i-BEAM
steering ahead

Implement Content Filtering

Lay Minh (Makito)

CCIE # 47682, MikroTik Certified Trainer, MikroTik Consultant

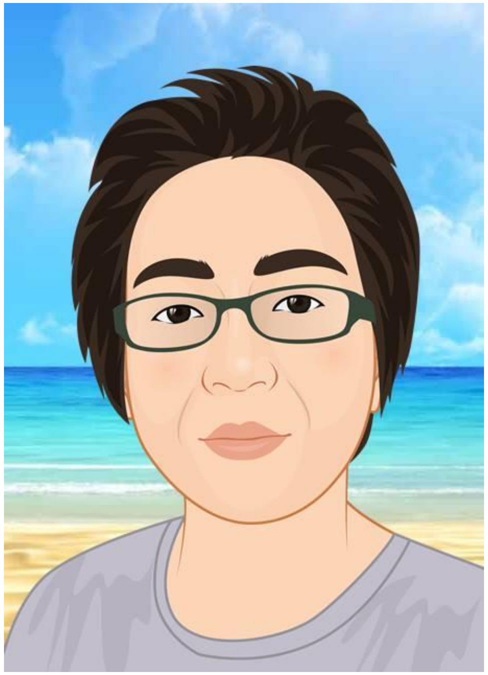
MikroTik User Meeting (Yangon, Myanmar)

About Me

i-BEAM
steering ahead

- **Lay Minh**
  - My nick name is **Makito**
  - CCIE # 47682
  - Chief Technology Officer (CTO) at i-BEAM
  - MikroTik Certified Trainer & Consultant
  - Experiences:
    - 10 years in ISP industry since 2005
    - Billing solutions for service providers
    - ISP core network design and operations
  - MikroTik Certifications:
  - Areas of interest: BGP, MPLS, IPv6

CISCO CERTIFIED CCIE SERVICE PROVIDER



MikroTik MTCNA | MikroTik MTCRE | MikroTik MTCWE | MikroTik MTCTCE | MikroTik MTCUME | MikroTik MTCINE

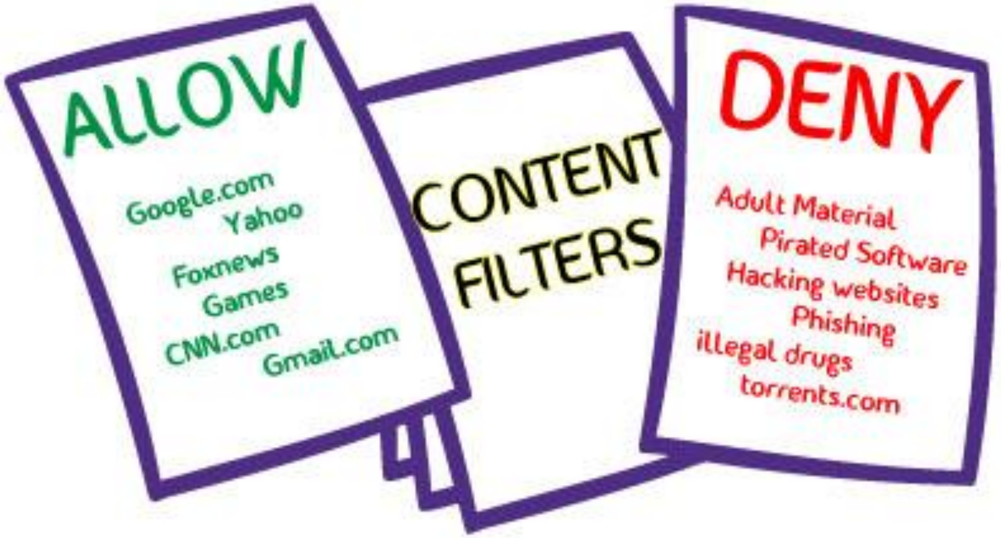
About i-BEAM steering ahead

- **Initially found in year 2003, renovated in 2015!**
- One of the very first ICT training centers in Myanmar
- Basically we are doing:
  - MikroTik Certification Training
    - MTCNA, MTCRE, MTCWE, MTCTCE, MTCUME, MTCINE
  - MikroTik Products & Solutions
  - Cisco Certification Training
    - CCNA, CCNP, CCIE, CCDA, CCDP, CCDE..etc.
  - Linux & Network Fundamentals Training
  - IT/Network Consultation
  - ISP Billing Solution
  - ISP Design & Operations

What Is Content Filtering?

i-BEAM
steering ahead

- "Content" typically means web pages, e-mails, videos, files, or applications on the internet.
- Content Filtering restricts user's access to specific contents for some reasons:
  - Company Policies
  - Government Authority Requests
  - Parental Controls
  - Legality Issues
  - Security Purpose
  - …etc.



How To Do Content Filtering?

- Before you start, you have to know what kind of contents you wanna filter.
- Different techniques can be used depends on the nature of contents:
  - Routing Table
  - IP-based Filter
  - Keyword Filter
  - Layer 7 Filter
  - Web Proxy
  - DNS

Features Comparison

i-BEAM
steering ahead

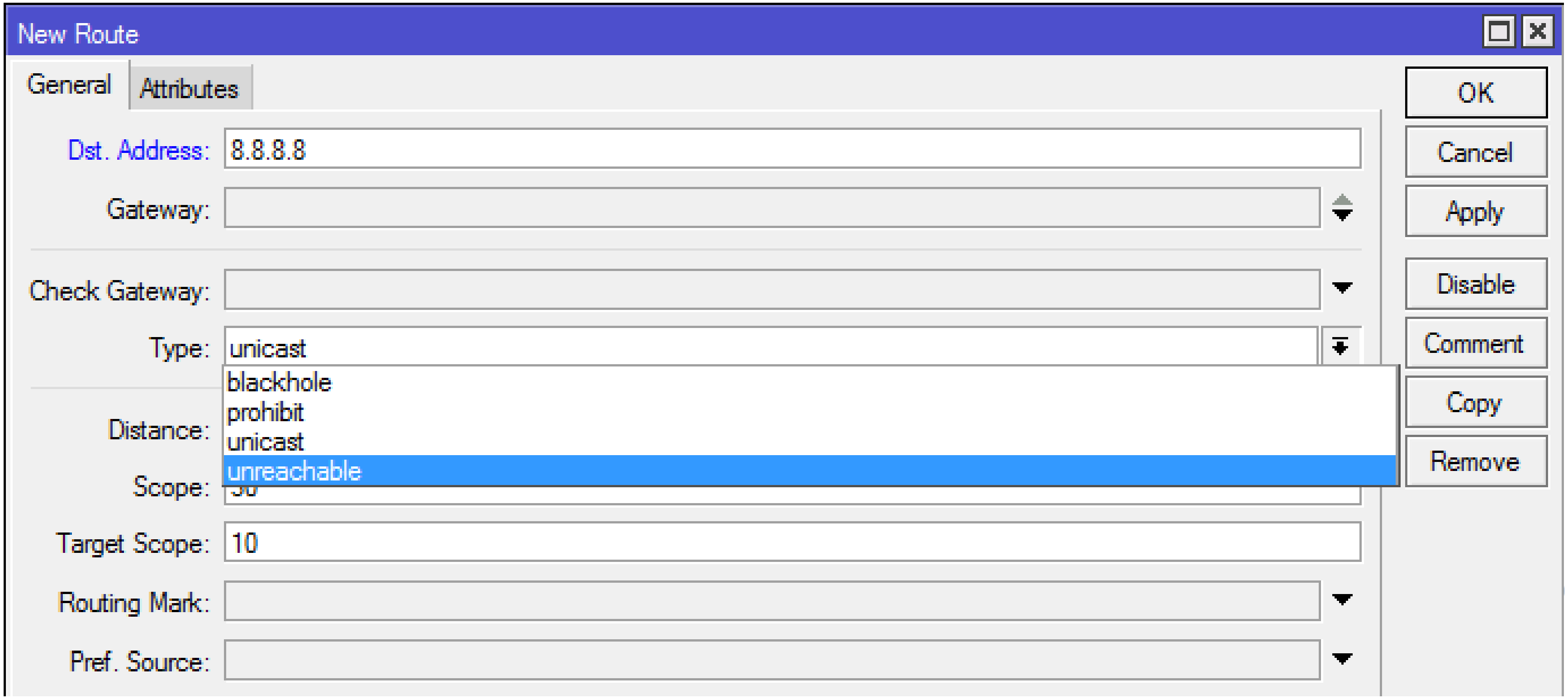
Features	Routing Table	IP-based Filter	Keyword Filter	Layer 7 Filter	Web Proxy	DNS
Filter specific IP address	YES	YES	NO	NO	YES	NO
Filter specific domain name	NO	NO	YES	YES	YES	YES
Filter specific web page	NO	NO	MAYBE	MAYBE	YES	NO
Filter specific protocol	NO	YES	NO	MAYBE	NO	NO
Filter specific keyword	NO	NO	MAYBE	MAYBE	NO	NO
Filter specific packet format	NO	NO	NO	YES	NO	NO

Routing Table

- Implementation
  - Use IP routing table (RIB) to drop or reject packets to specific destination IP address/subnet
- Use Case
  - Filter exact IP address/subnet
- Pros
  - Easy to implement
- Cons
  - Cannot selectively filter by source IP, all users are effected
  - Cannot do protocol-based, domain-based, or content-based filtering.
  - Interfere other websites on the same shared hosting

Routing Table – Implement

- Go to menu **IP → Routes**, create a new route:
  - **[Dst. Address]** is the IP/subnet you wanna block
  - Select **[Type]** as "**unreachable**", "**blackhole**", or "**prohibited**".



New Route

Dst. Address: 8.8.8.8

Gateway:

Check Gateway:

Type: unicast

Distance:

Scope: 30

Target Scope: 10

Routing Mark:

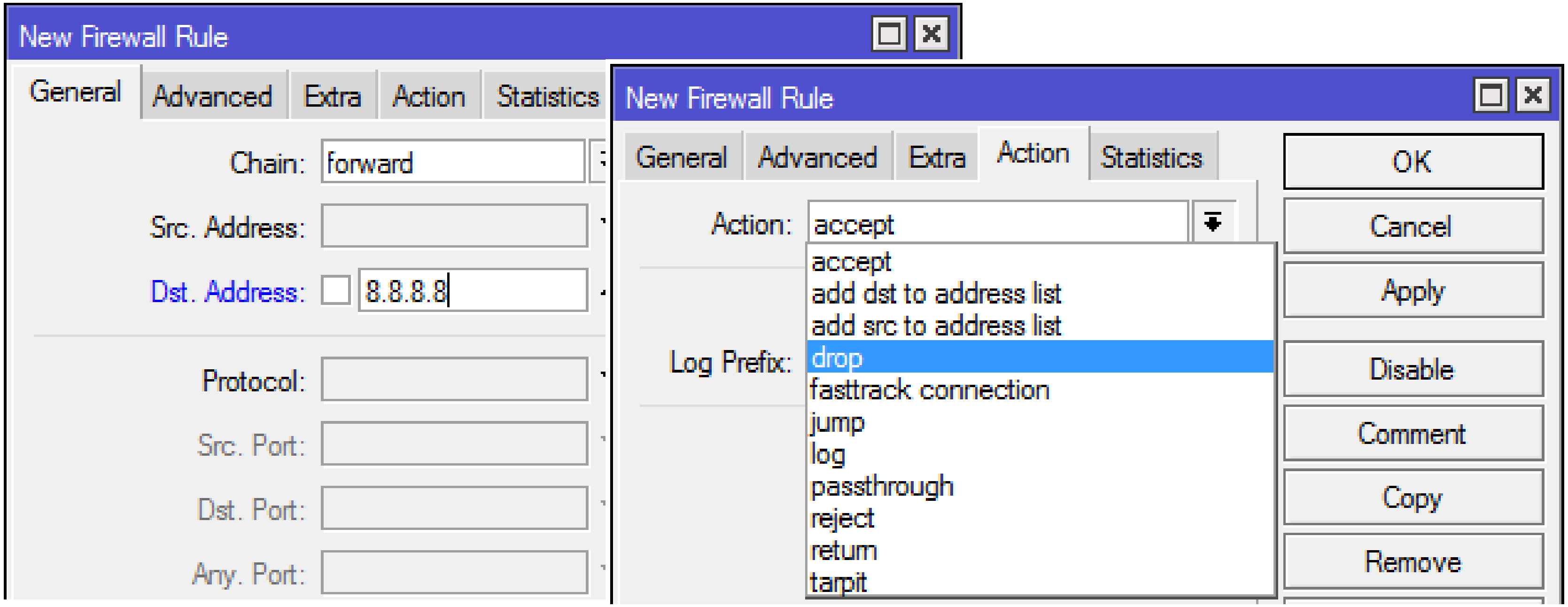
Pref. Source:

IP-BASED FILTER

- Implementation
  - Use IP Firewall to drop or reject packets based on source or destination address, protocol, and port.
- Use Case
  - Filter exact IP address/subnet
  - Filter protocol and port number
- Pros
  - Can do protocol-based filter
  - Can selectively apply in specific conditions (i.e. office hours)
- Cons
  - Cannot domain-based, or content-based filtering
  - Interfere other websites on the same shared hosting

IP-Based Filter – Implement

- Go to menu **IP → Firewall → Filter Rules**, create a new rule:
  - **[Dst. Address]** is the IP/subnet you wanna block
  - Specify **[Src. Address]** if you wanna block specific user only
  - **[Action]** can be "**drop**" or "**reject**"
  - Use **[Src. Address List]** and **[Dst. Address List]** for multiple IPs.



New Firewall Rule

Chain: forward

Src. Address:

Dst. Address: □ 8.8.8.8

Protocol:

Src. Port:

Dst. Port:

Any. Port:

New Firewall Rule

Action: accept

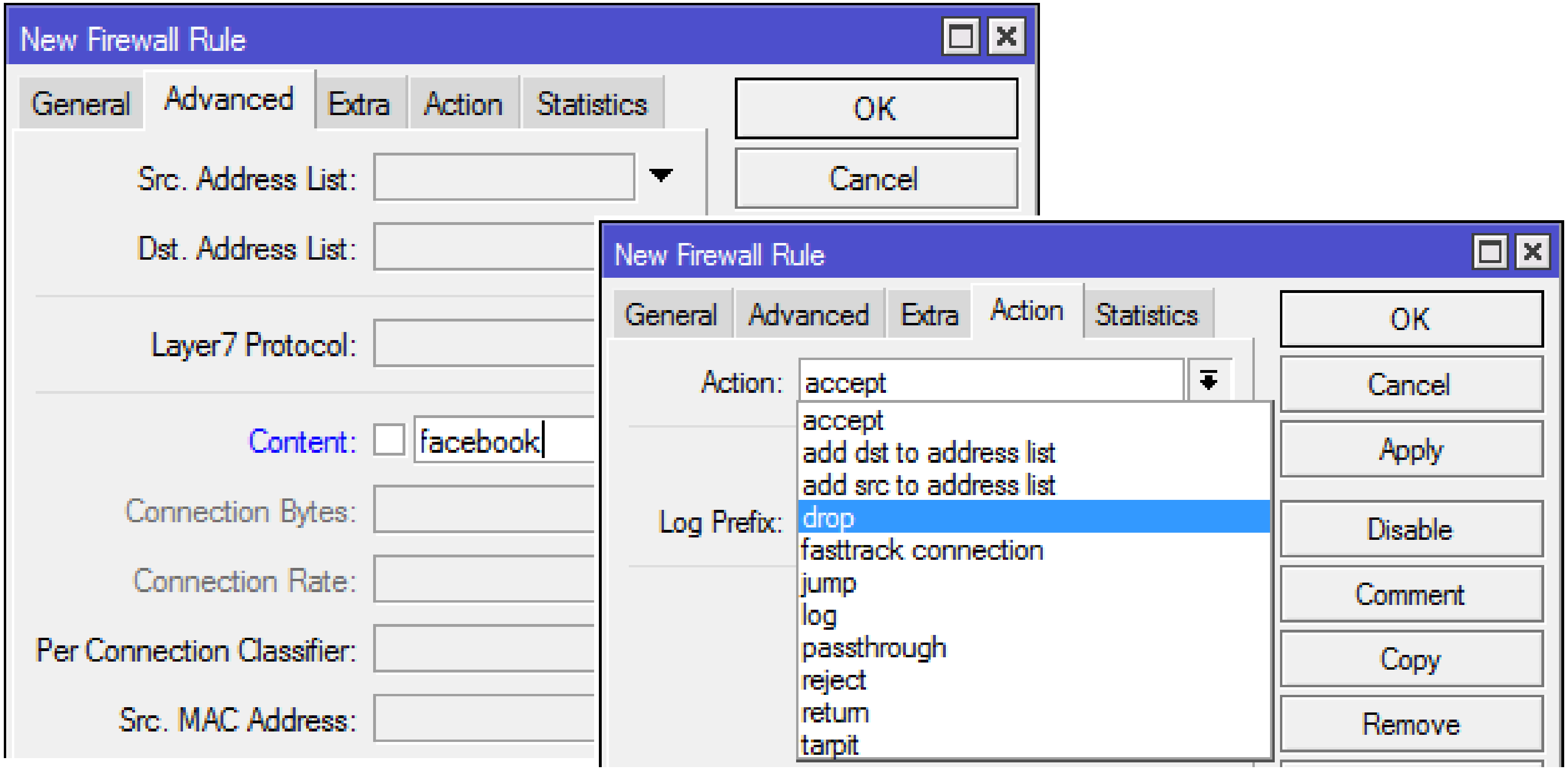
Log Prefix:

Keyword Filter

- Implementation
  - Use IP Firewall to drop or reject packets with specific keyword in the packet payload
- Use Case
  - Filter the word you don’t like
- Pros
  - You don’t really need to know what is the IP or domain of the content
- Cons
  - Encrypted contents (HTTPS) are not visible to the firewall rule
  - A bad website usually does not say they are bad ☺
  - Consider nature of packet fragmentation

Keyword Filter – Implement

- Go to menu **IP → Firewall → Filter Rules**, create a new rule:
  - Go to tab **[Advanced]**, fill in keyword in **[Content]**
  - **[Action]** can be "**drop**" or "**reject**"



New Firewall Rule

Src. Address List:

Dst. Address List:

Layer7 Protocol:

Content: □ facebook

Connection Bytes:

Connection Rate:

Per Connection Classifier:

Src. MAC Address:

New Firewall Rule

Action: accept

Log Prefix:

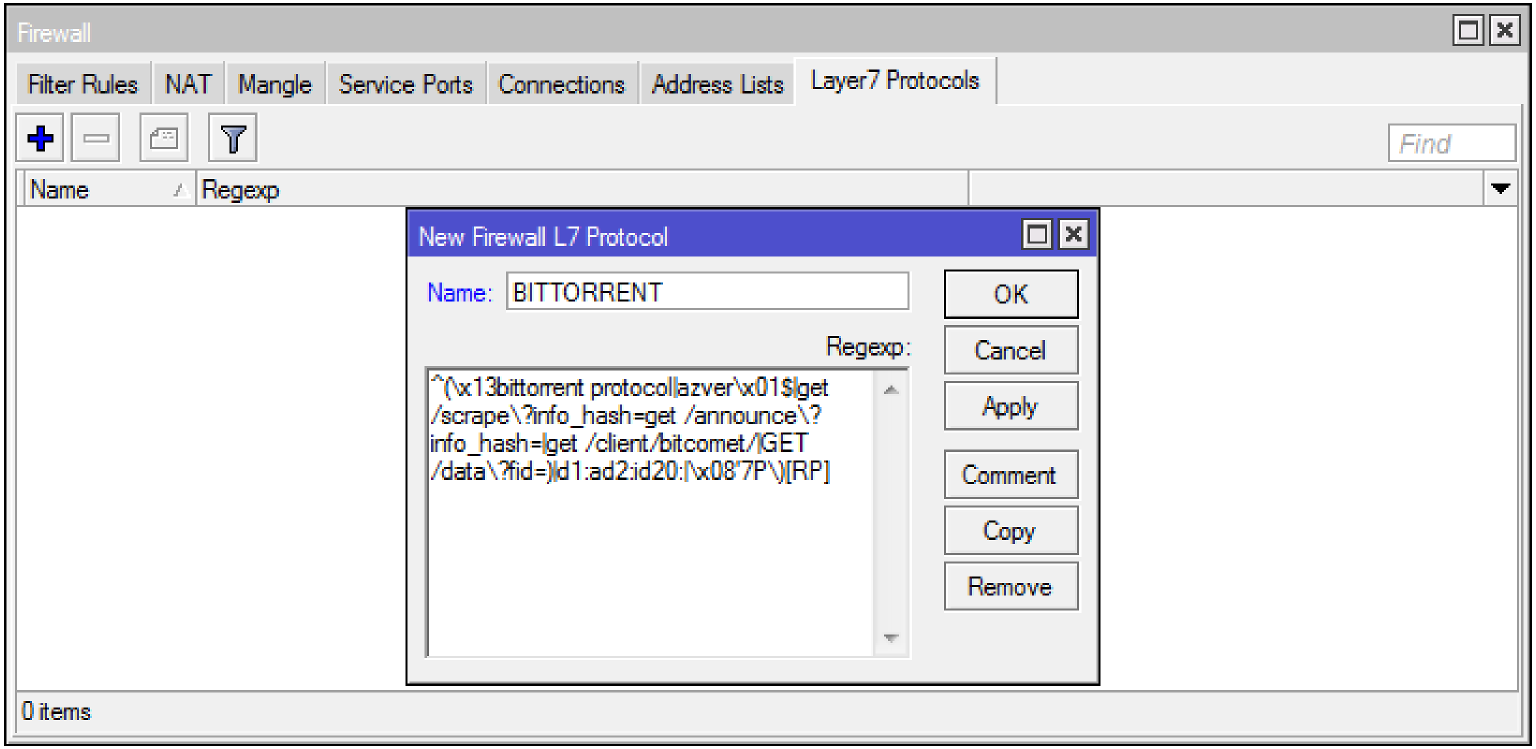
Layer 7 Filter

- Implementation
  - Use IP Firewall to drop or reject packets matched Layer 7 Regexp
- Use Case
  - Filter contents based on their packet format
  - Filter applications that don’t have specific port number
- Pros
  - Enhanced keyword matching in packet payload
  - Match various types of application, including some P2P software
- Cons
  - It is slow, high CPU consumption on matching Regular Expression
  - Cannot guarantee it will always work
  - Encrypted packets are not visible to Layer 7 filter

Layer 7 Filter – Implement (Step 1)

i-BEAM steering ahead

- Go to menu **IP → Firewall → Layer 7 Protocols**, create Layer 7 Protocol with Regular Expression for matching the packets.



Name: BITTORRENT

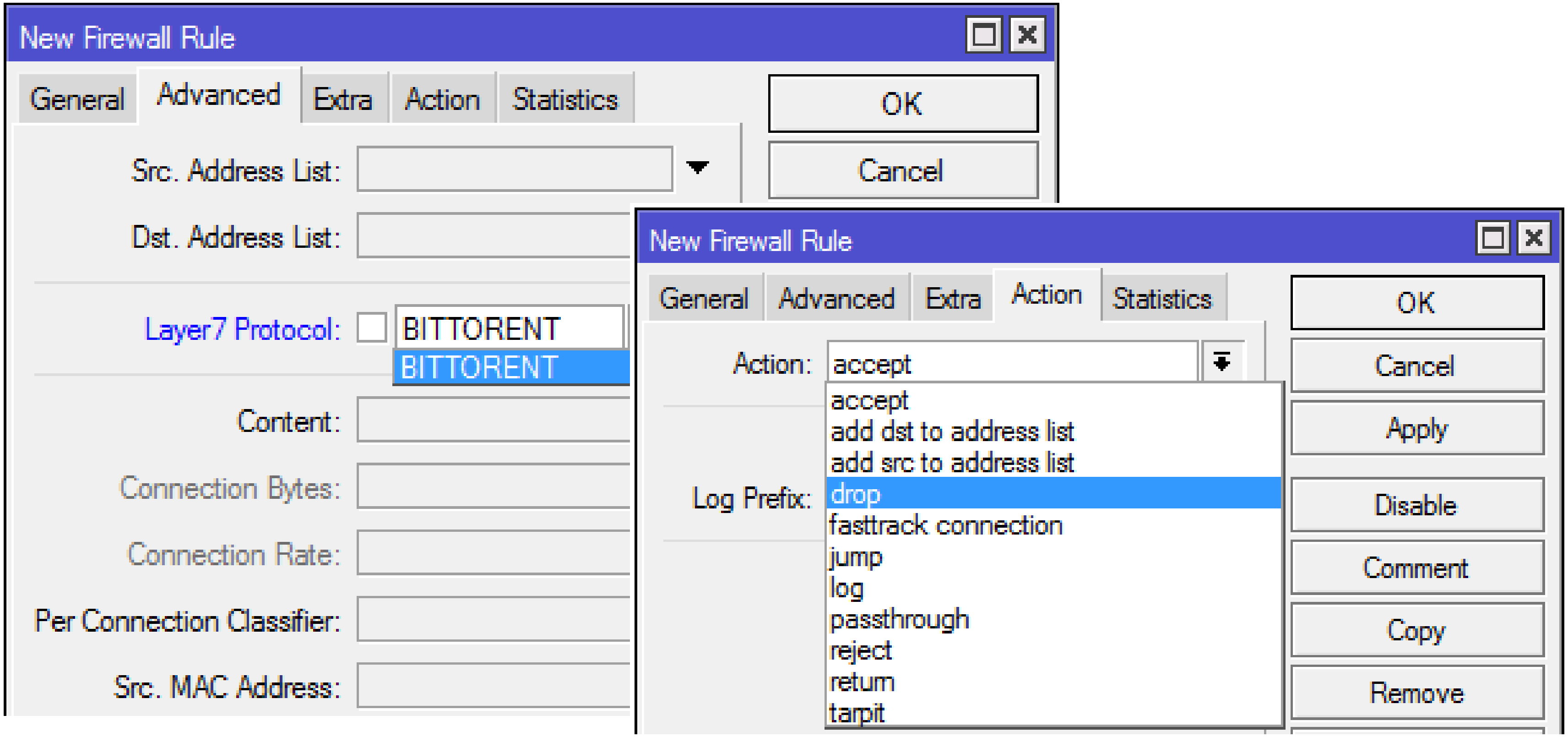
Regexp:

```
^(\x13bittorrent protocol|azver\x01$|get
/scrape\?info_hash=get /announce\?
info_hash=|get /client/bitcomet/|GET
/data\?fid=)|d1:ad2:id20:|\x08'7P\)[RP]
```

- Reference: http://l7-filter.sourceforge.net/protocols

Layer 7 Filter – Implement (Step 2)

- Go to menu **IP → Firewall → Filter Rules**, create a new rule:
  - Go to tab **[Advanced]**, select your created **[Layer 7 Protocol]**
  - **[Action]** can be “**drop**” or “**reject**”



New Firewall Rule

Src. Address List:

Dst. Address List:

Layer7 Protocol: □ BITTORENT

Content:

Connection Bytes:

Connection Rate:

Per Connection Classifier:

Src. MAC Address:

New Firewall Rule

Action: accept

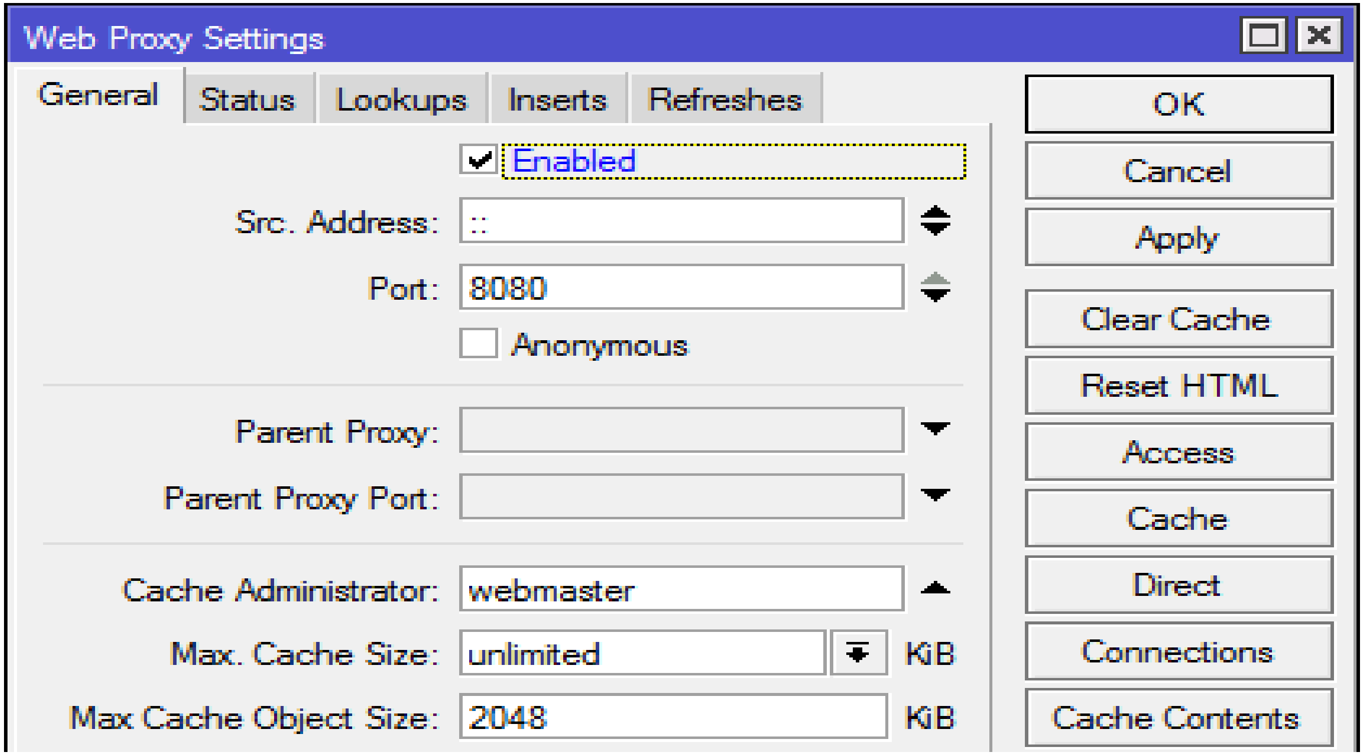
Log Prefix:

Web Proxy

- Implementation
  - Transparently redirect all HTTP requests to the router's Web Proxy, define Access rules to allow/deny websites
- Use Case
  - Filter specific website or web page
- Pros
  - Can block specific website without interfering other websites on shared hosting
  - Can block specific page of a website
  - Can do redirection
- Cons
  - So far it does not support HTTPS
  - Performance is not good for busy networks

WEB PROXY – IMPLEMENT (STEP 1)

- Enable Web Proxy in menu **IP → Web Proxy**



☑ Enabled

Src. Address: ::

Port: 8080

□ Anonymous

Parent Proxy:

Parent Proxy Port:

Cache Administrator: webmaster

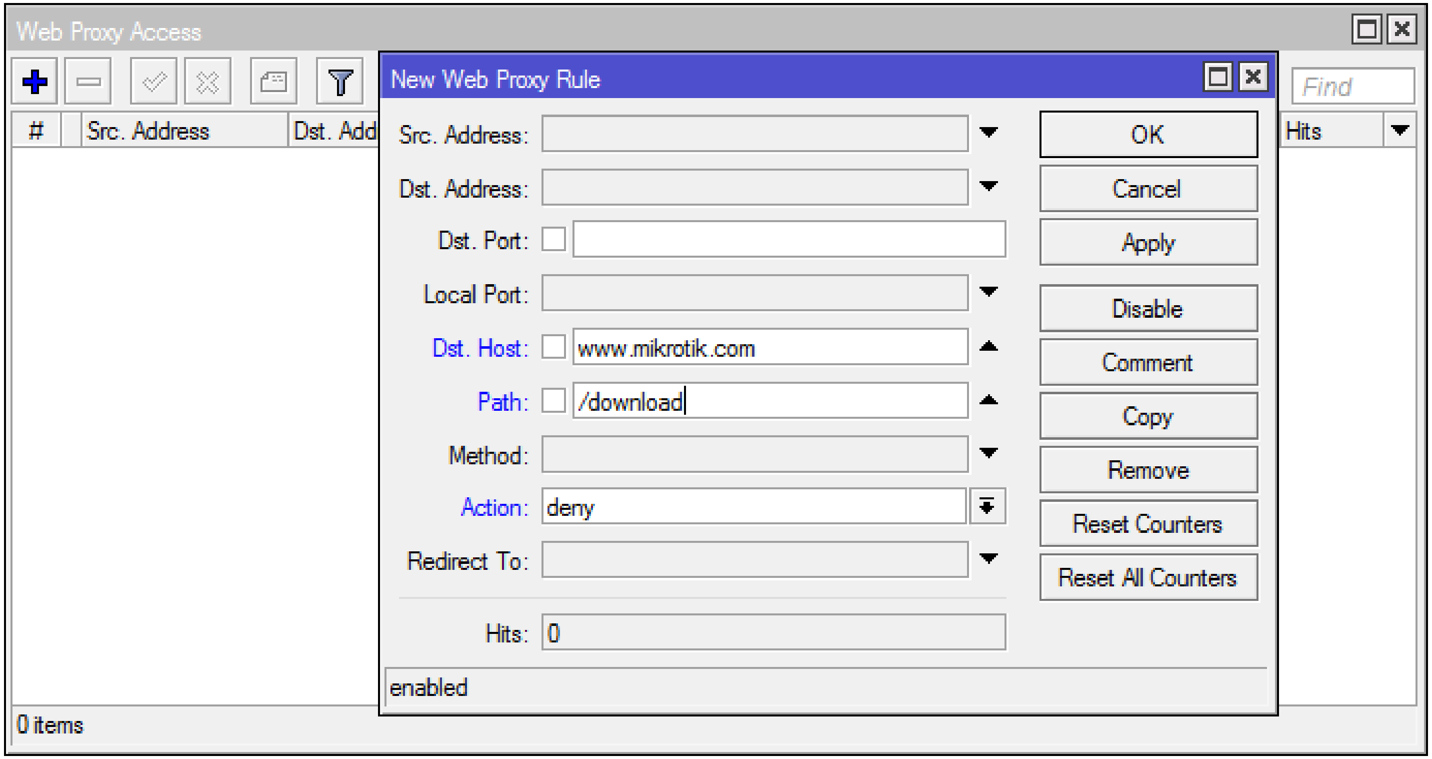
Max. Cache Size: unlimited KiB

Max Cache Object Size: 2048 KiB

- Click on **[Access]** button to configure filtering rules.

Web Proxy – Implement (Step 2)

- Create new Web Proxy Rule:
  - **[Dst. Host]** is domain name, **[Path]** is page URL
  - Select **[Action]** “**deny**”
  - Specify URL in **[Redirect To]** to redirect user to another site



Web Proxy Access

New Web Proxy Rule

Src. Address:

Dst. Address:

Dst. Port: □

Local Port:

Dst. Host: □ www.mikrotik.com

Path: □ /download

Method:

Action: deny

Redirect To:

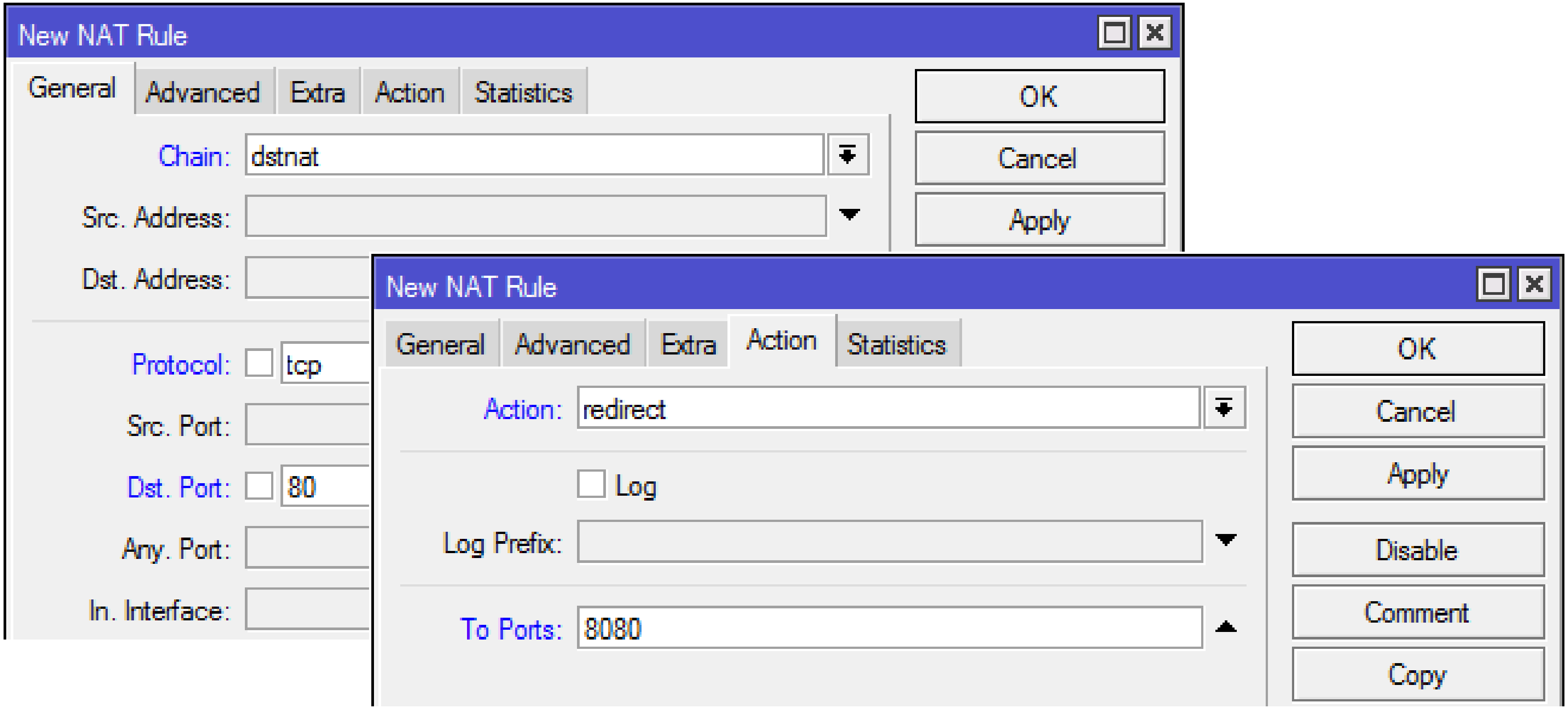
Hits: 0

enabled

0 items

WEB PROXY – IMPLEMENT (STEP 3)

- Go to menu **IP → Firewall → NAT**, create a new rule:
  - **[Chain]** is “**dstnat**”, **[Protocol]** is “**tcp**”, **[Dst. Port]** is “**80**”
  - **[Action]** is “**redrect**”, and **[To Ports]** “**8080**”



New NAT Rule — General

Chain: dstnat

Protocol: □ tcp

Dst. Port: □ 80

New NAT Rule — Action

Action: redirect

□ Log

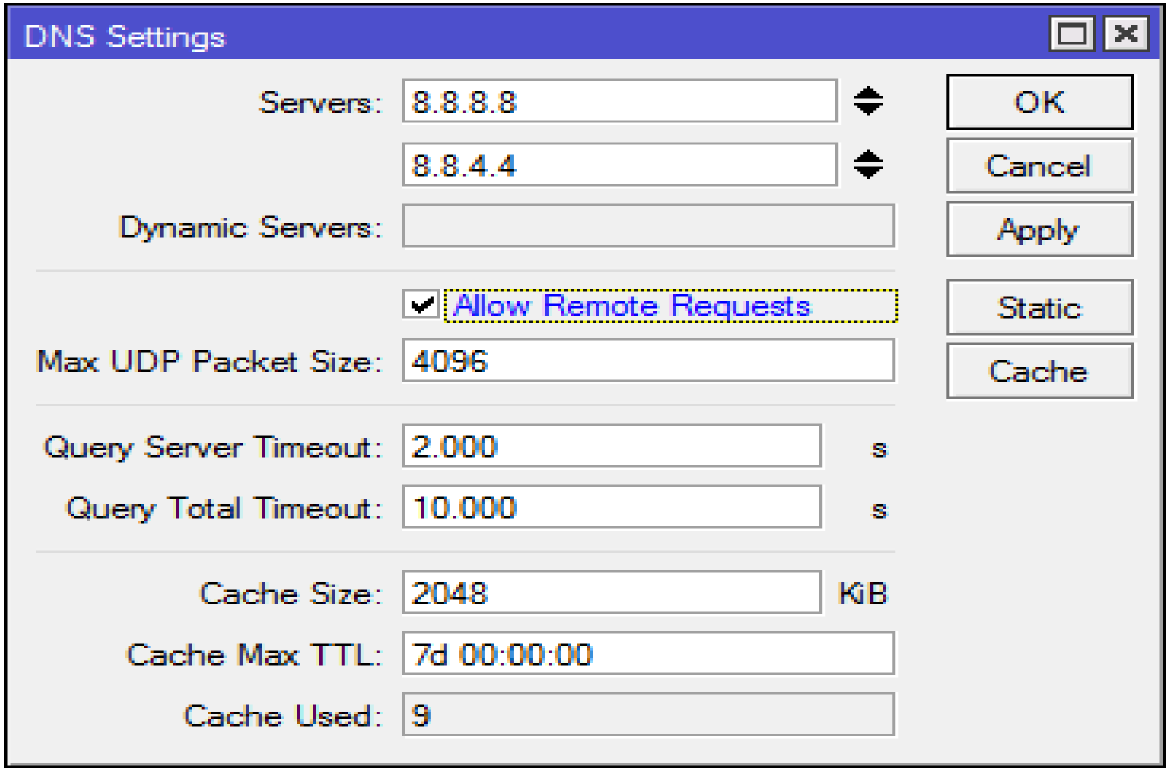
To Ports: 8080

DNS

- Implementation
  - Transparently redirect all DNS requests to the router’s DNS Server, create fake records to manipulate the DNS replies
- Use Case
  - Filter specific domain name
- Pros
  - It is fast and effective
  - Works on all protocols, as long as they use domain name
- Cons
  - Cannot selectively filter specific protocol or specific web page
  - Applications which connect directly to IP addresses won’t be filtered (i.e. Facebook App on smart phones)
  - Creates interference to the fake IPs that you manipulated

DNS – Implement (Step 1)

- Enable access to router DNS Server in menu **IP → DNS**



DNS Settings

Servers: 8.8.8.8

8.8.4.4

Dynamic Servers:

☑ Allow Remote Requests

Max UDP Packet Size: 4096

Query Server Timeout: 2.000 s

Query Total Timeout: 10.000 s

Cache Size: 2048 KiB

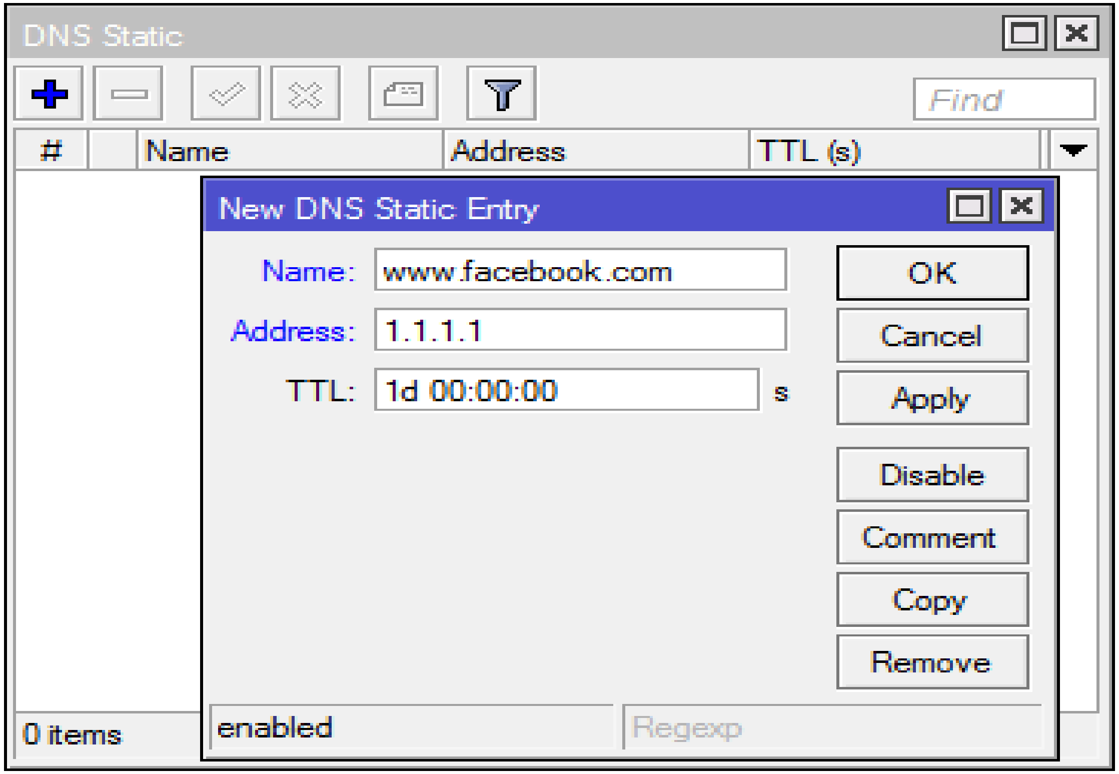
Cache Max TTL: 7d 00:00:00

Cache Used: 9

- Click on **[Static]** button to create fake records for domains

DNS – Implement (Step 2)

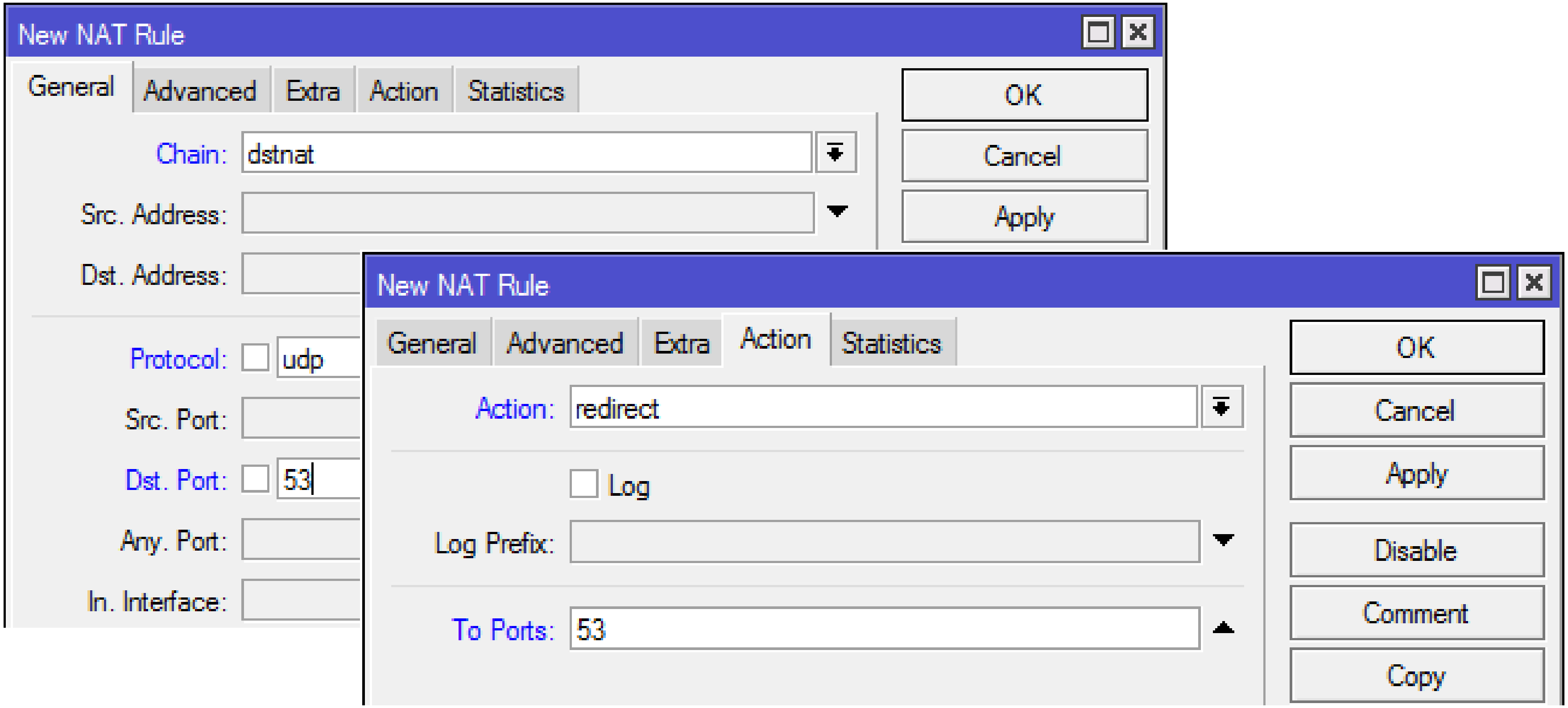
- Create new DNS Static Entry:
  - **[Name]** is domain name you wanna manipulate
  - Fill in fake IP in **[Address]** field, can be any IP that you think it will never be reachable by users.



Name: www.facebook.com
Address: 1.1.1.1
TTL: 1d 00:00:00 s
enabled

DNS – Implement (Step 3)

- Go to menu **IP → Firewall → NAT**, create a new rule:
  - **[Chain**] is “**dstnat**”, **[Protocol]** is “**udp**”, **[Dst. Port]** is “**53**”
  - **[Action]** is “**redrect**”, and **[To Ports]** “**53**”



New NAT Rule

Chain: dstnat

Src. Address:

Dst. Address:

Protocol: □ udp

Src. Port:

Dst. Port: □ 53

Any. Port:

In. Interface:

New NAT Rule

Action: redirect

□ Log

Log Prefix:

To Ports: 53

So…What Should We Use?

- **There is no single solution that can do everything ☺**
- Review your requirements and select the most suitable solutions.
- Filtering Suggestions:

Content	Solutions
Facebook	Routing Table, IP-based Filter, DNS
YouTube	DNS
HTTP websites	Web Proxy, Keyword Filter
HTTPS websites	DNS
Skype	Layer 7 Filter
LINE	Routing Table, IP-based Filter
Torrents	Layer 7 Filter

Useful Resources

- Facebook Address List
  - https://www.facebook.com/download/1635317286685519/address_list_FACEBOOK.txt
- Google Address List
  - https://www.facebook.com/download/1503947393258558/address_list_GOOGLE.txt

- i-BEAM Facebook Group
  - All presentations done by i-BEAM members (not only MUM) will be uploaded here!
  - https://www.facebook.com/groups/1481854632142914/
- i-BEAM Facebook Page
  - Check upcoming trainings and get our most special offers!
  - https://www.facebook.com/informationbeam

Questions & Answers

If you have any questions, please feel free to ask!

The End

Thanks for your attention!

Contact Me

makito@informationbeam.net

Skype: akn_makito

Phone: (+95) 09977423735

(+855) 011277300