WPA HOWTO

From FreeRADIUS Wiki

Contents

Introduction

About this HOWTO

This document is intended as a practical document with a view to getting WPA authentication up-and-running as quickly and as easily as possible. We therefore gloss over most of the theory behind 802.1x, WPA, cryptosystems, digital signatures and certificates, etc.

Why Would I Want WPA?

In short — security. **W**i-Fi **P**rotected **A**ccess implements a sub-set (or instance, if you like) of the IEEE's 802.1x authentication standards for wireless networks, and does so in a method compliant with the (at time of writing) forthcoming 802.11i standard. WPA provides for both *authentication* (assuring the identity of a client machine, the *supplicant*) and *encryption* (ensuring exchanges between the wireless access point and client are secure).

WPA Encryption

The encryption provided by WPA is considered to be much more secure than traditional WEP. WEP uses RC4 cryptography usually with a *fixed* key of 64 to 256 bits in length, and because of this an attack on a WEP-secured network can be mounted by collecting packets for analysis and extracting the key from them. Such a crack can take as long as a matter of days on a small, household network, to a few hours on a busy corporate system. WPA, on the other hand, uses the **T**emporal **K**ey **I**ntegrity **P**rotocol system, TKIP. This has a number of fancy features (with names like "per-packet mixing") that I don't understand, but most importantly the keys are changed over time to make an attack difficult. Some "unofficial" extensions to WPA (that I've found on my hardware, and will probably be on others' as well) also allow AES (alias

"Rijndael") encryption, which I believe to the strongest of the lot (although I might be wrong here).

WPA Authentication

WPA Pre-Shared Key (WPA-PSK, or "WPA Personal") is the first kind of WPA, and is trivial to set up (so it's not covered in this document). This uses a password (which can be up to 63 characters in length) to shared between access point and client (a "shared secret") to authenticate, and act as the starting point for the cryptographic process.

WPA with 802.1x and EAP authentication ("WPA Enterprise") is the second form, and it's what we'll be setting up in this document. The **E**xtensible **A**uthentication **P**rotocol is a provision of 802.1x that allows a variety of means of authenticating clients, and in our case we will be using TLS. This involves issuing potential client machines with *digital certificates* which have been *signed* by some authority in such a way that they cannot (for all practical purposes) be forged by an attacker. The access point achieves this by requesting the client's certificate and passing it to a RADIUS server, which then checks the certificate is genuine and whether the named client is allowed access. These certificates are also used as a starting point for the cryptographic process.

My Choice

Personally, I chose WPA because I realised I had all the necessary hardware and software to hand — i.e. I did it because I could. I'm currently under the impression that WPA with RADIUS is the most secure way to tie down a wireless LAN. It's also more convenient for me as I don't have to cook up and distribute new WPA-PSK keys every so often; I can also allow friends to use my network with a centrally-managed database and time-limited certificates, and thereby avoid having to divulge network secrets.

OK, so I'm a geek and I did it because I could.

For more information on WPA, visit the Wi-Fi Alliance's WPA official home-page at wi-fi.org, in particular their "WPA Overview" (from which much of this section was researched).

Assumptions

About You and Your System

I am assuming a basic level of competence/experience with UNIX/Linux system administration (i.e., no less basic than my own despotic experience as a home sysadmin), so all the usual `cp`, `mv` business, basic TCP/IP, networking, etc. I also assume that you, like I, might not know much about FreeRADIUS or the full extent of its capabilities except that it can be used to control access to wireless networks, and wish to use it exclusively for this purpose.

Hardware

- Something running Linux. For an ideas on specifications, see §§1.4.1
- A Wireless Access Point (AP) capable WPA (or 802.1x) authentication with RADIUS
- A Wireless Network Adapter connected to the Windows XP machine (at present, I have no experience with Linux clients and so cannot document this) with WPA ("Wi-Fi") capability.

Software

I assume that you have built and/or installed:

- Windows XP SP1 with Hotfix Q815485 installed (for WPA management). Service Pack 2 has since been installed, and makes no difference to the procedures outlined herein;
- All the relevant drivers and firmware updates for WPA support on your wireless network adapter;
- OpenSSL 0.9.7a; and
- FreeRADIUS 0.9.1

or better. For help building and installation, see either the documentation that came with the package ("Oh, really?!") or see Raymond McKay's HOWTO on this topic.

What I Use

Computers

The machine on which the RADIUS server resides started out as a stock Fedora Core 1 install. The software on it that as listed above, obtained as RPMs from the FC1 CDs, Fedora Updates, etc. You won't need a monster to run FreeRADIUS: my machine dates from 1998 and uses an 233 MHz AMD K6 CPU with 64 Mb EDO RAM and a 3.2 Gb HDD. It's no speed demon, but it manages to provide RADIUS, Samba, DNS, DHCP, IP routing/firewalling and printing services to a small bevy (of order 10) clients; you may need to scale specification in accordance with load.

My client is a Hewlett-Packard Pavilion zx5000-series notebook running Windows XP SP1 with a Broadcom 54g MaxPerformance 802.11g wireless ethernet adapter. In contrast, this has a 2.8 GHz Pentium 4 CPU with 512 Mb RAM, and still doesn't even know it's born.

Wireless Access Point

I use a U.S. Robotics SureConnect 9106 ADSL Wireless Gateway as a AP. It turns out that this is actually three separate devices rolled into one. Firstly, the DSL modem; secondly, an Ethernet switch. Finally, there is a computer in its own right in the unit running BusyBox Linux 0.60.4 (2.4 series kernel), complete with tools like IPTables for firewalling, etc. This computer has two physical network interfaces, `eth0`, hard-wired onto the switch, and `wl0`, the wireless interface. A bridge device `br0` sits across these two, facilitating wireless access.

Network Topology

My RADIUS authentication server also acts as the local DNS server and DHCP server. It has two Ethernet cards in it: `eth0` is connected to the wired network, and `eth1` goes into the back of the AP, both on different Class C subnets. The machine has a number of IPTables rules configured to negotiate traffic between these two subnets.

HOWTO Do It: An Outline

OK, here's what we'll do get our WPA Authentication working:

1. **Make certificates.** Certificates are a digital means of ensuring the identity of a machine or individual and providing keys for encryption. We'll need certificates for the client(s) and RADIUS server. These certificates also need to be certified by a root certification authority (CA), and we'll make one of these as well.
2. **Configure FreeRADIUS.** FreeRADIUS checks the certificate and tells the wireless access point whether or not to accept the connection request.
3. **Configure the AP.** Many modern APs can be configured as a NAS that refers to a RADIUS server for authentication.
4. **Configure the Client.**

Step 1: Create Certificates

Here, we create and install the digital certificates used to authenticate clients on the wireless network.

In Version 2 of the server, the certificate creation scripts are located in raddb/certs/. The README file in that directory describes how to create temporary certs for testing, and how to replace those certs with real ones.

On the Windows XP Client

Copy the files ca.der and p12/*client-name*.p12 to the XP box.

First, install the root certificate to establish ourselves as an authority.

1. Double-click on ca.der.
2. In the "Certificate" property box, click Install Certificate.
3. In the Wizard, click Next.
4. Choose Place all certificates in the following store, and choose "Trusted Root Certification Authorities".
5. Click Next to finish.

Next, install the client certificate.

1. Double-click on client-name.p12.
2. In the Wizard, click Next and Next again.
3. You will be asked for a password. This is the certificate password specified either when you invoked `CA.client`, or can be found in pass/client-name.pass.
4. Choose Automatically select the certificate store based on the type of certificate.
5. Click Next to finish.

On the RADIUS Server

The certficates are stored in the raddb/certs/ directory. We suggest leaving them there.

Other guides

See also [[1]]. That web page contains a simple series of steps that guide you through certificate creation.

Step 2: Configure FreeRADIUS

Before You Proceed

You'll need to create some files and know some parameters before proceeding to configure FreeRADIUS.

Random Files

The TLS element of FreeRADIUS's EAP module (the bit that does the real authentication, EAP itself is just a wrapper) requires two files with random data: /etc/wireless-auth/DH and /etc/wireless-auth/random. Any random data will do for this, and I use the `dns-keygen` program.

In Version 2, the scripts in the raddb/certs/ directory do this for you.

Keys and Shared Secrets

You will need to know the password for the server's private key. This was established in §3.2, where you either passed the password to the scripts as arguments or it was found or generated and stored in pass/server-name.pass.

In addition, you will need a *shared secret* known only to the RADIUS server and the AP allowing the latter to identify itself to the former. This can be up to 31 characters long and anything you like, but obviously the longer and crazier the better. So I used:

```
$ /usr/sbin/dnskeygen | head -c 31
```

Configuration Files

The following scheme assumes you will be using FreeRADIUS exclusively for WPA authentication, and as such it's pretty minimal (FreeRADIUS gurus in all likelihood won't be reading this HOWTO). I arrived at it by taking the advice in McKay's HOWTO, and then deleting bits until it broke FreeRADIUS.

You will need to adjust the following files in /etc/raddb/ (or wherever your FreeRADIUS is configured to search for its config files):

- **eap.conf** - the EAP configuration, given below
- **clients.conf** - controls which APs can access this RADIUS server, given below
- **users** - a list of client users, given below

FreeRADIUS EAP Configuration

Please note that there are several settings in this file you will have to enter in accordance with your local network.

In Version 2, the default configuration works with EAP. The only file you may need to edit is raddb/eap.conf, to update the password to the private keys. If you have put the certificates in a non-standard location, the filenames need to be updated, too.

See the "tls" subsection of eap.conf. All of the filenames, directories, and certificate passwords are located there.

The Access Point Database

Here you will need the shared secret mentioned in §§4.1.2. Also, try looking in the FreeRADIUS README file to see if there is a known NAS type for your AP. If it's not listed, try a NAS type of `other`, or keep trying different ones to see which works best (I find the USR 9106 seems to be OK with either `other` or `tc`).

Listing 4.2 - /etc/raddb/clients.conf

```
# clients.conf
# Network access points that authenticate through RADIUS specified here.
#
# IMPORTANT: THIS FILE CONTAINS SECRETS.
# This file should have -rw-r----- root:radiusd permissions.

# The wireless access point
client "(the AP's IP address)" {
    secret = (RADIUS shared secret)
    shortname = (a name for logging, etc.)
    nastype = (your AP's NAS type; if unknown, try "other")
}
```

The Users’ (Supplicant) Database

Simply add a user with a "known good' password to the "users" file. The server will use that to authenticate the user.

Note that setting Auth-Type is nearly always wrong.

Listing 4.3 - /etc/raddb/users

```
# users

# a sample user and password
username  Cleartext-Password := "password"
```

Starting radiusd

Starting as a Service

If you are confident in this configuration, you can start the `radiusd` service as normal:

```
# /sbin/service radiusd start
```

Directly with Debugging Options

If not, or this fails, invoke `radiusd` directly with the debugging options enabled to see what's going on:

```
# /usr/sbin/radiusd -X -A
```

Step 3: Configure the AP

This is perhaps the easiest step, but because APs vary in their configuration, the one I can talk on the least. However, most APs I've seen claim to have a web-based interface for configuration, and I assume you've accessed yours on a number of occasions when installing the device or bored. Here I'll try and describe things in terms as broad as possible.

- Point your web-browser to the AP's configuration page.
- Choose the page that deals with "Wireless Security", "Network Authentication", or similar.
- If you are presented a list of authentication methods, select "WPA".
- Enter the RADIUS settings:
  - **RADIUS Server IP Address:** Self-explanatory: the IP address of your RADIUS server.
  - **RADIUS Port:** This is normally 1812.
  - **RADIUS Key:** Enter the shared secret used in this AP's block in the FreeRADIUS clients.conf file.
- Choose an encryption method (typically one of WEP, TKIP or AES).

  I chose AES, although unlike TKIP this is not strictly part of the WPA specification. AES is expected to form part of WPA 2. I attribute having mutually compatible hardware strictly to serendipity, and you may not have AES available to you.
- Tell the AP to accept the changes.

Note that any given AP will be wildly different from this. For instance, some place authentication methods and RADIUS configuration may be in separate pages in your AP's configurator application; there are also things like WPA rekeying intervals (I use 3600 seconds) and perhaps other options in APs I've not had a chance around with which to play.

Step 4: Configure the Client

This section assumes that you have:

- installed and configured your 802.11 wireless hardware
- configured the wireless interface's TCP/IP settings to your liking (e.g. DHCP, firewalling, etc.)
- installed Windows XP SP1 and Hotfix Q815485 (available via http://download.microsoft.com) for WPA Authentication, plus any of the necessary tools provided by your wireless interface manufacturer.

  Note Hotfix Q815485 does *not* provide WPA support and wireless encryption through TKIP and AES. As far as I understand, this provides the mechanisms for Windows XP to *configure and manage* such features (as opposed to manufacturer-specific utilities). You'll still need WPA support from your wireless hardware drivers.

As an example of this, I normally use a built-in Broadcom 54g MaxPerformance 802.11g with my notebook. The drivers for this provide WPA support with WEP, TKIP and AES encryption, and this can be configured either with the standard Windows XP property boxes, or through Broadcom's own utility.

Establishing the Connection

The configuration of a WPA-authenticated connection can normally be carried out in in two ways. Firstly, many wireless adapter manufacturers provide utilities to manage wireless connections on their hardware. As this method depends on exactly what card one is using, it is not covered here; furthermore, I guess that those who plan on taking this route will probably not need to read this section!

The second route is to let Windows XP manage the authentication. This I can describe.

1. Plug in and/or activate your wireless hardware. A "two monitors" icon may appear in the Notification Area for the interface.
2. Right-click on the wireless interface's "two monitors" icon in the Notification Area, and choose *View Available Wireless Networks*. At this point, you will be presented with a list of available networks. If you configured your AP with "Disable SSID Broadcast" (or similar), you might not see any networks at all. In either case,
3. Click *Advanced...* in the box that opens. The wireless interface's properties box will open.
4. Check *Use Windows to configure my wireless network settings*.
5. In the "Preferred networks" group, choose the network with WPA authentication and click *Properties*. If its not listed, click *Add...*. At this point, the "Wireless network properties" box appears.
6. If it is absent, enter the network's SSID (*Association* tab).
7. Under the *Association* tab, in the "Wireless network key" group, set the following:
   - **Network Authentication**: *WPA*
   - **Data Encryption**: choose one of either *AES* or *TKIP* to reflect the settings on your AP.
8. Under the *Authentication* tab, set:
   - Check *Authenticate as computer*...
   - **Un**check *Authenticate as guest*...
   - **EAP Type**: *Smart Card or Other Certificate*.
9. Click *Properties* (under the *Authentication* tab).
10. In the "Smart Card or other Certificate Properties" box, set the following:
    - Choose *Use a certificate on this computer*
    - Check *Use simple certificate selection*
    - Check *Validate server certificate*
    - **Un**check *Connect to these servers:*
    - In the list of trusted root CAs, check *only the CA that corresponds to the certificate you have generated*
    - **Un**check *Use a different user name for the connection*

11. Click *OK* in all three boxes to set the connection properties.

If all is well-configured, everything should be working in minutes. (The process could take a minute or longer from cold; I find that activating the connection on my notebook *before* logging in seems to work the quickest.) To check progress, open up the Network Connections pseudo-folder from Control Panel. The status of the wireless connection should go from "Wireless connection unavailable" to "Attempting authentication" and then "Authentication succeeded" (along with an informative speech bubble from the Notification Area). If you use DHCP, check that the interface has acquired an IP address.

That's it!

Troubleshooting

radiusd Won't Start

Start `radiusd` directly with the debugging options, as per §§4.3.2. When setting up FreeRADIUS, I found I made the following common errors:

- **Unmatched `{` or `}` in the configuration files.** - `radiusd` normally reports a message to the effect of "file ended early" in this case.
- **Can't access configuration/key files.** - If you're running `radiusd` as non suid-root, make sure that the files' permissions are correct. (`radiusd` will tell you "permission denied" and the filename.)
- **Files not found.** Check that all the null files (see §4.2) exist.

Can't Authenticate

You've done all the steps above, and after about five minutes of waiting, Windows XP has popped up a little bubble (or put a flag in the "Network Connections" pseudo-folder) saying "Cannot log onto the network" (or "Authentication failed"). Maybe, if you're using DHCP, it's even gone ahead with that crazy "Zero Configuration" business.

First, try setting up a "dummy account" to test authentication. Add the following group to /etc/raddb/clients.conf:

Listing 7.1 - Added to /etc/raddb/users

```
client "127.0.0.1" {
        secret = test-secret
        shortname = localhost
}
```

Add the following line to /etc/raddb/users *before* the `DEFAULT` entry:

Listing 7.2 - Added to /etc/raddb/users

```
"test" Auth-Type := Local, User-Password == "test"
```

Start `radiusd` directly (see §§4.3.2), and test this configuration using:

```
$ radtest test test localhost 0 test-secret
```

Amongst the reams of RADIUSspeak produced, you should see a message informing you that an `Accept-Accept` message has been returned. If not, go through the configuration files again. Some things that could go wrong include:

- **Unmatched passwords/keys** - This could be in the EAP/TLS configuration in /etc/raddb/radiusd.conf (when accessing the server's private key), or a mismatch between your /etc/raddb/clients.conf and AP's settings.
- **Typos in IP addresses**
- **User names spelt incorrectly**

You should, at this point, feel quite patronised. But if this solves the problem, the test lines of Listings 7.1 and 7.2 above should be removed.

Still not working? Take a look at the client. Windows XP clearly wasn't designed with the dollar-prompt-and-dot-conf type in mind — things break, and you're lucky if you get an error message telling you even vaguely what has gone wrong. Try:

- **Rebooting** and try activating your wireless hardware *before* you log in.
  I don't know why this expedites the process on my machine; perhaps it prevents the XP supplicant from attempting to authenticate as the user, and authenticate as the machine instead.
- **Checking the logs.** These can be found in the *Event Viewer*:
  1. Click *Start*, *Run...*.
  2. Enter **`eventvwr.msc /s`**.
  3. Click *OK*.
     From here on in it's down to your own sysadmin's intuition. The "System" and "Application" logs will be of particular interest. In my experience, the messages given are very earnest, descriptive and long-winded, but more often than not seem to omit everything useful.
- **Reinstalling** all the software for your wireless adapter.

Miscellaneaous

Questions, Answers and Cry-For-Helps

How can I add more APs?

Easily. Simply configure the additional APs as described herein, and add corresponding blocks for their IP addresses in the FreeRADIUS *clients.conf* file. In the interest of security, I advise you use a different shared secret for each.

Can I use something other than WPA?

Yes. In the properties box for a given wireless network in Windows XP, you can choose from Open, Shared, WPA and WPA Pre-Shared Key. Provided you choose some from of encryption, you will be able to use 802.1x authentication in conjunction with it. As far as I have played with this, no adjustments need to be made with RADIUS, but you might need to adjust your AP's encryption settings and tell it to use 802.1x.

I can't find any mention of WPA in the Windows XP property boxes!

Check the following:

- Are you up to date with Windows XP Service Packs, Hotfixes and Patches?
- Have you tried the utilities that came with your wireless adapter?
- Are you using the latest drivers and configuration utilities? Perhaps your adapter doesn't support WPA *yet*.

I've found that XP doesn't present the WPA options for a network unless it has confirmed that the network requires WPA authentication. So you should also check that

- the wireless interface is activated (i.e., radio on); and
- you've spelt the SSID correctly (if broadcast is disabled).

Windows XP will also not present WPA options if your wireless network interface does not support WPA. This is the case with a number of Acer laptops.

If this is of no help, you could try using something other than WPA in the meantime (see above).

See Also

- WPA
- EAP
- HOWTO
- Wi-Fi Alliance's web-site

Retrieved from "http://wiki.freeradius.org/WPA_HOWTO"

This page has been accessed 122,444 times. This page was last modified on 9 July 2010, at 21:33.