Search and Rescue Thousand MikroTik Device from Intruders in One Night

12 June 2019, Kuala Lumpur
MikroTik User Meeting Malaysia 2019
By Michael Takeuchi

Michael Takeuchi

- Level 2 Escalation Support Engineer and Network Engineer at NetData
- Handling Hundreds Gigabit/s Networks (Local & International Networks) in NetData
- SDN & NFV Developer in NetData
- MikroTik Certified Consultant
- MikroTik Certified Engineer
  (MTCNA, MTCRE, MTCINE, MTCWE, MTCUME, MTCTCE, MTCIPv6E)

https://www.linkedin.com/in/michael-takeuchi

michael.takeuchi@nds.id or info@nds.id

https://www.facebook.com/mict404

Presentation Background

- Once upon time, MikroTik RouterOS have a **vulnerability** in their system, so I got a Project that I need to upgrade around 2000 MikroTik RouterOS in a month
- 2000 router make me lazier to do that, so after around 600 router I do upgrade manually, I start to looking another ways to upgrade the remain router automatically without big effort from my self
- Also as a services to several customer that need automation solutions

How We Do Search and Rescue?

- Search
  - Documentation
  - Scanner Tools
    - Scanning
    - Data Processing
- Rescue
  - Flowchart
  - Deployment
  - Checking

Search – Documentation

- You can look your devices from:

1. Network Monitoring System (NMS)
2. Excel/Word Documentation

Search – Scanner Tools (Scanning)

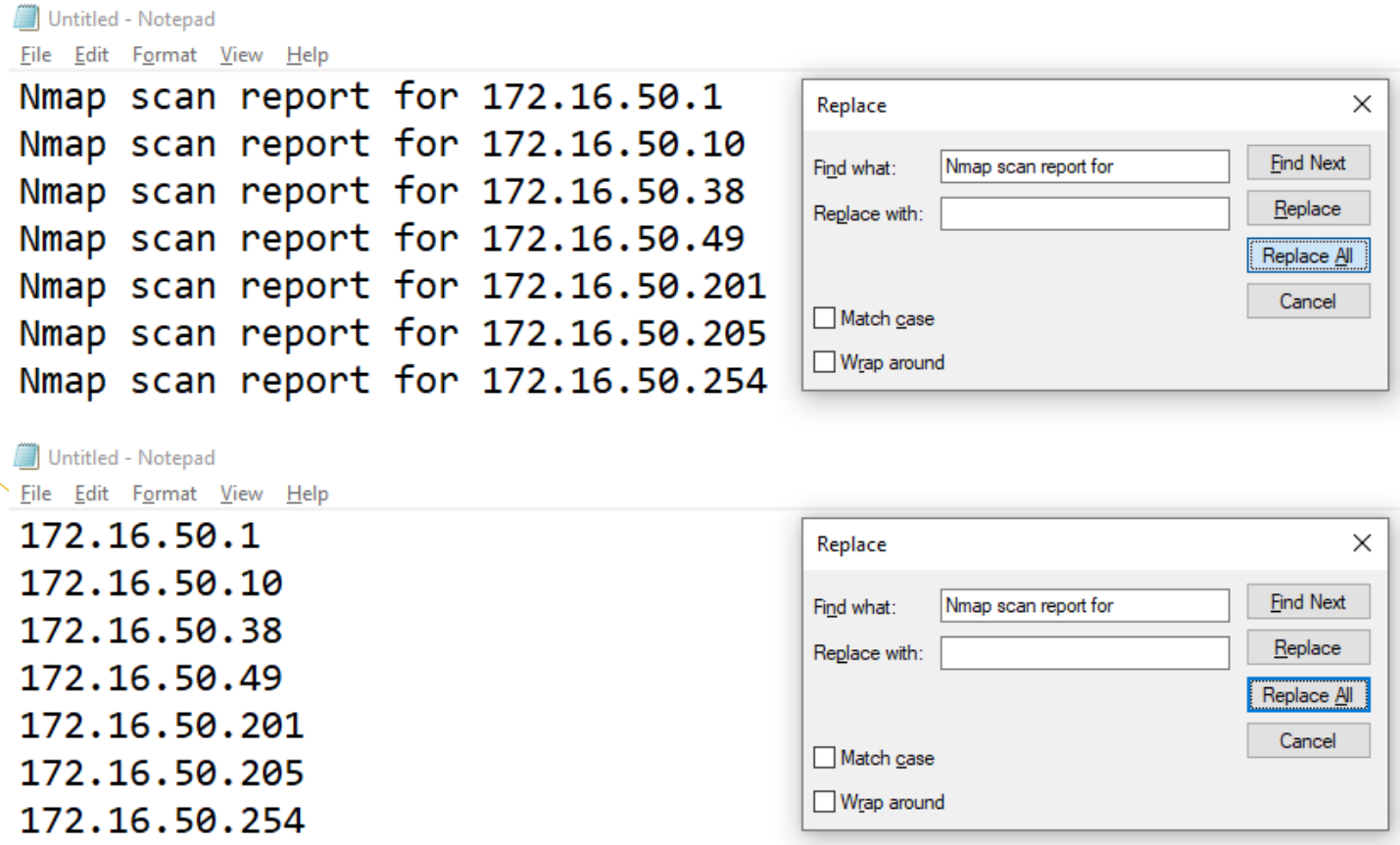
```
C:\Users\Takeuchi>nmap -p 8291 --open 172.16.50.0/24 | findstr "172.16.50."
Nmap scan report for 172.16.50.1
Nmap scan report for 172.16.50.10
Nmap scan report for 172.16.50.38
Nmap scan report for 172.16.50.49
Nmap scan report for 172.16.50.201
Nmap scan report for 172.16.50.205
Nmap scan report for 172.16.50.254

C:\Users\Takeuchi>
```

```
nmap -p 8291 --open 172.16.50.0/24 | findstr "172.16.50."
```

`-p`	:	Port Number
`--open`	:	Only Show Open Ports
`172.16.50.0/24`	:	IP Range
`|`	:	Pipe
`findstr`	:	Output Filter
`"172.16.50."`	:	String to Find

Search – Scanner Tools (Data Processing)



You can replace

"Nmap scan report for"

With "~~nothing~~" so you can filter out the IP Address only

Rescue – Start to Rescue

In this presentation I will talk about how to upgrade RouterOS version using Ansible, but you can explore more to patch your security issue such as configuring firewall

So the questions is, why **Ansible?**

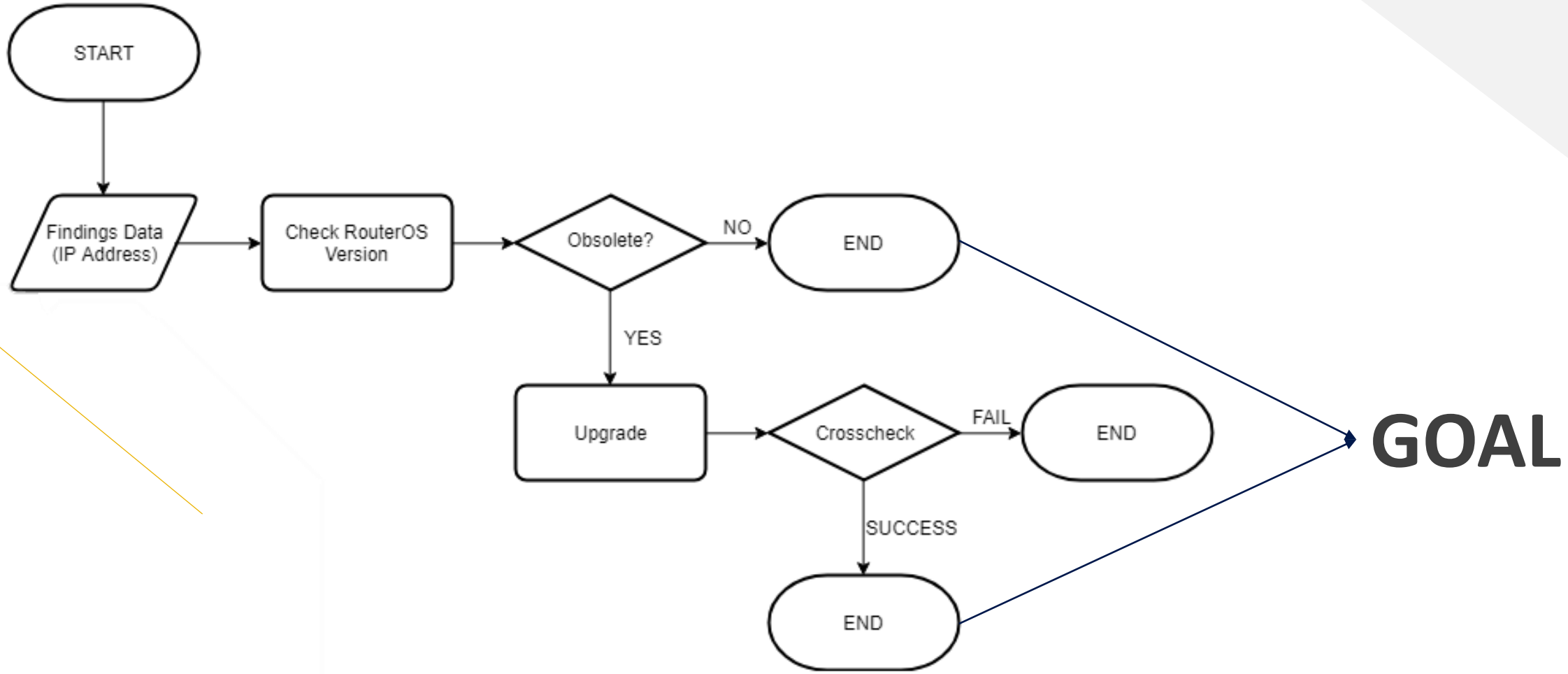
- Easier for Network Engineer that not really understand programming
- Script-less (compared with Python Paramiko or something similar)

To install **Ansible** you can refer to:
https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html

You can adjust with what operating system that you use (I will use Linux here)

Note: Ansible will remote your router with **SSH**

Rescue – Flowchart



Rescue – Ansible Setup

```
root@ansible:~/ansible# ls
ansible.cfg  hosts
root@ansible:~/ansible# cat ansible.cfg
[defaults]
inventory = ./hosts
host_key_checking = false
timeout = 5
remote_port = 2288
root@ansible:~/ansible# cat hosts
[group1]
172.16.50.1
172.16.50.10
172.16.50.38
172.16.50.49
172.16.50.201
172.16.50.205
172.16.50.254

root@ansible:~/ansible#
```

- In **Ansible** we have 2 file:

`ansible.cfg` = **Ansible Config File**

`hosts` = **Hosts IP Address List**

Rescue – Ansible Setup

- **ansible.cfg**

```
[defaults]
inventory = ./hosts
host_key_checking = false
timeout = 5
remote_port = 2288
```

==========

```
inventory              = Host List
host_key_checking      = Avoid SSH Key Checking
timeout                = SSH Timeout
remote_port            = SSH Port
```

Rescue – Ansible Setup

- **hosts**

```
[group1]
172.16.50.1
172.16.50.10
172.16.50.38
172.16.50.49
172.16.50.201
172.16.50.205
172.16.50.254
```

Rescue – RouterOS Repository

- Downloading NPK from mikrotik.com for 2000 Router is costing our internet bandwidth so much, so we will create local repository and download from mikrotik.com once to our local repository and make our internet bandwidth is not exhausted for downloading NPK File from mikrotik.com
- To create RouterOS Repository you only need to create a simple web server, and upload all of RouterOS NPK file to your repository server and make sure that your server are reachable from router
- Please note that CHR Architecture didn’t give us NPK file ☺ so for CHR we need to upgrade the RouterOS directly to mikrotik.com

Rescue – RouterOS Repository

```
root@ansible:/var/www/html# ls
index.html                  routeros-mmips-6.44.2.npk    routeros-tile-6.44.2.npk
routeros-arm-6.44.2.npk     routeros-powerpc-6.44.2.npk  routeros-x86-6.44.2.npk
routeros-mipsbe-6.44.2.npk  routeros-smips-6.44.2.npk
root@ansible:/var/www/html# service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2019-04-25 11:01:23 WIB; 5h 53min ago
  Process: 793 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
  Process: 402 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 431 (apache2)
    Tasks: 55 (limit: 4915)
   CGroup: /system.slice/apache2.service
           ├─431 /usr/sbin/apache2 -k start
           ├─797 /usr/sbin/apache2 -k start
           └─798 /usr/sbin/apache2 -k start

Apr 25 11:01:23 ansible systemd[1]: Starting The Apache HTTP Server...
Apr 25 11:01:23 ansible apachectl[402]: AH00558: apache2: Could not reliably determine the server's fully
Apr 25 11:01:23 ansible systemd[1]: Started The Apache HTTP Server.
Apr 25 11:06:18 ansible systemd[1]: Reloading The Apache HTTP Server.
Apr 25 11:06:18 ansible apachectl[793]: AH00558: apache2: Could not reliably determine the server's fully
Apr 25 11:06:18 ansible systemd[1]: Reloaded The Apache HTTP Server.
root@ansible:/var/www/html#
```

Rescue – RouterOS Script

1. Check RouterOS Version & Identity Name

```
:put ("Router ".[/system identity get name]. " has RouterOS Version ".[/system resource get version]);
```

2. Download RouterOS NPK File based on router architecture from local repository

```
/tool fetch url=("http://172.16.50.100/routeros-".[/system resource get architecture-name]."-6.44.2.npk");
```

3. NPK File Check

```
:if ([:len [/file find name=("routeros-".[/system resource get architecture-name]."-6.44.2.npk")]] !=0)
do={:put ("Router ".[/system identity get name]." NPK File Downloaded")} else={
:put ("Router " .[/system identity get name]." NPK File Not Found")};
```

4. Reboot to Upgrade (DOWNTIME OCCURS!!!)

```
:execute {/system reboot};
```

5. Crosscheck RouterOS Version

```
:put ("Router ".[/system identity get name]. " has RouterOS Version ".[/system resource get version]);
```

Rescue – RouterOS Script with CHR

- If you have few Cloud Hosted Router (CHR) router you can separate it on Ansible Host Group, but here I just made a script on RouterOS that can identify your router architecture is CHR or not
- If router architecture is not CHR (x86_64) then download from local repository, else download from mikrotik.com

```
:if ([/system resource get architecture-name] !="x86_64") do={/tool fetch
url=("http://172.16.50.100/routeros-".[/system resource get architecture-name]."-6.44.2.npk");}
else={/system package update check-for-updates;/system package update download;}
```

Rescue – Execute with Ansible

```
ansible group -m raw -a 'script;:delay 1;quit' -u takeuchi -k
```

group = Your Group List, i write **group1** in hosts file
script = Your MikroTik Script
takeuchi = MikroTik Username

Rescue – Execute with Ansible

- RouterOS Version Checking

```
root@ansible:~/ansible# ansible group1 -m raw -a ':put ("Router ".[/system identity get name]. " has RouterOS Version "
.[/system resource get version]);:delay 1;quit;' -u takeuchi -k | grep "RouterOS Version"
SSH password:
Router Gateway has RouterOS Version 6.44.1 (stable)
Router R1 has RouterOS Version 6.44.1 (stable)
Router R3 has RouterOS Version 6.44.1 (stable)
Router R2 has RouterOS Version 6.44.1 (stable)
Router R4 has RouterOS Version 6.44.1 (stable)
Router R5 has RouterOS Version 6.44.1 (stable)
Router R6 has RouterOS Version 6.44.1 (stable)
root@ansible:~/ansible#
```

Rescue – Execute with Ansible

- RouterOS NPK Download

```
root@ansible:~/ansible# ansible group1 -m raw -a '/tool fetch url=("http://172.16.50.100/routeros-"
.[/system resource get architecture-name]."-6.44.2.npk");:delay 1;quit;' -u takeuchi -k
SSH password:
172.16.50.1 | SUCCESS | rc=0 >>
      status: finished
  downloaded: 20067KiB
       total: 20067KiB
    duration: 2s

Shared connection to 172.16.50.1 closed.


172.16.50.10 | SUCCESS | rc=0 >>
      status: finished
  downloaded: 20067KiB
       total: 20067KiB
    duration: 3s

interrupted
Shared connection to 172.16.50.10 closed.


172.16.50.38 | SUCCESS | rc=0 >>
      status: finished
  downloaded: 20067KiB
       total: 20067KiB
    duration: 3s

interrupted
Shared connection to 172.16.50.38 closed.
```

Rescue – Execute with Ansible

- NPK File Check

```
root@ansible:~/ansible# ansible group1 -m raw -a ':if ([:len [/file find name=("routeros-".[/system resource get architecture
-name]."-6.44.2.npk")]] !=0) do={:put ("Router ".[/system identity get name]." NPK File Downloaded")} else={:put ("Router " .
[/system identity get name]." NPK File Not Found")};delay 1;quit' -u takeuchi -k | grep "Router"
SSH password:
Router Gateway NPK File Downloaded
Router R2 NPK File Downloaded
Router R1 NPK File Downloaded
Router R4 NPK File Downloaded
Router R3 NPK File Downloaded
Router R5 NPK File Downloaded
Router R6 NPK File Downloaded
root@ansible:~/ansible#
```

Rescue – Execute with Ansible

- Router Upgrade Reboot (DOWNTIME OCCURS!!!)

```
root@ansible:~/ansible# ansible group1 -m raw -a ':execute {/system reboot};delay 1;quit' -u takeuchi -k
SSH password:
172.16.50.38 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.38 closed.\r\n",
    "unreachable": true
}
172.16.50.1 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.1 closed.\r\n",
    "unreachable": true
}
172.16.50.201 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.201 closed.\r\n",
    "unreachable": true
}
172.16.50.10 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.10 closed.\r\n",
    "unreachable": true
}
172.16.50.49 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.49 closed.\r\n",
    "unreachable": true
}
172.16.50.205 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.205 closed.\r\n",
    "unreachable": true
}
172.16.50.254 | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: Shared connection to 172.16.50.254 closed.\r\n",
    "unreachable": true
}
root@ansible:~/ansible#
```

Rescue – Execute with Ansible

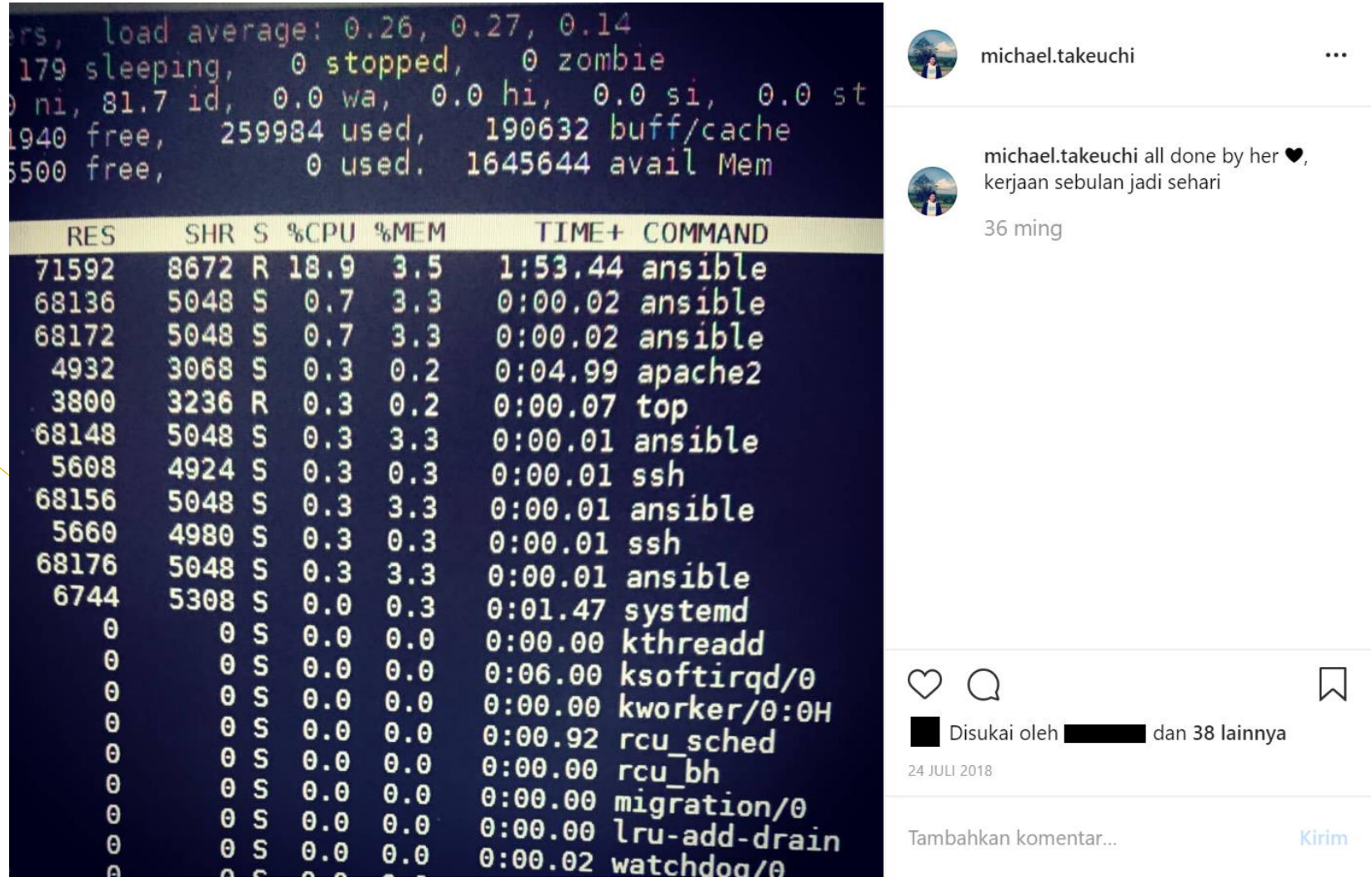
- RouterOS Version Crosscheck

```
root@ansible:~/ansible# ansible group1 -m raw -a ':put ("Router ".[/system identity get name]. " has RouterOS Version "
.[/system resource get version]);delay 1;quit' -u takeuchi -k | grep "RouterOS Version"
SSH password:
Router Gateway has RouterOS Version 6.44.2 (stable)
Router R1 has RouterOS Version 6.44.2 (stable)
Router R2 has RouterOS Version 6.44.2 (stable)
Router R3 has RouterOS Version 6.44.2 (stable)
Router R4 has RouterOS Version 6.44.2 (stable)
Router R5 has RouterOS Version 6.44.2 (stable)
Router R6 has RouterOS Version 6.44.2 (stable)
root@ansible:~/ansible#
```

Pros, Cons & Solution

- Pros:
  - Easy to Setup
  - Script-less
- Cons:
  - Problem if you didn’t implement RADIUS AAA yet for Router Authentication or;
  - Having different username or password each devices
- Solution for Cons
  - Integrate Ansible with Databases
  - Script from Scratch (usually using python)

Success Story



```
rs,  load average: 0.26, 0.27, 0.14
179 sleeping,   0 stopped,   0 zombie
 ni, 81.7 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
940 free,   259984 used,   190632 buff/cache
500 free,        0 used.  1645644 avail Mem

  RES    SHR S %CPU %MEM     TIME+ COMMAND
71592   8672 R 18.9  3.5   1:53.44 ansible
68136   5048 S  0.7  3.3   0:00.02 ansible
68172   5048 S  0.7  3.3   0:00.02 ansible
 4932   3068 S  0.3  0.2   0:04.99 apache2
 3800   3236 R  0.3  0.2   0:00.07 top
68148   5048 S  0.3  3.3   0:00.01 ansible
 5608   4924 S  0.3  0.3   0:00.01 ssh
68156   5048 S  0.3  3.3   0:00.01 ansible
 5660   4980 S  0.3  0.3   0:00.01 ssh
68176   5048 S  0.3  3.3   0:00.01 ansible
 6744   5308 S  0.0  0.3   0:01.47 systemd
    0      0 S  0.0  0.0   0:00.00 kthreadd
    0      0 S  0.0  0.0   0:06.00 ksoftirqd/0
    0      0 S  0.0  0.0   0:00.00 kworker/0:0H
    0      0 S  0.0  0.0   0:00.92 rcu_sched
    0      0 S  0.0  0.0   0:00.00 rcu_bh
    0      0 S  0.0  0.0   0:00.00 migration/0
    0      0 S  0.0  0.0   0:00.00 lru-add-drain
    0      0 S  0.0  0.0   0:00.02 watchdog/0
```


https://www.instagram.com/p/BlmfC0nHPi2k3bBZyWe-FGoGiPiAfU6lyMns440/

References

https://wiki.mikrotik.com/wiki/Automated_Backups

https://wiki.mikrotik.com/wiki/Manual:Scripting

https://wiki.mikrotik.com/wiki/Scripts

https://wiki.mikrotik.com/wiki/Manual:Scripting-examples

https://docs.ansible.com/

https://docs.ansible.com/ansible/latest/user_guide/index.html

https://github.com/mict404/ansible-mikrotik-auto-upgrade

Interesting? Questions?

Leave us a messages on info@nds.id ☺
We can script from scratch and make a solution for our beloved customer

And also create another services that suitable to solving our customer problem

Thank You

NETDATA Team