My Cloud OS5 Firmware Notes

Cerberus Nov 24

This thread is a place for me to post random odds and ends about My Cloud OS5, and a good place to start is the structure of firmware bin file itself.

Firmware File Structure:

```
+--------------------------+
| CONTROL_HEADER           | 128 bytes
+--------------------------+
|                          |
| FILE_1_DATA              | uImage (kernel)
|                          |
+--------------------------+
|                          |
| FILE_2_DATA              | uRamdisk (initramfs)
|                          |
+--------------------------+
|                          |
| FILE_3_DATA              | image.cfs (root filesystem)
|                          |
+--------------------------+
|                          |
| FILE_4_DATA              | default.tar.gz (configuration)
|                          |
+--------------------------+
| FILE_EXTENDED_HEADER     | 96 bytes
+--------------------------+
|                          |
| FILE_EXTENDED_DATA       | uP.bin (ROM file)
|                          |
+--------------------------+
| FILE_EXTENDED_HEADER     | 96 bytes
+--------------------------+
|                          |
| FILE_EXTENDED_DATA       | grub.tgz (bootloader)
|                          |
+--------------------------+
:                          :
: ADDITIONAL_FILES         :
:                          :
+--------------------------+
```

Cerberus Nov 24

My Cloud OS5 Firmware (Extract and Install):

```
VERSION="5.27.157";
DEVICE="PR4100";

cp /shares/Public/My_Cloud_${DEVICE}_${VERSION}.bin /usr/local/upload;
upload_firmware -D -n My_Cloud_${DEVICE}_${VERSION}.bin;
```

My Cloud OS5 Firmware (Extract Only):

```
VERSION="5.27.157";
DEVICE="PR4100"
```

```
mkdir /shares/Public/My_Cloud_${DEVICE}_${VERSION};
cd /shares/Public/My_Cloud_${DEVICE}_${VERSION};

upload_firmware -D -f My_Cloud_${DEVICE}_${VERSION}.bin;
mv -f /usr/local/upload/uImage /shares/Public/My_Cloud_${DEVICE}_${VERSION};
mv -f /usr/local/upload/uRamdisk /shares/Public/My_Cloud_${DEVICE}_${VERSION};
mv -f /usr/local/upload/image.cfs /shares/Public/My_Cloud_${DEVICE}_${VERSION}
mv -f /tmp/uP.bin /shares/Public/My_Cloud_${DEVICE}_${VERSION};
mv -f /tmp/default.tar.gz /shares/Public/My_Cloud_${DEVICE}_${VERSION};
mv -f /tmp/grub /shares/Public/My_Cloud_${DEVICE}_${VERSION};

dd if="image.cfs" of="image.squashfs" bs=2048 skip=1;
unsquashfs -d image image.squashfs;
```

Cerberus Nov 24

My Cloud OS5 Apps (Extract and Install):

```
DEVICE="PR4100";

cd /shares/Public;
cp AppName_1.0.0_My_Cloud_${DEVICE}.bin /usr/local/apps_upload;
upload_apkg -d -m -t 2 -p AppName_1.0.0_My_Cloud_${DEVICE}.bin;
```

My Cloud OS5 Apps (Extract and Reinstall):

```
DEVICE="PR4100";

cd /shares/Public;
cp AppName_1.0.0_My_Cloud_${DEVICE}.bin /usr/local/apps_upload;
upload_apkg -d -f 1 -g 1 -r AppName_1.0.0_My_Cloud_${DEVICE}.bin;
```

Cerberus Nov 24

My Cloud OS5 Samba Share (Create):

```
SHARE_PATH="/mnt/HD/HD_a2/Share_Name";
SHARE_BASE="$(basename "${SHARE_PATH}")";

smbif -a "${SHARE_PATH}";
smbif -t "${SHARE_BASE}"; # Public Off
smbif -p "${SHARE_BASE}"; # Public On
```

My Cloud OS5 Samba Share (Delete):

```
SHARE_PATH="/mnt/HD/HD_a2/Share_Name";
SHARE_BASE="$(basename "${SHARE_PATH}")";
smbif -b "${SHARE_BASE}";
```

My Cloud OS5 Samba Share (Rename):

```
SHARE_PATH_OLD="/mnt/HD/HD_a2/Share_Old";
SHARE_BASE_OLD="$(basename "${SHARE_PATH_OLD}")";
SHARE_PATH_NEW="/mnt/HD/HD_a2/Share_New";
SHARE_BASE_NEW="$(basename "${SHARE_PATH_NEW}")";

if [ ! -d "${SHARE_PATH_NEW}" ]; then
    smbif -o "${SHARE_BASE_OLD}" -n "${SHARE_BASE_NEW}" -x;
fi
```

Skip to main content

My Cloud OS5 Samba Share (Permissions):

```
SHARE_PATH="/mnt/HD/HD_a2/Share_Name";
SHARE_BASE="$(basename "${SHARE_PATH}")";
SHARE_USER="user_name";
SHARE_PERM=2; # 1:Read Only  2:Read Write  3:Deny

smbif -m "${SHARE_PERM}" -s "${SHARE_BASE}" -u "${SHARE_USER}";
```

Cerberus Nov 24

My Cloud EX2 Ultra MTD Partitions (Backup):

```
nanddump /dev/mtd0 -f mtd0_U-Boot.img
nanddump /dev/mtd1 -f mtd1_uImage.img
nanddump /dev/mtd2 -f mtd2_uRamdisk.img
nanddump /dev/mtd3 -f mtd3_image.cfs.img
nanddump /dev/mtd4 -f mtd4_rescue_fw.img
nanddump /dev/mtd5 -f mtd5_config.img
nanddump /dev/mtd6 -f mtd6_reserve_1.img
nanddump /dev/mtd7 -f mtd7_reserve_2.img
```

My Cloud EX2 Ultra MTD Partitions (Restore):

```
nandwrite /dev/mtd1 mtd1_uImage.img
nandwrite /dev/mtd2 mtd2_uRamdisk.img
nandwrite /dev/mtd3 mtd3_image.cfs.img
nandwrite /dev/mtd4 mtd4_rescue_fw.img
nandwrite /dev/mtd5 mtd5_config.img
nandwrite /dev/mtd6 mtd6_reserve_1.img
nandwrite /dev/mtd7 mtd7_reserve_2.img
```

Erase Kernel:

```
# Ignore erase counters
flash_eraseall /dev/mtd1

# Preserve erase counters
ubiformat /dev/mtd1
```

My Cloud EX2 Ultra Config Files (Erase):

```
umount -l /usr/local/config
ubidetach /dev/ubi_ctrl -m 5
flash_eraseall /dev/mtd5
```

My Cloud EX2 Ultra Config UBI Volume (Create):

```
ubiformat /dev/mtd5
ubiattach /dev/ubi_ctrl -m 5 -O 2048
ubimkvol /dev/ubi0 -N config -s 20MiB
busybox mount -t ubifs ubi0:config /usr/local/config
```

My Cloud EX2 Ultra MTD Partitions:

```
 dev: size     erasesize  name      hsize
mtd0: 00500000 00020000  "U-Boot"       5 MiB
mtd1: 00500000 00020000  "uImage"       5 MiB
mtd2: 00500000 00020000  "uRamdisk"     5 MiB
mtd3: 0b900000 00020000  "image.cfs"  185 MiB
                   20000  "rescue fw"   15 MiB
mtd5: 01400000 00020000  "config"      20 MiB
```

Skip to main content

```
mtd6: 00a00000 00020000   "reserve1"    10 MiB
mtd7: 00a00000 00020000   "reserve2"    10 MiB
```

My Cloud EX2 Ultra MTD Partitions with UBI Volumes:

```
/dev/mtd5 <------------------ MTD Partition (config)
    /dev/ubi0 <-------------- UBI Device
        /dev/ubi0_0 <-------- UBI Volume

/dev/mtd6 <------------------ MTD Partition (reserve1)
    /dev/ubi1 <-------------- UBI Device
        /dev/ubi1_0 <-------- UBI Volume

/dev/mtd7 <------------------ MTD Partition (reserve2)
    /dev/ubi2 <-------------- UBI Device
        /dev/ubi2_0 <-------- UBI Volume
```

My Cloud EX2 Ultra UBI Volumes:

```
UBI: attached mtd5 (name "config", size 20 MiB) to ubi0
UBIFS: mounted UBI device 0, volume 0, name "config"
UBI: attached mtd6 (name "reserve1", size 10 MiB) to ubi1
UBIFS: mounted UBI device 1, volume 0, name "reserve1"
UBI: attached mtd7 (name "reserve2", size 10 MiB) to ubi2
UBIFS: mounted UBI device 2, volume 0, name "reserve2"

ubiattach creates a UBI device (ubi0, ubi1, ubi2, etc) from an MTD partition.
ubimkvol creates a volume (ubi0_0, ubi0_1, ubi0_2, etc) on a UBI device.
```

My Cloud EX2 Ultra Project Features:

```
PROJECT_FEATURE_MFG_CFG_IN_UBI=1
PROJECT_FEATURE_CONFIG_FILESYSTEM="UBIFS"
PROJECT_FEATURE_ROOTFS_ON_NAND=1
```

Cerberus Nov 24

My Cloud PR4100 Power Supply 3-Pin AC Power Cable:

> - Length: 900mm
> - NEMA 5-15P 3 pin plug
> - IEC-60320-1 C5 3-pin plug
> - Type A to C5
> - 18AWGX3C
> - Rated: 7A 125V

My Cloud PR4100 90 Watt Power Supply:

> - 19 volts
> - 4.74 amps
> - 90 watts
> - IEC 60320-1 C6 3-pin Socket
> - Inner Diameter: 2.5mm
> - Outer Diameter: 5.5mm
> - Barrel Length: 11mm

Cerberus Nov 24

Skip to main content

The following is a list of My Cloud PR4100 and PR2100 Power Management Controller (PMC) chip commands.

WARNING: Some of these commands are potentially dangerous, if you screw something up by using them, it’s not my problem.

LCD Backlight Brightness (BKL):

```
0x00 = 000% Brightness (Min)
0x19 = 025% Brightness
0x32 = 050% Brightness
0x3C = 060% Brightness (Default)
0x64 = 100% Brightness (Max)

Type: Get and Set
Note: Level 0-100

# echo "BKL=00" > /dev/ttyS2
```

LCD Display Text (LN1, LN2):

```
Type: Set only
Note: 16 Characters Max Per Line

# echo "LN1=XXXXXXXXXXXXXXXX" > /dev/ttyS2
# echo "LN2=XXXXXXXXXXXXXXXX" > /dev/ttyS2
```

Fan Speed Level 0-100% (FAN):

```
0x00 : 0x0000 -> ((00 / 2) * 60) = 0000 RPM
0x05 : 0x0009 -> ((09 / 2) * 60) = 0270 RPM
0x10 : 0x000E -> ((14 / 2) * 60) = 0420 RPM
0x15 : 0x0013 -> ((19 / 2) * 60) = 0570 RPM
0x20 : 0x001C -> ((28 / 2) * 60) = 0840 RPM
0x25 : 0x0021 -> ((33 / 2) * 60) = 0990 RPM
0x30 : 0x0029 -> ((41 / 2) * 60) = 1230 RPM
0x35 : 0x002D -> ((45 / 2) * 60) = 1350 RPM
0x40 : 0x0035 -> ((53 / 2) * 60) = 1590 RPM
0x45 : 0x0039 -> ((57 / 2) * 60) = 1710 RPM
0x50 : 0x0040 -> ((64 / 2) * 60) = 1920 RPM
0x55 : 0x0043 -> ((67 / 2) * 60) = 2010 RPM
0x60 : 0x0049 -> ((73 / 2) * 60) = 2190 RPM
0x64 : 0x004A -> ((74 / 2) * 60) = 2220 RPM

Type: Get and Set
Note: Level 0-64

# echo "FAN=25" > /dev/ttyS2
```

Fan TAC Frequency (TAC):

```
0x00 : 0x0000 -> ((00 / 2) * 60) = 0000 RPM
0x05 : 0x0009 -> ((09 / 2) * 60) = 0270 RPM
0x10 : 0x000E -> ((14 / 2) * 60) = 0420 RPM
0x15 : 0x0013 -> ((19 / 2) * 60) = 0570 RPM
0x20 : 0x001C -> ((28 / 2) * 60) = 0840 RPM
0x25 : 0x0021 -> ((33 / 2) * 60) = 0990 RPM
0x30 : 0x0029 -> ((41 / 2) * 60) = 1230 RPM
0x35 : 0x002D -> ((45 / 2) * 60) = 1350 RPM
```

Skip to main content

```
0x40 : 0x0035 -> ((53 / 2) * 60) = 1590 RPM
0x45 : 0x0039 -> ((57 / 2) * 60) = 1710 RPM
0x50 : 0x0040 -> ((64 / 2) * 60) = 1920 RPM
0x55 : 0x0043 -> ((67 / 2) * 60) = 2010 RPM
0x60 : 0x0049 -> ((73 / 2) * 60) = 2190 RPM
0x64 : 0x004A -> ((74 / 2) * 60) = 2220 RPM

Type: Get only
Note: RPM = ((TAC / 2) * 60)

# echo "TAC" > /dev/ttyS2
```

Fan TAC Frequency as RPM (RPM):

```
0x0000 = 0000 RPM
0x01FE = 0510 RPM
0x08AC = 2220 RPM

Type: Get only

# echo "RPM" > /dev/ttyS2
```

Interrupt Mask Register (IMR):

```
0x01 -> Fan Failure
0x02 -> Power Supply 1
0x04 -> Power Supply 2
0x08 -> USB Copy Button State
0x10 -> Drive Presence Detection
0x20 -> LCD Up Button Pressed
0x40 -> LCD Down Button Pressed
0x80 -> Buzzer State

Type: Get and Set
Note: Must set IMR=FF before ISR use
Response: IMR=FF

# echo "IMR=FF" > /dev/ttyS2
```

Interrupt Status Register (ISR):

```
0x01 -> Fan Failure
0x02 -> Power Supply 1
0x04 -> Power Supply 2
0x08 -> USB Copy Button State
0x10 -> Drive Presence Detection
0x20 -> LCD Up Button Pressed
0x40 -> LCD Down Button Pressed
0x80 -> Buzzer State

Type: Get only
Note: After boot ISR=10
Note: Must set IMR=FF before ISR use
Note: ISR resets to 0x00 after value retrieved
Note: ALERT sent when a button is pressed

# echo "ISR" > /dev/ttyS2
```

Skip to main content

LED Configuration (LED, BLK):

```
0x00 = 0000 0000 -> Power All Off, USB All Off
0x01 = 0000 0001 -> Power Blue On
0x02 = 0000 0010 -> Power Red On
0x04 = 0000 0100 -> Power Green On
0x08 = 0000 1000 -> USB Red On
0x09 = 0000 1001 -> USB Red On, Power Blue On
0x0A = 0000 1010 -> USB Red On, Power Red On
0x0C = 0000 1100 -> USB Red On, Power Green On
0x10 = 0001 0000 -> USB Blue On
0x11 = 0001 0001 -> USB Blue On, Power Blue On
0x12 = 0001 0010 -> USB Blue On, Power Red On
0x14 = 0001 0100 -> USB Blue On, Power Green On
0x1F = 0001 1111 -> Power All On, USB All On

Type: Get and Set
Note: BLK is Blink
Note: Set LED=00 and PLS=00 before setting BLK
Note: Set BLK=00 and PLS=00 before setting LED

echo "BLK=00" > /dev/ttyS2
echo "LED=00" > /dev/ttyS2
```

LED Configuration Bits (LED, BLK):

```
0 0 0 0  0 0 0 1 --> 0x01 (Hex)
- - - -  - - - -
| | | |  | | | |
| | | |  | | | +---> Power Blue
| | | |  | | +-----> Power Red
| | | |  | +-------> Power Green
| | | |  +---------> USB Red
| | | +------------> USB Blue
| | +--------------> Unused
| +----------------> Unused
+------------------> Unused
```

LED Pulse (PLS):

```
0x00 -> LED Pulse Off
0x01 -> LED Pulse On

Type: Get and Set
Note: Power LED Only
Note: Set LED=00 and BLK=00 before setting PLS

# echo "PLS=00" > /dev/ttyS2
```

Drive Status (DE0):

```
0x0F = 0000 1111 -> hd1,hd2,hd3,hd4
0x1F = 0001 1111 -> hd2,hd3,hd4
0x2F = 0010 1111 -> hd1,hd3,hd4
0x3F = 0011 1111 -> hd3,hd4
0x4F = 0100 1111 -> hd1,hd2,hd4
0x5F = 0101 1111 -> hd2,hd4
0x6F = 0110 1111 -> hd1,hd4
0x7F = 0111 1111 -> hd4
                 -> hd1,hd2,hd3
0x9F = 1001 1111 -> hd2,hd3
```

Skip to main content

```
0xAF = 1010 1111 -> hd1,hd3
0xBF = 1011 1111 -> hd3
0xCF = 1100 1111 -> hd1,hd2
0xDF = 1101 1111 -> hd2
0xEF = 1110 1111 -> hd1
0xFF = 1111 1111 -> None

Type: Get and Set
Note: Red LED On/Off
Note: Zeros in right half of byte powers down drive(s)
Note: DLC/DLS are inverse of DE0

# echo "DE0=FF" > /dev/ttyS2
```

Drive Enable Register (DE0):

```
0x00 - 0000 0000 -> hd1,hd2,hd3,hd4
0x01 - 0000 0001 -> hd1,hd2,hd3
0x02 - 0000 0010 -> hd1,hd3,hd4
0x03 - 0000 0011 -> hd3,hd4
0x04 - 0000 0100 -> hd1,hd2,hd4
0x05 - 0000 0101 -> hd2,hd4
0x06 - 0000 0110 -> hd1,hd4
0x07 - 0000 0111 -> hd4
0x08 - 0000 1000 -> hd2,hd3,hd4
0x09 - 0000 1001 -> hd2,hd3
0x0A - 0000 1010 -> hd1,hd3
0x0B - 0000 1011 -> hd3
0x0C - 0000 1100 -> hd1,hd2
0x0D - 0000 1101 -> hd2
0x0E - 0000 1110 -> hd1
0x0F - 0000 1111 -> None

Type: Get and Set
Note: Blue LED On/Off
Note: Zeros in right half of byte powers down drive(s)
Note: DLC/DLS are inverse of DE0

# echo "DE0=0F" > /dev/ttyS2
```

Drive LED Status (DLC, DLS):

```
0x10 = 0001 0000 -> hd1
0x20 = 0010 0000 -> hd2
0x30 = 0011 0000 -> hd1,hd2
0x40 = 0100 0000 -> hd3
0x50 = 0101 0000 -> hd1,hd3
0x60 = 0110 0000 -> hd2,hd3
0x70 = 0111 0000 -> hd2,hd3,hd4
0x80 = 1000 0000 -> hd4
0x90 = 1001 0000 -> hd1,hd4
0xA0 = 1010 0000 -> hd2,hd4
0xB0 = 1011 0000 -> hd1,hd2,hd4
0xC0 = 1100 0000 -> hd3,hd4
0xD0 = 1101 0000 -> hd1,hd3,hd4
0xE0 = 1110 0000 -> hd1,hd2,hd3
0xF0 = 1111 0000 -> hd1,hd2,hd3,hd4

Type: Set only
e LED Set Register
Note: DLC = Drive LED Clear Register
```

Skip to main content

```
Note: Red LED On/Off
Note: DLC/DLS are inverse of DE0
Note: Use DLC for on, or DLS for off
Note: Ones in right half of byte powers down drive(s)

# echo "DLC=F0" > /dev/ttyS2
# echo "DLS=F0" > /dev/ttyS2
```

Drive Power (DLC, DLS):

```
0x01 - 0000 0001 -> hd1
0x02 - 0000 0010 -> hd2
0x03 - 0000 0011 -> hd1,hd2
0x04 - 0000 0100 -> hd3
0x05 - 0000 0101 -> hd1,hd3
0x06 - 0000 0110 -> hd2,hd3
0x07 - 0000 0111 -> hd2,hd3,hd4
0x08 - 0000 1000 -> hd4
0x09 - 0000 1001 -> hd1,hd4
0x0A - 0000 1010 -> hd2,hd4
0x0B - 0000 1011 -> hd1,hd2,hd4
0x0C - 0000 1100 -> hd3,hd4
0x0D - 0000 1101 -> hd1,hd3,hd4
0x0E - 0000 1110 -> hd1,hd2,hd3
0x0F - 0000 1111 -> hd1,hd2,hd3,hd4

Type: Set only
Note: Blue LED On/Off
Note: Ones in right half of byte powers down drive(s)
Note: DLC/DLS are inverse of DE0
Note: Use DLC for on, or DLS for off

# echo "DLC=0F" > /dev/ttyS2
# echo "DLS=0F" > /dev/ttyS2
```

Drive Status and Power Bits (DE0, DLC, DLS):

```
0 0 0 1  0 0 0 0 --> 0x10 (Hex)
- - - -  - - - -
| | | |  | | | |
| | | |  | | | +---> hd1 Power
| | | |  | | +-----> hd2 Power
| | | |  | +-------> hd3 Power
| | | |  +---------> hd4 Power
| | | +------------> hd1 Status LED (Red)
| | +--------------> hd2 Status LED (Red)
| +----------------> hd3 Status LED (Red)
+------------------> hd4 Status LED (Red)
```

Drive LED Blink (DLB):

```
0x00 = 0000 0000 -> None
0x10 = 0001 0000 -> hd1
0x20 = 0010 0000 -> hd2
0x30 = 0011 0000 -> hd1,hd2
0x40 = 0100 0000 -> hd3
0x50 = 0101 0000 -> hd1,hd3
0x60 = 0110 0000 -> hd2,hd3
Skip to main content -> hd1,hd2,hd3
0x80 = 1000 0000 -> hd4
```

```
0x90 = 1001 0000 -> hd1,hd4
0xA0 = 1010 0000 -> hd2,hd4
0xB0 = 1011 0000 -> hd1,hd2,hd4
0xC0 = 1100 0000 -> hd3,hd4
0xD0 = 1101 0000 -> hd1,hd3,hd4
0xE0 = 1110 0000 -> hd2,hd3,hd4
0xF0 = 1111 0000 -> hd1,hd2,hd3,hd4

Type: Get and Set
Note: Red LED On/Off

echo "DLB=00" > /dev/ttyS2
```

Drive LED Blink Bits (DLB):

```
0 0 0 1  0 0 0 0 --> 0x10 (Hex)
- - - -  - - - -
| | | |  | | | |
| | | |  | | | +---> Always 0
| | | |  | | +-----> Always 0
| | | |  | +-------> Always 0
| | | |  +---------> Always 0
| | | +------------> Status, hd1 (Red LED Blink)
| | +--------------> Status, hd2 (Red LED Blink)
| +----------------> Status, hd3 (Red LED Blink)
+------------------> Status, hd4 (Red LED Blink)
```

Drive Presence Detection Register (DP0):

```
0x90 - 1001 0000 -> hd1,hd2,hd3,hd4 Present
0x91 - 1001 0001 -> hd2,hd3,hd4 Present
0x92 - 1001 0010 -> hd1,hd3,hd4 Present
0x93 - 1001 0011 -> hd3,hd4 Present
0x94 - 1001 0100 -> hd1,hd2,hd4 Present
0x95 - 1001 0101 -> hd2,hd4 Present
0x96 - 1001 0110 -> hd1,hd4 Present
0x97 - 1001 0111 -> hd4 Present
0x98 - 1001 1000 -> hd1,hd2,hd3 Present
0x99 - 1001 1001 -> hd2,hd3 Present
0x9A - 1001 1010 -> hd1,hd3 Present
0x9B - 1001 1011 -> hd3 Present
0x9C - 1001 1100 -> hd1,hd2 Present
0x9D - 1001 1101 -> hd2 Present
0x9E - 1001 1110 -> hd1 Present
0x9F - 1001 1111 -> No Drives Present

Type: Get only
Note: ALERT sent when a drive is inserted or removed

# echo "DP0" > /dev/ttyS2
```

Drive Presence Detection Register Bits (DP0):

```
1 0 0 1  0 0 0 0 --> 0x90 (Hex)
- - - -  - - - -
| | | |  | | | |
| | | |  | | | +---> hd1
| | | |  | | +-----> hd2
Skip to main content ---> hd3
| | | |  +---------> hd4
| | | +------------> Bit always 1
| | +--------------> Bit always 0
| +----------------> Bit always 0
+------------------> Bit always 1

Note: 0=Present, 1=Absent
```

PMC Temperature (TMP, tmp):

```
0x1C -> (28C x 1.8) + 32 = 082.4F
0x1D -> (29C x 1.8) + 32 = 084.2F
0x1E -> (30C x 1.8) + 32 = 086.0F
0x1F -> (31C x 1.8) + 32 = 087.8F
0x20 -> (32C x 1.8) + 32 = 089.6F
0x21 -> (33C x 1.8) + 32 = 091.4F
0x22 -> (34C x 1.8) + 32 = 093.2F
0x23 -> (35C x 1.8) + 32 = 095.0F
0x24 -> (36C x 1.8) + 32 = 096.8F
0x25 -> (37C x 1.8) + 32 = 098.6F
0x26 -> (38C x 1.8) + 32 = 100.4F
0x27 -> (39C x 1.8) + 32 = 102.2F
0x28 -> (40C x 1.8) + 32 = 104.0F
0x29 -> (41C x 1.8) + 32 = 106.0F
0x30 -> (48C x 1.8) + 32 = 118.4F

Type: Get only
Note: TMP is hex (upper case)
Note: tmp is decimal (lower case)

# echo "TMP" > /dev/ttyS2
```

Buzzer (BZR):

```
Type: Set only
Note: On BZR=FF / Off BZR=00

# echo "BZR=FF" > /dev/ttyS2
```

Status (STA):

```
0x6A = 0110 1010 = 106 -> Power Cable in Jack #2
0x6C = 0110 1100 = 108 -> Power Cable in Jack #1
0x6E = 0110 1110 = 110 -> Power Cable in Jack #1 and #2
0x62 = 0110 0010 = 098 -> USB Button Pushed and Power Cable in Jack #2
0x64 = 0110 0100 = 100 -> USB Button Pushed and Power Cable in Jack #1
0x66 = 0110 0110 = 102 -> USB Button Pushed and Power Cable in Jack #1 and
0x4A = 0100 1010 = 074 -> LCD Up Button Pushed and Power Cable in Jack #2
0x4C = 0100 1100 = 076 -> LCD Up Button Pushed and Power Cable in Jack #1
0x4E = 0100 1110 = 078 -> LCD Up Button Pushed and Power Cable in Jack #1
0x2A = 0010 1010 = 042 -> LCD Down Button Pushed and Power Cable in Jack #
0x2C = 0010 1100 = 044 -> LCD Down Button Pushed and Power Cable in Jack #
0x2E = 0010 1110 = 046 -> LCD Down Button Pushed and Power Cable in Jack #

Type: Get only
Response: STA=6c

# echo "STA" > /dev/ttyS2
```

Skip to main content

Status Bits (STA):

```
0 1 1 0  1 1 1 0 --> 0x00 (Hex)
- - - -  - - - -
| | | |  | | | |
| | | |  | | | +---> Unused
| | | |  | | +-----> Power Cable #2
| | | |  | +-------> Power Cable #1
| | | |  +---------> LCD Button
| | | +------------> Unused
| | +--------------> LCD Down Button [ OR ] USB Button
| +----------------> LCD Up Button [ OR ] USB Button
+------------------> Unused

Note: 0=Bit Off, 1=Bit On
Note: LCD and USB share two bits
Note: LCD has two bits, one up, one down
Note: USB has two bits, both on or both off
```

Timeout Event Counter (TEC):

```
Type: Get only
Response: TEC=00
Note: Records reboot timeout events.

# echo "TEC" > /dev/ttyS2
```

Unknown (ECH):

```
Type: Get and Set
Response: ALERT

# echo "ECH=00" > /dev/ttyS2
```

Version (VER):

```
Type: Get only
Response: VER=WD BBC v02
Note: Two versions are known to exist.
Note: My Cloud OS3 includes VER=WD BBC v01
Note: My Cloud OS5 includes VER=WD BBC v02
Note: My Cloud OS5 uses pmc_update to update the PMC chip.

# echo "VER" > /dev/ttyS2
```

Configuration (CFG):

```
Type: Get and Set
Note: CFG=1 ->  Reset PMC
Note: CFG=2 ->  Disable automatic drive presence detection DP0
Note: CFG=3 ->  Enable automatic drive presence detection DP0
Note: Default value is 0x03

# echo "CFG=02" > /dev/ttyS2
```

Flash Write Test (FWT):

Skip to main content

> **WARNING:** This command is potentially dangerous and should not be used.
>
> ```
> Type: Set only
>
> # echo "FWT=00" > /dev/ttyS2
> ```

Update (UPD):

> This command puts the PMC chip into a special update or reset mode, with an interactive menu. Of all the commands one may issue, this appears to be the most dangerous. Yet, it may afford the possibility of dumping the PMC firmware for analysis.
>
> **WARNING:** Using this command could potentially corrupt the PMC chip firmware, rendering the fan control and other hardware control functions inoperative.
>
> ```
> # echo "UPD" > /dev/ttyS2
>
> ========= WDPMC Update Menu v1.0 =============
> Reset PMC -------------------------------- 0
> Write Image To PMC Internal Flash -------- 1
> Read Image From PMC Internal Flash ------- 2
> Execute The New Program ------------------ 3
> ==============================================
>
> Note: To select a number, echo it to the PMC via /dev/ttyS2
>
> # echo 1 > /dev/ttyS2
> # echo 2 > /dev/ttyS2
> # echo 3 > /dev/ttyS2
> ```

Cerberus Nov 26

My Cloud PR4100:

BNFA - Black Canyon
Intel Pentium N3710 Quad Core 1.6 GHz 4 GB RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 42 6C 61 63 6b 43 79 00 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 42 6C 61 63 6b 43 79 00 55 AA
MAGIC_NUMBER_uP ------> 55 AA 42 6C 61 63 6b 43 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 42 6C 61 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 42 6C 61 63 6b 43 79 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 7
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud PR2100

BBCL - Bryce Canyon
Intel Pentium N3710 Quad Core 1.6 GHz, 4 GB RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 42 72 79 63 65 43 79 00 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 42 72 79 63 65 43 79 00 55 AA
MAGIC_NUMBER_uP ------> 55 AA 42 72 79 63 65 43 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 42 72 79 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 42 72 79 63 65 43 79 5A

PRODUCT_ID -----------> 0
```

Skip to main content

```
CUSTOM_ID ------------> 20
MODEL_ID -------------> 8
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud DL4100:

BNEZ - Sprite
Intel Atom C2338 Dual Core 1.7 GHz, 2 GB RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 53 70 72 69 74 65 00 00 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 53 70 72 69 74 65 00 00 55 AA
MAGIC_NUMBER_uP ------> 55 AA 53 70 72 69 74 65 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 53 70 72 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 53 70 72 69 74 65 00 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 5
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud DL2100:

BBAZ - Aurora
Intel Atom C2350 Dual Core 1.7 GHz, 1 GB RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 41 75 72 6F 72 61 00 00 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 41 75 72 6F 72 61 00 00 55 AA
MAGIC_NUMBER_uP ------> 55 AA 41 75 72 6F 72 61 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 41 75 72 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 41 75 72 6F 72 61 00 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 6
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud EX4100:

BWZE - Yellowstone
Marvell Armada A388 Dual Core 1.6 GHz, 2 GB RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 59 65 6C 6C 6F 77 00 00 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 59 65 6C 6C 6F 77 00 00 55 AA
MAGIC_NUMBER_uP ------> 55 AA 59 65 6C 6C 6F 77 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 59 65 6C 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 59 65 6C 6C 6F 77 00 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 3
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud EX2100:

BWAZ - Yosemite
Marvell Armada A385 Dual Core 1.3 GHz, 1 GB RAM

Skip to main content

```
MAGIC_NUMBER_OS5 -----> 55 BB 59 6F 73 65 6D 69 74 65 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 59 6F 73 65 6D 69 74 65 55 AA
MAGIC_NUMBER_uP ------> 55 AA 59 6F 73 65 6D 69 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 59 6F 73 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 59 6F 73 65 6D 69 74 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 4
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud EX4:

LT4A - Lightning
Marvell Armada A300 Single Core 2.0 GHz, 512 MB RAM

```
MAGIC_NUMBER_OS3 -----> 55 AA 4C 69 67 68 74 6e 69 00 55 AA
MAGIC_NUMBER_uP ------> 55 AA 4C 69 67 75 50 00 00 00 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 4C 69 67 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 4C 69 67 68 74 6e 69 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 0
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud EX2 Ultra:

BVBZ - Ranger Peak (see Grand Teton)
Marvell Armada A385 Dual Core 1.3 GHz, 1 GB RAM
Same hardware as My Cloud Mirror Gen 2, but with more RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 47 72 61 6E 64 54 65 74 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 47 72 61 6E 64 54 65 74 55 AA
MAGIC_NUMBER_uP ------> 55 AA 47 72 61 6E 64 54 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 47 72 61 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 47 72 61 6E 64 54 65 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID -------------> 1
HARDWARE_ID ----------> 1
SUB_ID ---------------> 1
```

My Cloud Mirror Gen 2:

BWVZ - Grand Teton
Marvell Armada A385 Dual Core 1.3 GHz, 512 MB RAM
Same hardware as My Cloud EX2 Ultra, but with less RAM

```
MAGIC_NUMBER_OS5 -----> 55 BB 47 72 61 6E 64 54 65 74 55 AA
MAGIC_NUMBER_OS3 -----> 55 AA 47 72 61 6E 64 54 65 74 55 AA
MAGIC_NUMBER_uP ------> 55 AA 47 72 61 6E 64 54 75 50 55 AA
MAGIC_NUMBER_RESCUE --> 55 AA 47 72 61 52 65 73 63 75 72 65
MAGIC_NUMBER_APKG ----> 47 72 61 6E 64 54 65 5A

PRODUCT_ID -----------> 0
CUSTOM_ID ------------> 20
MODEL_ID         ----> 1
```

```
 HARDWARE_ID ----------> 1
 SUB_ID ---------------> 1
```

My Cloud EX2:

KC2A - Kings Canyon
Marvell Armada A370 Single Core 1.2 GHz, 512 MB RAM
Same hardware as My Cloud Mirror Gen 1

```
 MAGIC_NUMBER_OS3 -----> 55 AA 4B 69 6E 67 73 43 61 00 55 AA
 MAGIC_NUMBER_uP ------> 55 AA 4B 69 6E 67 73 75 50 00 55 AA
 MAGIC_NUMBER_RESCUE --> 55 AA 4B 69 6E 52 65 73 63 75 72 65
 MAGIC_NUMBER_APKG ----> 4B 69 6E 67 73 43 61 5A

 PRODUCT_ID -----------> 0
 CUSTOM_ID ------------> 20
 MODEL_ID -------------> 1
 HARDWARE_ID ----------> 1
 SUB_ID ---------------> 1
```

My Cloud Mirror Gen 1:

BZVM - Zion (see Kings Canyon)
Marvell Armada A370 Single Core 1.2 GHz, 512 MB RAM
Same hardware as My Cloud EX2

```
 MAGIC_NUMBER_OS3 -----> 55 AA 4B 69 6E 67 73 43 61 00 55 AA
 MAGIC_NUMBER_uP ------> 55 AA 4B 69 6E 67 73 75 50 00 55 AA
 MAGIC_NUMBER_RESCUE --> 55 AA 4B 69 6E 52 65 73 63 75 72 65
 MAGIC_NUMBER_APKG ----> 4B 69 6E 67 73 43 61 5A

 PRODUCT_ID -----------> 0
 CUSTOM_ID ------------> 20
 MODEL_ID -------------> 1
 HARDWARE_ID ----------> 1
 SUB_ID ---------------> 1
```

My Cloud Gen 2:

GLCR - Glacier
Marvell Armada A375 Dual Core 1.0 GHz, 512 MB RAM
Part Number: WDBCTLxxxxHWT-10. Firmware 02.xx.xx

```
 MAGIC_NUMBER_OS5 -----> 55 BB 47 6C 61 63 69 65 72 00 55 AA
 MAGIC_NUMBER_OS3 -----> 55 AA 47 6C 61 63 69 65 72 00 55 AA
 MAGIC_NUMBER_uP ------> 55 AA 47 6C 61 63 69 75 50 00 55 AA
 MAGIC_NUMBER_RESCUE --> 55 AA 47 6C 61 52 65 73 63 75 72 65
 MAGIC_NUMBER_APKG ----> 47 6C 61 63 69 65 72 5A

 PRODUCT_ID -----------> 0
 CUSTOM_ID ------------> 20
 MODEL_ID -------------> 2
 HARDWARE_ID ----------> 1
 SUB_ID ---------------> 1
```

My Cloud Gen 1:

SQ - Sequoia

Mindspeed Comcerto 2000 M86261G-12 Dual Core 650 MHz, 256 MB RAM
Part Number: WDBCTLxxxxHWT-00. Firmware 03.xx.xx and 04.xx.xx
F [illegible] K pages and firmware 04.xx.xx has 64K pages

Skip to main content

WD Cloud:

BAGX - Mirrorman (See Glacier, Japan Only)
Part Number: WDBAGXxxxxHWT-JESN

WDMyCloudEX1100:

BG2Y - BlackIce (See Glacier)

Medion LifeCloud:

BLHW - Alpina (See Glacier)

Cerberus Nov 26

RAID Mode Translations:

Value	Text
standard	JBOD
linear	Spanning
raid0	RAID 0
raid1	RAID 1
raid5	RAID 5
spare	Spare
raid10	RAID 10

RAID Status Translations:

Value	Text
clean	Healthy
clean	Good
degraded	Degraded
damaged	Damaged
resize	Resizing
resize_wait	Wait to resize
migrate	Migrating
migrate	Wait to migrate
resize	Manual Rebuild
expansion	Expanding
reshape	Expanding
recovery	Rebuilding
resync	Rebuilding
resync1	Rebuilding
resync5	Verifying RAID parity

RAID XML Data Source:

`/var/www/xml/used_volume_info.xml`

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
  <volume_info>
    <item>
      <volume>1</volume>
      <web_cmd>1</web_cmd>
      <sdx3_flag>0</sdx3_flag>
```

Skip to main content

```
      <raid_mode>standard</raid_mode>
      <file_type>ext4</file_type>
      <raid_status>0</raid_status>
      <size>1917674508</size>
      <used_size>245890816</used_size>
      <free_size>1671783692</free_size>
      <percentage>12</percentage>
      <device>sda</device>
      <spare/>
      <mount>/dev/sda2</mount>
      <to_be_sync/>
      <used_device>/dev/sda2 </used_device>
      <dev_num>1</dev_num>
      <volume_encrypt>0</volume_encrypt>
      <mount_status>1</mount_status>
      <scsi_mapping>0</scsi_mapping>
      <raid_uuid>WD-WCCXXXXXXXXX</raid_uuid>
      <is_roaming_volume>0</is_roaming_volume>
      <scsi0_serial>sda:WD-WCCXXXXXXXXX</scsi0_serial>
    </item>
  </volume_info>
</config>
```

RAID XML Data Source:

`/var/www/xml/sysinfo.xml`

```
<config>
        <vols>
                <vol id="1">
                        <num>1</num>
                        <name>Volume_1</name>
                        <label>Volume_1</label>
                        <mnt>/mnt/HD/HD_a2</mnt>
                        <encrypted>0</encrypted>
                        <dev>/dev/sda2</dev>
                        <unlocked>1</unlocked>
                        <mounted>1</mounted>
                        <size>1963698696192</size>
                        <uuid>WD-WCCXXXXXXXXX</uuid>
                        <roaming>0</roaming>
                        <raid_level>standard</raid_level>
                        <raid_state>clean</raid_state>
                        <raid_state_detail></raid_state_detail>
                        <used_size>231814733824</used_size>
                        <state></state>
                </vol>
        </vols>
</config>
```

Cerberus Nov 26

Serial Number Breakdown (My Cloud):

```
   W = Western Digital
   N = Drives (N = None, U = 4 TB)
  AQ = Product code
   2 = Manufactured in Thailand
   Y = Year 8 = 2018 (WD fiscal)
  WW = Work Week (WD fiscal)
SSSS = Building S/N
```

Skip to main content

Serial Number Examples:

```
W-N-AP-2-7-20-XXXX (PR2100, Diskless) - 2017
W-N-AQ-2-8-50-XXXX (PR4100, Diskless) - 2018
W-N-AQ-2-8-20-XXXX (PR4100, Diskless) - 2018
W-N-AQ-2-9-19-XXXX (PR4100, Diskless) - 2019
W-U-Q2-4-3-00-XXXX (EX2 Ultra, 4 TB) - 2013
W-U-BM-3-6-23-XXXX (EX2 Ultra, 8 TB) - 2016
W-N-AN-3-8-17-XXXX (EX2 Ultra, Diskless) - 2018
W-N-AN-3-0-20-XXXX (EX2 Ultra, Diskless) - 2020
W-N-AN-3-1-49-XXXX (EX2 Ultra, Diskless) - 2021
W-N-AN-3-1-51-XXXX (EX2 Ultra, Diskless) - 2021
```

Serial Number Decode (S/N or E/N):

```
W-N-AQ-2-7-19-XXXX
| | |  | | |  |
| | |  | | |  +---- Serial Number
| | |  | | +------- Work Week (WD Fiscal)
| | |  | +--------- Work Year (WD Fiscal)
| | |  +----------- Country of Origin (2, 3) (Taiwan, Thailand, Malaysia)
| | +-------------- Product Code
| +---------------- Drives (N = None, U = Drives Installed)
+------------------ Manufacturer (W = Western Digital)
```

WD Fiscal Years:

```
WD Fiscal Year 2016 (07/01/2015 - 06/29/2016)
WD Fiscal Year 2017 (07/01/2016 - 06/29/2017)
WD Fiscal Year 2018 (07/01/2017 - 06/29/2018)
WD Fiscal Year 2019 (07/01/2018 - 06/29/2019)
WD Fiscal Year 2020 (07/01/2019 - 06/29/2020)
WD Fiscal Year 2021 (07/01/2020 - 06/29/2021)
WD Fiscal Year 2022 (07/01/2021 - 06/29/2022)
WD Fiscal Year 2023 (07/01/2022 - 06/29/2023)
WD Fiscal Year 2024 (07/01/2023 - 06/29/2024)
```

Date Examples:

```
FISCAL YEAR + WORK WEEKS = DATE

S/N: WNAQ2719XXXX (2017 Model) 11/11/2016
S/N: WNAQ2830XXXX (2018 Model) 01/27/2018
E/N: WUBP2851XXXX (2018 Model) 06/23/2018
S/N: WNAQ2941XXXX (2019 Model) 04/14/2019
S/N: WNAQ2030XXXX (2020 Model) 01/27/2020
```

Keerti_01 WDStaff Nov 28

Hi @Cerberus ,

Have you opened a Support Case? If not opened, for more information, please contact the WD Technical Support team for the best assistance and troubleshooting:

Western Digital Support Ask A Question

Not finding the answer to your Western Digital storage product question? You can ask your Western Digital Support question here.

Cerberus Nov 30

Skip to main content

The following Bash script will check the EMMC (Embedded Multi-Media Card) flash memory device of the PR4100 or PR2100 for errors.

```
#!/bin/bash

mknod /dev/mmcblk0p7 b 179 7 > /dev/null 2>&1
mknod /dev/mmcblk0p8 b 179 8 > /dev/null 2>&1

blkdev="$(blkid | grep 'wdnas_efi' | awk -F: '{print $1}' | cut -c1-13)";
echo "Checking integrity of system partitions (${blkdev}X)";

# Ignore the following fsck errors, they're caused by My Cloud OS5 not includi
# Cannot initialize conversion from codepage 850 to ANSI_X3.4-1968: Invalid ar
# Cannot initialize conversion from ANSI_X3.4-1968 to codepage 850: Invalid ar

# EFI vfat Partition
if stat "${blkdev}1" >/dev/null 2>&1; then
        umount /dev/mmcblk0p1 2>/dev/null;
        if [ $(blkid | grep "${blkdev}1" | awk -F' ' '{print $6}') = 'TYPE="v
                # fsck.fat -p ${blkdev}1;
                echo "${blkdev}1 ($(blkid | grep ${blkdev}1 | awk -F'"' '{prin
        else
                echo "${blkdev}1 ($(blkid | grep ${blkdev}1 | awk -F'"' '{prin
        fi;
else
        echo "${blkdev}1 ($(blkid | grep ${blkdev}1 | awk -F'"' '{print $2}'))
fi;

# Linux ext4 Partitions
for i in $(seq 2 9); do
        if stat "${blkdev}$i" >/dev/null 2>&1; then
                umount /dev/mmcblk0p$i 2>/dev/null;
                if [ $(blkid | grep "${blkdev}$i" | awk -F' ' '{print $5}') =
                        # e2fsck -p "${blkdev}$i" 2>/dev/null;
                        echo "$blkdev$i ($(blkid | grep $blkdev$i | awk -F'"'
                else
                        echo "${blkdev}1 ($(blkid | grep ${blkdev}1 | awk -F'
                fi
        else
                echo "${blkdev}1 ($(blkid | grep ${blkdev}1 | awk -F'"' '{prin
        fi;
done;
```

Cerberus Nov 30

The following are My Cloud PR4100 and PR2100 customized temperature thresholds used to make automatic adjustments to the fan speed.

CPU Temperature Range:

- 80C/194F = Critical
- 70C/158F = Danger
- 60C/140F = Hot
- 40C/104F = Normal

RAM Temperature Range:

- 80C/176F = Critical
- 70C/158F = Danger
- 50C/122F = Hot
- al

Skip to main content

PMC Temperature Range:

- 80C/194F = Critical
- 70C/158F = Danger
- 50C/122F = Hot
- 40C/104F = Normal

HDD Temperature Range:

- 70C/158F = Critical
- 60C/140F = Danger
- 50C/122F = Hot
- 40C/104F = Normal

Cerberus Nov 30

My Cloud OS5 Firmware Version Notes

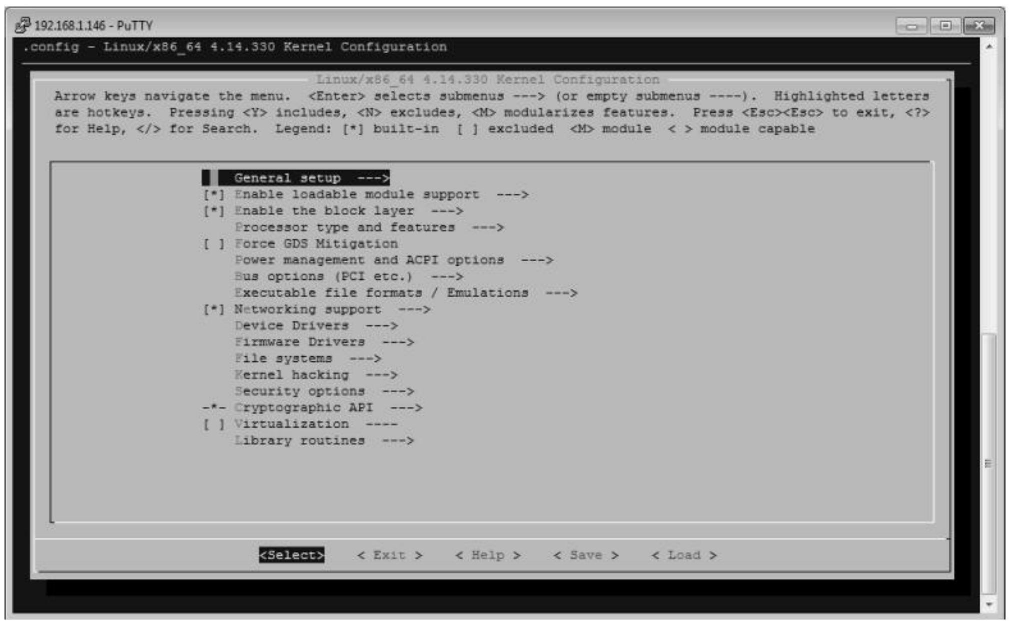
Version	Date	Notes
5.27.157	2023-10-19	Debian 11 (“Bullseye”), glibc 2.31 (undocumented)
5.26.300	2023-06-20	
5.26.202	2023-05-15	
5.26.119	2023-01-05	
5.25.132	2022-12-19	
5.25.124	2022-11-29	
5.24.108	2022-09-20	Required to update prior firmware to latest (99% problem)
5.23.114	2022-07-19	
5.22.113	2022-05-17	
5.21.104	2022-03-22	
5.20.113	2022-02-08	
5.19.117	2022-01-10	Deprecated Netatalk
5.18.117	2021-10-28	
5.17.107	2021-09-09	
5.16.105	2021-08-12	
5.15.106	2021-07-13	
5.14.105	2021-06-08	
5.13.115	2021-05-12	
5.12.108	2021-04-13	
5.11.112	2021-03-18	Added disable HTTPS redirect
5.10.122	2021-02-25	
5.09.115	2021-01-19	
5.08.115	2020-12-17	
5.07.118	2020-12-09	Added exFAT driver for USB storage
5.06.115	2020-11-19	Removed dashboard subnet access restriction
5.05.111	2020-11-10	
5.04.114	2020-10-27	Debian 10 (“Buster”), Kernel 4.14.22, glibc 2.28
5.03.103	2020-10-13	Limited public release (Guinea Pig Release x3)
5.02.134	2020-09-25	Limited public release (Guinea Pig Release x2), unlisted
5.02.104	2020-10-01	Limited public release (Guinea Pig Release x1)

My Cloud OS5 version 5.27.157 updated glibc to version 2.31 (undocumented), but it’s still far too old. Versions 5.02.134 and 5.02.104 have date discrepancies.

Skip to main content

Cerberus Dec 2

Since the Linux kernel (4.14.22) included with My Cloud OS5 is way past it’s shelf life, I decided to compile a customized version of Linux kernel (4.14.330) and successfully tested it on the PR4100 without any problems.



```
.config - Linux/x86_64 4.14.330 Kernel Configuration
                    Linux/x86_64 4.14.330 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters
  are hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?>
  for Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                    General setup  --->
                [*] Enable loadable module support  --->
                [*] Enable the block layer  --->
                    Processor type and features  --->
                [ ] Force GDS Mitigation
                    Power management and ACPI options  --->
                    Bus options (PCI etc.)  --->
                    Executable file formats / Emulations  --->
                [*] Networking support  --->
                    Device Drivers  --->
                    Firmware Drivers  --->
                    File systems  --->
                    Kernel hacking  --->
                    Security options  --->
                -*- Cryptographic API  --->
                [ ] Virtualization  ----
                    Library routines  --->

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

This is strictly for personal use and I won’t be releasing any custom Linux kernels for My Cloud NAS devices, because there are risks involved, and I simply don’t want to deal with potential headaches.

Most users can barely establish an SSH connection and run basic Linux commands, let alone fix Linux kernel problems, which is infinitely more complex.

Cerberus Dec 3

The hard drive health status displayed on the My Cloud OS5 dashboard originates as follows.

`home_mgr.cgi --> xmldb --> sysinfo.xml --> sysinfod --> smartctl`

- `/usr/local/modules/cgi/home_mgr.cgi`
- `/usr/local/modules/usr/sbin/xmldb`
- `/var/www/xml/sysinfo.xml`
- `/usr/local/modules/wd/usr/sbin/sysinfod`
- `/usr/local/modules/usr/bin/smartctl`

To determine if a drive is **NOT** healthy, `sysinfod` merely looks for the following string plus the word “`FAILED`” in the `smartctl` program output, which is a sloppy and error-prone way of doing it because it neglects to consider the actual S.M.A.R.T. status of the drive.

- `SMART overall-health self-assessment test result`
- `FAILED`

An example of the full output of the `smartctl` program command is shown below.

```
# smartctl --smart on --info --health /dev/sda

smartctl 7.2 2020-12-30 r5155 [x86_64-linux-4.14.22] (local build)
Copyright (C) 2002-20, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF INFORMATION SECTION ===
Model Family:     Western Digital Black
Device Model:     WDC WD2003FZEX-00Z4SA0
Serial Number:    WD-XXXXXXXXXXXX
LU WWN Device Id: 5 0014ee 20a3f5855
Firmware Version: 01.01A01
User Capacity:    2,000,398,934,016 bytes [2.00 TB]
Sector Sizes:     512 bytes logical, 4096 bytes physical
Rotation Rate:    7200 rpm
Device is:        In smartctl database [for details use: -P show]
                  ACS-2 (minor revision not indicated)
SATA Version is:  SATA 3.0, 6.0 Gb/s (current: 6.0 Gb/s)
```

Skip to main content

```
Local Time is:    Sun Dec  3 02:32:24 2023 EST
SMART support is: Available - device has SMART capability.
SMART support is: Enabled

=== START OF ENABLE/DISABLE COMMANDS SECTION ===
SMART Enabled.

=== START OF READ SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED
```

Cerberus Dec 3

The WD naming convention for My Cloud NAS devices is a master class in how **NOT** to do it, where calling everything My Cloud only sets the stage for endless device identification problems, and many of the device sub-names only make it worse.

At the very least, having unique My Cloud sub-names would have been better than the mixed-bag that WD came up with. Each My Cloud NAS device has a unique code, and there's absolutely no reason why the WD marketing knuckleheads couldn't have followed suit. They had 26 letters and 10 numbers to choose from, and this is what they came up with?

Very Confusing:

- `My Cloud <-- Catches everything`
- `My Cloud Home <-- Catches everything, but WORSE`
- `My Cloud EX2`
- `My Cloud EX2 Ultra <-- Easily confused with EX2`

Less Confusing:

- `My Cloud PR4100`
- `My Cloud PR2100`
- `My Cloud DL4100`
- `My Cloud DL2100`
- `My Cloud EX4100`
- `My Cloud EX2100`

A simple solution would have been to have a My Cloud network, and unique device names that use the My Cloud network. Thus, branding is preserved, and everything makes sense. For example…

Not Confusing:

- `SX1 (My Cloud)`
- `HX1 (My Cloud Home)`
- `HX2 (My Cloud Home Duo)`
- `EX2`
- `FX2 (EX2 Ultra)`
- `PR4100`
- `PR2100`
- `DL4100`
- `DL2100`
- `EX4100`
- `EX2100`

New & Unread Topics

Topic	Replies	Views	Activity
Albums sort	1	223	Jul 15

Skip to main content

Topic	Replies	Views	Activity
My Cloud EX2 Ultra · Red Light on Disc 1 Error 36	1	288	Jul 25
My Cloud OS5 Firmware 5.27.157 Problems	18	1.5k	May 24
The RAID Fallacy	1	344	Nov 26
☑ Pr2100 RAID 1 is expansion…i think I screwed up my-cloud, update	2	230	Nov 20

Want to read more? Browse other topics in or view latest topics.