

Porting of drivers/net/wireless/ath/wcn36xx/smd.c

June 5, 2017

1 Introduction

The driver drivers/net/wireless/ath/wcn36xx/smd.c was introduced in 2013 in the commit 8e84c25. It is part of a mac80211 driver for Qualcomm WCN3660/WCN3680 hardware. This driver was accompanied by changes in MAINTAINERS, Kconfig, and a makefile, as well as a new Kconfig a new makefile and numerous new .c and .h files. Gcc produces 1 error.

2 IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE

Gcc reports that the constant IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE is not known. We try the following patch query (step1.cocci):

```
@bad depends on after@
symbol IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE;
@@

    IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE

@depends on !bad@
@@

- IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE
```

The first commit is ea1b2b45 at 10%. This commit log says that no driver ever sets this flag. The relevant expression in our code is

```
!(WCN36XX_FLAGS(wcn) &
    IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE)
```

Since the flag is never set, the result of the bit and should be 0. And then its negation would be 1. Elsewhere in the commit we see numerous tests that are specialized in this way, eg an if on a bit and is completely dropped, because the bit and is false. The change is as follows:

```
@@
expression e;
@@

- !(e & IEEE80211_HW_2GHZ_SHORT_PREAMBLE_INCAPABLE)
+ 1
```

The actual v3.6 code contains `true` instead of `1`. This probably makes more sense given the name of the field that stores the result of the above expression, which is `short_preamble_supported`. Nevertheless, the type of that field is `u8`, not `bool`. Still, the behavior is the same in either case, and `true` and `1` are interchanged in many places in the kernel. There are no introductions of `true` in the example commit, because all of the illustrated expressions are in if tests, not stored.

`== success (1/2)`