Dynamic Simple Queue Script (ARP Table)

PT Maxindo Mitra Solusi
http://www.maxindo.net.id

Maxindo Mitra Solusi
Internet Service Provider

Maxindo Mitra Solusi

Head Office

Jalan Marina Raya
Ruko Cordoba Blok H No. 77
Pantai Indah Kapuk
Jakarta Utara
DKI Jakarta 14470
☎(021) 5698 3483
☎(021) 5698 3484
sales@maxindo.net.id

Network Operation Center

Jalan Cendrawasih Raya No. 61
Cengkareng Barat
Jakarta Barat
DKI Jakarta 11730
☎(021) 5438 0777
☎(021) 5406 0001

Branch Offices

Jalan Ngagel Jaya Selatan
Ruko RMI Blok J No. 27
Surabaya
Jawa Timur 60283
☎(031) 502 9025
☎(031) 9902 2727

Jalan Puncak Jaya Ujung Kav. 8
Dieng, Malang
Jawa Timur
☎(0341) 555 133

Jalan Ngagel Jaya Selatan
Ruko RMI Blok J No. 27
Surabaya
Jawa Timur 60283
☎(031) 502 9025
☎(031) 9902 2727

Ruko Magnetica Square
Jalan Majapahit Blok D No. 9
Lippo Cikarang
Bekasi

Jalan Raya Serang KM 26
Kampung Pos Sentul RT9/RW4
Balaraja, Tanggerang
☎(021) 2987 0911
☎(021) 2987 0910

Profile

Erick Setiawan

Network Analyst

erick@maxindo.net.id

- Bina Nusantara University – Computer Engineering, 2009
- MikroTik Certified Consultant Asia Region
- MTCINE – MTCRE – MTCWE – MTCTCE - MTCNA

Outline

- Objective Overview
- How to use
- Screenshots
- How it works
- Script breakdown

Objective Overview

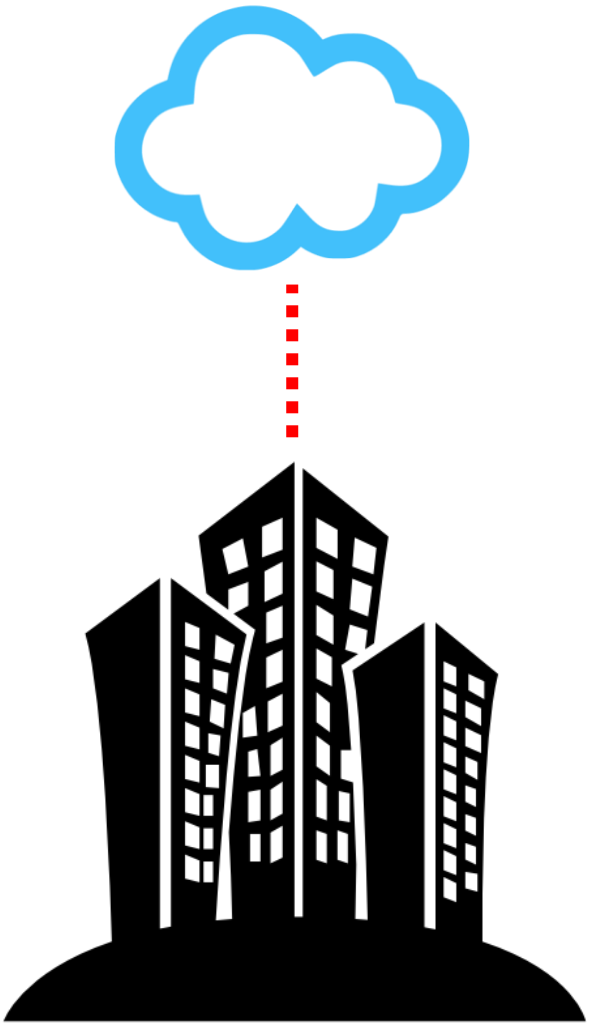
Example Case #1

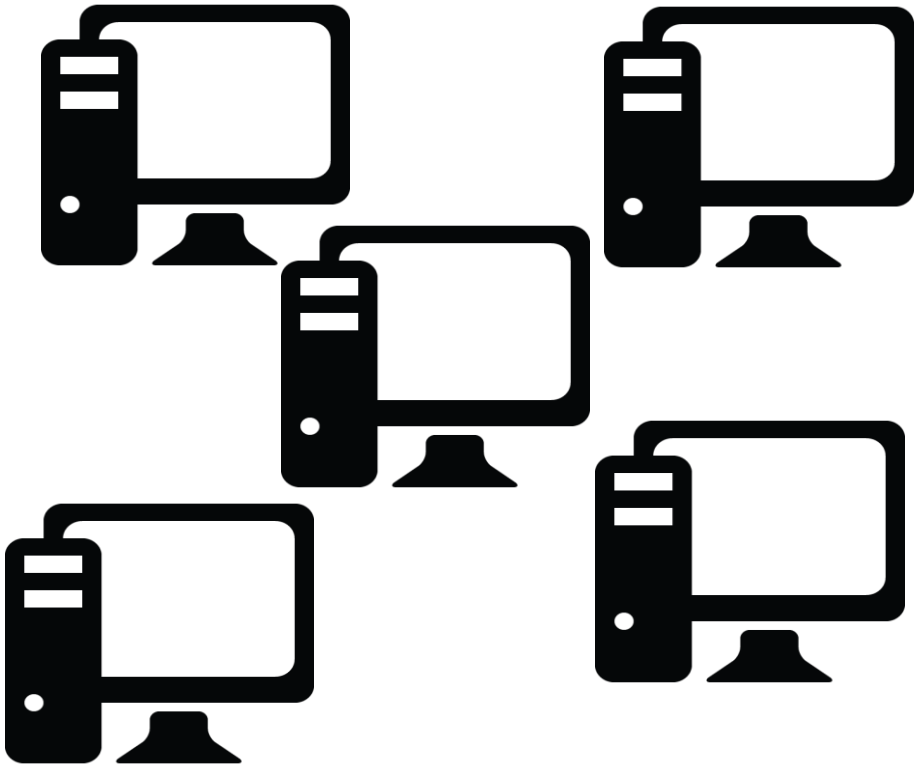
Avg. 100-150 customers per day

Need QoS / Bandwidth Mgmt

Objective Overview

Example Case #2





80-100 fixed PCs + BYOD

WiFi for guest

Need QoS / Bandwidth Mgmt

Expected Goal

- MikroTik RouterBoard/RouterOS
- To keep it as simple as possible, regarding to network device performance and cost

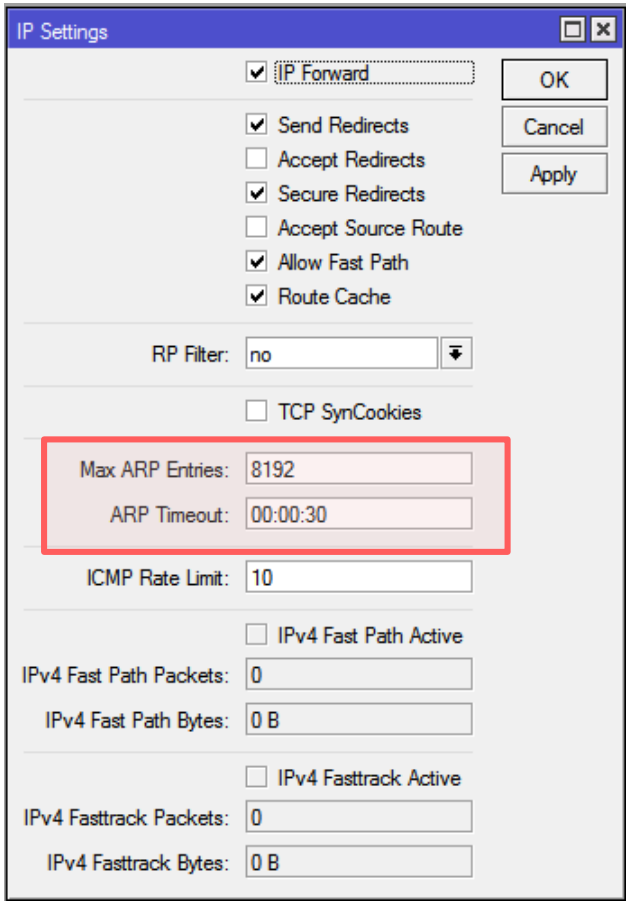
Possible Proposed Solution

- No QoS ☺
- Simple Queue (for each IP in the network)
- PCQ rate
- MikroTik's Hotspot w/wo RADIUS

Ideas

- Using simple queues to limit any traffic (or could be a packet mark) from specific target IP
- Only make simple queues rule for any active IP on the network (taken from ARP table)
- Utilize MikroTik's advantage to runs script and task scheduling to do the job.

Ideas


IP Settings
IP Forward
OK
Cancel
Apply
Send Redirects
Accept Redirects
Secure Redirects
Accept Source Route
Allow Fast Path
Route Cache
RP Filter: no
TCP SynCookies
Max ARP Entries: 8192
ARP Timeout: 00:00:30
ICMP Rate Limit: 10
IPv4 Fast Path Active
IPv4 Fast Path Packets: 0
IPv4 Fast Path Bytes: 0 B
IPv4 Fasttrack Active
IPv4 Fasttrack Packets: 0
IPv4 Fasttrack Bytes: 0 B


Maxindo RouterOS 6.36 (c) 1999-2016 http://www.maxindo.net.id
[?] Gives the list of available commands
command [?] Gives help on the command and list of arguments
[Tab] Completes the command/word. If the input is ambiguous,
a second [Tab] gives possible options
/ Move up to base level
.. Move up one level
/command Use command at the base level
[erick@ini-router] > /queu sim add target=192.168.

Simple Queue

wiki.mikrotik.com

The simplest way to limit data rate for specific IP addresses and/or subnets, is to use simple queues.

Objective Overview

Creating simple queue rule for each ARP entry exists on specific interface :

- *Runs as a background services*
- *Updates its condition per time-range specified*
- *Changes based on ARP entries exists on router's ARP table*
- *Rules made will contain any parameter (packet-marks, max-limit, limit-at) that you need*

*Will be done using **/system script** and **/system scheduler***

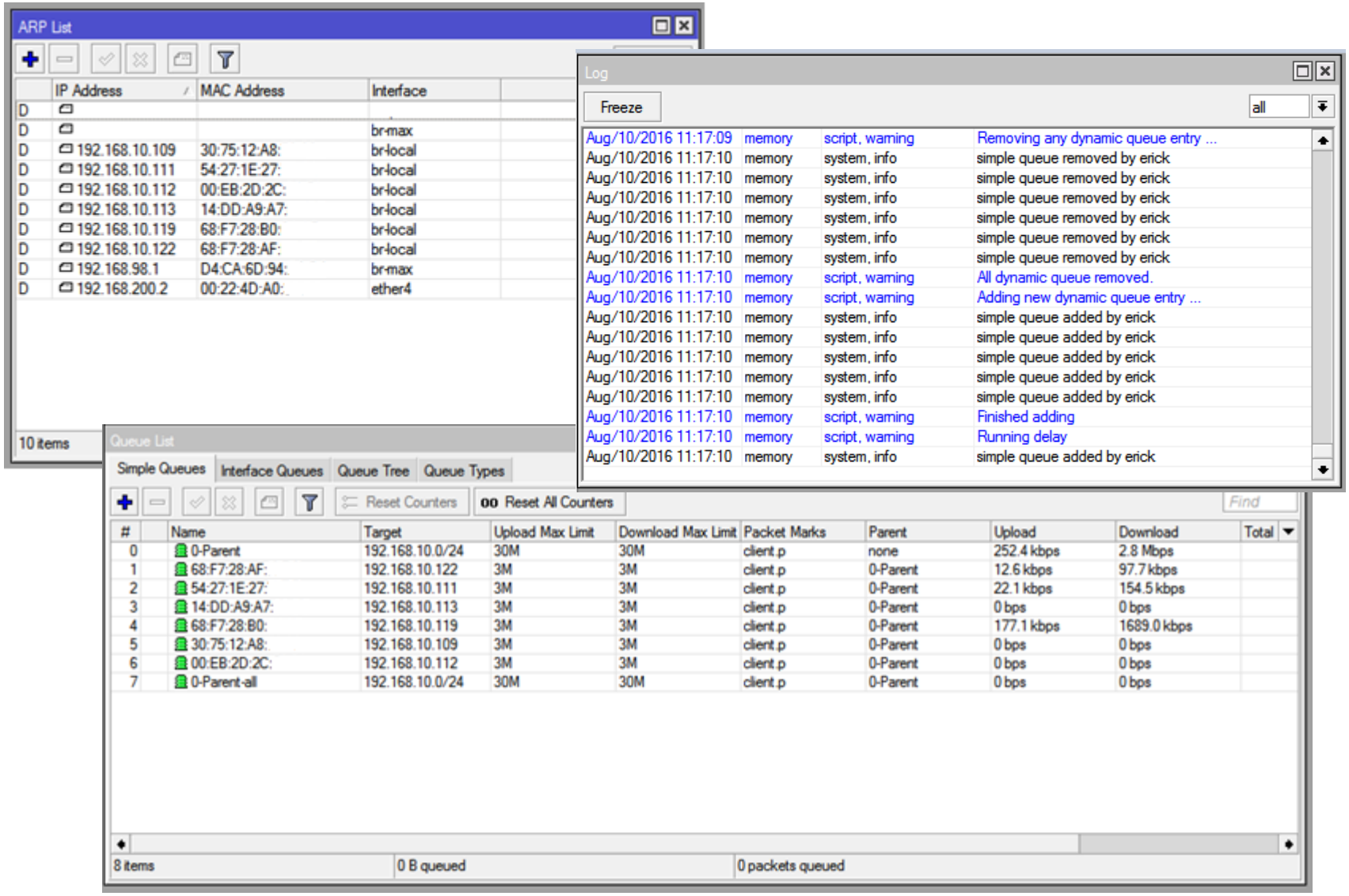
How to use

- *Create your mangle*

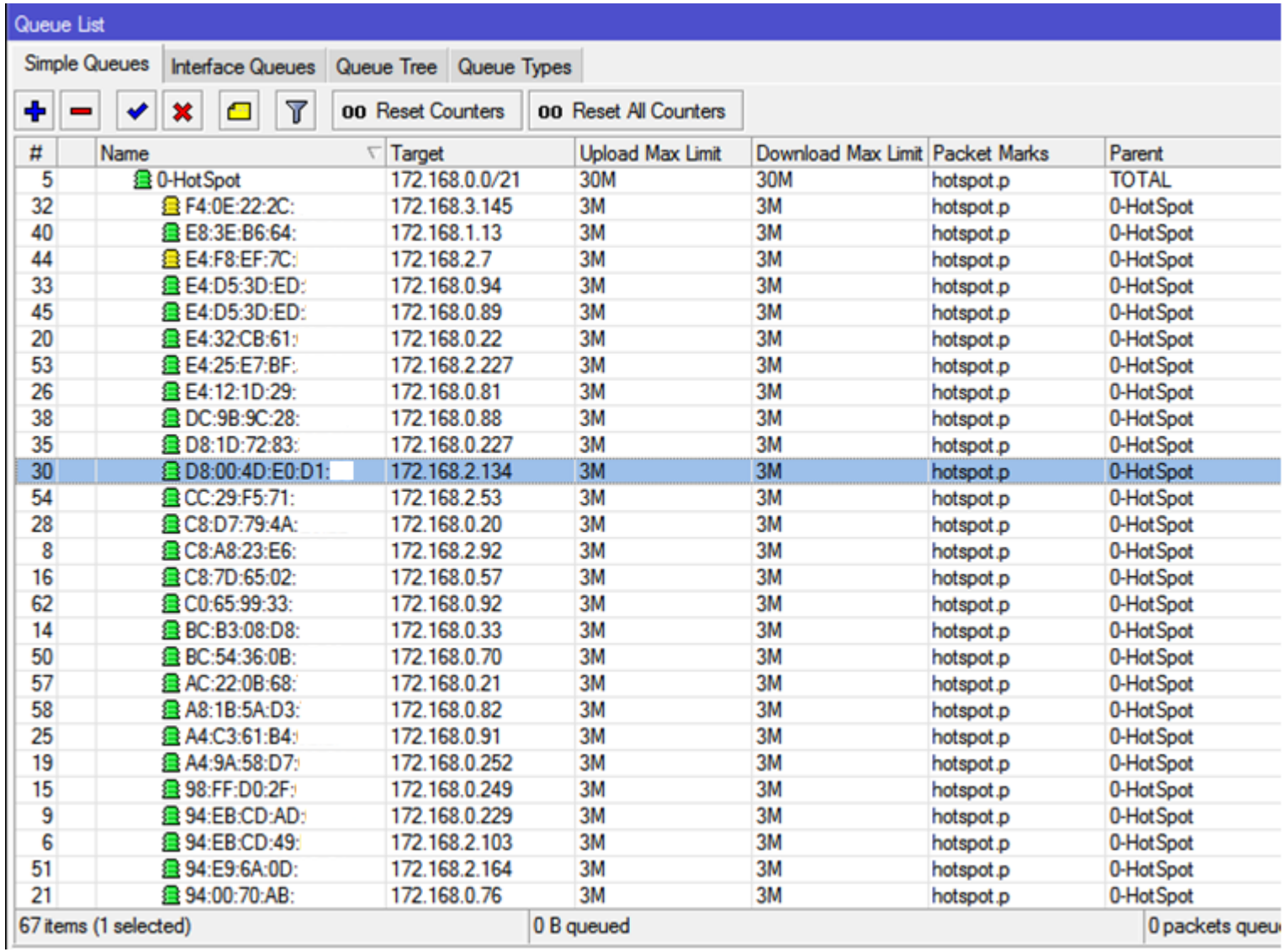
```
/ip firewall mangle
add action=mark-connection chain=prerouting \
    new-connection-mark=client.c passthrough=yes src-address-list=lan-addr
add action=mark-packet chain=prerouting connection-mark=client.c \
    new-packet-mark=client.p passthrough=no
/ip firewall address-list
add list=lan-addr address=192.168.10.0/24
```

- *Customize based on your specific need/network configuration*
- *Drag-drop-import*

Screenshots



Screenshots



Queue List

Simple Queues | Interface Queues | Queue Tree | Queue Types

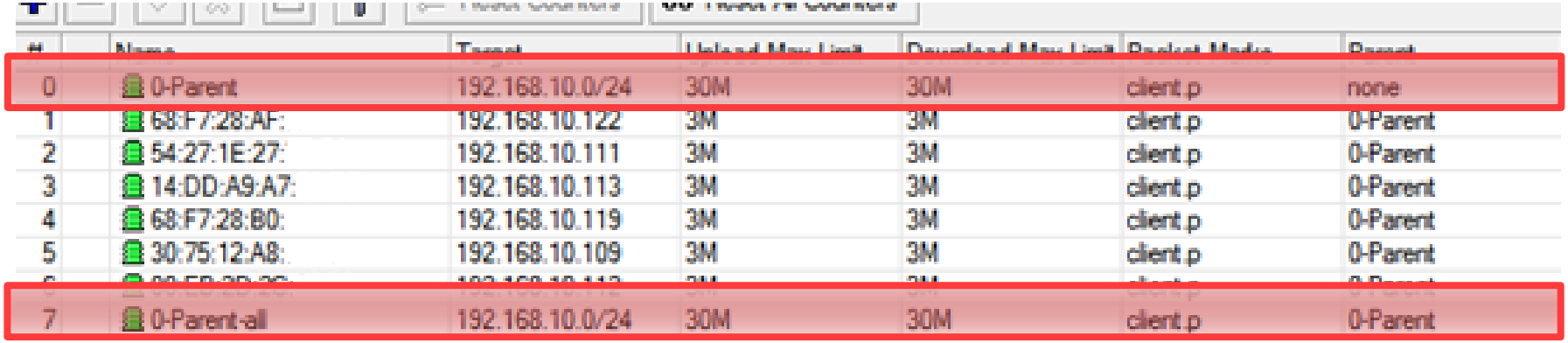
Reset Counters | Reset All Counters

#	Name	Target	Upload Max Limit	Download Max Limit	Packet Marks	Parent
5	0-HotSpot	172.168.0.0/21	30M	30M	hotspot.p	TOTAL
32	F4:0E:22:2C:	172.168.3.145	3M	3M	hotspot.p	0-HotSpot
40	E8:3E:B6:64:	172.168.1.13	3M	3M	hotspot.p	0-HotSpot
44	E4:F8:EF:7C:	172.168.2.7	3M	3M	hotspot.p	0-HotSpot
33	E4:D5:3D:ED:	172.168.0.94	3M	3M	hotspot.p	0-HotSpot
45	E4:D5:3D:ED:	172.168.0.89	3M	3M	hotspot.p	0-HotSpot
20	E4:32:CB:61:	172.168.0.22	3M	3M	hotspot.p	0-HotSpot
53	E4:25:E7:BF:	172.168.2.227	3M	3M	hotspot.p	0-HotSpot
26	E4:12:1D:29:	172.168.0.81	3M	3M	hotspot.p	0-HotSpot
38	DC:9B:9C:28:	172.168.0.88	3M	3M	hotspot.p	0-HotSpot
35	D8:1D:72:83:	172.168.0.227	3M	3M	hotspot.p	0-HotSpot
30	D8:00:4D:E0:D1:	172.168.2.134	3M	3M	hotspot.p	0-HotSpot
54	CC:29:F5:71:	172.168.2.53	3M	3M	hotspot.p	0-HotSpot
28	C8:D7:79:4A:	172.168.0.20	3M	3M	hotspot.p	0-HotSpot
8	C8:A8:23:E6:	172.168.2.92	3M	3M	hotspot.p	0-HotSpot
16	C8:7D:65:02:	172.168.0.57	3M	3M	hotspot.p	0-HotSpot
62	C0:65:99:33:	172.168.0.92	3M	3M	hotspot.p	0-HotSpot
14	BC:B3:08:D8:	172.168.0.33	3M	3M	hotspot.p	0-HotSpot
50	BC:54:36:0B:	172.168.0.70	3M	3M	hotspot.p	0-HotSpot
57	AC:22:0B:68:	172.168.0.21	3M	3M	hotspot.p	0-HotSpot
58	A8:1B:5A:D3:	172.168.0.82	3M	3M	hotspot.p	0-HotSpot
25	A4:C3:61:B4:	172.168.0.91	3M	3M	hotspot.p	0-HotSpot
19	A4:9A:58:D7:	172.168.0.252	3M	3M	hotspot.p	0-HotSpot
15	98:FF:D0:2F:	172.168.0.249	3M	3M	hotspot.p	0-HotSpot
9	94:EB:CD:AD:	172.168.0.229	3M	3M	hotspot.p	0-HotSpot
6	94:EB:CD:49:	172.168.2.103	3M	3M	hotspot.p	0-HotSpot
51	94:E9:6A:0D:	172.168.2.164	3M	3M	hotspot.p	0-HotSpot
21	94:00:70:AB:	172.168.0.76	3M	3M	hotspot.p	0-HotSpot

67 items (1 selected) | 0 B queued | 0 packets queu

How It Works

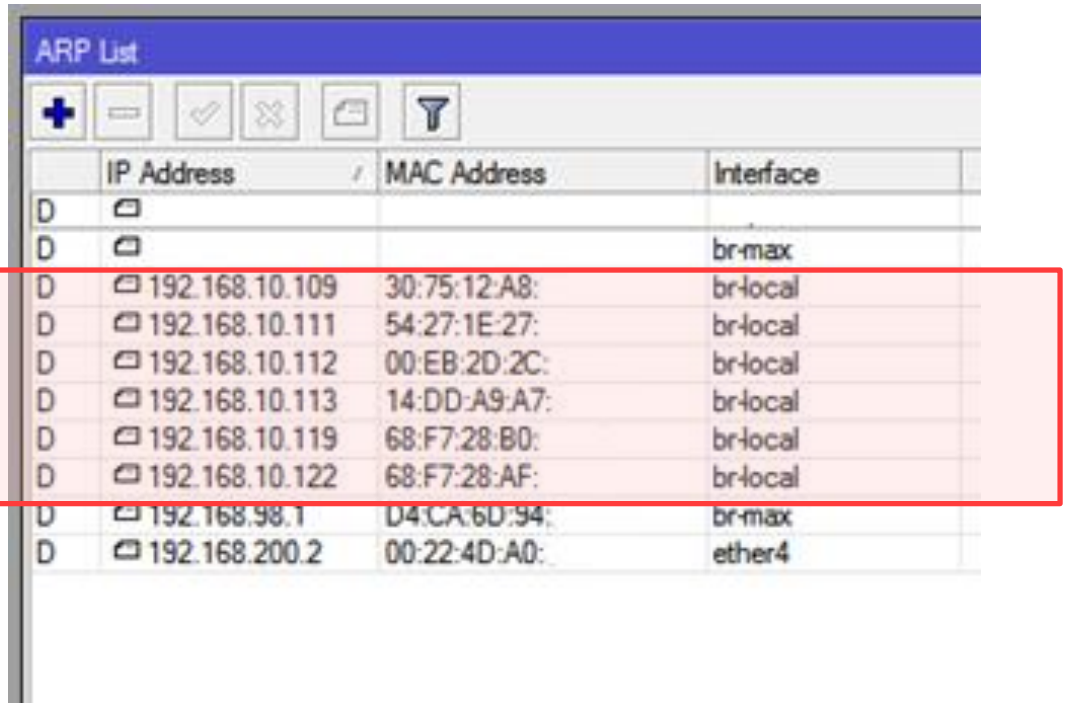
1. *The script will make a parent queue[i] and a catch-all rule[ii] for our workaround with all parameter which we've already set.*



#	Name	Target	Upload Max Limit	Download Max Limit	Packet Marks	Parent
0	0-Parent	192.168.10.0/24	30M	30M	client.p	none
1	68:F7:28:AF:	192.168.10.122	3M	3M	client.p	0-Parent
2	54:27:1E:27:	192.168.10.111	3M	3M	client.p	0-Parent
3	14:DD:A9:A7:	192.168.10.113	3M	3M	client.p	0-Parent
4	68:F7:28:B0:	192.168.10.119	3M	3M	client.p	0-Parent
5	30:75:12:A8:	192.168.10.109	3M	3M	client.p	0-Parent
7	0-Parent-all	192.168.10.0/24	30M	30M	client.p	0-Parent

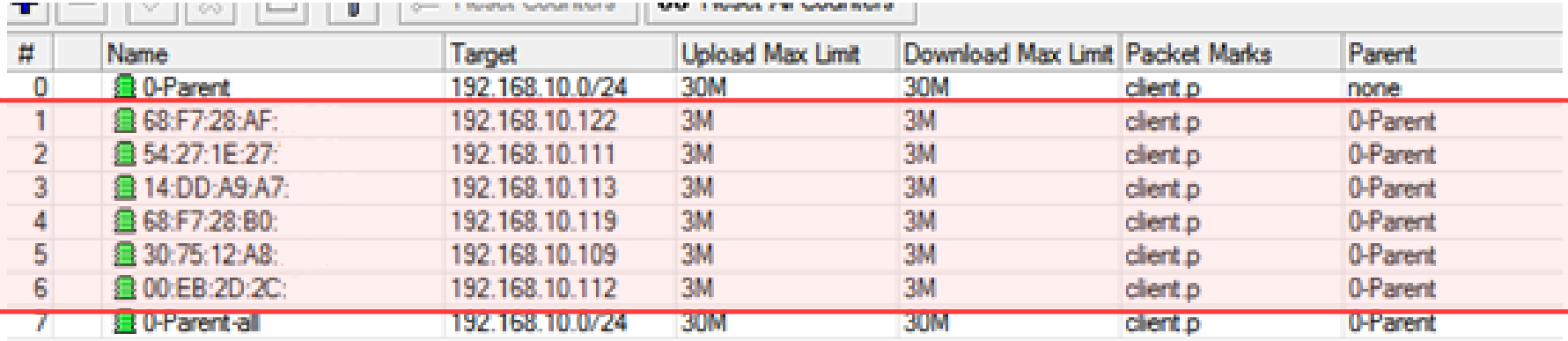
How It Works

2. *Takes all IP addresses and MAC address with specified filter (interface) from **/ip arp** table.*



How It Works

3. *Removes all dynamic rules previously created, if available. Then creates simple queue rules using parameters we've already set.*



#	Name	Target	Upload Max Limit	Download Max Limit	Packet Marks	Parent
0	0-Parent	192.168.10.0/24	30M	30M	client p	none
1	68:F7:28:AF:	192.168.10.122	3M	3M	client p	0-Parent
2	54:27:1E:27:	192.168.10.111	3M	3M	client p	0-Parent
3	14:DD:A9:A7:	192.168.10.113	3M	3M	client p	0-Parent
4	68:F7:28:B0:	192.168.10.119	3M	3M	client p	0-Parent
5	30:75:12:A8:	192.168.10.109	3M	3M	client p	0-Parent
6	00:EB:2D:2C:	192.168.10.112	3M	3M	client p	0-Parent
7	0-Parent-all	192.168.10.0/24	30M	30M	client p	0-Parent

How It Works

4. *Delay for specified time (default is 60 secs) and re-run the steps from number 2.*

5. *The scheduler at **/system scheduler** checks and ensures that the script is running in the background.*

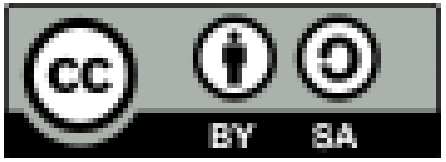
Script Breakdown

```
:local filterARP "br-local";
:local targetAddress "192.168.10.0/24";
:local limitClient "3M";
:local limitParent "30M";
:local parentLimit "0-Parent";
:local packetMark "client.p";
:local delay "60s";

:local enabled true;
:local enableLog true;

/queue simple
:if ($enabled) do={
    :if ([find name=($parentLimit)] = "") do={  add name=($parentLimit) packet-mark=$packetMark \
        target=$targetAddress max-limit=($limitParent."/".$limitParent);    }
    :if ([find name=($parentLimit."-all")] = "") do={ add name=($parentLimit."-all") parent=$parentLimit packet-mark=$packetMark \
        target=$targetAddress max-limit=($limitParent."/".$limitParent); }
}

:while (true) do={

    :local arp [:toarray [/ip arp print as-value where dynamic && interface=$filterARP ]];
    :local queue [:toarray [/queue simple print as-value]];

    :if ($enableLog) do={ :log warning message= "Removing any dynamic queue entry ..."; }

    :if ($enabled) do={ /queue simple remove [find (parent=$parentLimit) && (name!=($parentLimit."-all"))]; }

    :if ($enableLog) do={ :log warning message= "All dynamic queue removed."; :log warning message= "Adding new dynamic queue entry ..."; }

    :foreach a in=$arp do={
        :local ip ($a->"address");
        :local zz ($a->"mac-address");
        :if ($enabled) do={/queue simple add name=($zz) target=$ip max-limit=($limitClient."/".$limitClient) parent=$parentLimit \
        packet-mark=$packetMark place-before=($parentLimit."-all"); }

    }

    :if ($enableLog) do={ :log warning message= "Finished adding"; :log warning message= "Running delay"; }
    :delay $delay;
}
```

Please share and contribute your idea

https://github.com/ericksetiawan/dynamic-simple-queue

"Learn like Newbie, work like a Pro" - LucuBRB



PT Maxindo Mitra Solusi

http://www.maxindo.net.id