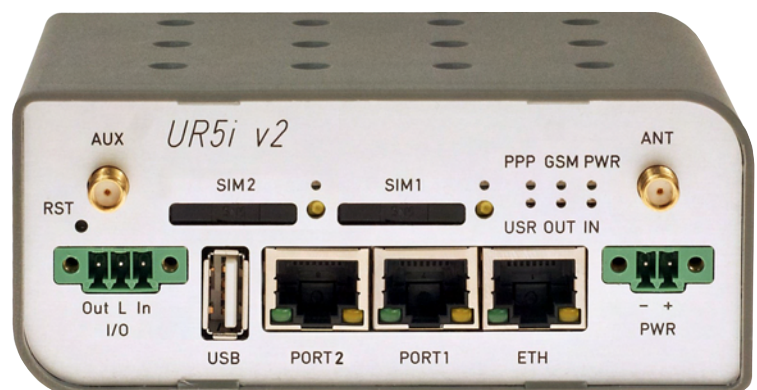
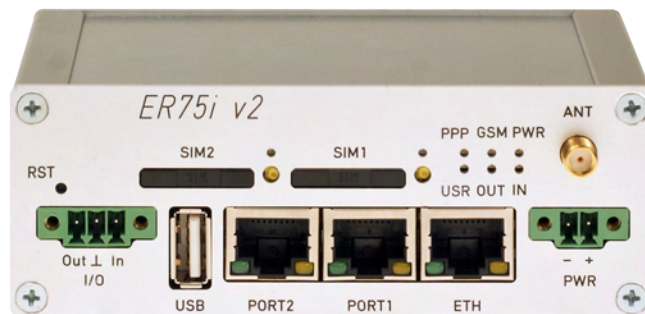








Commands and script for v2 routers

APPLICATION GUIDE



Used symbols

-  Danger – important notice, which may have an influence on the user’s safety or the function of the device.
-  Attention – notice on possible problems, which can arise in specific cases.
-  Information, notice – information, which contains useful advice or special interest.
-  Example – example of function, command or script.

Firmware version

Actual version of firmware is 3.0.6 (18.10.2012).

GPL license

Source codes under GPL license are available free of charge by sending an email to info@conel.cz.

Routers version

- ⤴ Properties and settings of router associated with the GSM connection is not available in industrial router XR5i v2.
- ⤴ PPPoE configuration item is only available on the industrial router XR5i v2, used to set the PPPoE connection over Ethernet.



**Declared quality system
ISO 9001**

Conel s.r.o., Sokolska 71, 562 04 Usti nad Orlici, Czech Republic
Issue in CZ, 11/5/2012





Contents

1.Commands	1
1.1.arp	1
1.2.awk	2
1.3.brctl	2
1.4.cat	3
1.5.cd	3
1.6.chmod	4
1.7.conntrack	4
1.8.cp	5
1.9.date	6
1.10.defaults	6
1.11.df	6
1.12.dmesg	7
1.13.echo	7
1.14.email	8
1.15.ethtool	8
1.16.free	8
1.17.grep	9
1.18.gsmat	10
1.19.gsminfo	10
1.20.gsmpwr	10
1.21.gsmsms	11
1.22.gunzip	11
1.23.gzip	11
1.24.hwclock	12
1.25.ifconfig	12
1.26.io	13
1.27.ip	13
1.28.iptables	14
1.29.kill	14
1.30.killall	15
1.31.led	15
1.32.ln	16
1.33.logger	16
1.34.ls	17
1.35.mac	17
1.36.mkdir	18
1.37.mount	18
1.38.mv	19
1.39.ntpdate	19
1.40.openssl	20
1.41.passwd	20
1.42.ping	20
1.43.portd	21
1.44.ps	21



CONTENTS

1.45.pwd	21
1.46.reboot	22
1.47.restore	22
1.48.rm	22
1.49.rmdir	23
1.50.route	23
1.51.sed	24
1.52.service	24
1.53.sleep	24
1.54.slog	25
1.55.snmptrap	25
1.56.tail	26
1.57.tar	26
1.58.tcpdump	27
1.59.telnet	27
1.60.touch	28
1.61.traceroute	28
1.62.umount	29
1.63.vi	29
1.64.wget	30
2.Script examples	31
2.1.Send SMS	31
2.2.SMS command 1	31
2.3.SMS command 2	32
2.4.Send information email 1	32
2.5.Send information SMNP trap 1	32
2.6.Send information email 2	33
2.7.Send information SNMP trap 2	33
2.8Automatic reboot	33
2.9.Switch between WAN and PPP	34

**Table list**

Table 1: arp options	1
Table 2: awk options	2
Table 3: brctl commands	3
Table 4: cat options	3
Table 5: cd options	3
Table 6: chmod options	4
Table 7: contrack comands	4
Table 8: contrack table	4
Table 9: contrack options	5
Table 10: expectation options	5
Table 11: contrack and expectation options	5
Table 12: cp options	5
Table 13: date options	6
Table 14: df options	6
Table 15: dmesg options	7
Table 16: echo options	7
Table 17: email options	8
Table 18: grep options	9
Table 19: Description of GSM information	10
Table 20: gunzip options	11
Table 21: gzip options	11
Table 22: hwclock options	12
Table 23: ifconfig options	13
Table 24: io options	13
Table 25: IP options	14
Table 26: IP object	14
Table 27: kill options	15
Table 28: killall options	15
Table 29: led options	15
Table 30: ln options	16
Table 31: logger options	16
Table 32: ls options	17
Table 33: mkdir options	18
Table 34: mount flags	18
Table 35: mount options	18
Table 36: mv options	19
Table 37: ntpdate options	19
Table 38: ping options	20
Table 39: portd options	21
Table 40: reboot options	22
Table 41: rm options	22
Table 42: route options	23
Table 43: sed options	24
Table 44: slog options	25



TABLE LIST

Table 45: snmptrap options	25
Table 46: tail options	26
Table 47: tar options	26
Table 48: touch options	28
Table 49: traceroute options	28
Table 50: umount options	29
Table 51: vi options	29
Table 52: wget options	30

1. Commands

1.1. arp

Description:

The arp program displays and modifies the Internet-to-Ethernet address translation tables used by the address resolution protocol.

Synopsis:

```
arp [-a <hostname>] [-s <hostname> <hw_addr>] [-d <hostname>] [-v] [-n] [-i <if>]
[-D <hostname>] [-A ] [-f <filename>]
```

Options:

Option	Description
-a	The entries will be displayed in alternate (BSD) style.
-s	Manually create an ARP address mapping entry for host hostname with hardware address set to hw_addr.
-d	Remove any entry for the specified host.
-v	Tell the user what is going on by being verbose.
-n	Shows numerical addresses instead of trying to determine symbolic host, port or user names.
-i	Select an interface.
-D	Use the interface ifa's hardware address.
-f	Similar to the -s option, only this time the address info is taken from file filename set up. The name of the data file is very often <i>/etc/ethers</i> , but this is not official. If no filename is specified <i>/etc/ethers</i> is used as default. The format of the file is simple; it only contains ASCII text lines with a hardware address and a hostname separated by whitespace. Additionally the pub, temp and netmask flags can be used.

Table 1: arp options

With no flags, the program displays the current ARP entry for hostname. The host may be specified by name or by number, using Internet dot notation.



For detail description this command, visit Linux manual pages.

Examples:



View arp table without translating IP addresses to domain names
arp -n

1.2. *awk*

Description:

Awk scans each input *file* for lines that match any of a set of patterns specified literally in *program-text* or in one or more files specified as **-f progfile**.

Synopsis:

```
awk [-v] [-F] [-f] ... [<program-text>] [<file> ...]
```

Options:

Option	Description
-v	Assign the value <i>val</i> to the variable <i>var</i> , before execution of the program begins. Such variable values are available to the BEGIN block of an AWK program.
-F	Use for the input field separator (the value of the FS predefined variable).
-f	Read the AWK program source from the file <i>program-file</i> , instead of from the first command line argument. Multiple -f (or --file) options may be used.

Table 2: *awk* options

Examples:



Show IP address of Gateway

```
route -n | awk '/^0\.0\.0\.0/ { print $2 }'
```

1.3. *brctl*

Description:

Brctl is used to set up, maintain, and inspect the Ethernet bridge configuration in the Linux kernel.

An Ethernet bridge is a device commonly used to connect different networks of Ethernets together, so that these Ethernets will appear as one Ethernet to the participants.

Each of the Ethernets being connected corresponds to one physical interface in the bridge. These individual Ethernets are bundled into one bigger ('logical') Ethernet, this bigger Ethernet corresponds to the bridge network interface.

Synopsis:

```
brctl [<commands>]
```

Options:

Command	Parameters	Description
addbr	<bridge>	Add bridge.
delbr	<bridge>	Delete bridge.
addif	<bridge> <device>	Add interface to bridge.
delif	<bridge> <device>	Delete interface from bridge.
setageing	<bridge> <time>	Set ageing time.
setbridgepri	<bridge> <prio>	Set bridge priority.
setfd	<bridge> <time>	Set bridge forward delay.
sethello	<bridge> <time>	Set hello time.
setmaxage	<bridge> <time>	Set max message age.
setpathcost	<bridge> <port> <cost>	Set path cost.

setportprio	<bridge> <port> <prio>	Set port priority.
show		Show list of bridges.
showmacs	<bridge>	Show list of mac address.
showstp	<bridge>	Show bridge stp info.
stp	<bridge> {on off}	Turn stp on/off.

Table 3: brctl commands

Examples:

Create bridge between eth0 and eth1.

```
brctl addbr br0
brctl addif br0 eth0
brctl addif br0 eth1
```

1.4. cat

Description:

This command concatenates files and print on the standard output.

Synopsis:

```
cat [-u] [<file>] ...
```

Options:

Option	Description
-u	Ignored since unbuffered i/o is always used.

Table 4: cat options

Examples:

View the contents of file /proc/tty/driver/spear_serial (information about serial ports v2 routers).

```
cat /proc/tty/driver/spear_serial
```

Copy the contents of the router configuration files in /tmp/my.cfg.

```
cat /etc/settings.* > /tmp/my.cfg
```

1.5. cd

Description:

This command is used to change the current working directory.

Synopsis:

```
cd [-P] [-L] [<directory>]
```

Options:

Option	Description
-P	Do not follow symbolic links.
-L	Follow symbolic links (default).

Table 5: cd options

Examples:

Move to home directory (/root).

```
Cd
```

Move to directory /mnt.

```
cd /mnt
```

1.6. *chmod*

Description:

This command is used to change file mode bits.

Synopsis:

`chmod [-R] <mode> <filename>`

Options:

Option	Description
-R	Change files and directories recursively.

Table 6: chmod options

Examples:

Settings rights (permit execution) of script `/tmp/script`.

`chmod 755 /tmp/script`



1.7. *conntrack*

Description:

This program is user interface to netfilter connection tracking system.

Synopsis:

`conntrack [commands] [option]`

Options:

Commnads	Description
-L [table] [option]	List conntrack or expectation table.
-G [table]	Get conntrack or expectation.
-D [table]	Delete conntrack or expectation.
-I [table]	Create a conntrack or expectation.
-U [table]	Update a conntrack.
-E [table]	Show events.
-F [table]	Flush table.

Table 7: conntrack comands

Table	Description
conntrack	This is the default table. It contains a list of all currently tracked connections through the system.
expect	This is the table of expectations. Connection tracking expectations are the mechanism used to "expect" RELATED connections to existing ones.

Table 8: conntrack table

Option	Description
-n <ip>	source NAT ip
-g <ip>	destination NAT ip
-m <mark>	Set mark
-e <eventmask>	Event mask, eg. NEW,DESTROY
-z	Zero counters while listing
-o <type[,...]>	Output format, eg. xml

Table 9: conntrack options

Option	Description
--tuple-src <ip>	Source address in expect tuple.
--tuple-dst <ip>	Destination address in expect tuple.
--mask-src <ip>	Source mask address.
--mask-dst <ip>	Destination mask address.

Table 10: expectation options

Option	Description
-s <ip>	Source address from original direction.
-d <ip>	Destination address from original direction.
-r <ip>	Source address from reply direction.
-q <ip>	Destination address from reply direction.
-p <proto>	Layer 4 Protocol, eg. 'tcp'.
-f <proto>	Layer 3 Protocol, eg. 'ipv6'.
-t <timeout>	Set timeout.
-u <status>	Set status, eg. ASSURED.

Table 11: contrack and expectation options

Examples:



Display content of contrack table.
contrack -L



Delete content of contrack table.
contrack -F

1.8. cp

Description:

This command is used to copy files and directories.

Synopsis:

`cp [<option>] <source> <dest>`

Options:

Option	Description
-a	Preserve the all attributes
-d, -P	Never follow symbolic links
-H, -L	Follow command-line symbolic links
-p	Preserve the mode, ownership, timestamps attributes
-f	If an existing destination file cannot be opened, remove it and try again
-i	Prompt before overwrite
-R, -r	Copy directories recursively

Table 12: cp options

Examples:



Copy the system log to directory /mnt.
cp /var/log/messages /mnt*



Copy configuration profile "Alternative 1" to profile "Standard".
cp -r /etc/alt1/ /etc*

1.9. *date*

Description:

This command is used to displays the current time in the given FORMAT, or sets the system date.

Synopsis:

`date [-R] [-d <string>] [-s] [-r <file>] [-u] [MMDDhhmm[[CC]YY][.ss]]`

Options:

Option	Description
-R	Output date and time in RFC 2822 format
-d <string>	Display time described by STRING, not 'now'
-s	Set time described by STRING
-r <file>	Display the last modification time of FILE
-u	Print or set Coordinated Universal Time

Table 13: date options

Examples



Display the current date and time.
`date`



Setting the date and time on December 24, 2011 20:00.
`date 122420002011`

1.10. *defaults*

Description:

The script is used to restore the default configuration.

Synopsis:

`defaults`

1.11. *df*

Description:

This command is used to view report file system disk space usage.

Synopsis:

`df [-k] [<filesystem> ...]`

Options:

Option	Description
-k	Print sizes in kilobytes.

Table 14: df options

1.12. *dmesg*

Description:

This command is used to print or control the kernel ring buffer.

Synopsis:

```
dmesg [-c] [-n <level>] [-s <size>]
```

Options:

Option	Description
-c	Clears the ring buffer's contents after printing.
-n <level>	Set the level at which logging of messages is done to the console.
-s <size>	Use a buffer of size SIZE to query the kernel ring buffer. This is 16392 by default.

Table 15: dmesg options

Examples:

View the latest news and subsequent deletion of the kernel ring buffer.

```
dmesg -c
```



1.13. *echo*

Description:

This command prints the strings to standard output.

Synopsis:

```
echo [-n] [-e] [-E] [<string> ...]
```

Options:

Option	Description
-n	do not output the trailing newline
-e	enable interpretation of backslash escapes
-E	disable interpretation of backslash escapes (default)

Table 16: echo options

Examples:

Switch profile to "Standard".

```
echo "PROFILE=" > /etc/settings
reboot
```



Switch profile to "Alternative 1".

```
echo "PROFILE=alt1" > /etc/settings
reboot
```



Send a sequence of bytes 0x41,0x54,0x0D,0x0A to serial line (write data in octal).

```
echo -n -e "\101\124\015\012" > /dev/ttyS0
```



1.14. *email*

Description:

The program used for sending email.

Synopsis:

`email -t <to> [-s <subject>] [-m <message>] [-a <attachment>] [-r <retries>]`

Options:

Option	Description
-t	Email of recipient.
-s	Subject of email.
-m	Message of email.
-a	Attachment of email.
-r	Number of retries.

Table 17: email options

Examples:



Send system logs to the address `john.doe@email.com`.
`email -t john.doe@email.com -s "System Log" -a /var/log/messages`

1.15. *ethtool*

Description:

This command is used to display or change Ethernet card settings.

Synopsis:

`ethtool [<option> ...] <devname> [<commands>]`

Options:

For detail description this command, visit Linux manual pages.



Examples:



View the status of the interface `eth0`.
`ethtool eth0`



Switch interface `eth0` to mode 10 Mbit/s, half duplex.
`ethtool -s eth0 speed 10 duplex half autoneg off`



Turn on autonegacion on the interface `eth0`.
`ethtool -s eth0 autoneg on`

1.16. *free*

Description

This command is used to display information about free and used memory.

Synopsis

`free`

1.17. *grep*

Description:

Grep searches the named input FILES (or standard input if no files are named, or the file name - is given) for lines containing a match to the given PATTERN. By default, grep prints the matching lines.

Synopsis:

```
grep [<options> ...] <pattern> [<file> ...]
```

Options :

Option	Description
-H	Print the filename for each match.
-h	Suppress the prefixing of filenames on output when multiple files are searched.
-i	Ignore case distinctions
-l	Suppress normal output; instead print the name of each input file from which output would normally have been printed.
-L	Suppress normal output; instead print the name of each input file from which no output would normally have been printed.
-n	Prefix each line of output with the line number within its input file.
-q	Quiet; do not write anything to standard output. Exit immediately with zero status if any match is found, even if an error was detected. Also see the -s or --no-messages option.
-v	Invert the sense of matching, to select non-matching lines.
-s	Suppress error messages about nonexistent or unreadable files.
-c	Suppress normal output; instead print a count of matching lines for each input file.
-f	Obtain patterns from FILE, one per line.
-e	Use PATTERN as the pattern; useful to protect patterns beginning with -.
-F	Interpret PATTERN as a list of fixed strings, separated by new lines, any of which is to be matched.

Table 18: grep options

Examples:



See all lines of system log in which occurs the word "error"

```
grep error /var/log/messages
```



View all processes whose name the contents of the string "ppp".

```
ps | grep ppp
```

1.18. *gsmat*

Description:

The program used for sending AT command to GSM module.

Synopsis:

`gsmat <AT command>`

Examples:



Determine the type and firmware version of GSM module.

`gsmat ATI`



Determine the IMEI code of module.

`gsmat "AT+GSN"`

1.19. *gsminfo*

Description:

The program used to display information about the signal quality.

Synopsis:

`gsminfo`

Description of item:

Item	Description
PLMN	Code of operator
Cell	The cell to which the router is connected
Channel	The channel on which the router communicates
Level	The signal quality of the selected cell
Neighbours	Signal quality of neighboring hearing cells
Uptime	Time to establish PPP connection

Table 19: Description of GSM information

1.20. *gsmplr*

Description:

The program used to control the supply of GSM module.

Synopsis:

`gsmplr [on | off]`

Examples:



Power of GSM module is turning on.

`gsmplr on`



Power of GSM module is turning off.

`gsmplr off`

1.21. *gsmsms*

Description:

The program used to send SMS message.

Synopsis:

`gsmsms <phone number> <text>`

Examples:

 Send SMS "Hello word" on telephone number +420123456789.
`gsmsms +420123456789 "Hello word"`

1.22. *gunzip*

Description:

This program is used to decompress FILE (or standard input if filename is '-').

Synopsis:


`gunzip [-c] [-f] [-t] <filename>`

Options :

Option	Description
-c	Write output on standard output
-f	Force decompression even if the file has multiple links or the corresponding file already exists, or if the compressed data is read from or written to a terminal.
-t	Test. Check the compressed file integrity.

Table 20: gunzip options

Examples:

 Decompression of file test.tar.gz (creates file test.tar).
`gunzip test.tar.gz`

1.23. *gzip*

Description:

This program is used to compress FILE with maximum compression.

Synopsis:


`gzip [-c] [-d] [-f] <filename>`

Options :

Option	Description
-c	Write output on standard output.
-d	Decompress.
-f	Force compression even if the file has multiple links or the corresponding file already exists, or if the compressed data is read from or written to a terminal.

Table 21: gzip options

Examples:

 Compression of file test.tar (creates file test.tar.gz).
`gzip test.tar`

1.24. *hwclock*

Description:

This program is used to query and set the hardware clock (RTC).

Synopsis:

`hwclock [-r] [-s] [-w] [-u] [-l]`

Options :

Option	Description
-r	Read hardware clock and print result.
-s	Set the System Time from the Hardware Clock.
-w	Set the Hardware Clock to the current System Time.
-u	The hardware clock is kept in coordinated universal time.
-l	The hardware clock is kept in local time.

Table 22: *hwclock* options

Examples:

Set the hardware clock to the current system time.

`hwclock -w -u`



1.25. *ifconfig*

Description:

This command is used to configure a network interface.

Synopsis:

`ifconfig [-a] <interface> [<option> ...]`

Options :

Option	Description
<code>broadcast <address></code>	If the address argument is given, set the protocol broadcast address for this interface.
<code>pointtopoint <address></code>	This keyword enables the point-to-point mode of an interface, meaning that it is a direct link between two machines with nobody else listening on it.
<code>netmask <address></code>	Set the IP network mask for this interface.
<code>dstaddr <address></code>	Set the remote IP address for a point-to-point link (such as PPP).
<code>metric <NN></code>	This parameter sets the interface metric.
<code>mtu <NN></code>	This parameter sets the Maximum Transfer Unit (MTU) of an interface.
<code>trailers</code>	This flag used to cause a non-standard encapsulation of inet packets on certain link levels.
<code>arp</code>	Enable or disable the use of the ARP protocol on this interface.
<code>allmulti</code>	Enable or disable all-multicast mode. If selected, all multicast packets on the network will be received by the interface.
<code>multicast</code>	Set the multicast flag on the interface. This should not normally be needed as the drivers set the flag correctly themselves.

promisc	Enable or disable the promiscuous mode of the interface. If selected, all packets on the network will be received by the interface.
txqueuelen <NN>	Set the length of the transmit queue of the device.
up down	This flag causes the interface to be activated. This flag causes the driver for this interface to be shut down.

Table 23: ifconfig options

Examples:



View the status of all interfaces.

ifconfig



Activation of loopback with IP address 127.0.0.1/8

ifconfig lo up



Activation of virtual interface eth0:0 with IP address 192.168.2.1/24

ifconfig eth0:0 192.168.2.1 netmask 255.255.255.0 up

1.26. io

Description:

The program is used to control outputs and read inputs. Supports reading state of binary outputs and setting state of counters.

Synopsis:

io [get <pin>] | [set <pin> <value>]

Options :

Option	Description
get	Set output.
set	Determine state of input.

Table 24: io options

Examples:



Set the state of binary output OUT0 to 1.

io set out0 1



Determine the state of digital input BIN0.

io get bin0



Determine the state of analog input AN1 on expansion port XC-CNT.

io get an1



Determine the state of counter input CNT1 on expansion port XC-CNT.

io get cnt1

1.27. ip

Description:

This command is used to configure a network interface.

Synopsis:

ip [-f] [-o] { address | link | route | tunnel } <command>

Options :

Option	Description
-f	Followed by protocol family identifier: inet, inet6 or link.
-o	Output each record on a single line, replacing line feeds with the '?' character.

Table 25: IP options

Option	Description
link	Show or manipulate with network device.
address	Show or manipulate with protocol IP or IPv6 address on device.
route	Show or manipulate with routing table.
tunnel	Show or manipulate tunnel over IP.

Table 26: IP object



For detail description of all commands command, visit Linux manual pages.

Examples:



View the status of all interfaces.

```
ip link show
```



View the route table.

```
ip route list
```



Add routing networks 192.168.3.0/24 through interface eth0.

```
ip route add 192.168.3.0/24 dev eth0
```



Add routing IP address 192.168.3.1 through gateway 192.168.1.2.

```
ip route add 192.168.3.1 via 192.168.1.2
```



Add default gateway 192.168.1.2.

```
ip route add default via 192.168.1.2
```

1.28. iptables

Description:

This command is used to administration tool for IP packet filtering and NAT.

Synopsis:

```
iptables [<options>]
```

Options:



For detail description this command, visit Linux manual pages.

Examples:



Redirect incoming TCP connections to port 8080 on IP address 192.168.1.2 and port 80.

```
iptables -t nat -A napt -p tcp --dport 8080 -j DNAT --to-destination 192.168.1.2:80
```

1.29. kill

Description:

This command is used to terminate process.

Synopsis:

```
kill [ -<signal> ] <process-id> [ <process-id> ... ]
```

```
kill -l
```

Options :

Option	Description
-l	Print a list of signal names. These are found in /usr/include/linux/signal.h

Table 27: kill options

Examples:



End the process with PID 1234 by sending signal SIGTERM.
kill 1234



End the process with PID 1234 by sending signal SIGKILL.
kill -9 1234

1.30. *killall*

Description:

This command is used to kill all process with process name.

Synopsis:

killall [-q] [-<signal>] <process-name> [<process-name> ...]

Options :

Option	Description
-l	Print a list of signal names. These are found in /usr/include/linux/signal.h
-q	Do not complain if no processes were killed

Table 28: killall options

Examples:



End the all processes with name pppd by sending signal SIGTERM.
killall pppd



End the all processes with name pppd by sending signal SIGKILL.
killall -9 pppd

1.31. *led*

Description:

The program used to control the USB LED on the front panel of the router.

Synopsis:

led [on | off]

Options :

Option	Description
on	User LED is on.
off	User LED is off.

Table 29: led options

Examples:



Turn on USB LED.
led on



Turn off USB LED.
led off

1.32. *ln*

Description:

The program used to make links between files.

Synopsis:

`ln [option] < target >... < link_name > | < directory >`

Options :

Option	Description
-s	Make symbolic links instead of hard links.
-f	Remove existing destination files.
-n	No dereference symlinks - treat like normal file.
-b	Make a backup of the target (if exists) before link operation.
-S	Use suffix instead of ~ when making backup files.

Table 30: *ln* options

Examples:

Creating a symbolic link to file `/var/log/messages` called `my.log`.

`ln -s /var/log/messages my.log`



1.33. *logger*

Description:

The program makes entries in the system log. It provides a shell command interface to the system log module.

Synopsis:

`logger [option] [message ...]`

options :

Option	Description
-i	Log the process id of the logger process with each line.
-s	Log the message to standard error, as well as the system log.
-f <i>file</i>	Log the specified file.
-p <i>pri</i>	Enter the message with the specified priority. The priority may be specified numerically or as a facility.level pair.
-t <i>tag</i>	Mark every line in the log with the specified <i>tag</i>
-u <i>sock</i>	Write to socket as specified with <i>socket</i> instead of builtin syslog routines.
-d	Use a datagram instead of a stream connection to this socket.

Table 31: *logger* options

Examples:

Send the message `System rebooted` to the syslogd daemon:

`logger System rebooted`



Send the message `System going down immediately!!!` to the syslog daemon, at the emerg level and user facility:

`logger -p user.emerg "System going down immediately!!!"`



1.34. *ls*

Description:

The program used to list directory contents.

Synopsis:

`ls [option] < filename > ...`

Options :

Option	Description
-l	List files in a single column
-A	Do not list implied . and ..
-a	Do not hide entries starting with .
-C	List entries by columns
-c	With -l: show ctime
-d	List directory entries instead of contents
-e	List both full date and full time
-i	List the i-node for each file
-l	Use a long listing form
-n	List numeric UIDs and GIDs instead of names
-L	List entries pointed to by symbolic links
-r	Sort the listing in reverse order
-S	Sort the listing by file size
-s	List the size of each file, in blocks
-t	With -l: show modification time
-u	With -l: show access time
-v	Sort the listing by version
-x	List entries by lines instead of by columns
-X	Sort the listing by extension

Table 32: *ls* options

Examples:

View detailed content of directory `/mnt`.
`ls -l /mnt`



View list contents of actually directory.
`ls`

1.35. *mac*

Description:

The program used to display the MAC address of `eth0`.

Synopsis:

`mac [<separator>]`

Examples:

Display the MAC address of `eth0`. Will be used as the separator character "-" instead of ":".
`mac -`

1.36. *mkdir*

Description:

This program used to make directories.

Synopsis:

`mkdir [<option>] directory ...`

Options :

Option	Description
-m	Set permission mode (as in chmod), not <code>rwxrwxrwx - umask</code> .
-p	No error if existing, make parent directories as needed.

Table 33: mkdir options

Examples:

Create directory `/tmp/test/example`.
`mkdir -p /tmp/test/example`



1.37. *mount*

Description:

This program used to mount a file system.

Synopsis:

`mount [-a] [-o] [-r] [-t] [-w] <DEVICE> <NODE> [-o <option>, ...]`

Options :

Flags	Description
-a	Mount all filesystems in fstab
-o	One of many filesystem options, listed below
-r	Mount the filesystem read-only
-t	Specify the filesystem type
-w	Mount for reading and writing (default)

Table 34: mount flags

Option	Description
async/sync	Writes are asynchronous / synchronous
atime/noatime	Enable / disable updates to inode access times
dev/nodev	Allow use of special device files / disallow them
exec/noexec	Allow use of executable files / disallow them
suid/nosuid	Allow set-user-id-root programs / disallow them
remount	Re-mount a mounted filesystem, changing its flags
ro/rw	Mount for read-only / read-write
bind	Bind a directory to an additional location
move	Relocate an existing mount point.

Table 35: mount options

For detail description this command, visit Linux manual pages.

Examples:

Connect a contents of USB flash drive to the directory `/mnt`.
`mount -t vfat /mnt /dev/sda1`



1.38. *mv*

Description:

This program is used to move or rename files.

Synopsis:

```
mv [-f] [-i] <source> ... <dest>
```

Options :

Option	Description
-f	Don't prompt before overwriting.
-i	Interactive, prompt before overwrite.

Table 36: mv options

Examples:

Rename file abc.txt na def.txt.
mv abc.txt def.txt



Move all files with the extension txt to the directory /mnt.
*mv *.txt /mnt*

1.39. *ntpdate*

Description:

The program is used to set the system time from NTP server.

Synopsis:

```
ntpdate [-p <probes>] [-t <timeout>] <server>
```

Options :

Option	Description
-p	Specify the number of samples to be acquired from each server as the integer samples, with values from 1 to 8 inclusive.
-t	Specify the maximum time waiting for a server response as the value timeout, in seconds and fraction.

Table 37: ntpdate options

Examples:

Set the system time according to the NTP server time.windows.com.
ntpdate time.windows.com

1.40. openssl

Description:

The openssl program is a command line tool for using the various cryptography functions of OpenSSL's crypto library from the shell. It can be used for:

- Creation of RSA, DH and DSA key parameters
- Creation of X.509 certificates, CSRs and CRLs
- Calculation of Message Digests
- Encryption and Decryption with Ciphers
- SSL/TLS Client and Server Tests
- Handling of S/MIME signed or encrypted mail

Synopsis:

openssl [<option> ...]

Options:

For detail description this command, visit Linux manual pages.

Examples:

Generate a new key for the SSH server.

```
openssl genrsa -out /etc/certs/ssh_rsa_key 512
```

Generate a new certificate for the HTTPS server.

```
openssl req -new -out /tmp/csr -newkey rsa:1024 -nodes -keyout /etc/certs/https_key
openssl x509 -req -setstart 700101000000Z -setend 400101000000Z -in /tmp/csr -signkey
/etc/certs/https_key -out /etc/certs/https_cert
```

1.41. passwd

Description:

This program is used to change password for user root.

Synopsis:

passwd

1.42. ping

Description:

This program is used to send ICMP echo request to network host.

Synopsis:

ping [-c <count>] [-s <size>] [-q] <hosts>

Options :

Option	Description
-c	Send only COUNT pings
-s	Send SIZE data bytes in packets (default=56)
-q	Quiet mode, only displays output at start and when finished
-i	Selects outgoing interface

Table 38: ping options

Examples:

Send one ICMP packet Echo Request with size 500 B on IP address 10.0.0.1.

```
ping -c 1 -s 500 10.0.0.1
```

1.43. *portd*

Description:

The program is used for transparent transfer of data from the serial line by protocol TCP or UDP.

Synopsis:

```
portd -c <device> [-b <baudrate>] [-d <databits>] [-p <parity>] [-s <stopbits>] [-l <split timeout>] [-4] [-h <hostname>] [-o <proto>] -t <port> [-k <keepalive time>] [-i <keepalive interval>] [-r <keepalive probes>] [-x] [-z]
```

Options :

Option	Description
-c	Serial line device
-b	Baudrate
-d	Number of data bits
-p	Parity – even, odd or none.
-s	Number of stop bits.
-l	Split timeout.
-4	Forced detection Expansion port 485.
-h	Hostname
-o	Protocol TCP or UDP.
-t	TCP or UDP port.
-k	Keepalive time.
-i	Keepalive interval.
-r	Keepalive probes.
-x	Use signal CD as indicator of the TCP connection.
-z	Use DTR as control TCP connection.

Table 39: portd options

Examples:

Running a TCP server listening on port 1000th After a TCP connection, the program transparently transmit data from the serial port settings 115200 bit / s, 8N1.

```
portd -c /dev/ttyS0 -b 115200 -t 1000 &
```

1.44. *ps*

Description:

This program is used to view report process status.

Synopsis:

```
ps
```

1.45. *pwd*

Description:

This program used to view current directory.

Synopsis:

```
pwd
```

1.46. *reboot*

Description:

This program is used to reboot the router.

Synopsis:

`reboot [-d <delay>] [-n <nosync>] [-f <force>]`

Options :

Option	Description
-d	Delay interval for rebooting
-n	No call to sync()
-f	Force reboot, do not call shutdown.

Table 40: reboot options

Examples:

Reboot router after 10 second.

`reboot -d 10`



1.47. *restore*

Description:

This program is used to restore configuration from file.

Synopsis:

`restore <filename>`

Examples:

Restore configuration from file /tmp/my.cfg

`restore /tmp/my.cfg`



1.48. *rm*

Description:

This program is used to remove files or directories.

Synopsis:

`rm [-i] [-f] [-r] <file>...`

Options :

Option	Description
-i	Always prompt before removing each destination.
-f	Remove existing destinations, never prompt.
-r	Remove the contents of directories recursively.

Table 41: rm options

Examples:

Remove all files with extension txt in the current directory.

`rm *.txt`



Remove directory /tmp/test and all subdirectories.

`rm -rf /tmp/test`



1.49. *rmdir*

Description:

This program is used to remove empty directories.

Synopsis:

`rmdir <filename>`

Examples:

Remove empty directory `/tmp/test`.
`rmdir /tmp/test`



1.50. *route*

Description:

This program is used to show and manipulate the IP routing table.

Synopsis:

`route [-n] [-e] [-A] [add | del | delete]`

Options :

Option	Description
-n	Dont resolve names.
-e	Display other/more information.
-A	Select address family.

Table 42: route options



For detail description this command, visit Linux manual pages.

Examples:

View the routing table without translating IP addresses to domain names.
`route -n`



Add routing networks `192.168.3.0/24` through `eth0`.
`route add -net 192.168.3.0/24 dev eth0`



Add routing IP addresses `192.168.3.1` through `192.168.1.2` gateway.
`route add -host 192.168.3.1 gw 192.168.1.2`



Add default gateway `192.168.1.2`
`route add default gw 192.168.1.2`



1.51. sed

Description:

This program is used for filtering and transforming text.

Synopsis:

```
sed [-e] [-f] [-i] [-n] [-r] pattern [-files]
```

Options :

Option	Description
-e	Add the script to the commands to be executed.
-f	Add script-file contents to the commands to be executed.
-i	Edit files in place (makes backup if extension supplied).
-n	Suppress automatic printing of pattern space.
-r	Use extended regular expression syntax.

Table 43: sed options



If no -e or -f is given, the first non-option argument is taken as the sed script to interpret. All remaining arguments are names of input files; if no input files are specified, then the standard input is read. Source files will not be modified unless -i option is given.

Examples:



Change parameter PPP_APN in file /etc/settings.ppp to value "internet".
`sed -e "s/(PPP_APN=).*\1internet/" -i /etc/settings.ppp`

1.52. service

Description:

This program is used to start, stop or restart specified service.

Synopsis:

```
service < service name > <start | stop | restart>
```

Examples:



Start service cron.
`service cron start`



Restart service ppp.
`service ppp restart`

1.53. sleep

Description:

This program is used to delay for a specified amount of time.

Synopsis:

```
sleep <time>
```

Examples:



Pause for 30 second.
`pause 30`

1.54. *slog*

Description:

This script used to show system log (file /var/log/message).

Synopsis:

`slog [-n <number>] [-f]`

Options :

Option	Description
-n	Print last N lines instead of last 10.
-f	Output data as the file grows.

Table 44: slog options

Examples:

Continuous listing the system log. Listing stops when reaching the maximum number of lines of log.

`slog -f`



1.55. *snmptrap*

Description:

This program is used to sending SNMP trap.

Synopsis:

`snmptrap [-c <community>] [-g <generic>] [-s <specific>] <hostname> [<oid> <type> <value>]`

Options :

Option	Description
-c	Community
-g	Specifies generic trap types <ul style="list-style-type: none"> • 0 - coldStart • 1 - warmStart • 2 - linkDown • 3 - linkUp • 4 - authenticationFailure • 5 - egpNeighborLoss • 6 - enterpriseSpecific
-r	Sends MAC address of eth0 interface
-s	Specifies user definition trap types in the enterpriseSpecific

Table 45: snmptrap options

Examples:

Send TRAP with information about the status of a digital input BIN0 to the IP address 192.168.1.2.

`snmptrap 192.168.1.2 1.3.6.1.4.1.30140.2.3.1.0 u `io get bin0``



Send TRAP "warm start" to the IP address 192.168.1.2.

`snmptrap -g 1 192.168.1.2`



1.56. *tail*

Description:

This program is used to output the last part of files.

Synopsis:

```
tail [ -n <number> ] [ -f ]
```

Options :

Option	Description
-n	Print last N lines instead of last 10.
-f	Output data as the file grows.

Table 46: tail options

Examples:

Show last 30 lines of /var/log/messages.

```
tail -n 30 /var/log/messages
```



1.57. *tar*

Description:

This program is used to create, extract or list files from a tar file.

Synopsis:

```
tar -[czxtv0] [ -f tarfile ] [ -C dir ] [ file ] ...
```

Options :

Option	Description
c	Create
x	Extract
t	List
z	Filter the archive through gzip.
-f	Name of TARFILE or "-" for stdin
0	Extract to stdout.
-C	Change to directory DIR before operation.
v	Verbosely list files processed.

Table 47: tar options

Examples:

Creating log.tar archive that contains files from the directory /var/log.

```
tar -cf log.tar /var/log
```



Extract files from the archive log.tar.

```
tar -xf log.tar
```



1.58. *tcpdump*

Description:

This program is used to dump traffic on a network.

Synopsis:

```
tcpdump [-AdDeflLnNOpqRStuUvxX] [-c <count>] [--C <file size>] [-E algo:secret] [-F <file>] [-i <interface>] [-r <file>] [-s <snaplen>] [-T type] [-w <file>] [-y <datalinktype>] [expression]
```

Options:



For detail description this command, visit Linux manual pages.

Examples:



View traffic on interface ppp0.

```
tcpdump -n -i ppp0
```



View traffic on interface eth0 except protocol Telnet.

```
tcpdump -n not tcp port 23
```



View UDP traffic on interface eth0.

```
tcpdump -n udp
```



View HTTP traffic on interface eth0.

```
tcpdump -n tcp port 80
```



View all traffic from/to IP address 192.168.1.2.

```
tcpdump -n host 192.168.1.2
```



View traffic from/to IP address 192.168.1.2 except protocol Telnet.

```
tcpdump -n host 192.168.1.2 and not tcp port 23
```

1.59. *telnet*

Description:

This program is used to establish interactive communication with another computer over a network using the TELNET protocol.

Synopsis:

```
telnet <host> [<port>]
```

Examples:



Connect to 192.168.1.2 by protocol Telnet.

```
telnet 192.168.1.2
```

1.60. touch

Description:

This program used to update timestamp of file.

Synopsis:

touch [-c] <file> [<file> ...]

Options :

Option	Description
-c	Do not create any files.

Table 48: touch options

Examples:

Create a file, respectively update timestamp of file /tmp/test.

touch /tmp/test



1.61. traceroute

Description:

This program is printed the route packets trace to network host.

Synopsis:

traceroute [-F|dnrv] [-f <1st_ttl>] [-m <max_ttl>] [-p <port#>] [-q <nqueries>] [-s <src_addr>] [-t <tos>] [-w <wait>] [-g <gateway>] [-i <iface>] [-z <pausesecs>] host [data size]

Options :

Option	Description
-F	Set the don't fragment bit
-l	Use ICMP ECHO instead of UDP datagrams
-l	Display the ttl value of the returned packet
-d	Enable socket level debugging
-n	Print hop addresses numerically rather than symbolically
-r	Bypass the normal routing tables and send directly to a host
-v	Verbose output
-m	Set the max time-to-live (max number of hops)
-p	Set the base UDP port number used in probes (default is 33434)
-q	Set the number of probes per ``ttl" to nqueries (default is 3)
-s	Use the following IP address as the source address
-t	Set the type-of-service in probe packets to the following value (default 0)
-w	Set the time (in seconds) to wait for a response to a probe (default 3 sec)
-g	Specify a loose source route gateway (8 maximum)

Table 49: traceroute options

Examples:

Start traceroute on IP address 10.0.0.1 (without translation IP addresses to domain names).

traceroute -n 10.0.0.1



1.62. *umount*

Description:

This program is used to unmount file systems.

Synopsis:

`umount [-a] [-r] [-l] [-f] <file system> | <directory>`

Options :

Option	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).

Table 50: unmount options

Examples:

Disconnecting the disc connected to the directory /mnt.

`umount /mnt`

1.63. *vi*

Description:

This program is used to edit and read text file.

Synopsis:

`vi [-R] [<file> ...]`

Options :

Option	Description
-R	Read only, do not write to the file.

Table 51: vi options

Option:

Open file /etc/rc.local in the text editor vi.

`vi /etc/rc.local`

1.64. wget

Description:

This program is used to retrieve files via HTTP or FTP.

Synopsis:

```
wget [-c] [-q] [-O <document file>] [--header 'header: value'] [-Y on/off] [-P <DIR>] <url>
```

Options :

Option	Description
-c	Continue retrieval of aborted transfers
-q	Quiet mode - do not print
-P	Set directory prefix to DIR
-O	Save to filename ('-' for stdout)
-Y	Use proxy ('on' or 'off')

Table 52: wget options

Examples:



Download a file my.cfg from HTTP server with IP address 10.0.0.1.
`wget http://10.0.0.1/my.cfg`



2. Script examples

2.1. Send SMS

Send incoming SMS to the email.

Startup Script

```
EMAIL=john.doe@email.com
cat > /var/scripts/sms << EOF
#!/bin/sh
/usr/bin/email -t $EMAIL -s "Received SMS from \2" -m "Authorized: \1, Text: \3 \4 \5 \6 \7 \8"
EOF
```

2.2. SMS command 1

Implementations of new SMS command "IMPULSE", which activates binary output OUT0 for 5 second. SMS will be processed, if it comes from one of three numbers defined on the web interface or phone number +420123456789.

Startup Script

```
PHONE=+420123456789
cat > /var/scripts/sms << EOF
#!/bin/sh
if [ "\1" = "1" ] || [ "\2" = "$PHONE" ]; then
  if [ "\3" = "IMPULSE" ]; then
    /usr/bin/io set out0 1
    sleep 5
    /usr/bin/io set out0 0
  fi
fi
EOF
```

2.3. SMS command 2

This script implementations new SMS command "PPP", which sets item Network type, Default SIM card and Backup SIM card. PPP command has follow structure PPP <AUTO/GPRS/UMTS> <1/2>. First parameter sets network type. If second parameter is 1, Default SIM card sets as primary SIM card. If second parameter is 2, Default SIM card sets as secondary SIM card.

Startup Script

```
cat > /var/scripts/sms << EOF
STARTUP=#!/bin/sh
if [ "$1" = "1" ]; then
  if [ "$3" = "PPP" ]; then
    if [ "$4" = "AUTO" ]; then
      sed -e "s^(PPP_NETTYPE=\\).*^10/" -e "s^(PPP_NETTYPE2=\\).*^10/" -i /etc/settings.ppp
    elif [ "$4" = "GPRS" ]; then
      sed -e "s^(PPP_NETTYPE=\\).*^11/" -e "s^(PPP_NETTYPE2=\\).*^11/" -i /etc/settings.ppp
    elif [ "$4" = "UMTS" ]; then
      sed -e "s^(PPP_NETTYPE=\\).*^12/" -e "s^(PPP_NETTYPE2=\\).*^12/" -i /etc/settings.ppp
    fi
  if [ "$5" = "1" ]; then
    sed -e "s^(PPP_DEFAULT_SIM=\\).*^11/" -e "s^(PPP_BACKUP_SIM=\\).*^12/" -i /etc/settings.ppp
  elif [ "$5" = "2" ]; then
    sed -e "s^(PPP_DEFAULT_SIM=\\).*^12/" -e "s^(PPP_BACKUP_SIM=\\).*^11/" -i /etc/settings.ppp
  fi
  reboot
fi
fi
EOF
```

2.4. Send information email 1

Send information email about establishing of PPP connection.

Up Script

```
EMAIL=john.doe@email.com
/usr/bin/email -t $EMAIL -s "Router has established PPP connection. IP address: $4"
```

2.5. Send information SNMP trap 1

Send information SNMP trap about establishing of PPP connection.

Up Script

```
SNMP_MANAGER=192.168.1.2
/usr/bin/snmptrap -g 3 $SNMP_MANAGER
```

2.6. Send information email 2

Send information email about switch binary input BIN0.

Startup Script

```
EMAIL=john.doe@email.com  
MESSAGE="BIN0 is active"
```

```
while true  
do  
  /usr/bin/io get bin0  
  VAL=$?  
  if [ "$VAL" != "$OLD" ]; then  
    [ "$VAL" = "0" ] && /usr/bin/email -t $EMAIL -s "$MESSAGE"  
    OLD=$VAL  
  fi  
  sleep 1  
done
```

2.7. Send information SNMP trap 2

Send information SNMP trap about change state of binary input BIN0.

Startup Script

```
SNMP_MANAGER=192.168.1.2
```

```
while true  
do  
  /usr/bin/io get bin0  
  VAL=$?  
  if [ "$VAL" != "$OLD" ]; then  
    /usr/bin/snmptrap $SNMP_MANAGER 1.3.6.1.4.1.30140.2.3.1.0 u $VAL  
    OLD=$VAL  
  fi  
  sleep 1  
done
```

2.8. Automatic reboot

Automatic reboot at the definition time. (23:55)

Startup Script

```
echo "55 23 * * * root /sbin/reboot" > /etc/crontab  
service cron start
```

2.9. Switch between WAN and PPP

Switching between WAN and PPP. PPP connection is active, if PING on the defined IP address does not pass through.

Startup Script

```
WAN_PING=192.168.2.1
WAN_GATEWAY=192.168.2.1
WAN_DNS=192.168.2.1
```

```
./etc/settings.eth
```

```
/sbin/route add $WAN_PING gw $WAN_GATEWAY
/sbin/iptables -t nat -A PREROUTING -i eth1 -j napt
/sbin/iptables -t nat -A POSTROUTING -o eth1 -p ! esp -j MASQUERADE
```

```
LAST=1
while true
do
ping -c 1 $WAN_PING
PING=$?
if [ $PING != $LAST ]; then
LAST=$PING
if [ $PING = 0 ]; then
/etc/init.d/ppp stop
sleep 3
/sbin/route add default gw $WAN_GATEWAY
echo "nameserver $WAN_DNS" > /etc/resolv.conf
/usr/sbin/contrack -F
/etc/scripts/ip-up - - - $ETH2_IPADDR
else
/etc/scripts/ip-down - - - $ETH2_IPADDR
/usr/sbin/contrack -F
/sbin/route del default gw $WAN_GATEWAY
/etc/init.d/ppp start
fi
fi
sleep 1
done
```