Radmin® Radius User Administration

Installation and reference manual for RAdmin version 1.19

RAdmin

Table of Contents

1. Introduction

This document describes RAdmin, and shows how to install and configure RAdmin.

RAdmin is a Radius user administration package. It allows you to manage the users in a Radius users database. You can therefore use RAdmin to control the users that are allowed to connect to your network through Radius-compliant terminal servers and routers. You can add, change and remove users, and you can view the connection summaries and history of users, and investigate problems with your modem banks and your Radiator radius server. You can set up time limits, static IP addresses, simultaneous-use limits and automatically lock out users after a number of bad logins, and you can set arbitrary Radius Check and Reply items for users and groups of users.

RAdmin works in conjunction with the Radiator® AAA server from Radiator Software (see https://radiatorsoftware.com/products/radiator/) and an SQL database. Its web-based interface allows you to manage your user database using a web browser.

Radiator and RAdmin run on Unix-like systems, Windows or any combination of those operating systems.

RAdmin is not a billing or invoicing system. If you need to be able to bill or invoice your users, you should consider one of the third-party ISP billing packages that Radiator works with.

In order to install and configure RAdmin, you will need to be (or have access to) a competent system administrator. You will also need to have an understanding of Radius and your authentication requirements.

Installing RAdmin requires that a number of other components be installed first. Do not skip these stages, they are all required.

‘RAdmin’ and the RAdmin logo are registered trademarks of Open System Consultants Pty Ltd, a subsidiary of Radiator Software Oy

2. Overview

This section the describes the prerequisites and overall deployment of RAdmin

2.1. Prerequisites

RAdmin consists of a set of perl CGI scripts that operate on tables in an SQL database. Your Radiator Radius server will be configured to use the information in those SQL tables to allow your authorized users to log into your network. Therefore, RAdmin has the following prerequisites:

1. Perl. Perl is a freely available program for all Unix-like, Windows and other systems. Perl is typically pre-installed on Unix type systems. Full source code and binaries for are available at https://www.perl.org. For Windows, we recommend Strawberry Perl or ActivePerl from ActiveState.

2. Perl modules for accessing SQL. The DBI and DBD modules for most free and commercial databases are included with your Perl installation or are available from CPAN. You will need to install the DBI module, and the DBD module that corresponds to the SQL database you plan to use.

3. SQL Server. RAdmin uses an SQL database to store details about your users. Your Radiator Radius server will be configured to get user details from that SQL database. RAdmin and Radiator work with a very wide range of free and commercial SQL servers on a wide variety of platforms, including:

    - MySQL and MariaDB
    - PostgreSQL
    - Oracle
    - Microsoft SQL Server

  - SQLite
  - many, many others

   No matter which SQL database you choose, you will probably have to nominate one of your staff to become expert in maintaining it, especially for backups of your precious user database. You will need to have a significant amount of disk space available on your SQL server to hold the RAdmin database. For most systems, you will need at least 100Mb of space for your RAdmin tables.

4. Web server. On Unix, we recommend Apache (www.apache.org), but RAdmin will work with many other CGI compliant web servers. On Windows, it will work with Microsoft IIS.
5. A Web browser. Any web browser on any platform will work fine.

To deploy on Unix or Linux, all of the above tools are freely available, often with easy installation from you OS vendors repositories. Consult you OS documentation for instructions on how to install the support packages you need.

2.2. Deployment

A complete Radiator/RAdmin system consists of 3 largely independent components:

- Radiator, running on a “Radiator Host”
- SQL Server, running on a “SQL Host”
- RAdmin and a web server, running on a “RAdmin host”

These 3 components may all be installed on a single computer, or installed on 2 or 3 separate computers. You will need to decide how to deploy these 3 components on the computers in your network. Some of the factors that could influence your decision may be:

- Available hardware and performance
- Available SQL server licenses and costs
- The architecture of your network, especially with respect to firewalls
- The size of your user population

For a small user population (say 1000 or less), it is common to run all 3 components on the one computer (i.e. Radiator Host, SQL Host and RAdmin Host are all the same computer). For very large user populations it is common to run a separate SQL Host, a RAdmin Host, and multiple Radiator Hosts.

3. Installation

> **Note**
>
> If you are upgrading a previously installed RAdmin to a more recent version, follow the instructions in doc/migration.html in your distribution, or at https://www.open.com.au/radmin/migration.html [https://www.open.com.au/radmin/migration.html].

No matter how many computers you plan to deploy your system on, installation and configuration consists of the following steps.

1. Make sure you are (or have access to) a system administrator and someone who understands your Radius authentication and accounting requirements. Assistance with installing or configuring RAdmin is not included in the price of RAdmin.

2. Purchase and download RAdmin
3. Purchase and download Radiator
4. Install your selected SQL server on your SQL host
5. Install a web server on your RAdmin host
6. Install Perl on your Radiator host and RAdmin host
7. Install DBI and the DBD module for your selected SQL server on your Radiator host and RAdmin host.
8. Unpack your RAdmin distribution (see below)
9. Create the RAdmin database and tables (see below)
10. Install the RAdmin CGI scripts on your RAdmin host (see below)
11. Install and configure Radiator on your Radiator host(s) (see below)
12. Test the entire Radiator/SQL/RAdmin system (see below)
13. Arrange for backups of your SQL database to be taken periodically

The detailed steps are somewhat different for Unix and Windows systems, so we have given detailed instructions separately for each below. In a mixed environment, you may have to follow instructions for part of the install for Unix and part for Windows.

3.1. Installation on Unix

This section the describes the installation of RAdmin to Unix.

3.1.1. Unpack your RAdmin distribution

The Radmin distribution is supplied as a tarred and gzipped file called something like Radmin-1.19.tgz. You will need to move it to a work area and unpack it.

```
cat Radmin-1.19.tgz|gunzip -c|tar xvf -
```

This will create a directory Radmin-1.19 in the current directory.

```
cd Radmin-1.19
```

3.1.2. Create the RAdmin database and tables

Instructions for specific databases follow

3.1.2.1. MySQL

1. Edit installMysql.sh, check and possibly change the configurable variables near the top to suit your site.
2.
   ```
   sh installMysql.sh
   ```

3.1.2.2. Oracle

1. Edit installOracle.sh, check and possibly change the configurable variables near the top to suit your site.
2.
   ```
   sh installOracle.sh
   ```

3.1.2.3. Sybase

1. Edit installSybase.sh, check and possibly change the configurable variables near the top to suit your site.

2.
```
sh installSybase.sh
```

3.1.2.4. Postgresql

1. Edit installPostgresql.sh, check and possibly change the configurable variables near the top to suit your site.
2.
```
sh installPostgresql.sh
```

3.1.2.5. DBD::SQLite

1. Edit installSQLite.sh, check and possibly change the configurable variables near the top to suit your site.
2.
```
sh installSQLite.sh
```

3.1.2.6. Other Databases

We have not provided installation scripts for other databases. You will need to follow these steps:

1. Following your SQL vendor's instructions, create an empty database
2. Create a database user that has permission to create tables in the new database
3. Set the new database users password
4. Edit Radmin/Sql.pm, and configure it so that RAdmin can connect to the SQL database, using the connection name rules for your selected DBD module, plus the user name and password you created above.
5. Create the tables by running

   ```
   perl createdb.pl -create
   ```

   This will attempt to log in to the database using the details you entered into Radmin/Sql.pm, then create the tables described in Radmin/Schema.pm
6. If the previous step fails, it may be due to unusual table creation syntax for your SQL server. If so, you may need to hand edit a file to create the tables for your database. You can see what createdb.pl is trying to do by running it like this:

   ```
   perl createdb.pl -create -n -d
   ```
7. Install the basic data into the database with:

   ```
   perl createdb.pl basicdb.dat radattrs.dat
   ```
8. Try to automate the entire process and send us an installXXX.sh so others can benefit from your experience.

3.1.3. Install the RAdmin CGI scripts on your RAdmin host

1. Become root
2. Start the installation with

   ```
   perl install.pl
   ```
3. When prompted for “Web server html directory”, enter the full path to the root directory of your web server (i.e. the main directory for the HTML documents. The installation will create a directory “Radmin” there, containing documentation and other static files.

4. When prompted for “Web server CGI directory”, enter the full path your web servers cgi-bin directory. The installation will create a directory “Radmin” there, containing the RAdmin CGI scripts.
5. When prompted for “User name that your web server runs as”, enter the Unix user name that your web server is configured to run as. All the files that RAdmin installs in your web server will be owned by this user.
6. Radmin will now install the CGI scripts and documentation in the web server and the Radmin perl modules in your Perl site library directory (typically `/usr/local/lib/perl5/site_perl`>). It will also record the answers you gave above in `Radmin/Paths.pm`, and will reuse the answers if you run install.pl again.

3.1.4. Install and configure Radiator on your Radiator host(s)

See the Radiator installation instructions. There is an example Radiator configuration file in `Radiator/goodies/radmin.cfg`. You will need to edit it and set DBSource, DBUsername and DBAuth to suit your SQL server.

3.1.5. Test the entire Radiator/SQL/RAdmin system

1. Add a test user, using

   ```
   http://localhost/cgi-bin/Radmin/private/editUser.pl
   ```

2. start radiusd (the Radiator server process)
3. Using radpwtst (the Radiator test program), try to authenticate your test user with something like:

   ```
   radpwtst -user mytestuser -password mytestuserpassword
   ```

3.2. Installation on Windows

You should already have installed your selected SQL server and your web server.

Note

These instructions were tested with Windows 2012R2 as the RAdmin web server and SQL Server Express 2014 running on another host. SQL Server Management Studio v17.6 was used for managing SQL Server.

3.2.1. Install Perl and required support modules

1. Download and install Strawberry Perl or ActiveState Perl. During installation, accept all the defaults. Allow setup to reboot your computer.
2. Connect your computer to the Internet so you can download the required Perl modules from ActiveState using PPM in the next step.
3. Run ppm (the Perl package manager) from the command prompt. You will get a command line screen running ppm with a PPM> prompt or a graphical user interface.
4. Make sure you have DBI and DBD-ODBC packages installed.
5. On command line you can type “install DBD-ODBC” or “install DBI”. This will download and install DBD-ODBC and DBI packages, if required. These will allow RAdmin to connect to any ODBC compliant SQL server
6. Close the PPM window. Perl is now installed.

3.2.2. Unpack your RAdmin distribution

The Radmin distribution is supplied as a tarred and gzipped file called something like Radmin-1.19.tgz. You will need to move it to a work area and unpack it. We recommend you unpack it into a directory called something like c:\Radmin-1.19

The distribution can be unpacked with 7-Zip, WinZip, or other file archiver.

3.2.3. Create the RAdmin database and tables

Here we will describe how to create a new RAdmin database on Microsoft SQL Server. Depending on your Windows and SQL Server version, you may adjust the steps. If you use other database, follow the database vendor's instructions.

First, create an empty database for the RAdmin tables. To do this, use SQL Server Management Studio. If it's not already installed, download it from Microsoft.

3.2.3.1. Creating an empty database with SQL Server Management Studio

1. SQL Server Management Studio.
2. In the left hand window, find your server.
3. Right click on the Databases folder. Choose "New Database". Specify a database name (we recommend "radmin").
4. Click OK. The database device will be created and sized automatically.

3.2.3.2. Enable SQL Server 'Windows Authentication mode'

By default, SQL Server has 'Windows Authentication mode' enabled. You need to change this to 'SQL Server and Windows Authentication mode'.

1. Right click on Server (top of tree object) and choose Properties -> Security
2. Enable 'SQL Server and Windows Authentication mode'.

3.2.3.3. Create the RAdmin database user

You must now create a database user that the RAdmin scripts will use to log in to the database you created above.

These examples create a new database user called "radmin", with a password "ra+9+9dminpw". You may wish to choose a different password, but we recommend that you keep the user name radmin. The default database for the "radmin" user is set to the "radmin" database, which is also owned by radmin. Record these names and password, you will need them later.

1. Right click on the Security folder. Choose New -> Login
2. Specify a "Login Name" (we recommend "radmin").
3. Choose "SQL Server authentication" and set password to "ra+9+9dminpw".
4. Click on "Default database" and select "radmin".
5. Click OK.

You can also use command line utility "sqlcmd" to create the user and set the default database:

1. Run the "sqlcmd" command line application. For example, to connect to local server that listens on port 5161:

```
sqlcmd -S 127.0.0.1,5161 -d radmin
```

2. Run these commands to create a user and set the default database:

   ```
   EXEC sp_addlogin 'radmin', 'ra+9+9dminpw', 'radmin';
   go
   EXEC sp_changedbowner radmin,true
   go
   ```

3. Run “exit” to leave sqlcmd.

3.2.3.4. Create an ODBC System Data Source Name (DSN)

Run “ODBC Data Source Administrator”. For more information about locating it, see: http://msdn.microsoft.com/en-us/library/ms188691.aspx [http://msdn.microsoft.com/en-us/library/ms188691.aspx]

1. Select “System DSN” tab and click Add.
2. Choose “SQL Server” if there are multiple choices, choose something like SQL Server (for example, Version 6.03.9600.17415, SQLSRV32.DLL, date 10/29/2014)
3. - Name: Radmin
   - Description: RAdmin radius database
   - Server: (local) from drop down menu did not work. Interface name worked (you will see this after authentication dialog)

     If drop down menu gives nothing, check you have "SQL Server Browser" started and running from SQL Server Configuration
   - Manager. Click Next
4. - Authentication dialog: Choose “With SQL Server authentication using a login ID and password entered by the user”.

     Use Login ID: radmin and Password: ra+9+9dminpw
5. Click through with Next while checking the options. “Test Data Source” should succeed. If not, check server, password etc.

3.2.3.5. Create the RAdmin tables and complete the installation

By now, you should have installed Perl, plus the DBI and DBD-ODBC perl modules. If so, you can now use the createdb.pl program supplied with RAdmin to create the database tables that RAdmin needs.

1. Start a Command Prompt window.
2. Change directories to the RAdmin distribution
3. Edit the file `Radmin\Sql.pm` in your RAdmin distribution directory. This is the file that tells all the RAdmin programs how to connect to the SQL database. Ensure that it is set up like this:

   ```
   $Radmin::config{DBSource} = 'dbi:ODBC:Radmin';
   $Radmin::config{DBUsername} = 'radmin';
   $Radmin::config{DBAuth} = ra+9+9dminpw';
   ```

   Where “Radmin” is the name of the ODBC System DSN you created above, “radmin” is the SQL database login name you created above, and “ra+9+9dminpw” is the password for that user.
4. Run the command:

```
perl createdb.pl -create -- basicdb.dat radattrs.dat
```

This will create all the tables and data that RAdmin needs

3.2.4. Check CGI is enabled on your RAdmin hos

You need to verify that CGI is enabled and there is a mapping that tells IIS to use Perl to run RAdmin's CGI scripts.

Verify that IIS is installed. If not, choose "CGI" under Role Services during the installation. If IIS is already installed, verify that "CGI" enabled:

1. ServerManager -> Tools -> Internet Information Server (IIS) Manager
2. Click the Server and see that CGI available. If not, go to ServerManager and click Manage -> Add Roles and Features -> Server Roles -> Web Server (IIS) and locate and select CGI.

For CGI to work, you must set up CGI directory and create a mapping so that the CGI scripts are processed by Perl.

1. Right click on your chosen website and choose "Add Virtual Directory". Create a mapping for the CGI scripts directory. For example, map "/Scripts" to a newly created "C:\Inetpub\Scripts".
2. Click on the new virtual directory "Scripts" and double click on "Handler Mappings". Choose "Add Script Map". Enter "*.pl" for Request path and enter the full path to the installed perl.exe. For example, 'C:\path\to\perl.exe "%s" %s' where %s are literal parameters, including double quotes for the first parameter.
3.

   Tip

   For more information about ensuring that RAdmin perl scripts run with IIS, see http://community.activestate.com/forum-topic/configuring-perl-iis-7-0 [http://community.activestate.com/forum-topic/configuring-perl-iis-7-0]

3.2.5. Install the RAdmin CGI scripts on your RAdmin host

1. Start a Command Prompt window
2. Change to the directory where your RAdmin distribution was unpacked, typically with something like: `cd c:\Radmin-1.19`
3. Start the installation with

   ```
   perl install.pl
   ```

4. When prompted for "Web server html directory", enter the full path to the root directory of your web server (i.e. the main directory for the HTML documents. The installation will create a directory "Radmin" there, containing documentation and other static files.
5. When prompted for "Web server CGI directory", enter the full path your web servers CGI scripts directory. The installation will create a directory "Radmin" there, containing the RAdmin CGI scripts.
6. Radmin will now install the CGI scripts and documentation in the web server and the Radmin perl modules in your Perl site library directory (typically `c:\perl\lib\site`). It will also record the answers you gave above in Radmin\Paths.pm, and will reuse the answers if you run install.pl again.

3.2.6. Install and configure Radiator on your Radiator host(s)

See the Radiator installation instructions. There is an example Radiator configuration file in `Radiator/goodies/radmin.cfg`. You will need to edit it and set DBSource, DBUsername and DBAuth to suit your SQL server.

3.2.7. Test the entire Radiator/SQL/RAdmin system

1. Add a test user, using

   ```
   http://localhost/cgi-bin/Radmin/private/editUser.pl
   ```

2. start radiusd (the Radiator server process) in an MS-DOS command window
3. In another MS-DOS command window, use radpwtst (the Radiator test program), to try to authenticate your test user with something like:

   ```
   radpwtst -user mytestuser -password mytestuserpassword
   ```

4. Configuring and customizing RAdmin

This section the describes configuring and customizing RAdmin

4.1. RAdmin user Permissions

The RAdmin system administrator is able to configure who is permitted to use RAdmin administrative web pages, and what functions they are allowed to perform. There are 2 levels to this control:

- RAdmin user authentication
- RAdmin user permissions

4.1.1. RAdmin user authentication

In web-based administration packages such as RAdmin, it is common to require administrators to log in before they can use the system. RAdmin has a flexible system for authenticating access to RAdmin web pages. The RAdmin system administrator is able to configure one of 3 types of authentication:

- No authentication.
- Web-Server authentication.
- RAdmin authentication.

> **Tip**
>
> There should never be any need to configure both Web-Server authentication and RAdmin authentication.

4.1.2. No Authentication

This is the default configuration for RAdmin.

In this option, Administrative users are not required to log in before using the RAdmin web pages. You would only choose this if there were only one or 2 administrators, and then only access to the RAdmin web pages was from within the local network. This option is not recommended for production systems.

In this method every administrative user is logged in as ‘anonymous’, and will receive the permissions profile configured for ‘anonymous’ on the ‘Edit Administrative User’ page.

4.1.3. Web-Server authentication

With this option, RAdmin user authentication is done by your web server, using whatever methods and systems have been configured into the web server. You might choose this option if you have a pre-existing system for controlling and authenticating access to web server pages for your staff. With this option, the Password field on the ‘Edit Administrator User’’ page is not used. To enable this option, you must configure your web server appropriately but you do not need to enable any authentication options with RAdmin.

In this method every administrative user will receive the permissions profile configured into RAdmin for the username with which they authenticated to the web server (or ‘anonymous’ if no such Administrative user has been configured into RAdmin). See the ‘Edit Administrative User’ page for each web-server authenticated user.

Like any other web application, you can configure your web server so that only specific users can access particular pages. You can choose whether or not to do this, based on a number of factors:

- Do you intend to let your users access the public/changePassword.pl and public/showUsage.pl pages and change their own password and view their own usage?
- Do you intend to impose user-specific permissions to RAdmin users, i.e. to permits different staff members to do different things?

In order to distinguish between your users (usually the public, with few privileges), and your authorized RAdmin administrative users (usually your internal staff, responsible for administering the end users), or to distinguish between individual RAdmin users (perhaps with different access levels or permissions) you will need to enable web server access control on your web server.

The way to do this depends on what type of web server you are running, and is usually different for each type. If you are running the Apache, you can add access configuration to its configuration file for each location and directory you wish to protect. In the following example, only users specified in the password file *and* coming from one of the two IP subnets are permitted to access the contents of the location.

```
# Example Apache 2.4 configuration
<Location "/cgi-bin">
        Options None
        AuthType Basic
        AuthName "RAdmin system"
        AuthBasicProvider file
        AuthUserFile /path/to/htpasswd-radmin
        <RequireAll>
                Require valid-user
                <RequireAny>
                        Require ip 10.20.30.0/24
                        Require ip 172.16.130.0/24
               </RequireAny>
        </RequireAll>
</Location>
```

Consult your web server vendor documentation for details on other web servers.

It is common practice to enable access control for the RAdmin private scripts (usually in cgi-bin/Radmin/private), and to have no access control for the publicly runnable scripts (usually in cgi-bin/Radmin/public).

Tip

It is possible to configure Apache (and some other web server) to authenticate web users by access a Radius server (such as Radius) to authenticate users. This can be very convenient, because you would not have to maintain a separate AuthUserFile for your Apache web access.

4.1.4. RAdmin authentication

With this option, Administrative User authentication is done by Radmin. To enable this option, set the ‘Authenticate Admin users?’ option on the ‘Edit RAdmin Configuration’. When this option is set, RAdmin users will be required to log in to RAdmin before any RAdmin functions can be used. The first time an administrative page is access, a RAdmin Login page will be presented and the user will be required to enter a valid administrative user name and password (configured using the ‘Add Administrative User’ page).

Having logged in, the administrative user will not need to re-authenticate until the ‘Maximum Admin User session time (mins)’ expires.

This options requires cookies to be enabled in Administrative user's web browsers.

4.1.5. RAdmin user permissions

Regardless of how RAdmin users are authenticated, RAdmin applies restrictions to what RAdmin users are able to do using the RAdmin web pages.

You can impose different levels of access to the RAdmin system to different users by using RAdmin permissions. These allow you to specify what RAdmin functions your RAdmin users are permitted to access. (Do not confuse this with what your users are allowed to do: RAdmin users are usually your staff members charged with adding, changing and administering your users).

You can create any number of Permission Profiles, using the Add Permissions Profile page. Each Permission Profile lists what actions RAdmin users with that profile are permitted to do. Then you can add and configure RAdmin users using the Add Admin User page. There you can select which Permissions Profile the user gets, and therefore what things that RAdmin user is permitted to do.

When a RAdmin page is protected by web access control, RAdmin will attempt to match their web user name with a RAdmin Administrator User Name, in order to find out what permissions they have. The exceptions to this are:

- If there is no web access control on that page, the name “anonymous” will be used. This is usually the general public who have few, if any permissions.
- If there is no exact matching RAdmin user name, the profile for the Radmin user “DEFAULT” will be used. DEFAULT is therefor a catchall for users that do not have their own RAdmin Admin user entry.

RAdmin is delivered with a number of standard Permission profiles and 2 RAdmin users” “anonymous” and “DEFAULT”, each with the “Everything” permissions profile. This means that any user person with access to your web server has permission to do anything in RAdmin. You must change this before commissioning your system. It is common practice to give “View own usage” and “Change own password” to “anonymous”, and more extensive permissions to your staff. At least one of your staff should have the “All” profile, so that they can administer the permissions profiles of other users. The you can add, change and delete as many Permissions profiles as you need to suit your organization's requirements.

4.1.6. Subscriptions

RAdmin can optionally manage a set of web page subscriptions. With this system, you can establish a set of subscription products, each with its own Apache htpasswd style password file. You can then enable or disable access to each product on a per-user basis. RAdmin will then automatically add or remove the user from the web server access file, and optionally email the user with access details. Automatic expiry is supported by the goodies/expire program.

To enable Subscriptions, you need to set $Radmin::config{Subscriptions} to 1 in your Site.pm file. There are a number of other Site.pm configuration options that relate only to Subscriptions, such as SubscriptionEmailBody, SubscriptionEmailSubject that you may wish to configure to suit your site.

4.1.7. Digipass Support

RAdmin can optionally support OneSpan, formerly known as Vasco, Digipass tokens (https://www.onespan.com [https://www.onespan.com]). Digipass tokens are small handheld devices that generate one-time-passwords. They can be purchased from OneSpan and issued to your users. Such tokens provide much higher levels of security than static passwords. Digipass is supported by RAdmin on Solaris, Linux and Windows. Support for Digipass requires the installation of the Authen::Digipass module from Radiator Software. The Authen::Digipass module is not available in the standard Radiator distribution. Contact Radiator Software to obtain Authen::Digipass module.

You can add new tokens into the RAdmin database with the RAdmin ‘Import Digipass Tokens’ page. After a token is imported, it must be allocated to a user before that user can use the token to authenticate.

In order to enable Digipass token support:

1. Install Radiator in the usual way
2. Follow the instructions in Radiator/goodies/digipass-install.txt to install the Authen-Digipass support module. This is required on both the Radiator host and on the RAdmin web server host (if they are different).
3. Install RAdmin as described above in this document.
4. On the ‘Edit Radmin Configuration’ page enable the ‘Support Vasco Digipass?’ option. Click ‘**Update**’.
5. RAdmin web pages will now include ‘List Digipass Tokens’ and ‘Import Digipass Tokens’, and the Edit User page will include some new options for allocating and listing Digipass tokens.
6. Configure Radiator based on the example configuration file `Radiator/goodies/radminDigipass.cfg` which shows how to authenticate using Digipass token data held in the RAdmin database.

Exactly which Digipass actions are available to a particular RAdmin user depends on the Permissions profile assigned to them. The Permissions profile individually controls whether a user can List, Allocate, Deallocate, Unlock, Reset, Reset the static password (PIN), Set the static password (PIN) and Import tokens.

4.1.8. TOTP and HOTP (OATH) token Support

RAdmin can optionally TOTP and HOTP tokens developed by OATH [https://openauthentication.org/ [https://openauthentication.org/]]. TOTP and HOTP are defined by RFCs 6238 and 4226, respectively. Such tokens provide much higher levels of security than static passwords. OATH tokens are supported on all RAdmin platforms. These tokens are typically provisioned to a mobile device app, such as Google Authenticator or Microsoft Authenticator.

Each token has a secret seed value and associated information, such as counter for detecting replay attempts. In order to authenticate a TOTP or HOTP token, the RAdmin database must contain a table named RADOATH with a token record that stores the seed and its associated information. You can add new tokens into the RAdmin database when creating or listing users. When a token is created, it must be allocated to a user before that user can use the token to authenticate. You can view all currently existing tokens with the RAdmin ‘List OATH tokens’ page.

In order to enable TOTP and HOTP token support:

1. Install Radiator in the usual way
2. Install the MIME-Base32 and Imager-QRCode modules from CPAN (www.cpan.org [www.cpan.org]) on the RAdmin host.
3. Install RAdmin as described above in this document.
4. On the ‘Edit Radmin Configuration’ page enable the ‘Support OATH (TOTP and HOTP)’ option. Click ‘**Update**’.
5. RAdmin web pages will now include ‘List OATH tokens’ and the Edit User page will include some new options for allocating and listing allocated OATH tokens.
6. Configure Radiator based on the example configuration file `Radiator/goodies/radmin-totp.cfg` which shows how to authenticate using TOTP token data held in the RAdmin database. HOTP token authentication is in hotp.cfg and it can be updated to work with RAdmin’s RADOATH table schema. Exactly which OATH token actions are available to a particular RAdmin user depends on the Permissions profile assigned to them. The Permissions profile individually controls whether a user can List, Allocate, Deallocate, and Delete tokens.

4.1.9. Yubikey Support

RAdmin can optionally support Yubikey tokens from Yubico (http://www.yubico.com [http://www.yubico.com]). Yubico tokens are small USB devices that act like a keyboard and which type in a one-time-password when the button is pressed. They can be purchased from Yubico and issued to your users. Such tokens provide much higher levels of security than static passwords. Yubikey is supported on all RAdmin platforms.

Each Yubikey token has a unique Token ID (also called the public identity in Yubico documentation), and a secret AES cryptographic key. In order to authenticate a Yubikey token, the RAdmin database must contain a Yubikey record containing both the Token ID and the AES secret for that key. You can add new tokens into the RAdmin database with the RAdmin ‘Import Yubikey Tokens’ page. After a token is imported, it must be allocated to a user before that user can use the token to authenticate.

In order to enable Yubikey token support:

1. Install Radiator in the usual way
2. Install the Auth-Yubikey_Decrypter module from CPAN (www.cpan.org) and the Crypt::Rijndael module, also available from CPAN on the Radiator host.
3. Install RAdmin as described above in this document.
4. On the ‘Edit Radmin Configuration’ page enable the ‘Support Yubikey?’ option. Click ‘**Update**’.
5. RAdmin web pages will now include ‘List Yubikey Tokens’ and ‘Import Yubikey Tokens’, and the Edit User page will include some new options for allocating and listing allocated Yubikey tokens.
6. Configure Radiator based on the example configuration file `Radiator/goodies/radminYubikey.cfg` which shows how to authenticate using Yubikey token data held in the RAdmin database.

Exactly which Yubikey actions are available to a particular RAdmin user depends on the Permissions profile assigned to them. The Permissions profile individually controls whether a user can List, Allocate, Deallocate, and Import tokens.

The administrator will need to use YubiKey Manager or YubiKey Personalization Tool to program the Yubikey with a new 6 byte Token ID (public identity) and a random AES Secret (AES key), then enter the programmed Token ID and AES Secret into the Import Yubikey Tokens’ page.

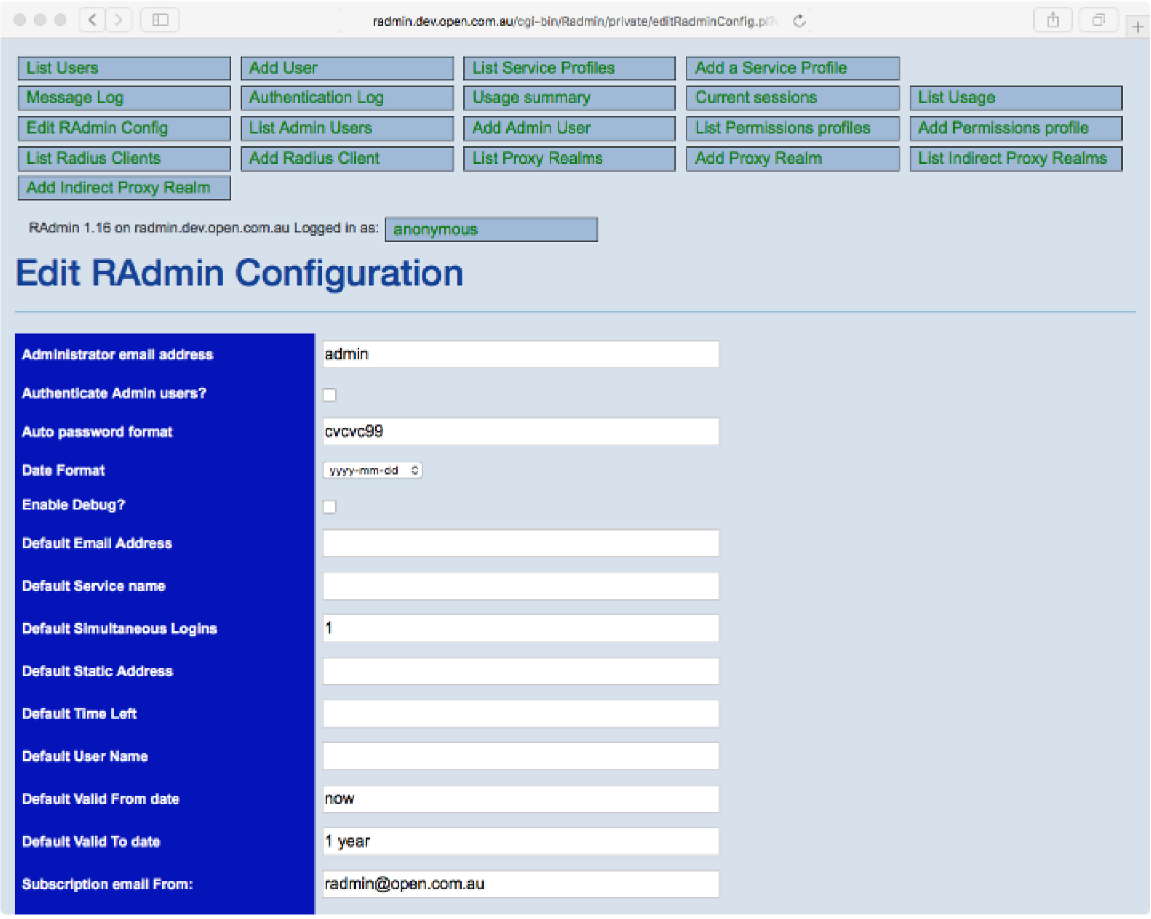
Note

Support for Windows COM/ActiveX browser plugin for programming Yubikeys was removed in RAdmin 1.16.

4.2. Edit RAdmin Configuration

Suitably privileged users can use the ‘Edit RAdmin Config’ page to alter the configuration of your RAdmin system. The configuration options on this page overrides any manually set options in the Site.pm file.

Figure 1. Edit RAdmin Configuration



RAdmin 1.16 on radmin.dev.open.com.au Logged in as: anonymous

Edit RAdmin Configuration

Administrator email address	admin
Authenticate Admin users?	☐
Auto password format	cvcvc99
Date Format	yyyy-mm-dd
Enable Debug?	☐
Default Email Address	
Default Service name	
Default Simultaneous Logins	1
Default Static Address	
Default Time Left	
Default User Name	
Default Valid From date	now
Default Valid To date	1 year
Subscription email From:	radmin@open.com.au

The following chapters describe the available configuration options.

4.2.1. Administrator email address

The email address of the RAdmin system administrator. Used to generate the link in the contact address at the bottom of every RAdmin page.

4.2.2. Authenticate Admin users?

If this option is enabled, Administrative users will be required to log in to RAdmin before being able to use RAdmin.

CAUTION

Do not enable this option until you have configured the required Administrative users and their passwords.

Tip

The default passwords for the default Administrative users ‘anonymous’ and ‘DEFAULT’ are empty.

4.2.3. Auto password format

The format for automatically generated passwords. Used in the ‘Add User’ page. The format consists of a sequence of special characters, which specify how to generate a random password. The following special characters are available:

Table 1. Auto password format special characters

Special Character	Replaced by
a	Any lowercase letter or digit
c	A lowercase consonant (any of bcdfghjklmnpqrstvwxyz)
v	A lowercase vowel (any of aeiou)
9	A digit
any other character	not replaced

For example, if Auto password format is set to *cvcvc99* it would generate passwords like: *hobela56* or *pemedo25*, which are relatively easy for English speakers to pronounce and remember.

4.2.4. Date format

The format that will be used to display dates, and to recognize input dates. Options include:

- yyyy-mm-dd
- dd/mm/yy
- mm/dd/yy

4.2.5. Enable Debug?

Causes RAdmin to print some internal debugging information to STDERR. It is sometimes useful for finding and fixing database access problems. Most web servers will capture this information and log it to a log file. For Apache, this would typically be a file like `/etc/httpd/logs/error_log`. The exact location may depend on your local web server configuration.

4.2.6. Default Email Address

The default Email Address for the ‘Add User’ page.

4.2.7. Default Service name

The default Service Profile for the ‘Add User’ page.

4.2.8. Default Static Address

The default Static IP Address for the ‘Add User’ page.

4.2.9. Default User Name

The default User Name for the ‘Add User’ page. If most of your user names have a common feature (such as an @domain.com), this can be used to set up the default domain suffix.

4.2.10. Default Valid From date

The default Valid From for the ‘Add User’ page. Any of the special relative dates such as ‘today’, ‘tomorrow’ etc. can be used.

4.2.11. Default Valid To date

The default Valid To for the ‘Add User’ page. Any of the special relative dates such as ‘1 week’, ‘1 year’ etc. can be used.

4.2.12. Subscription email From:

The ‘From:’ address that will be used on all automatically generated emails sent by the RAdmin subscription system. Only used if ‘Enable Subscription Management’ is enabled.

4.2.13. Hide Passwords?

If this flag is enabled, the Edit User page will disguise all user passwords, rather than showing them in plaintext. The passwords will be disguised with the string entered in ‘Password Mask’ below.

4.2.14. Time Interval Format

The format used to display time intervals, such as in the ‘List Usage’ and ‘Usage Summary’ pages.

4.2.15. Logo image

The name of your site's log image file. Must be an absolute or a relative URL.

4.2.16. Maximum Admin User session time (mins)

This option specifies the maximum time (in minutes) that an Administrative User authenticated by RAdmin can use RAdmin without re-authenticating. After this time has expired the user will be required to re-authenticate. If set to 0, session times are not limited, and the user will not be required to reauthenticate for as long as they continue to use the same browser.

4.2.17. Password storage format

Specifies how end-user passwords are to be stored in the RAdmin database. Options are shown in table below.

Table 2. Password Storage Formats

Format	Meaning
Plaintext	Stores the exact plain text password.
Unix Crypt	Stored the encrypted password, using the standard unix crypt() function. Note: it is not possible to recover the correct plaintext password from Unix Crypt.

RCrypt	Stores the password encrypted with the Open System Consultants RCrypt encryption algorithm. Requires the ‘Rcrypt key’ to decode.

CAUTION

If you change this, you will also need to alter your Radiator configuration so that it expects the new format. Failure to do this may result in users being unable to log on through Radiator.

4.2.18. Password Mask

When ‘Hide Passwords’ is enabled, this specifies the string that will be used to hide passwords in the ‘Edit User’ page.

4.2.19. Rcrypt Key

When ‘Password storage format’ is set to RCrypt, this is the key that will be used to encrypt and decrypt passwords from the RAdmin database. If you are using RCrypt storage, your Radiator system will have to be configured to use exactly the same Rcrypt key that you enter here.

4.2.20. Subscription email SMTP Server

If ‘Enable Subscription management’ is set, this field specifies the name or address of the SMTP mail server that will be used to send automatically generated email.

4.2.21. Subscription Email Access

If ‘Enable Subscription management’ is set, this field specifies the text that will be used in the body of emails sent when access has been granted to one or more subscription products. It will typically be used to tell the user their username and password. IN the text, the special character %0 will be replaced with the users User Name, and %1 will be replaced with their password.

4.2.22. Subscription Email Body

If ‘Enable Subscription management’ is set, this field specifies the text of email that is sent whenever access to a subscribed product(s) changes. %0 is replaced by password details (see Subscription Email Access above). If access to any product has been added or extended %1 is replaced by the product access change details.

4.2.23. Subscription Email Subject

If ‘Enable Subscription management’ is set, this field specifies the Subject: line that will be used on all outgoing emails about subscription changes.

4.2.24. Enable Subscription management

This flag enables RAdmin subscription management option. When enabled, administrators are able to define subscription products, and then select which product(s) end users are subscribed to.

4.2.25. Support OneSpan Digipass

If this flag is set and the Authen-Digipass Perl module is installed on the RAdmin server host, then a number of new menu items and links will appear for importing, listing and assigning OneSpan, formerly known as Vasco, Digipass tokens for each user. See Section 4.1.7. Digipass Support on page 12 for more details.

4.2.26. Support OATH (TOTP and HOTP)

If this flag is set and the MIME-Base32 and Imager-QRCode Perl modules are installed on the RAdmin server host, then a new menu item and links will appear for importing, listing and assigning TOTP and HOTP tokens for each user. See Section 4.1.8. TOTP and HOTP (OATH) token Support on page 12 for more details.

4.2.27. Support Yubikey

If this flag is set and the Auth-Yubikey_Decrypter Perl module is installed on the RAdmin server host, then a number of new menu items and links will appear for importing, listing and assigning Yubico (www.yubico.com) Yubikey tokens for each user. See Section 4.1.9. Yubikey Support on page 13 for more details

4.2.28. Table heading attributes

Customizes the look of headings in the tables on various list pages. Any HTML that can be used inside <th>...</th> tags is valid.

4.2.29. Table heading font

Customizes the font of headings in the tables on various list pages. Any HTML that can be used inside a <font ...> tag is valid.

4.2.30. User help document

Absolute or relative URL for the location of the RAdmin user administrator help document.

4.2.31. Volume Interval Format

The format used to display data volumes, such as in the ‘List Usage’ and ‘Usage Summary’ pages.

4.3. Site.pm

You can configure some of RAdmin’s behaviour by editing the file Radmin/Site.pm (Radmin\Site.pm on Windows) and reinstalling it. Some of the things that can be customized there are:

Your date format Your auto-password format Default From and To dates for new users Advanced administrators with special site-specific requirements can also override RAdmin functions by adding replacement functions to Site.pm. That way you can trigger your own site specific code during the operation of RAdmin. When RAdmin is first installed, all the configurable behaviour of RAdmin is determined by Site.pm. If the ‘Edit RAdmin Config’ page is used to customize RAdmin, then it overrides Site.pm.

- Your date format
- Your auto-password format
- Default From and To dates for new users

Advanced administrators with special site-specific requirements can also override RAdmin functions by adding replacement functions to Site.pm. That way you can trigger your own site specific code during the operation of RAdmin.

When RAdmin is first installed, all the configurable behaviour of RAdmin is determined by Site.pm. If the ‘Edit RAdmin Config’ page is used to customize RAdmin, then it overrides Site.pm.

4.4. Database selection

In some organizations, it is required to manage several distinct RAdmin databases, each with its own set of users, permissions and RAdmin users. This allows you to outsource or onsell administration functions for distinct groups of users.

RAdmin can be configured to allow this, however to configure it requires advanced system administration ability. Contact info@radiatorsoftware.com for details.

5. Converting databases

You may need to populate your new RAdmin database with information about existing users from an existing database. RAdmin comes with a program that will convert users from an existing simple Radiator SQL user database or a Unix password style file.

5.1. From a Radiator SQL user database

The convert program in the goodies directory does this conversion. It understands the following arguments:

```
convert [-h] [-v] [-update] [-password] [-dbsource dbi:drivername:option]
[-dbusername dbusername] [-dbauth auth] [-validfrom YYYYMMDD]
[-validto YYYYMMDD] [filename ...]
```

-
  ```
  -h
  ```

  Print help information and exit

-
  ```
  -v
  ```

  Print verbose information about inserts and updates

-
  ```
  -update
  ```

  If the user already exists in the target RAdmin database, update their information. Default is to print a warning and not update the user

-
  ```
  -password
  ```

  Interprets the filenames specified on the command line as Unix style password files, and inserts or updates the user details, including user name, encrypted password, full name and added date. Disables reading from SQL, and causes -dbsource, -dbusername and -dbauth to be ignored.

-
  ```
  -dbsource dbi:drivername:option
  ```

  Specifies the data source name of the database to read from. Must be specified unless -password is used. Use standard DBI connection strings, such as for example

  ```
  dbi:Oracle:radiusdb
  dbi:FreeTDS:database=master;host=fred;port=1433;
  dbi:Pg:dbname=radiator
  ```

-
  ```
  -dbusername username
  ```

  Specifies the username to use to connect to the SQL database to read from. dbauth password Specifies the password for dbusername. Not required for some database types.

-
  ```
  -dbsource
  ```

  Print help information and exit

-
  ```
  -validfrom
  ```

  Specifies an optional Valid From date for all users affected by this command. Format is any date/time format supported by RAdmin. The default is to not set Valid From date.

-
  ```
  -validto
  ```

  Specifies an optional Valid To date for all users affected by this command. Format is any date/time format supported by RAdmin. If -validfrom is set, the default is 1 year from the Valid From date, else the default is to not to set Valid To date.

Examples:

```
perl goodies/convert -password -validfrom tod -validto '6 months' /etc/passwd
```

User details are read from /etc/passwd. Users that are not already in the RAdmin database are added. Username and encrypted password and full name are added, added date is set to today. The valid from date is set to midnight at the beginning of today, and the valid to date is set to 6 months from today.

```
perl goodies/convert -dbsource dbi:Oracle:radiusdb \
-dbusername mikem -dbauth fred -update
```

Connects to the Radiator database on the Oracle SID “radiusdb”, using the Oracle user name mikem, and Oracle password “fred”. All users found in the database are inserted or updated into the RAdmin database. For new users, the Added Date is set to today, but the Valid From and Valid To dates are not set for any users.

6. Altering RAdmin

RAdmin is delivered with full source code, allowing system administrators to alter the behaviour of RAdmin at a number of levels. The following chapters show some ways.

6.1. Look and Feel

Every RAdmin web page loads the CSS style sheet /Radmin/radmin.css from the web server, if it is present. Adding a CSS style sheet file at this location is an easy way to control the look and feel of the RAdmin web pages. Using CSS you can customise colours, layout, images and much more. Consult a CSS reference manual for more details on how to do this.

6.2. Site.pm

This Perl module is loaded by every RAdmin web page and application. You can customise it to alter various configurable variables or override various RAdmin Perl functions. See Site.pm for some examples.

7. Getting Help

The following chapters describe support possibilities.

7.1. Support contract holders

Support may be purchased at the time you purchase RAdmin. For more information, see https://radiatorsoftware.com/support/ for details. Support contracts last for a limited period (typically one year) and may include a limited amount of pre-paid email support.

Radiator Software will respond promptly to support email from support contract holders during business hours. We will keep track of the effort required to answer your support email, and inform you when your prepaid support time has expired.

If you have a RAdmin or Radiator support contract, you may send email to support@radiatorsoftware.com

If you don’t have a support contract, we will not respond to your query on this address.

If you need an urgent response outside of the standard email support hours, you may want to post to the public Radiator mailing list instead. Someone will be sure to be awake somewhere in the world.

7.2. No support contract

The standard RAdmin license does not include support, but it does include the full source code and free access to the Radiator mailing list. This means you can help yourself, and you can work with other RAdmin users in the user community. In order to participate with others in this effort, you can join the public Radiator mailing list at https://lists.open.com.au/mailman/listinfo/radiator

The staff of Radiator Software monitor the Radiator mailing list and frequently answer questions. Its very active so don't hesitate to use it.

7.3. What to do if you need help

Before you post to the support address or mailing list asking for assistance, we suggest you go through the following check list:

1. Consult this reference manual.
2. Consult the RAdmin FAQ for extra hints.
3. Check that you are using the latest version of RAdmin. See https://www.open.com.au/radmin/downloads, use the username and password we have issued to you. Upgrade if you need to.
4. Check whether there are any patches that address your problem. Apply the patch if you think you might need it.
5. If you still have the problem, post to the public mailing list by mailing to: radiator@lists.open.com.au. If you have a support contract, send email to support@radiatorsoftware.com. Be sure to include at least the following information:
    a. A detailed description of the problem.
    b. Your Radiator configuration file (remove any secrets and passwords first).
    c. An extract from your Radiator log file (with Trace level of 4) illustrating the problem, or at least what is happening at the time of the problem.
    d. Details of the computer type, operating system etc.

This information helps people to understand your problem and help find a solution more quickly.

7.4. Bug reports

We are interested in your feedback, both positive and negative, and bug reports. Please send them to info@radiatorsoftware.com. Licensees are entitled to free upgrades, and we do fix bugs that are reported to us, so if you report a bug, you can expect to get an upgrade with a fix. If you don't report it, it might never get fixed.