use **mod_auth_xradius**.

It is also recommended that you consider using **mutual https authentication** for web applications that are worthy of two-factor authentication. Strong mutual authentication means that the targeted website is authenticated to the user in some cryptographically secure manner, thwarting most man-in-the-middle attacks. The use of cryptography is key. While some sites use an image in an attempt to validate a server, it should be noted that any man-in-the-middle could simply replay such an image.

The WiKID software token performs mutual authentication by retrieving a hash of the website's SSL certificate from the WiKID server and comparing a hash of the downloaded SSL certificate. If the two match, the token will launch the default browser to the target site for the user. If they don't match an error will be displayed, much like SSH. To configure mutual authentication for web applications, see **this tutorial**.

Our configuration was as follows:

- Ubuntu 8.1
- Apache 2.2.9-7
- libapache2-mod-auth-radius 1.5.7-8. *NB: Earlier versions will NOT work.*
- For two-factor authentication, we were using WiKID, in this case, the commercial version.

Here's how it will work, when the user clicks on a two-factor protected link, they will be prompted for a username and password. The user generates the one-time passcode on their WiKID token and enters it into the password prompt. Apache will route the username and one-time password to the WiKID server via mod_auth_radius. If the username and one-time password match what WiKID expects, the server will tell Apache to grant access. First, we add Apache to the WiKID Strong Authentication Server as a network client, then add radius to Apache. I assume you already have a WiKID domain and users setup.

So, start by adding a new Radius network client to the WiKID server for your web server:

- Log into WiKID server web interface (*http://yourwikidserver/WiKIDAdmin*).
- Select **Network Clients** tab.
- Click on **Create New Network Client.**
- Fill in the requested information.

- For the IP Address, use the web server IP address
- For Protocol, select Radius
- Hit the Add button, and on the next page, enter a shared secret
- Do not enter anything into the Return Attribute box
- From the terminal or via ssh, run 'stop' and then 'start' to load the network client into the built-in WiKID radius server

That is it for the WiKID server.

Now to get Apache ready for two-factor authentication. I started from a fresh Ubuntu 8.1 install so I had to install both apache and mod_auth_radius.

```
$ sudo apt-get install libapache-mod-auth-radius
```

Now you need to add two more things to your *apache2.conf* and *httpd.conf*. First create a directory that will be protected by two-factor authentication. In this case, the entire site is protected. Enter this into your apache2.conf:

```
<Directory /var/www>
  Options Indexes FollowSymlinks
  AuthType Basic
  AuthName "WiKID RADIUS authentication"
  AuthBasicAuthoritative Off
  AuthBasicProvider radius
  AuthRadiusAuthoritative on
  AuthRadiusActive On
  Require valid-user
</Directory>
```

Note the "AuthBasicProvider radius" directive. That stops the browser from re-submitting cached credentials to the WiKID server, which clearly will not work for one-time passwords.

Now, in *httpd.conf*, enter:

```
AddRadiusAuth wikid_server_address:1812 wikidserver_shared_secret 5
AuthRadiusCookieValid 60
```

You will want to change *wikid_server_address* to the IP address of the WiKID server and *wikidserver_shared_secret* to the shared secret you configured above in the WiKID server. Note that the the AddRadiusAuth line ends with 5 and not 5:3. The 3 in the later setting is for the number of times to attempt a password use. For one-time passwords, we only want them tried once, therefore we leave it empty. The 5 is for a 5 second time out. The AuthRadiusCookieValid directive is set for 60 minutes.

That should be all you need. You can use a .htaccess file, but that is frowned upon. The Directory method is deemed more secure.

Links

- WiKID Strong Authentication - **Two-Factor Authentication**
- Mod-auth-radius - **mod-auth-radius**
- Apache - **The Apache Webserver**

Related Tutorials

- **Add WiKID two-factor authentication to the Astaro Security Gateway**
- **Two-factor authentication for SSH using Freeradius and WiKID**
- **How to configure OpenVPN for two-factor authentication from WiKID**
- **How to configure an SSL VPN for two-factor authentication and mutual https authentication**