Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'eapol_test.8' command

$ man eapol_test.8

EAPOL_TEST(8) EAPOL_TEST(8)

NAME

eapol_test - EAP peer and RADIUS client testing

SYNOPSIS

eapol_test [-nWS] [-cconfig file] [-aserver IP address] [-Aclient IP address] [-pUDP port] [-sshared secret] [-rre-authentications] [-ttimeout] [-CConnect-Info] [-MMAC address] [-ofile] [-Nattr spec]

eapol_test scard

eapol_test sim [PIN] [num triplets]

OVERVIEW

eapol_test is a program that links together the same EAP peer implementation that wpa_supplicant is using and the RADIUS authentication client code from hostapd. In addition, it has minimal glue code to combine these two components in similar ways to IEEE 802.1X/EAPOL Authenticator state machines. In other words, it integrates IEEE 802.1X Authenticator (normally, an access point) and IEEE 802.1X Supplicant (normally, a wireless client) together to generate a single program

that can be used to test EAP methods without having to setup an access point and a wireless client.

The main uses for eapol_test are in interoperability testing of EAP methods against RADIUS servers and in development testing for new EAP methods. It can be easily used to automate EAP testing for interoperability and regression since the program can be run from shell scripts without require additional test components apart from a RADIUS server. For example, the automated EAP tests described in eap_testing.txt are implemented with eapol_test. Similarly, eapol_test could be used to implement an automated regression test suite for a RADIUS authentication server.

As an example:

```
eapol_test -ctest.conf -a127.0.0.1 -p1812 -ssecret -r1
```

tries to complete EAP authentication based on the network configuration from test.conf against the RADIUS server running on the local host. A re-authentication is triggered to test fast re-authentication. The configuration file uses the same format for network blocks as wpa_supplicant.

COMMAND ARGUMENTS

-c configuration file path

A configuration to use. The configuration should use the same format for network blocks as wpa_supplicant.

-a AS address

IP address of the authentication server. The default is '127.0.0.1'.

-A client address

IP address of the client. The default is to select an address automatically.

-p AS port

UDP port of the authentication server. The default is '1812'.

-s AS secret

Shared secret with the authentication server. The default is 'radius'.

-r count

Number of reauthentications.

-t timeout

Timeout in seconds. The default is 30.

-C info

RADIUS Connect-Info. The default is 'CONNECT 11Mbps 802.11b'.

-M mac address

Client MAC address (Calling-Station-Id). The default is '02:00:00:00:00:01'.

-o file

Location to write out server certificate.

-N attr spec

Send arbitrary attribute specific by attr_id:syntax:value, or attr_id alone. attr_id should be the numeric ID of the attri? bute, and syntax should be one of 's' (string), 'd' (integer), or 'x' (octet string). The value is the attribute value to send. When attr_id is given alone, NULL is used as the attribute value. Multiple attributes can be specified by using the option

several times.

-n Indicates that no MPPE keys are expected.

-W Wait for a control interface monitor before starting.

-S Save configuration after authentication.

SEE ALSO

wpa_supplicant(8)

LEGAL

wpa_supplicant is copyright (c) 2003-2022, Jouni Malinen <j@w1.fi> and contributors. All Rights Reserved.

This program is licensed under the BSD license (the one with advertise? ment clause removed).