NAME

wpa_cli − WPA command line client

SYNOPSIS

wpa_cli [**-p** *path to ctrl sockets*] [**-g** *path to global ctrl_interface socket*] [**-i** *ifname*] [**-hvB**] [**-a** *action file*] [**-P** *pid file*] [**-G** *ping interval*] [*command ...*]

OVERVIEW

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

wpa_cli can show the current authentication status, selected security mode, dot11 and dot1x MIBs, etc. In addition, it can configure some variables like EAPOL state machine parameters and trigger events like reassociation and IEEE 802.1X logoff/logon. wpa_cli provides a user interface to request authentication information, like username and password, if these are not included in the configuration. This can be used to implement, e.g., 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 control interface of wpa_supplicant can be configured to allow non-root user access (ctrl_interface GROUP= parameter in the configuration file). This makes it possible to run wpa_cli with a normal user account.

wpa_cli 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 including the command as a command line parameter. Commands are then entered on the wpa_cli prompt. In command line mode, the same commands are entered as command line arguments for wpa_cli.

INTERACTIVE AUTHENTICATION PARAMETERS REQUEST

When wpa_supplicant need authentication parameters, like username and password, which are not present in the configuration file, it sends a request message to all attached frontend programs, e.g., wpa_cli in interactive mode. wpa_cli shows these requests with "CTRL-REQ-<type>-<id>:<text>" prefix. <type> is IDENTITY, PASSWORD, or OTP (one-time-password). <id> is a unique identifier for the current network. <text> is description of the request. In case of OTP request, it includes the challenge from the authentication server.

The reply to these requests can be given with **identity**, **password**, and **otp** commands. <id> needs to be copied from the matching request. **password** and **otp** commands can be used regardless of whether the request was for PASSWORD or OTP. The main difference between these two commands is that values given with **password** are remembered as long as wpa_supplicant is running whereas values given with **otp** are used only once and then forgotten, i.e., wpa_supplicant will ask frontend for a new value for every use. This can be used to implement one-time-password lists and generic token card -based authentication.

Example request for password and a matching reply:

```
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
```

COMMAND ARGUMENTS

-p path Change the path where control sockets should be found.

-g control socket path
Connect to the global control socket at the indicated path rather than an interface-specific control socket.

-i ifname
Specify the interface that is being configured. By default, choose the first interface found with a control socket in the socket path.

-h Help. Show a usage message.

-v Show version information.

-B Run as a daemon in the background.

-a file Run in daemon mode executing the action file based on events from wpa_supplicant. The specified file will be executed with the first argument set to interface name and second to "CONNECTED" or "DISCONNECTED" depending on the event. This can be used to execute networking tools required to configure the interface.

Additionally, three environmental variables are available to the file: WPA_CTRL_DIR, WPA_ID, and WPA_ID_STR. WPA_CTRL_DIR contains the absolute path to the ctrl_interface socket. WPA_ID contains the unique network_id identifier assigned to the active network, and WPA_ID_STR contains the content of the id_str option.

-P file Set the location of the PID file.

-G ping interval
Set the interval (in seconds) at which wpa_cli pings the supplicant.

command
Run a command. The available commands are listed in the next section.

COMMANDS

The following commands are available:

status get current WPA/EAPOL/EAP status

mib get MIB variables (dot1x, dot11)

help show this usage help

interface [ifname]
show interfaces/select interface

level <debug level>
change debug level

license show full wpa_cli license

logoff IEEE 802.1X EAPOL state machine logoff

logon IEEE 802.1X EAPOL state machine logon

set set variables (shows list of variables when run without arguments)

pmksa show PMKSA cache

reassociate
force reassociation

reconfigure
force wpa_supplicant to re-read its configuration file

preauthenticate <BSSID>
force preauthentication

identity <network id> <identity>
configure identity for an SSID

password <network id> <password>
configure password for an SSID

pin <network id> <pin>
configure pin for an SSID

otp <network id> <password>
configure one-time-password for an SSID

bssid <network id> <BSSID>
set preferred BSSID for an SSID

list_networks
list configured networks

terminate
terminate **wpa_supplicant**

quit exit wpa_cli

SEE ALSO

wpa_supplicant(8)

LEGAL