A deeper journey into MikroTik routers v2

Now with 53% unseen content

possible.lv
RouterOS v6.37

You have connected to a router. Administrative access only. If this device is not in your possession, please contact your local network administrator.

WebFig Login:

Login: admin
Password: 

Authentication failed: invalid username or password.

- Winbox
- Telnet
- Graphs
- License
- Help

It's good that you are not in my hotel. Lol

— Tanoy Bose
Us

- Kirils – passionate about ... things and stuff
- Jānis – maker ... who sometimes breaks things
Goal of this research is to achieve the interoperability of computer programs (i.e. software running on MikroTik routers) with other computer programs.
ACK: Prior research

- “ANTONY++” from awmn.net
  - Initial NPK analysis

- “DRUBICZA”
  - NPK file unpacking

- Paul McCall
  - Initial supout.rif analysis

- OpenWRT team
  - Kernel config files
ACK: The team

- **Kirils Solovjovs**
  - Dynamic binary analysis, jailbreak scripts, internal file format analysis

- **Jānis Jansons**
  - Static binary analysis, webfig analysis, bootup sequence, testing

- **You?**
  - Interested in MikroTik sw/hw
  - Experience in Linux or reverse engineering
Content outline

- **RouterOS intro**
- **RouterOS boot process and binaries**
- **Package format**
- **supout.rif**
- **Config files**
- **Lots of demos, of course!**
Who uses MikroTik?

1. Brazil
2. Indonesia
3. Russian Federation
4. China
5. United States
6. Iran
7. Poland
8. India
9. Czech Republic
10. Ukraine
Do they update?
RouterOS externals
RouterOS is ...

- **LINUX (KERNEL) + STARTUP SCRIPTS + NOVÅ BINARIES + CONFIG**
GPL to the rescue?

HTTPS://GITHUB.COM/WSXARCHER/ROUTEROS-LINUX-PATCH
History: the beginnings

• 1999
  – MikroTik™ v2.0 Router Software
    • Initial release
    • Works on 486
    • Upgrades available as packages

• 2001
  – MikroTik™ v2.3 Router Software
    • NPK first mentioned as method for extending functionality
History: not just x86

- **12 Feb 2004**
  - **MikroTik RouterOS™ V2.8**
    - SOFTWARE KEY SYSTEM CHANGED
    - HAS NOT BEEN CHANGED SINCE!

- **1 Aug 2005**
  - **MikroTik RouterOS™ V2.9**
    - FIRST NEW ARCHITECTURE INTRODUCED
      - mipsel for RB500
History: “backdoor”

- 15 Nov 2005
  - 2.9.8
    - A WILD “/NOVA/ETC/DEVEL-LOGIN” APPEARS IN /NOVA/BINLOGIN
    - [ -f /NOVA/ETC/DEVEL-LOGIN &&
      USERNAME == DEVEL && PASSWORD ==
      ADMIN.PASSWORD ] && /BIN/ASH
    - FUN FACT: PREVIOUSLY USERNAME WAS “BASH”
History: the ghost & signing

- **8 Feb 2009**
  - 3.21
    - WHAT’S UP WITH THIS VERSION?
    - WHY HAS IT VANISHED FROM THE INTERNET?

- **16 Mar 2009**
  - 3.22
    - NPK VERIFICATION AND SIGNING ADDED
    - CHECKSUM AND SIGNATURE CHECKED BY /NOVA/BIN/INSTALLER
    - NO MORE FREE LUNCHES
History: SquashFS in NPK

- **7 May 2013**
  - **6.0 (SINCE BETA3)**
    - SquashFS employed in npk files
    - Zerofill blocks added
      - So that actual SquashFS start is located at addresses divisible by 4096

- **6 Nov 2015**
  - 6.33
    - Packages now include distribution channel
      - Bugfix | Current | Development | Release-candidate
¿Development branch?

GET /routeros/LATEST.7 HTTP/1.1
Host: upgrade.mikrotik.com
User-Agent: RouterOS 6.40

HTTP/1.1 403 Forbidden
Content-Type: application/xml

[admin@MikroTik] /system package update> print
  channel: current
  installed-version: 6.40
[admin@MikroTik] /system package update> set channel=development
[admin@MikroTik] /system package update> check-for-updates
  channel: development
  current-version: 6.40
  status: ERROR: file not found

<?xml version="1.0" encoding="UTF-8"?>
<Error><Code>AccessDenied</Code><Message>Access Denied</Message><RequestId>20C70</RequestId><HostId>U+NhxKind+0jlzr52scm3d80</HostId></Error>
RouterOS ecosystem revisited
DEMO: RouterOS console
RouterOS command tree

- xviewer memory requirements (.png)
- /ip 3.7 GiB
- /interface 3.5 GiB
- /routing 2.1 GiB
- /tool 1.9 GiB
- /system 1.2 GiB
- /caps-man 1.1 GiB
- /ipv6 0.9 GiB
Example: /log command

- `log`
  - `get`
    - `value-name`
      - `message`
      - `buffer`
      - `topics`
  - `number`
    - `print`
      - `without-paging`
        - `message`
        - `follow-only`
        - `count-only`
        - `value-list`
        - `as-value`
      - `append`
        - `interval`
        - `where`
        - `follow`
        - `detail`
      - `terse`
        - `brief`
        - `file`
      - `info`
        - `debug`
        - `warning`
      - `error`
        - `message`
      - `find`
        - `message`
        - `nextid`
        - `buffer`
        - `topics`
        - `dead`
      - `time`
      - `id`
RouterOS internals
RouterOS boot process

Kernel -> /init -> /etc/rc.d/rc.start -> /etc/rc.d/run.d/S...

/etc/rc.d/run.d/S10nova -> /nova/bin/loader

/nova/bin/loader -> /nova/bin/sys2

/nova/bin/loader -> /nova/bin/sermgr

/nova/bin/loader -> /nova/bin/www

/nova/bin/loader -> ../nova/bin/...
A nice feature for jailbreakers...

```bash
#!/bin/bash
[echo "Starting..."
/etc/rc.d/rc.sysinit || exit 1
export PATH=$(path --colon /sbin):$(path --colon /bin)
export LD_LIBRARY_PATH=/rw/lib:$(path --colon /lib)

# disable core files
ulimit -c 0

# set maximum opened files per process
ulimit -n 500000

# start syslog daemon
for i in $(path --prefix S /etc/rc.d/run.d); do
  if [ -f $i ]; then
    $i || exit 1
  fi
done
exit 0

/etc/rc.d/rc.start
```
A nice feature for jailbreakers...

- "PATH" looks for specified path in prefixed directories
  - Used throughout their scripts
  - Makes using custom scripts easier

```
# path --colon /etc/rc.d/
/pckg/dude/etc/rc.d/:/pckg/ntp/etc/rc.d/:/pckg/ups/etc/rc.d/:/pckg/ipv6/etc/rc.d/:/pckg/security/etc/rc.d/:/etc/rc.d/
# path --colon /bin/
/flash//bin/:/pckg/kvm/bin/:/bin/
```
RouterOS boot process

- Kernel
- /init
- /etc/rc.d/rc.start
  - /etc/rc.d/run.d/S...
  - /etc/rc.d/run.d/S10nova
    - /nova/bin/loader
      - /nova/bin/sys2
      - /nova/bin/sermgr
      - /nova/bin/www
      - ..//nova/bin/...
nova binaries

• **LOADER**
  – **Spawns processes and manages communication between them**

• **WATCHDOG**
  – **Restarts the device if a critical process stops working**

• **SYS2**
  – **Manages device settings and parses received commands**

• **SERMGR**
  – **Super-server daemon that provides internet services**
sermgr ≈ inetd
nova binaries

- **NET**
  - **Deals with network configuration, tunnels, AT commands**

- **MODULER**
  - **Manages loading of firmware for external devices**
    - e.g. USB2SERIAL adaptors, 3G modems

- **MODPROBED**
  - **Symlink to moduler, used for loading kernel modules**

- **MANAGER**
  - **User and group management**
nova binaries

- **LOG**
  - **LOG DAEMON**

- **MPROXY**
  - **WINBOX DAEMON**

- **QUICKSET**
  - **SEPARATE DAEMON FOR MANAGEMENT OF QUICKSET SETTINGS**

- **UNDO**
  - **SAFE MODE SUPPORT**

- **WWW**
  - **WEB INTERFACE DAEMON**
Package format
NPK format

- **Numeric values are unsigned little endian**
- **File consists of header, file size, parts and footer.**
- **File size is 8b less**
- **Each part consist of:**
  - **Part type (short)**
  - **Payload size (long)**
  - **Payload**

```
| FB | 0F | 10 | A1 | 1F | 01 | 00 | 00 | 01 | 00 | 20 | 00 | 00 | 00 | 72 | 65 | 73 | 74 | 72 | 69 |
| 63 | 74 | 69 | 6F | 6E | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 06 | D9 | B4 | 82 | 59 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 02 | 00 | 27 | 00 | 00 | 00 | 50 | 72 |
| 6F | 76 | 69 | 64 | 65 | 73 | 20 | 72 | 65 | 73 | 74 | 72 | 69 | 63 | 74 | 65 | 64 | 20 | 76 | 65 |
| 72 | 73 | 69 | 6F | 6E | 20 | 6F | 66 | 20 | 72 | 6F | 75 | 74 | 65 | 72 | 6F | 73 | 03 | 00 | 02 |
| 00 | 00 | 00 | 00 | 00 | 04 | 00 | 68 | 00 | 00 | 00 | 78 | 9C | 7B | EB | CA | 00 | 06 | F7 | FF |
| 5B | 07 | 33 | A4 | 31 | B0 | DD | DC | D2 | 14 | C9 | 00 | 05 | 2C | 0C | 79 | F9 | 65 | 89 | 6F |
| A1 | 2A | 5E | 62 | 51 | C1 | 01 | 56 | A1 | 9F | 93 | 99 | 04 | 53 | B5 | B2 | 0B | 28 | 8B | A6 |
| 4A | 10 | AE | 4A | BF | 3C | B3 | 28 | 35 | 27 | B5 | B8 | 78 | 49 | 23 | 76 | E5 | 8C | 40 | 2C |
| 85 | A9 | 5C | 3F | AD | 2B | B5 | 30 | 37 | 3F | 25 | D5 | 08 | 97 | 46 | 26 | 20 | 96 | C4 | A2 |
| 31 | 31 | AF | 24 | 3D | 31 | 33 | CF | D0 | 12 | 00 | 99 | 5D | 3F | 86 | 09 | 00 | 44 | 00 | 00 |
| 00 | 20 | F1 | 64 | 5E | 73 | 76 | 2A | A2 | 95 | BF | 93 | 84 | F2 | BA | BA | 73 | F0 | 02 | 08 |
| 44 | EC | 3A | 17 | 29 | BD | D8 | BA | A3 | 94 | 49 | 1B | 30 | 66 | 82 | 84 | A6 | 8A | BC | 06 |
| 24 | A2 | BD | E4 | 9A | C0 | 6D | EC | F9 | 25 | 80 | C3 | C9 | B3 | 85 | BD | 3F | 6E | E3 | EB |
```

possible.lv

@KirilsSolovjovs
@Janamaja
NPK format

- **At least two types of current NPKs:**
  - **Package**
    - 0..3 header 1E F1 D0 BA
    - Footer 10 00 01 00 00 00 49
      - Footer since 3.22
  - **Restriction (invisible package)**
    - 0..3 header FB 0F 10 A1
    - Footer 03 00 00 00 00 00
# Part types

<table>
<thead>
<tr>
<th>N</th>
<th>Type</th>
<th>Meaning</th>
<th>First seen</th>
<th>Last seen</th>
<th>Mandatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00</td>
<td>Part info</td>
<td>forever</td>
<td>now</td>
<td>yes</td>
</tr>
<tr>
<td>2</td>
<td>00</td>
<td>Part description</td>
<td>forever</td>
<td>now</td>
<td>yes</td>
</tr>
<tr>
<td>3</td>
<td>00</td>
<td>Dependencies</td>
<td>forever</td>
<td>now</td>
<td>yes</td>
</tr>
<tr>
<td>4</td>
<td>00</td>
<td>File container</td>
<td>forever</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>5</td>
<td>00</td>
<td>Install script (libinstall)</td>
<td>forever</td>
<td>2.7.xx</td>
<td>no</td>
</tr>
<tr>
<td>6</td>
<td>00</td>
<td>? Uninstall script (libinstall)</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>7</td>
<td>00</td>
<td>Install script (bash)</td>
<td>forever</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>8</td>
<td>00</td>
<td>Uninstall script (bash)</td>
<td>forever</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>9</td>
<td>00</td>
<td>Signature</td>
<td>3.22</td>
<td>now</td>
<td>yes</td>
</tr>
<tr>
<td>10</td>
<td>00</td>
<td>unused</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>11</td>
<td>00</td>
<td>unused</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>12</td>
<td>00</td>
<td>unused</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>13</td>
<td>00</td>
<td>unused</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>14</td>
<td>00</td>
<td>unused</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>15</td>
<td>00</td>
<td>unused</td>
<td>never</td>
<td>never</td>
<td>no</td>
</tr>
<tr>
<td>16</td>
<td>00</td>
<td>Architecture</td>
<td>2.9</td>
<td>now</td>
<td>yes</td>
</tr>
<tr>
<td>17</td>
<td>00</td>
<td>Package conflicts</td>
<td>3.14</td>
<td>3.22</td>
<td>no</td>
</tr>
<tr>
<td>18</td>
<td>00</td>
<td>Package info</td>
<td>2.9</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>19</td>
<td>00</td>
<td>Part features</td>
<td>2.9</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>20</td>
<td>00</td>
<td>Package features</td>
<td>2.9</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>21</td>
<td>00</td>
<td>SquashFS block</td>
<td>6.0beta3</td>
<td>now</td>
<td>package only</td>
</tr>
<tr>
<td>22</td>
<td>00</td>
<td>Zero padding</td>
<td>6.0beta3</td>
<td>now</td>
<td>no</td>
</tr>
<tr>
<td>23</td>
<td>00</td>
<td>Digest</td>
<td>6.30</td>
<td>now</td>
<td>package only</td>
</tr>
<tr>
<td>24</td>
<td>00</td>
<td>Channel</td>
<td>6.33</td>
<td>now</td>
<td>package only</td>
</tr>
</tbody>
</table>
supout.rif
What is supout.rif?

- **Support Output**
  - *Ridiculously Intricate Format*
  - or **RouteOS Information File**, maybe, idk ＿( ´•`)_／

```
[admin@MikroTik] > /system sup-output
  created: 1%
-- [Q quit|D dump|C-z pause]
```
supout.rif from outside

---BEGIN ROUTEROS SUPOUT SECTION
oVWy9sRHaAgHnjXuAAAgJAgB=
--END ROUTEROS SUPOUT SECTION
--BEGIN ROUTEROS SUPOUT SECTION
w9Wzt8Wd0BAecukSMFFS0/czNx8SRh8SM3UVog8TVBNyJz8SVBjBKROlmbeKYkxayFAAccOD1D==
--END ROUTEROS SUPOUT SECTION
--BEGIN ROUTEROS SUPOUT SECTION
sNGZ09WdjhGA4x58xZXUwdX9z1gc0HFC21QCxT/cPYe5KpETRhKzP3cTMvUUlVEzNVFyJ5UUQjcy
MvUVwEgSkTp5mnCmoJXAAasy1SOE=
--END ROUTEROS SUPOUT SECTION
supout.rif section decoding

- **SWAP BITS AROUND**
  - PER THREE BYTES

- **BASE64**

- **SECTION DECODES TO:**
  - NAME + ‘\0’ +
  - ZLIB_COMPRESSED_CONTENT
supout.rif section decoding

tribitmap=\[10,11,0,1,2,3,4,5,14,15,16,17,6,7,8,9,18,19,20,21,22,23,12,13\]

def tribit(content):
    result=""
    for i in xrange(0, len(content) - 1, 3):
        goodtrubit=0
        badtrubit=ord(content[i])*10000+ord(content[i+1])*100+ord(content[i+2])
        for mangle in tribitmap:
            goodtrubit = (goodtrubit<<1) + (1 if ((badtrubit & (0x800000>>mangle))>0) else 0)
        for move in [16,8,0]:
            result=result+chr((goodtrubit >> move)& 0xff)
    return result
supout.rif from inside

- **What does it contain?**
  - **Your whole configuration**
  - `/proc/ folder`
  - **Memory addresses**
  - **Your log**
  - **And more**

```bash
$ ls supout.rif_contents/
01_debug 16_arp 31_profile 46_wirelesselog
02_profile 17_ip 32_dhcp 47_bfd
03_proc 18_nexthop 33_neighbor 48_bgp
04_startup 19_route 34_dhcp6 49_mme
05_livetrace 20_user 35_license 50_mpls
06_resource 21_firewall 36_package 51_ntp-client
07_pci 22_firewall-stats 37_instchk 52_ospf
08_usb 23_bridge 38_oops 53_ppp
09_log 24_mesh 39_backtrace 54_ipsec
10_export 25_queue 40_store 55_health
11_interface 26_queue-packets 41_hotspot 56_poe-out
12_ethernet 27_queue-bytes 42_routerboard 57_lcdtouch
13_switch 28_queue-stats 43_webproxy
14_address 29_ippool 44_wireless
15_port 30_certificate 45_wirelessdump
```

@KirilsSolovjovs
@Janamaja
mikrotik.com has a reader ...

Supout.rif reader

Upload your supout.rif file

Browse...  No file selected.  Upload
... but it won’t show you everything
Demo: mikrotik.com XSS

Demo: decode_supout.py
Config file format
Configuration

- **IDX = index**
- **DAT = data**
IDX format

- **Record ID (long)**
  - if ID is 0xFFFFFFFF, field has no content
  - used for offsetting

- **Length (long)**

- **Separator (long)**
  - usually 0x05000000
DAT format

- **LENGTH (SHORT)**
- **M2 RECORD OF LENGTH**
  - Config ID (3 bytes)
  - Type (1 byte)
    - Content depends on type

```
btype ..........  
00000000, - boolean  
,,1,1,,, - M2 block (len = short)  
,,11,,,, - binary data block (len = short)  
,,,,,,1, - one byte  
,,,,,,1, - ???  
,,,,,1,, - ???  
,,,,11,,, - 128 bit int  
,,,,1,,, - int (four bytes)  
,,,,1,,, - long (8 bytes)  
,1,,,,, - string  
,1,,,,, - ??? unused? or long array of?  
1,,,,,,, - short array of
```

types (MT notation)
(CAPITAL X = list of x)

- a,A, (0x18) IPv6 address (or duid)
- b,B, bool
- M, multi
- q,Q, (0x10) big number
- r,R, (0x10) mac address
- s,S, (0x21) string
- u,U, unsigned integer
Peculiarities / features

- **Field IDs shared with web**
- **Winbox protocol derived from DAT format**
  - Working directly with files?
  - Dangerous!
function post(req, cb) {
    console.log(req);
    if (window.ArrayBuffer) {
        request('POST', '/jsproxy', session.encryptUint8Array(msg2buffer(req)), function(r) {
            session.decryptUint8Array(new Uint8Array(r), cb);
            session.dequeue();
            if (cb == receive) {
                post({}, receive);
            }
        }, null, true);
    } else {
        request('POST', '/jsproxy', session.encrypt(msg2json(req)), function(r) {
            session.decrypt(r, cb);
            session.dequeue();
            if (cb == receive) {
                post({}, receive);
            }
        });
    }
}
user.dat has your password?

- Yep!
283i4jfkai3389

```python
key = md5(username + "283i4jfkai3389")
password = password xor key
```
Rooting the router
Getting shell

1) **CREATE /nova/etc/devel-login**

2) **TELNET TO 192.168.88.1 AS DEVEL**
   - YAAY! :)

3) **LS**
   - FAIL:(

```bash
$ telnet 192.168.88.1
Trying 192.168.88.1...
Connected to 192.168.88.1.
Escape character is '^]'.

MikroTik v6.39.2 (stable)
Login: devel
Password:

BusyBox v1.00 (2017.05.31-11:35+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# ls
bash: ls: not found
# 
```
[TAB] to the rescue

- **No ls? No problem!**
  - cat, space, tab, tab

- **Or, you know, do it properly, and upload busybox**
  - statically linked, for the right architecture
    - **uname -m**
  - **This might be of interest:**
    - [https://busybox.net/downloads/binaries/1.21.1/](https://busybox.net/downloads/binaries/1.21.1/)
Can we speed this up?

- **Of course.**
- **A VirtualBox appliance!**
  - Does the work for you
- **This should work out nicely**
  - If your CPU is AR9344 and device has at least two ethernet ports
    - RB951G-2HnD, RB951Ui-2HnD <= tested
    - CRS109-8G-1S-2HnD-IN, CRS125-24G-1S-IN, CRS125-24G-1S-2HnD-IN
    - OmniTIK 5, OmniTIK 5 PoE
How to use the appliance

Demo: MT_JB_0.89.ova

1) **Import the appliance**

2) **Make sure bridged network card is set to ethernet**

3) **Disconnect all wires from the router, power it up**

4) **Start the virtual machine and follow instructions**

5) **Be ready to swiftly re-plug the cable when prompted**
So, what’s new?

- **What if I've forgotten my password?**
  
  **Not a problem! ;)**
DEMO

PLEASE DON'T

BRICK THE ROUTER
Question time

• **Tools are available**
  
  https://github.com/0ki/

• **Current appliance:**
  
  http://02.lv/f/2017/09/15/MT_JB_0.89.0VA

  – GOOD LUCK GUESSING

  WHICH LETTERS ARE CAPITAL ;P