```
#! /bin/bash
#

set -e

progname=$(basename $0)
function help
{
    cat <<EOF
Usage: $progname COMMAND DEVICEs [ARGS]

Command for manipulating the pairing/authentication credentials of a
Wireless USB device that supports wired-mode Cable-Based-Association.

Works in conjunction with the wusb-cba.ko driver from http://linuxuwb.org.


DEVICE

 sysfs path to the device to authenticate; for example, both this
 guys are the same:

 /sys/devices/pci0000:00/0000:00:1d.7/usb1/1-4/1-4.4/1-4.4:1.1
 /sys/bus/usb/drivers/wusb-cbaf/1-4.4:1.1

COMMAND/ARGS are

 start

   Start a WUSB host controller (by setting up a CHID)

 set-chid DEVICE HOST-CHID HOST-BANDGROUP HOST-NAME

   Sets host information in the device; after this you can call the
   get-cdid to see how does this device report itself to us.

 get-cdid DEVICE

   Get the device ID associated to the HOST-CHDI we sent with
   'set-chid'. We might not know about it.

 set-cc DEVICE

   If we allow the device to connect, set a random new CDID and CK
   (connection key). Device saves them for the next time it wants to
   connect wireless. We save them for that next time also so we can
   authenticate the device (when we see the CDID he uses to id
   itself) and the CK to crypto talk to it.

CHID is always 16 hex bytes in 'XX YY ZZ...' form
BANDGROUP is almost always 0001

Examples:

  You can default most arguments to '' to get a sane value:
```

```
  $ $progname set-chid '' '' '' "My host name"

  A full sequence:

  $ $progname set-chid '' '' '' "My host name"
  $ $progname get-cdid ''
  $ $progname set-cc ''

EOF
}


# Defaults
# FIXME: CHID should come from a database :), band group from the host
host_CHID="00 11 22 33 44 55 66 77 88 99 aa bb cc dd ee ff"
host_band_group="0001"
host_name=$(hostname)

devs="$(echo /sys/bus/usb/drivers/wusb-cbaf/[0-9]*)"
hdevs="$(for h in /sys/class/uwb_rc/*/wusbhc; do readlink -f $h; done)"

result=0
case $1 in
    start)
        for dev in ${2:-$hdevs}
          do
          echo $host_CHID > $dev/wusb_chid
          echo I: started host $(basename $dev) >&2
        done
        ;;
    stop)
        for dev in ${2:-$hdevs}
          do
          echo 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 > $dev/wusb_chid
          echo I: stopped host $(basename $dev) >&2
        done
        ;;
    set-chid)
        shift
        for dev in ${2:-$devs}; do
            echo "${4:-$host_name}" > $dev/wusb_host_name
            echo "${3:-$host_band_group}" > $dev/wusb_host_band_groups
            echo ${2:-$host_CHID} > $dev/wusb_chid
        done
        ;;
    get-cdid)
        for dev in ${2:-$devs}
          do
          cat $dev/wusb_cdid
        done
        ;;
    set-cc)
        for dev in ${2:-$devs}; do
            shift
            CDID="$(head --bytes=16 /dev/urandom  | od -tx1 -An)"
            CK="$(head --bytes=16 /dev/urandom  | od -tx1 -An)"
```

```
            echo "$CDID" > $dev/wusb_cdid
            echo "$CK" > $dev/wusb_ck

            echo I: CC set >&2
            echo "CHID: $(cat $dev/wusb_chid)"
            echo "CDID:$CDID"
            echo "CK:  $CK"
        done
        ;;
    help|h|--help|-h)
        help
        ;;
    *)
        echo "E: Unknown usage" 1>&2
        help 1>&2
        result=1
esac
exit $result
```