

# Tech Note - RADIUS Troubleshooting

## Overview

RADIUS Authentication, Authorisation and Accounting (AAA) is a core component of the amigopod platform and therefore being able to effectively troubleshoot any authentication issues between Access Controllers (RADIUS NAS devices) and the amigopod is essential.

By default amigopod is configured to communicate RADIUS traffic on the following ports:

- Authentication transaction on UDP port 1812
- Accounting transactions on UDP port 1813
- RFC 3576 transactions on UDP port 3799

The inbuilt firewall rules that protect the amigopod kernel are automatically provisioned to allow traffic to flow to and from these ports.

## Basic Configuration

When configuring your network environment you must ensure that your Access Controller's RADIUS server definitions for both Authentication and Accounting are configured to match the above port settings. If your deployment demands that this default ports be changed from 1812 and 1813 these can be modified through the amigopod web interface under *RADIUS Services* → *Server Configuration* as shown in the screenshot below:

The screenshot displays the 'radius server configuration' page in the amigopod web interface. The page title is 'radius server configuration' and the subtitle is 'Use this page to set advanced configuration options for the RADIUS server.' The main content area contains a 'RADIUS Server Configuration' form with the following fields and options:

- Port Number:** 1812 (required field). Description: Base port number to use for RADIUS authentication. Accounting will use the next consecutive port number, and proxying will use the next port number again.
- Options:**  Include active sessions when calculating total account usage. Description: If checked, a user's active sessions will be included when calculating the cumulative accounting session times for a user account.
- Advanced Configuration:** A text area containing the following configuration options:

```
# Uncomment these lines to enable these options:  
nas.default_secret = "nwaradius"  
nas.default_name = "nwa"  
#security.reject_delay = 0  
#proxy_requests = no
```
- Server Options:** A text area for additional RADIUS server options. Description: Additional RADIUS server options. Enter name = value pairs on separate lines. Comments may be entered on lines starting with a "#".

A 'Save Changes' button is located at the bottom right of the form. Below the form, there are links for 'RADIUS Services' and 'Back to main'. A legend indicates that an asterisk (\*) denotes a required field.

The sidebar menu on the left includes the following categories and items:

- Home**
  - Start Here
  - Language
  - Time Zone
  - Translation Assistant
- Guest Manager**
  - Start Here
  - Create Account
  - Create Multiple
  - List Accounts
  - Edit Accounts
  - Active Sessions
  - Import Accounts
  - Export Accounts
  - Print Templates
  - Customization
- Hotspot Manager**
  - Start Here
  - Self Provisioning
  - Self Service
  - Manage Hotspot
  - Manage Plans
  - Manage Customer Info
  - Manage Invoice
  - Manage User Interface
- Reporting Manager**
  - Start Here
  - List Reports
- Administrator**
  - Start Here
  - Backup & Restore
  - Content Manager
  - Network Setup

## Basic Diagnostics

A basic display of the most recent RADIUS transactions is logged to the screen of the amigopod interface found under *RADIUS Services* → *Server Control* as shown in the screenshot below. Simple error messages such as unknown NAS & Incorrect passwords can be quickly gleaned from this screen.

**radius server control**

Control the local RADIUS server using these command links.

The local RADIUS server needs to be restarted to complete the changes made.

The RADIUS server is currently running.

**Restart RADIUS Server**  
Restart the local RADIUS server.

**Stop RADIUS Server**  
Stop the local RADIUS server.

**Debug RADIUS Server**  
Run the local RADIUS server and see detailed log output.

**RADIUS Server Time**

The RADIUS server time is currently: **Thursday, 23 April 2009 4:35:16 PM +1000**

**RADIUS Log Snapshot**

The most recent entries in the RADIUS server log file are shown below.

```
Thu Apr 23 16:34:38 2009 : Auth: Login OK: [amigopod] (from client sig-mcd-col-1 port 143)
Thu Apr 23 16:34:37 2009 : Auth: Login OK: [amigopod] (from client sig-mcd-col-1 port 120)
Thu Apr 23 16:34:24 2009 : Auth: Login OK: [amigopod] (from client sig-mcd-col-1 port 336)
Thu Apr 23 16:34:21 2009 : Auth: Login OK: [amigopod] (from client sig-mcd-col-1 port 257)
Thu Apr 23 16:34:20 2009 : Auth: Login OK: [amigopod] (from client sig-mcd-col-1 port 246)
```

If you are attempting to authenticate from your configured Access Controller and no entries are being display is the log, please consider double checking the following settings and scenarios:

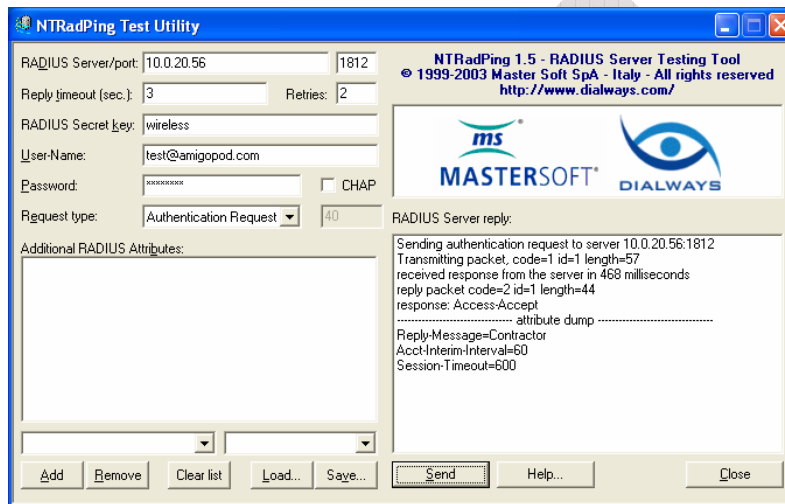
- The RADIUS configuration on your NAS is pointing to the correct IP address defined on the amigopod
- Basic IP connectivity can be verified between the NAS and the amigopod using ping or traceroute
- Are there any firewalls or VLAN configurations between the NAS and the amigopod preventing basic IP communications

## Diagnostic Tools

A great tool for testing basic IP connectivity and RADIUS availability on the amigopod is to download an open source RADIUS test client such as NTRadPing. This tool is available for download from Novell at the following URL:

<http://www.novell.com/coolsolutions/tools/14377.html>

The NTRadPing tool allows you to define a RADIUS destination IP Address, port number and a test user. This is incredibly useful for determining whether the RADIUS transactions are successfully making it to the amigopod or being blocked by VLAN configurations or firewall rules. See screenshot below of a successful authentication attempt from NTRadPing to the amigopod.



In order to successfully authenticate to the amigopod you must define a NAS entry under the *RADIUS Services* → *NAS List* with the IP Address of your test computer. As with all RADIUS configurations the shared secret and 1812 must match on both the client and the amigopod.

## Detailed Troubleshooting

If deeper troubleshooting is required, you can enable a detailed RADIUS debugger session by clicking on the *Debug RADIUS Server* button shown in the amigopod screenshot below.



When debugging RADIUS problems, the #1 problem to watch out for is **incorrect shared secrets**. This is almost always the problem when diagnosing a "Login incorrect" message.

The following section shows some typical failure scenarios and what you can expect to see in the amigopod RADIUS Debug screens.

For completeness, we have tried all 8 combinations of PAP/CHAP authentication, correct password/incorrect password, and correct shared secret/incorrect shared secret.

The results are as follows:

Correct Password - Correct shared secret - CHAP ==> Access-Accept

Correct Password - Correct shared secret - PAP ==> Access-Accept

Correct password - Incorrect shared secret - CHAP ==> Access-Accept

Correct password - Incorrect shared secret - PAP ==> Access-Reject Login incorrect:  
[demo@example.com]

Incorrect password - Correct shared secret - CHAP ==> Access-Reject Login incorrect  
(rlm\_chap: Wrong user password): [demo@example.com]

Incorrect password - Correct shared secret - PAP ==> Access-Reject Login incorrect:  
[demo@example.com]

Incorrect password - Incorrect shared secret - CHAP ==> Access-Reject Login incorrect  
(rlm\_chap: Wrong user password): [demo@example.com]

Incorrect password - Incorrect shared secret - PAP ==> Access-Reject Login incorrect:  
[demo@example.com]

There are 2 things to note:

1. When using PAP, it is impossible to distinguish between an incorrect shared secret and an incorrect user password. But in 2 out of 3 cases above the problem is the shared secret is wrong, not that the password is wrong.
2. When using CHAP, if the shared secret is incorrect the server is still able to authenticate and returns Access-Accept!

This is believed to be a result of the Message-Authenticator attribute was not present in our test scenario ; when this attribute is present in requests from the client, the RADIUS server is able to use it to determine if the shared secret is in fact correct or not (as the Message-Authenticator value is calculated using the shared secret, if it's wrong then a mismatch will occur). It is understood that when using Message-Authenticator the server will log a message about the shared secret being incorrect.

**Moral of the story:** When debugging a RADIUS authentication problem, one of the **FIRST steps should ALWAYS be to check and reset the shared secrets for clients if necessary.** This is likely to solve 90% of authentication problems.

Complete logs are below for reference:

```

////////////////////////////////////
// Correct Password - Correct shared secret - CHAP

Ready to process requests.
rad_recv: Access-Request packet from host 192.168.2.3:2388, id=1, length=76
User-Name = "demo@example.com"
CHAP-Password = 0x742779bdd72bef15defaf90121a2807799
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_chap: Setting 'Auth-Type := CHAP'
rlm_sql (sql): Reserving sql socket id: 3
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 3
rlm_chap: Login attempt by "demo@example.com" with CHAP password
rlm_chap: Using clear text password password for user demo@example.com authentication.
rlm_chap: chap user demo@example.com authenticated successfully
Exec-Program: /usr/bin/php /opt/ami_gopod/www/ami_gopod_request.php 2 1
Exec-Program-Wait: value-pairs: Reply-Message = "Guest", Session-Timeout = 874,
Exec-Program: returned: 0
Login OK: [demo@example.com] (from client hydra port 0 cli 00-17-31-57-d8-78)
rlm_sql (sql): Processing sql_postauth
rlm_sql (sql): Reserving sql socket id: 2
rlm_sql_postgresql: query: INSERT INTO radpostauth (username, pass, reply, authdate) VALUES
('demo@example.com', 'Chap-Password', 'Access-Accept', NOW())
rlm_sql_postgresql: Status: PGRES_COMMAND_OK
rlm_sql_postgresql: affected rows = 1
rlm_sql (sql): Released sql socket id: 2
Sending Access-Accept of id 1 to 192.168.2.3 port 2388
Reply-Message = "Guest"
Session-Timeout = 874

```

```

////////////////////////////////////
// Correct Password - Correct shared secret - PAP

```

```

rad_recv: Access-Request packet from host 192.168.2.3:2406, id=2, length=75
User-Name = "demo@example.com"
User-Password = "password"
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_sql (sql): Reserving sql socket id: 1
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 1
Exec-Program: /usr/bin/php /opt/ami_gopod/www/ami_gopod_request.php 2 1
Exec-Program-Wait: value-pairs: Reply-Message = "Guest", Session-Timeout = 823,
Exec-Program: returned: 0
Login OK: [demo@example.com] (from client hydra port 0 cli 00-17-31-57-d8-78)
rlm_sql (sql): Processing sql_postauth
rlm_sql (sql): Reserving sql socket id: 0
rlm_sql_postgresql: query: INSERT INTO radpostauth (username, pass, reply, authdate) VALUES
('demo@example.com', 'password', 'Access-Accept', NOW())
rlm_sql_postgresql: Status: PGRES_COMMAND_OK
rlm_sql_postgresql: affected rows = 1
rlm_sql (sql): Released sql socket id: 0
Sending Access-Accept of id 2 to 192.168.2.3 port 2406
Reply-Message = "Guest"
Session-Timeout = 823

```

```

////////////////////////////////////
// Correct password - Incorrect shared secret - CHAP

```

```

rad_recv: Access-Request packet from host 192.168.2.3:2423, id=3, length=76
User-Name = "demo@example.com"
CHAP-Password = 0x75a9b8959bf829fb06d916af17fc60e0e7
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_chap: Setting 'Auth-Type := CHAP'
rlm_sql (sql): Reserving sql socket id: 4
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 4
rlm_chap: Login attempt by "demo@example.com" with CHAP password
rlm_chap: Using clear text password password for user demo@example.com authentication.
rlm_chap: chap user demo@example.com authenticated successfully
Exec-Program: /usr/bin/php /opt/ami_gopod/www/ami_gopod_request.php 2 1
Exec-Program-Wait: value-pairs: Reply-Message = "Guest", Session-Timeout = 795,
Exec-Program: returned: 0
Login OK: [demo@example.com] (from client hydra port 0 cli 00-17-31-57-d8-78)
rlm_sql (sql): Processing sql_postauth
rlm_sql (sql): Reserving sql socket id: 3
rlm_sql_postgresql: query: INSERT INTO radpostauth (username, pass, reply, authdate) VALUES
('demo@example.com', 'Chap-Password', 'Access-Accept', NOW())
rlm_sql_postgresql: Status: PGRES_COMMAND_OK
rlm_sql_postgresql: affected rows = 1
rlm_sql (sql): Released sql socket id: 3
Sending Access-Accept of id 3 to 192.168.2.3 port 2423

```

Reply-Message = "Guest"  
 Session-Timeout = 795

////////////////////////////////////  
 // Correct password - Incorrect shared secret - PAP

```
rad_recv: Access-Request packet from host 192.168.2.3:2442, id=5, length=75
User-Name = "demo@example.com"
User-Password = "\365\336\267\236\253\243F= Jt\251\317\272`R"
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_sql (sql): Reserving sql socket id: 1
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 1
rlm_sql (sql): No matching entry in the database for request from user [demo@example.com]
Login incorrect: [demo@example.com] (from client hydra port 0 cli 00-17-31-57-d8-78)
Sending Access-Reject of id 5 to 192.168.2.3 port 2442
```

////////////////////////////////////  
 // Incorrect password - Correct shared secret - CHAP

```
rad_recv: Access-Request packet from host 192.168.2.3:2495, id=7, length=76
User-Name = "demo@example.com"
CHAP-Password = 0x069b51c50ccfd0c103623a0cae7952b9ca
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_chap: Setting 'Auth-Type := CHAP'
rlm_sql (sql): Reserving sql socket id: 4
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 4
rlm_chap: Login attempt by "demo@example.com" with CHAP password
rlm_chap: Using clear text password password for user demo@example.com authentication.
rlm_chap: Password check failed
Login incorrect (rlm_chap: Wrong user password): [demo@example.com] (from client hydra port 0
cli 00-17-31-57-d8-78)
Sending Access-Reject of id 7 to 192.168.2.3 port 2495
```

////////////////////////////////////  
 // Incorrect password - Correct shared secret - PAP

```
rad_recv: Access-Request packet from host 192.168.2.3:2514, id=8, length=75
User-Name = "demo@example.com"
User-Password = "incorrect"
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_sql (sql): Reserving sql socket id: 3
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
```

```

rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 3
rlm_sql (sql): No matching entry in the database for request from user [demo@example.com]
Login incorrect: [demo@example.com] (from client hydra port 0 cli 00-17-31-57-d8-78)
Sending Access-Reject of id 8 to 192.168.2.3 port 2514

```

```

////////////////////////////////////
// Incorrect password - Incorrect shared secret - CHAP

```

```

rad_recv: Access-Request packet from host 192.168.2.3:2568, id=9, length=76
User-Name = "demo@example.com"
CHAP-Password = 0xb6fbf9aea3255b30d81b5f49ae60a6f972
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_chap: Setting 'Auth-Type := CHAP'
rlm_sql (sql): Reserving sql socket id: 2
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql (sql): Released sql socket id: 2
rlm_chap: login attempt by "demo@example.com" with CHAP password
rlm_chap: Using clear text password password for user demo@example.com authentication.
rlm_chap: Password check failed
Login incorrect (rlm_chap: Wrong user password): [demo@example.com] (from client hydra port 0
cli 00-17-31-57-d8-78)
Sending Access-Reject of id 9 to 192.168.2.3 port 2568

```

```

////////////////////////////////////
// Incorrect password - Incorrect shared secret - PAP

```

```

rad_recv: Access-Request packet from host 192.168.2.3:2570, id=10, length=75
User-Name = "demo@example.com"
User-Password = "\345\030h\324JB\224\300\343^\033\207\337G\314v"
Calling-Station-Id = "00-17-31-57-d8-78"
rlm_sql (sql): Reserving sql socket id: 1
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radcheck WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupcheck.id, radgroupcheck.GroupName,
radgroupcheck.Attribute, radgroupcheck.Value, radgroupcheck.Op ??FROM radgroupcheck, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupcheck.GroupName ??ORDER BY radgroupcheck.id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT id, UserName, Attribute, Value, Op FROM radreply WHERE
Username='demo@example.com' ORDER BY id
rlm_sql_postgresql: Status: PGRES_TUPLES_OK
rlm_sql_postgresql: affected rows =
rlm_sql_postgresql: query: SELECT radgroupreply.id, radgroupreply.GroupName,
radgroupreply.Attribute, radgroupreply.Value, radgroupreply.Op ??FROM radgroupreply, usergroup
WHERE usergroup.Username = 'demo@example.com' AND usergroup.GroupName =
radgroupreply.GroupName ??ORDER BY radgroupreply.id

```



```
rlm_sql_postgresql: Status: PGRES_TUPLES_OK  
rlm_sql_postgresql: affected rows =  
rlm_sql (sql): Released sql socket id: 1  
rlm_sql (sql): No matching entry in the database for request from user [demo@example.com]  
Login incorrect: [demo@example.com] (from client hydra port 0 cli 00-17-31-57-d8-78)  
Sending Access-Reject of id 10 to 192.168.2.3 port 2570
```

DRAFT