NAME

wpa_cli - text-based frontend program for interacting with wpa_supplicant

SYNOPSIS

wpa_cli [**-p** *path_to_ctrl_sockets*] [**-i** *ifname*] [**-hvB**] [**-a** *action_file*] [**-P** *pid_file*] [**-g** *global_ctrl*] [**-G** *ping_interval*] *command ...*

DESCRIPTION

The **wpa_cli** utility is a text-based frontend program for interacting with wpa_supplicant(8). It is used to query current status, change configuration, trigger events, and request interactive user input.

The **wpa_cli** utility can show the current authentication status, selected security mode, dot11 and dot1x MIBs, etc. In addition, **wpa_cli** can configure EAPOL state machine parameters and trigger events such as reassociation and IEEE 802.1X logoff/logon.

The **wpa_cli** utility provides an interface to supply authentication information such as username and password when it is not provided in the wpa_supplicant.conf(5) configuration file. This can be used, for example, to implement one-time passwords or generic token card authentication where the authentication is based on a challenge-response that uses an external device for generating the response.

The **wpa_cli** utility supports two modes: interactive and command line. Both modes share the same command set and the main difference is in interactive mode providing access to unsolicited messages (event messages, username/password requests).

Interactive mode is started when **wpa_cli** is executed without any parameters on the command line. Commands are then entered from the controlling terminal in response to the **wpa_cli** prompt. In command line mode, the same commands are entered as command line arguments.

The control interface of wpa_supplicant(8) can be configured to allow non-root user access by using the *ctrl_interface_group* parameter in the wpa_supplicant.conf(5) configuration file. This makes it possible to run **wpa_cli** with a normal user account.

AUTHENTICATION PARAMETERS

When wpa_supplicant(8) needs authentication parameters, such as username and password, that are not present in the configuration file, it sends a request message to all attached frontend programs, e.g., **wpa_cli** in interactive mode. The **wpa_cli** utility shows these requests with a "CTRL-REQ-<*type*>-<*id*>:<*text*>" prefix, where <*type*> is IDENTITY, PASSWORD, or OTP (One-Time Password), <*id*> is a unique identifier for the current network, <*text*> is a description of the request. In the case of an OTP (One-Time Password) request, it includes the challenge from the authentication server.

A user must supply wpa_supplicant(8) the needed parameters in response to these requests.

For example,

CTRL-REQ-PASSWORD-1:Password needed for SSID foobar
> password 1 mysecretpassword

Example request for generic token card challenge-response:

CTRL-REQ-OTP-2:Challenge 1235663 needed for SSID foobar
> otp 2 9876

OPTIONS

These options are available:

-p *path*
Control sockets path. This should match the **ctrl_interface** in wpa_supplicant.conf(5). The default path is */var/run/wpa_supplicant*.

-i *ifname*
Interface to be configured. By default, the first interface found in the socket path is used.

-h Show help.

-v Show version information.

-B Run the daemon in the background.

-a *action_file*
Run in daemon mode, executing the action file based on events from wpa_supplicant(8).

-P *pid_file*
PID file location.

-g *global_ctrl*
Use a global control interface to wpa_supplicant(8) rather than the default Unix domain sockets.

-G *ping_interval*
Wait "ping_interval" seconds before sending each ping to wpa_supplicant(8). See the **ping** command.

command

See available commands in the next section.

COMMANDS

These commands can be supplied on the command line or at a prompt when operating interactively.

status Report the current WPA/EAPOL/EAP status for the current interface.

ifname

Show the current interface name. The default interface is the first interface found in the socket path.

ping Ping the wpa_supplicant(8) utility. This command can be used to test the status of the wpa_supplicant(8) daemon.

mib Report MIB variables (dot1x, dot11) for the current interface.

help Show usage help.

interface [*ifname*]

Show available interfaces and/or set the current interface when multiple interfaces are available.

level *debug_level*

Change the debugging level in wpa_supplicant(8). Larger numbers generate more messages.

license

Display the full license for **wpa_cli**.

logoff Send the IEEE 802.1X EAPOL state machine into the "logoff" state.

logon Send the IEEE 802.1X EAPOL state machine into the "logon" state.

set [*settings*]

Set variables. When no arguments are supplied, the known variables and their settings are displayed.

pmksa

Show the contents of the PMKSA cache.

reassociate

Force a reassociation to the current access point.

reconfigure
Force wpa_supplicant(8) to re-read its configuration file.

preauthenticate *BSSID*
Force preauthentication of the specified *BSSID*.

identity *network_id identity*
Configure an identity for an SSID.

password *network_id password*
Configure a password for an SSID.

new_password *network_id password*
Change the password for an SSID.

PIN *network_id pin*
Configure a PIN for an SSID.

passphrase *network_id passphrase*
Configure a private key passphrase for an SSID.

bssid *network_id bssid*
Set a preferred BSSID for an SSID

blacklist [*bssid* | *clear*]
Add a BSSID to the blacklist. When invoked without any extra arguments, display the blacklist. Specifying *clear* causes **wpa_cli** to clear the blacklist.

list_networks
List configured networks.

select_network *network_id*
Select a network and disable others.

enable_network *network_id*
Enable a network.

disable_network *network_id*

Disable a network.

add_network
Add a network.

remove_network *network_id*
Remove a network.

set_network [*network_id variable value*]
Set network variables. Shows a list of variables when run without arguments.

get_network *network_id variable*
Get network variables.

disconnect
Disconnect and wait for reassociate/reconnect command before connecting.

reconnect
Similar to **reassociate**, but only takes effect if already disconnected.

scan Request new BSS scan.

scan_results
Get the latest BSS scan results. This command can be invoked after running a BSS scan with **scan**.

bss [*idx* | *bssid*]
Get a detailed BSS scan result for the network identified by "bssid" or "idx".

otp *network_id password*
Configure a one-time password for an SSID.

terminate
Force wpa_supplicant(8) to terminate.

interface_add *ifname* [*confname driver ctrl_interface driver_param bridge_name*]
Add a new interface with the given parameters.

interface_remove *ifname*
Remove the interface.

interface_list
List available interfaces.

quit Exit **wpa_cli**.

SEE ALSO

wpa_supplicant.conf(5), wpa_supplicant(8)

HISTORY

The **wpa_cli** utility first appeared in FreeBSD 6.0.

AUTHORS

The **wpa_cli** utility was written by Jouni Malinen <*j@w1.fi*>. This manual page is derived from the *README* and *wpa_cli.c* files included in the **wpa_supplicant** distribution.