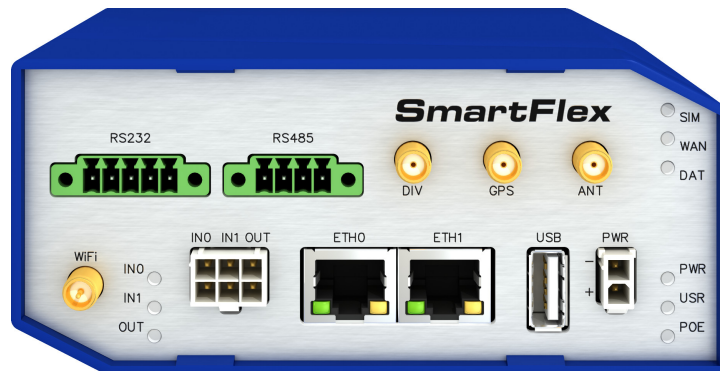


User module

Ext4_tools

APPLICATION NOTE



ADVANTECH

Used symbols



Danger – Information regarding user safety or potential damage to the router.



Attention – Problems that can arise in specific situations.



Information, notice – Useful tips or information of special interest.



Example – example of function, command or script.

Contents

1	Description of user module	1
2	Module installation	2
3	Commands set usage	3
3.1	Filesystem creating	3
3.2	Device mounting	6
3.3	Filesystem checking	8
3.4	Device unmounting	9
4	Recommended literature	10

List of Figures

1	Module installation	2
2	Module installed	2

List of Tables

1	mke2fs options	5
2	mount flags	6
3	mount options	6
4	e2fsck options	8
5	umount flags	9

1. Description of user module



This user module is compatible with **SmartFlex** and **SmartMotion** Advantech routers only and supported for FW of version **5.3.4** or newer.



Be careful when working with router's file systems. If the internal FLASH is formatted accidentally, the router will be no operable any more and service intervention will be required.

The *Ext4_tools* user module allows the router to create and check *ext3*, *ext2* and **ext4** file systems. This user module has no graphical user interface, all operations can be performed through the command line interface.

There are two new commands available within this module, **mke2fs** and **e2fsck** commands. Installation and usage of this module is described in following chapters.

2. Module installation

Module installation file can be downloaded from *Advantech* web pages under *Support&-Downloads* section or obtained from your local sales representative eventually.

The easy way for installation of the module is to use the router's web configuration pages. Access the web configuration main page and navigate to *User Modules* under *Customization* menu section, see figure 1. Click on *Browse...* button and locate the installation file named *ext4_tools.v3.tgz*.

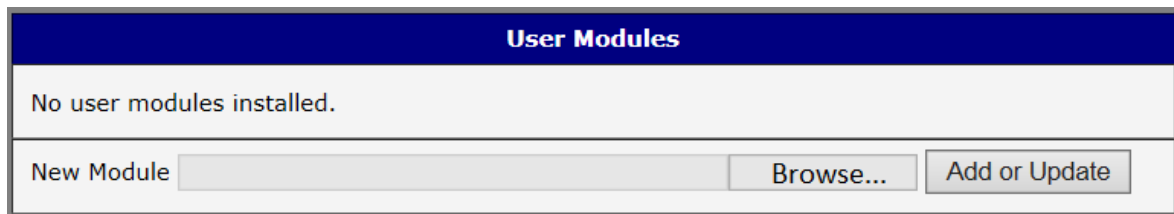


Figure 1: Module installation

To proceed with the installation click on *Add or Update* button. The module is now installed and listed as *Ext2/3/4 Filesystems Utilities* module, as shown on figure 2.

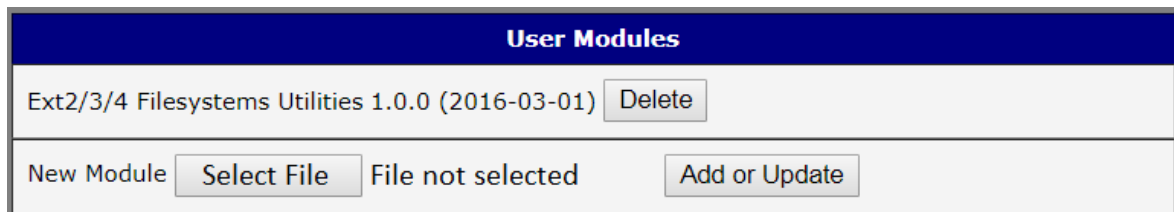


Figure 2: Module installed

3. Commands set usage

This chapter describes the usage of commands set for filesystems administration. **mount** and **umount** commands are pre-installed in the router, **mke2fs** and **e2fsck** commands were installed withing the *Ext4_tools* user module.

3.1 Filesystem creating

mke2fs command can be used to create a filesystem. It can create an ext2, ext3 or ext4 filesystem.

Synopsis:

```
mke2fs [-c|-l filename] [-b block-size] [-C cluster-size] [-i bytes-per-inode]
[-I inode-size] [-J journal-options] [-G flex-group-size]
[-N number-of-inodes] [-m reserved-blocks-percentage] [-o creator-os]
[-g blocks-per-group] [-L volume-label] [-M last-mounted-directory]
[-O feature[,...]] [-r fs-revision] [-E extended-option[,...]]
[-t fs-type] [-T usage-type ] [-U UUID] [-jnvDFKSV] device [blocks-count]
```

Options:

Option	Description
-b block-size	Specify the size of blocks in bytes. Valid block-size values are 1024, 2048 and 4096 bytes per block.
-c	Check the device for bad blocks before creating the file system. If this option is specified twice, then a slower read-write test is used instead of a fast read-only test.
-E extended-options	Set extended options for the filesystem. Extended options are comma separated, and may take an argument using the equals ('=') sign. The following extended options are supported: stride=stripe-size stripe-width=stripe-width resize=max-online-resize lazy_itable_init= (0 to disable, 1 to enable) test_fs - using experimental kernel code discard - discard blocks at mkfs time nodiscard - do not discard blocks at mkfs time

Continued on the next page

Continued from previous page

Option	Description
-F	Force mke2fs to create a filesystem, even if the specified device is not a partition on a block special device, or if other parameters do not make sense.
-g blocks-per-group	Specify the number of blocks in a block group. There is generally no reason for the user to ever set this parameter, as the default is optimal for the filesystem.
-G flex-group-size	Specify the number of block groups that will be packed together to create a larger virtual block group (or flex_bg group) in an ext4 filesystem.
-i bytes-per-inode	Specify the bytesinode ratio. mke2fs creates an inode for every bytes-per-inode bytes of space on the disk. The larger the bytes-per-inode ratio, the fewer inodes will be created.
-I inode-size	Specify the size of each inode in bytes. mke2fs creates 256-byte inodes by default.
-j	Create the filesystem with an ext3 journal. If the -J option is not specified, the default journal parameters will be used to create an appropriately sized journal (given the size of the filesystem) stored within the filesystem.
-J journal-options	Create the ext3 journal using options specified on the command-line. Journal options are comma separated, and may take an argument using the equals sign ('='). The following journal options are supported: size=journalsize device=externaljournal mke2fs -O journal_dev - external-journal
-K	Keep, do not attempt to discard blocks at mkfs time.
-l filename	Read the bad blocks list from filename. Note that the block numbers in the bad block list must be generated using the same block size as used by mke2fs.
-L volume-label	Set the volume label for the filesystem to new-volume-label. The maximum length of the volume label is 16 bytes.
-m reserved-blocks-percentage	Specify the percentage of the filesystem blocks reserved for the super-user. This avoids fragmentation, and allows root-owned daemons, such as syslogd(8), to continue to function correctly after non-privileged processes are prevented from writing to the filesystem. The default percentage is 5%.

Continued on the next page

Continued from previous page

Option	Description
-M last-mounted-directory	Set the last mounted directory for the filesystem.
-n	Causes mke2fs to not actually create a filesystem, but display what it would do if it were to create a filesystem..
-N number-of-inodes	Overrides the default calculation of the number of inodes that should be reserved for the filesystem .
-o creator-os	Overrides the default value of the "creator operating system" field of the filesystem.
-O feature[,...]	Create a filesystem with the given features (filesystem options), overriding the default filesystem options.
-q	Quiet execution.
-r fs-revision	Set the filesystem revision for the new filesystem.
-S	Write superblock and group descriptors only.
-t fs-type	Specify the filesystem type (i.e., ext2, ext3, ext4, etc.) that is to be created.
-T usage-type	Specify how the filesystem is going to be used, so that mke2fs can choose optimal filesystem parameters for that use.
-U UUID	Create the filesystem with the specified UUID..
-v	Verbose execution.
-V	Print the version number of mke2fs and exit.

Table 1: mke2fs options



For detailed description this command, visit [Linux manual pages](#).



Examples:

Force to create a filesystem of ext4 type on device `/dev/sda1`, which may be represented by an USB stick connected to the router.

```
mke2fs -F /dev/sda1 -t ext4
```

It is recommended not to do any filesystem operations on a mounted partition. But still you can force it to create a filesystem on in-use partition or mounted partition with option `-F` as shown in the example.

3.2 Device mounting

mount command can be used to mount a file system (on SD card, USB stick, etc.).

Synopsis:

```
mount [-afirw] [-t FSTYPE] [-O OPT] <DEVICE> <NODE> [ -o <option>, ...]
```

Options:

Flag	Description
-a	Mount all filesystems in fstab
-f	Dry run
-i	Don't run mount helper
-r	Read-only mount
-w	Read-write mount (default)
-t FSTYPE	Filesystem type
-O OPT	Mount only filesystems with option OPT (-a only)

Table 2: mount flags

Option	Description
loop	Ignored (loop devices are autodetected)
[a]sync	Writes are [a]synchronous
[no]atime	Disable/enable updates to inode access times
[no]diratime	Disable/enable atime updates to directories
[no]relatime	Disable/enable atime updates relative to modification time
[no]dev	(Dis)allow use of special device files
[no]exec	(Dis)allow use of executable files
[no]suid	(Dis)allow set-user-id-root programs
[r]shared	Convert [recursively] to a shared subtree
[r]slave	Convert [recursively] to a slave subtree
[r]private	Convert [recursively] to a private subtree
[un]bindable	Make mount point [un]able to be bind mounted
bind	Bind a directory to an additional location
move	Relocate an existing mount point
remount	Remount a mounted filesystem, changing its flags
ro/rw	Read-only/read-write mount

Table 3: mount options



For detailed description this command, visit Linux manual pages.



Examples:

Mount */dev/sda1* device, which can be represented by an USB flash stick connected to the router, to the directory */mnt/usb*. Make sure this destination directory exists.

```
mount -t vfat /dev/sda1 /mnt/usb
```

Note that the *-t* flag with the filesystem type is not mandatory. If the type is not specified, mount will try to guess the desired type.

3.3 Filesystem checking

e2fsck command is used to check the ext2, ext3 or ext4 file systems. Note that in general it is not safe to run e2fsck on mounted filesystems. The only exception is if the -n option is specified, and -c, -l, or -L options are not specified. However, even if it is safe to do so, the results printed by e2fsck are not valid if the filesystem is mounted.

Synopsis:

```
e2fsck [-pnycfv] [-C fd] [-b superblock] [-B blocksize]
        [-j external_journal] [-l|-L bad_blocks_file]
```

Options:

Option	Description
-p	Automatic repair (no questions)
-n	Make no changes to the filesystem
-y	Assume "yes" to all questions
-c	Check for bad blocks and add them to the badblock list
-f	Force checking even if filesystem is marked clean
-v	Be verbose
-C fd	This option causes e2fsck to write completion information to the specified file descriptor so that the progress of the filesystem check can be monitored
-b superblock	Use alternative superblock
-B blocksize	Force blocksize when looking for superblock
-j external_journal	Set location of the external journal
-l bad_blocks_file	Add to badblocks list
-L bad_blocks_file	Set badblocks list

Table 4: e2fsck options



For detailed description this command, visit Linux manual pages.



Examples:

Check for bad blocks on */dev/sda1* device.

```
mke2fs -c /dev/sda1
```

This will do a read-only test on the filesystem to make sure there are no bad blocks.

3.4 Device unmounting

umount command is used to manually unmount a mounted filesystem, informing the system to complete any pending read or write operations, and safely detaching it.

Synopsis:

```
umount [-arlfD] <FILESYSTEM> | <DIRECTORY>
```

Options:

Flag	Description
-a	Unmount all file systems
-r	Try to remount devices as read-only if mount is busy
-l	Lazy unmount (detach filesystem)
-f	Force unmount (i.e., unreachable NFS server)
-D	Don't free loop device even if it has been used

Table 5: umount flags



For detailed description this command, visit Linux manual pages.



Examples:

Unmount filesystem */dev/sda1*.

```
umount /dev/sda1
```

Note that mounted directory name can be used instead the filesystem path.

4. Recommended literature

- [1] Advantech B+B SmartWorx: **v2 Routers Configuration Manual** (MAN-0021-EN)
- [2] Advantech B+B SmartWorx: **SmartFlex Configuration Manual** (MAN-0023-EN)
- [3] Advantech B+B SmartWorx: **SmartMotion Configuration Manual** (MAN-0024-EN)
- [4] Advantech B+B SmartWorx: **SmartStart Configuration Manual** (MAN-0022-EN)
- [5] Advantech B+B SmartWorx: **ICR-3200 Configuration Manual** (MAN-0042-EN)



Product related documents can be obtained on *Engineering Portal* at <https://ep.advantech-bb.cz/> address.



Choose appropriate version of the manual matching model of your Advantech router.