

# 'Εξυπνες τεχνικές firewalling & traffic shaping

Topaloudis Michail – [mojiro.com](http://mojiro.com)

MUM in Athens

June 2015



# Λίγα λόγια για εμένα

- 10 χρόνια στο χώρο των δικτύων και τον προγραμματισμό
- MikroTik Certified Consultant με MTCNA και MTCWE πιστοποιήσεις
- Ειδίκευση σε
  - Firewall & IT Security
  - VPN Servers & Routing
  - Asset Monitoring
  - Centralized DHCP/DNS Control
  - Mikrotik API (integration with PHP, Drupal, .NET)

# Τι θα δούμε σήμερα;

- Αρχικά, μία σύντομη παρουσίαση της λογικής του Mikrotik Firewall
- Έπειτα, αναφορά στα βασικά εργαλεία που μας δίνει η Mikrotik
- Συνεχίζοντας, θα δούμε τις «έξυπνες» λειτουργίες του Firewall
- Τέλος, θα γίνει παρουσίαση των πρωτοκόλλων Skype & Bittorrent

# Σκοπός της παρουσίασης

- Η παρουσίαση απευθύνεται στους
  - Νέους στο χώρο της Mikrotik ώστε να αντιληφθούν τις εκτεταμένες δυνατότητες του
  - Επαγγελματίες που θέλουν να υλοποιήσουν περίεργα σενάρια και δε γνωρίζουν τις δυνατότητες του Firewall

# Mikrotik Firewall

- Αποτελείται από 3 μέρη
  - Filter – Ο τροχονόμος των πακέτων
  - NAT – Αναδρομολόγηση
  - Mangle – Επεξεργασία και τροποποίηση στα πακέτα
- Λογική IF-THEN
  - **IF** – Δημιουργούμε ένα κανόνα, ώστε να «περιγράψουμε» τα πακέτα
  - **THEN** – Στη συνέχεια ορίζουμε το Action που θα τους «συμβεί»

# Firewall Chains

- Το κάθε ένα από τα **Filter** / **NAT** / **Mangle**, διαχωρίζεται στις λεγόμενες «αλυσίδες»
- Η κάθε «αλυσίδα» έχει το ρόλο της και σε αυτές εισάγουμε τους κανόνες
  - πχ. στην **Input** του **Filter**, εισάγουμε κανόνες για πακέτα που φτάνουν στο router
  - ενώ στην **Forward** του **Filter**, εισάγουμε κανόνες για πακέτα που διέρχονται από τον router

Filter	NAT	Mangle
Input		Input
	Pre-Routing / DST-NAT	Pre-Routing
Forward		Forward
	Post-Routing / SRC-NAT	Post-Routing
Output	Output	Output

# Λογική IF – THEN (1)

- **ΕΑΝ** ένα πακέτο «έρχεται» από IP 192.168.1.10 και «πάει» προς 32.6.26.78
- **ΤΟΤΕ** άστο να περάσει
- **ΕΑΝ** ένα πακέτο «έρχεται» από IP 192.168.1.12
- **ΤΟΤΕ** μπλόκαρε το!

απλό έτσι;

# Λογική IF – THEN (2)

IF – THEN

The screenshot shows the 'New Firewall Rule' dialog box. The 'General' tab is selected and circled in red. The 'Chain' dropdown is set to 'input'. The 'Action' tab is highlighted with a green circle, and a green arrow points to it. The right side of the dialog contains buttons for OK, Cancel, Apply, Disable, Comment, Copy, Remove, Reset Counters, and Reset All Counters.

Field	Value
Chain	input
Src. Address	
Dst. Address	
Protocol	
Src. Port	
Dst. Port	
Any. Port	
P2P	

Buttons: OK, Cancel, Apply, Disable, Comment, Copy, Remove, Reset Counters, Reset All Counters

# Τι κριτήρια έχω διαθέσιμα;

- Θα έχετε ίσως δει ότι τα DSL Router διαφημίζουν ότι έχουν Firewall
  - Αφενώς είναι φτωχά και αφετέρου όποιος βασίζεται σε αυτά ΔΕΝ είναι επαγγελματίας
- Στο Mikrotik θα βρείτε πραγματικά άπειρα!
- Ναι, είναι πραγματικά πολλοί οι τρόποι με τους οποίους μπορούμε να εντοπίσουμε ένα πακέτο
  - Διευθύνσεις & Δυναμικές Λίστες Διευθύνσεων, Πόρτες, Πρωτόκολλα, Interfaces, Κατάσταση Σύνδεσης, Περιεχόμενο, Ημέρα/ώρα, κλπ

# Υπάρχουν όμως και τα special, ε;

- Ως special θα βαφτίζαμε αυτά τα οποία αλλάζουν λίγο τη «ροή» της ιστορίας ενός πακέτου εκτός από ένα απλό **Accept / Drop**
- Αυτά είναι:
  - Address Lists & Add to Address List σε συνδυασμό με το Timeout
  - Custom Chains & Jump to chain
  - Connection / Packet / Routing Marks & Marking

# Και γιατί είναι τόσο σημαντικά; (1)

- Σε ένα συνηθισμένο Firewall, οι κανόνες
  - τοποθετούνται ο ένας κάτω από τον άλλο σε συγκεκριμένες λίστες κανόνων
  - εκτελούνται με τη σειρά που έχουν τοποθετηθεί
  - δεν υπάρχει δυνατότητα να προστεθούν νέες διευθύνσεις IP μέσα στους κανόνες

## Main Firewall Chain



1. Accept HTTP



2. Drop Attacks



3. Drop Torrents

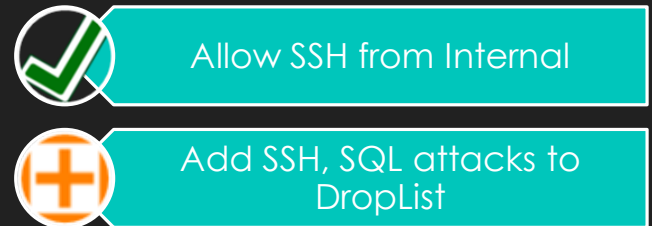
# Και γιατί είναι τόσο σημαντικά; (2)

- Στο Mikrotik όμως οι κανόνες είναι κάπως έτσι ...

## Main Firewall Chain



## Attack Chain (Custom)



## Traffic Shaping



## Routing Table



# Και γιατί είναι τόσο σημαντικά; (3)

- Δηλαδή στο Mikrotik, οι κανόνες εξακολουθούν και είναι σε λίστες και να εκτελούνται με τη σειρά, αλλά αλληλεπιδρούν!
  - μεταξύ τους
  - με τα **Queues**
  - τα πολλαπλά **Routing Tables**
- Άρα έχουν λειτουργίες που το καθιστούν ένα δυναμικό Firewall

# Trick A (1)

## Address Lists

- Όποιος προσπαθεί να συνδεθεί απευθείας από το Internet στο router μας, μέσω Telnet / SSH, πιθανόν να έχει μόνο κακόβουλες προθέσεις ...
- Άρα θα πρέπει
  - να λάβει γενικό αποκλεισμό από παντού
  - να αποκλειστεί και η δική μας πρόσβαση προς αυτόν
    - διότι μπορεί να έχει περάσει malware στο δίκτυο μας, το οποίο θα προσπαθήσει πιθανόν να συνδεθεί σε αυτόν

# Trick A (2)

## Address Lists – Κώδικας

Όποιος προσπαθεί να συνδεθεί στην Telnet / SSH Port θα λαμβάνει γενικό Block, εκτός και αν είναι στην accept-list

```
/ ip firewall filter
add action=accept      chain=input  src-address-list=accept-list
add action=drop        chain=input  src-address-list=drop-list

add action=add-src-to-address-list \
    address-list=drop-list \
    address-list-timeout=1d \
    chain=input \
    dst-port=22-23 \
    protocol=tcp
...
```

# Trick A (3)

## Address Lists – Κώδικας

Αυτός που επιχείρησε να συνδεθεί με Telnet / SSH λογικά είναι απρόσκλητος, άρα πρέπει να εμποδιστεί γενικώς!

```
/ ip firewall filter
add action=drop chain=input src-address-list=drop-list
add action=drop chain=forward src-address-list=drop-list
add action=drop chain=forward dst-address-list=drop-list
add action=drop chain=output dst-address-list=drop-list

/ ip firewall nat
add action=accept chain=dst-nat dst-address-list=drop-list
add action=accept chain=src-nat src-address-list=drop-list
```

# Trick A (4)

## Address Lists – Παρατηρήσεις

- Το `action=accept` στο `NAT` ισοδυναμεί με το `drop` του `Filter`
- Άμα η `address-list` που έχουμε ορίσει αρχίζει και αποκτά μεγάλο αριθμό διευθύνσεων, τότε θα πρέπει να ορίζουμε υποχρεωτικά `timeout`
  - Σε αντίθετη περίπτωση, θα γεμίζει ασταμάτητα, και θα αυξάνεται το φόρτο της CPU / RAM
  - Γενικά δεν υπάρχει λόγος να υπάρχει μία διεύθυνση εκεί για πάντα

# Trick B (1)

## Custom Chains

- Έστω ότι τους κανόνες που γράψαμε στο προηγούμενο Trick, θέλουμε να τους χρησιμοποιήσουμε και σε άλλα σημεία του **Filter**
- Τι θα κάνουμε; Θα γράφουμε συνέχεια τα ίδια και τα ίδια;
- Και αν στο μέλλον θέλουμε να κάνουμε μία αλλαγή;
  - Θααααααα την κάνουμε στα 4-5 διάσπαρτα σημεία...
- Για αυτό υπάρχουν τα **Custom Chains**, δηλαδή αλυσίδες σαν την **Input**, **Forward**, **Output**, που τις φτιάχνουμε εμείς

# Trick B (2)

## Custom Chains – Κώδικας

Οπότε οι κανόνες του προηγούμενου trick τροποποιούνται ως εξής

```
/ ip firewall filter
add action=jump          chain=input          jump-target=attack-chain
add action=jump          chain=forward        jump-target=attack-chain

add action=accept        chain=attack-chain    src-address-list=accept-list
add action=drop          chain=attack-chain    src-address-list=drop-list

add action=add-src-to-address-list \
    address-list=drop-list \
    address-list-timeout=1d \
    chain=attack-chain \
    dst-port=22-23 \
    protocol=tcp

add action=return        chain=attack-chain
```

# Trick B (3)

## Custom Chains – Παρατηρήσεις

- Αν σε μία **Custom Chain** δε βάλουμε ως τελευταίο κανόνα **action=return** τότε ο έλεγχος σταματά εκεί
  - Με το **action=return**, ο έλεγχος επιστρέφει στη προηγούμενη αλυσίδα και συνεχίζει μετά από το **jump**
- Ένα **Custom Chain** που έχει φτιαχτεί μέσα στο **Filter**, μπορεί να κληθεί μόνο από κανόνες του **Filter**, κ.ο.κ.
- Μπορούμε να έχουμε jumps μεταξύ **Custom Chain**
  - Προσοχή! να μη προκαλέσουμε κυκλικάς καλούμενες **Chains**
- Όλα τα **Filter Chains** μπορούν να καλούνται από το **Hotspot** και τα **VPN Profiles** ;)

# Trick C (1)

## Marking

- Με τη λειτουργία **Marking** (υλοποιείται από το Mangle του Firewall), μας δίνεται η δυνατότητα να «σημαδεύουμε» τα πακέτα που τηρούν κάποια κριτήρια
- Τα μαρκαρισμένα πακέτα στη συνέχεια μπορούμε να τα επεξεργαστούμε πολλαπλώς
  - Να τα χειριστούμε σε **Custom Routing Tables**, πχ. πολλές γραμμές DSL
  - Να καθορίσουμε προτεραιότητες μεταξύ τους
    - Τόσο με **Queue Trees**
    - όσο και με **Simple Queues**

# Trick C (2)

## Marking

- Ενδεικτικά θα αναφέρω πόσο πολύ μπορούν να μπλεχτούν κάποιες λειτουργίες στο Mikrotik
- Στον **DHCP Server**, συγκεκριμένες MAC-Addresses θα μπουν σε συγκεκριμένη **Address-List**, όταν συνδεθούν
  - Η **Address-List** θα σημαδευτεί με **Connection Mark** & **Routing Mark**
    - Το **Connection Mark** θα ελεγχθεί μέσω των **Queues**
    - Το **Routing Mark** θα βγει από διαφορετικό DSL Gateway

# Trick C (3) Marking

- Αντίστοιχα τρικ μπορούν να υλοποιηθούν και με
  - τα PPP User Profiles
  - τα Hotspot User Profiles
- Ωστόσο το Wireless δεν έχει απευθείας σύνδεση με Marks ή Address-Lists
  - Υπάρχει όμως τρόπος μέσω VLAN

# Trick D (1)

## Content Marking

- Όπως είπαμε η λειτουργία του **Marking** υλοποιείται μέσα από το **Mangle**
- Κάθε πακέτο μπορεί να σημαδεύεται με **Marks** όπως web, mail, chats, tunnels, κλπ
- Όποιο πακέτο έχει **Mark** πχ. mail θα πρέπει στα **Queues** να λάβει περισσότερο Bandwidth από τα web

# Trick D (2)

## Content Marking

### ○ Connections & Connection Marks

Firewall

Filter RulesNATMangleService PortsConnectionsAddress ListsLayer7 Protocols

Tracking

Find

	Src. Address	/	Dst. Address	Protocol	Conn...	Connection Mark	/	P2P	Timeout	TCP State	
U	:49727		:514	17 (udp)		unknown			00:00:08		
U	:1604		:5938	6 (tcp)		display			12:44:52	established	
U	:80		:10122	6 (tcp)		http			06:52:09	established	
U	:80		:55910	6 (tcp)		http			06:31:49	established	
U	:80		:23951	6 (tcp)		http			11:57:58	established	
A	:1116		:443	6 (tcp)		http			23:58:57	established	
A	:1413		:80	6 (tcp)		http			23:56:24	established	
A	:1534		:443	6 (tcp)		http			23:58:43	established	
A	:1575		:80	6 (tcp)		http			23:59:09	established	
A	:1576		:80	6 (tcp)		http			23:58:49	established	
A	:1578		:80	6 (tcp)		http			23:59:56	established	
A	:62102		:443	6 (tcp)		http			16:24:37	established	
A	:52104		:80	6 (tcp)		http			1d 00:00:00	established	
A	:34033		:443	6 (tcp)		http			23:59:07	established	
A	:50907		:443	6 (tcp)		http			23:58:49	established	
A	:42539		:443	6 (tcp)		http			23:58:49	established	
A	:62804		:5223	6 (tcp)		im			23:32:59	established	
A	:1524		:8291	6 (tcp)		mikrotik			00:04:10	established	
A	:41592		:179	6 (tcp)		routing			23:59:08	established	
U	:514		:514	17 (udp)		services			00:00:06		
A	:4620		:122	17 (udp)		services			00:00:02		

# Trick D (3)

## Content Marking - Κανόνες

- 1<sup>ος</sup> κανόνας
  - Κάθε Firewall & Traffic Shaping (Mangle) παραμετροποιείται βάση συνθηκών & απαιτήσεων του εκάστοτε έργου
    - ΔΕΝ βασιζόμαστε ποτέ σε templates
- 2<sup>ος</sup> κανόνας
  - Ποτέ δε θα είναι τέλειο, πάντα πρέπει να προσαρμόζεται

# Trick D (4)

## Content Marking - Μεθοδολογία

- Οι πρώτοι **Mark** κανόνες θα αφορούν πακέτα τα οποία είμαστε αρκετά σίγουροι για το περιεχόμενό τους
  - πχ. ICMP, Tunnels
- Οι επόμενοι κανόνες θα αφορούν υπηρεσίες οι οποίες συνηθίζουν να μασκαρεύονται και να κρυπτογραφούνται
  - πχ. Skype, TeamViewer, Torrents
- Επειδή ακριβώς αυτές οι υπηρεσίες δεν έχουν συγκεκριμένες πόρτες θα πρέπει να ανιχνεύονται με **L7 Patterns**
  - Τι είναι τα **L7 Patterns**;

# Trick D (5)

## Content Marking - Μεθοδολογία

- Γενικά πολλές υπηρεσίες τείνουν να χρησιμοποιούν παρατύπως τις πόρτες TCP 80 & 443 χωρίς όμως να περνάνε από αυτές κίνηση τύπου HTTP
- Με ένα **Layer 7 Pattern**, αντί να ελέγχουμε την πόρτα ενός πακέτου, διαβάζουμε το περιεχόμενό του!
- Υπάρχουν έτοιμα set για **L7**, ωστόσο θυμηθείτε τον 2<sup>ο</sup> κανόνα
  - Οι εταιρίες με κρυπτογραφημένο περιεχόμενο, τροποποιούν συνεχώς τη κρυπτογράφηση τους
    - Άρα και οι κανόνες **L7** μπορεί μετά από 1 έτος να μη λειτουργούν

# Trick D (6)

## Content Marking - Μεθοδολογία

- Οι επόμενοι κανόνες από τις κρυπτογραφημένες υπηρεσίες θα σχετίζονται με το καθαρό HTTP
- Ως τελευταίους κανόνες, συνήθως αφήνουμε ότι έχει να κάνει με Online Gaming καθώς, τα πακέτα αυτά είναι UDP και πολλές φορές συνεχούνται, είτε με SIP, είτε ακόμη χειρότερα με Torrents

# Λίγα λόγια για το Skype και το πρωτόκολλο Bittorrent

- Χρησιμοποιούν μονίμως κρυπτογράφηση
- Κρύβονται πίσω από πλήθος διαφορετικών πορτών (TCP/UDP) και διευθύνσεων
- Έχουν πάντα μικρά πακέτα
- Όμως ...
  - ανιχνεύουν το φόρτο του δικτύου με δικές τους μεθόδους
  - προσαρμόζουν το Bandwidth που απορροφούν ώστε να μην επιρρεάζουν τις υπόλοιπες λειτουργίες ενός δικτύου

# Τέλος;

- Ερωτήσεις;
- Απαντήσεις;