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

$ man wpa_priv.8

```
WPA_PRIV(8)                                        WPA_PRIV(8)

NAME

       wpa_priv - wpa_supplicant privilege separation helper

SYNOPSIS

       wpa_priv  [  -c ctrl path ]  [ -Bdd ]  [ -P pid file ]  [ driver:ifname

       [driver:ifname ...] ]

OVERVIEW

       wpa_priv is a privilege separation helper that minimizes  the  size  of

       wpa_supplicant code that needs to be run with root privileges.

       If  enabled,  privileged  operations  are  done in the wpa_priv process

       while leaving rest of the code (e.g., EAP authentication and WPA  hand?

       shakes) to operate in an unprivileged process (wpa_supplicant) that can

       be run as non-root user. Privilege separation restricts the effects  of

       potential  software errors by containing the majority of the code in an

       unprivileged process to avoid the possibility of a full system  compro?

       mise.

       wpa_priv  needs  to be run with network admin privileges (usually, root

       user). It opens a UNIX domain socket for each  interface  that  is  in?

       cluded  on the command line; any other interface will be off limits for

       wpa_supplicant in this kind of configuration. After  this,  wpa_suppli?

       cant  can be run as a non-root user (e.g., all standard users on a lap?

       top or as a special non-privileged user account created just  for  this

       purpose to limit access to user files even further).
```

EXAMPLE CONFIGURATION

The following steps are an example of how to configure wpa_priv to al?low users in the wpapriv group to communicate with wpa_supplicant with privilege separation:

Create user group (e.g., wpapriv) and assign users that should be able to use wpa_supplicant into that group.

Create /var/run/wpa_priv directory for UNIX domain sockets and control user access by setting it accessible only for the wpapriv group:

```
mkdir /var/run/wpa_priv
chown root:wpapriv /var/run/wpa_priv
chmod 0750 /var/run/wpa_priv
```

Start wpa_priv as root (e.g., from system startup scripts) with the en?abled interfaces configured on the command line:

```
wpa_priv -B -c /var/run/wpa_priv -P /var/run/wpa_priv.pid wext:wlan0
```

Run wpa_supplicant as non-root with a user that is in the wpapriv group:

```
wpa_supplicant -i ath0 -c wpa_supplicant.conf
```

COMMAND ARGUMENTS

-c ctrl path

Specify the path to wpa_priv control directory (Default: /var/run/wpa_priv/).

-B Run as a daemon in the background.

-P file

Set the location of the PID file.

driver:ifname [driver:ifname ...]

The <driver> string dictates which of the supported wpa_suppli?cant driver backends is to be used. To get a list of supported driver types see wpa_supplicant help (e.g, wpa_supplicant -h). The driver backend supported by most good drivers is wext.

The <ifname> string specifies which network interface is to be managed by wpa_supplicant (e.g., wlan0 or ath0).

wpa_priv does not use the network interface before wpa_suppli?cant is started, so it is fine to include network interfaces that are not available at the time wpa_priv is started. wpa_priv can control multiple interfaces with one process, but it is also possible to run multiple wpa_priv processes at the same time, if desired.

SEE ALSO

wpa_supplicant(8)

LEGAL