

Подход к настройке множества маршрутизаторов

Модульная параметризованная конфигурация

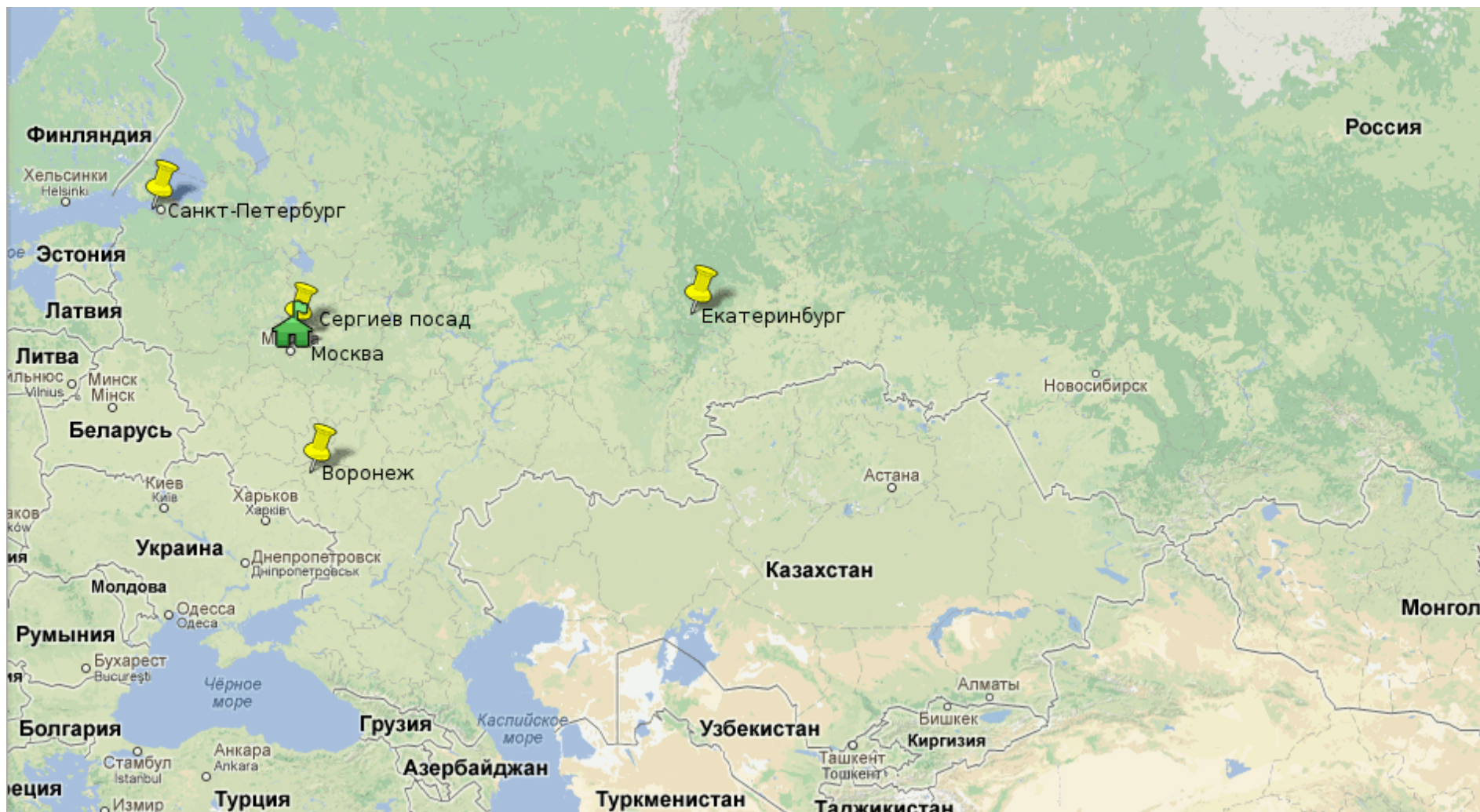
Михаил Москалёв



Routerz

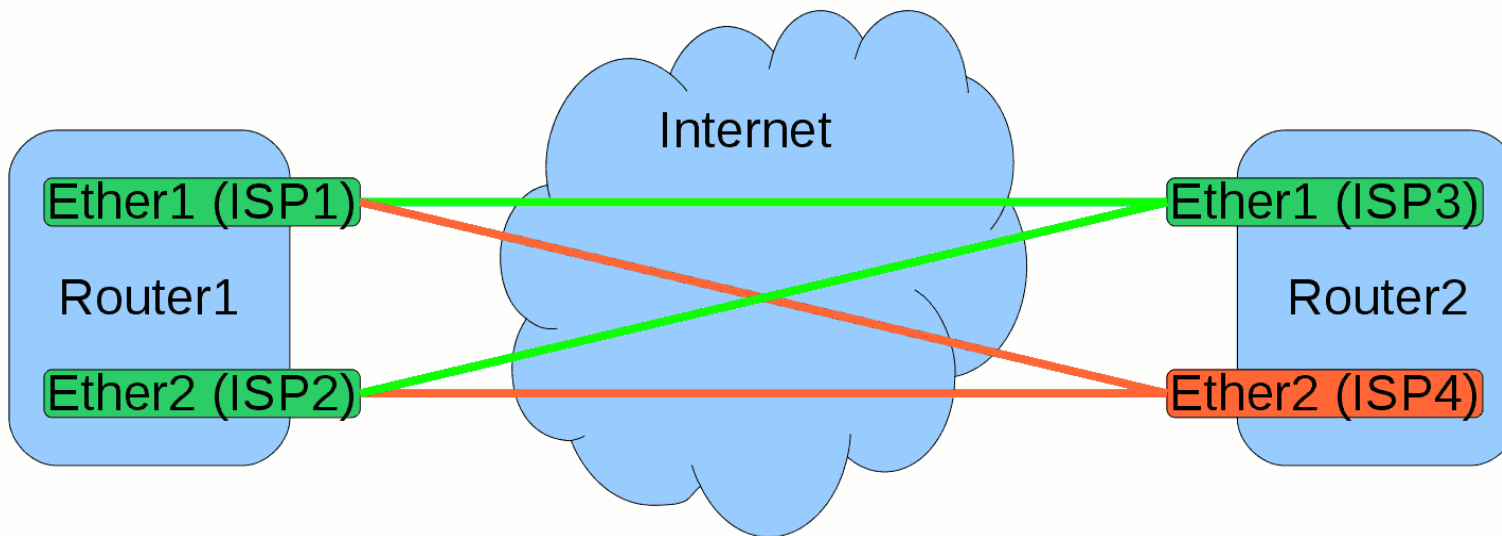


РОУТЕРЗ.РУ – учебный центр MikroTik в России и странах СНГ, дистрибьютор оборудования RouterBOARD.



RouterZ

Наша специализация — обучение и разработка решений на основе ПО и оборудования MikroTik.
Например: DualWAN VPN.



Москалёв Михаил –

- Технический директор РОУТЕРЗ.РУ.
- Владелец всех сертификатов MikroTik: MTCNA, MTCRE, MTCWE, MTSTCE, MTCUME, MTCINE.
- Первый и самый опытный тренер в РФ.
Сертифицированный тренер Mikrotik с 2011 года, сертификат # TR0125.
- Пользователь оборудования MikroTik с 2009 года.
- Занимаюсь сетями и базами данных с 1994 года.





ROUTERZ.RU —
координатор программы «[Mikrotik Academy](http://www.mikrotik.com/training/academy)¹»

Программа предполагает:

- бесплатное обучение и сертификацию преподавателей учебного заведения по курсу МТСНА и одному инженерному по выбору;
- интеграцию преподавания курса МТСНА в учебный план;
- бесплатное предоставление комплекта из 20 роутеров для выполнения лабораторных работ;
- возможность для студентов бесплатно получить международный сертификат МТСНА.

1 — подробнее <http://www.mikrotik.com/training/academy>

Цель

Ручное конфигурирование множества маршрутизаторов с помощью RSC-файлов.

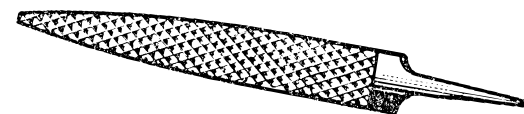
Рассмотрим несколько последовательно усложняющихся подходов.

Описанные способы могут быть интересны для развертывания от одного нового роутера в 1 — 3 месяца до десятков в день.

Конфигурирование RSC-файлом

Вариант 1а

- 1) Настраиваем эталонный роутер
- 2) Делаем `/export file=config`
- 3) Загружаем в новый роутер
- 4) При необходимости
«доработать напильником»



Конфигурирование RSC-файлом

Вариант 1а



Удобства

- Очевидность подхода
- Хорошо работает с простыми конфигурациями.



Неудобства

- В экспорт попадает много лишнего.
- Чем больше файл, тем больше столкновений с багами и несовместимостью.
- Доработка после заливки не удобна при множестве изменений.

Редактирование RSC-файлов

- Кодировка: ASCII, CP1251 или UTF-8 (для старых версий ROS только в однобайтовой кодировке).
- Unicode с BOM кодами в начале файла будут выдавать ошибку (такое часто случается при создании файла в Windows Notepad).
- Желательно чтобы концы строк были CR LF (стиль Windows).

Конфигурирование RSC-файлом

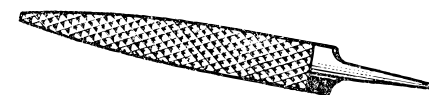
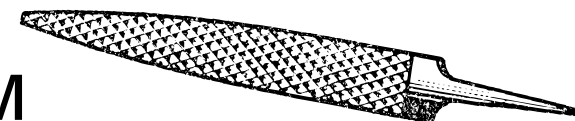
Вариант 1

```
/interface wireless
# managed by CAPsMAN
# channel: 2412/20-Ce/gn(18dBm), SSID: WI-FI, CAPsMAN forwarding
set [ find default-name=wlan1 ] adaptive-noise-immunity=ap-and-client-mode \
    ampdu-priorities=0,2 antenna-gain=2 band=2ghz-b/g/n channel-width=\
    20/40mhz-Ce country=russia distance=indoors frequency-mode=\
    regulatory-domain hw-protection-mode=cts-to-self hw-protection-threshold=\
    300 l2mtu=1600 mode=ap-bridge radio-name=ih-map-2n-2 ssid=MikroTik-85F956 \
    wireless-protocol=802.11
/ip neighbor discovery
# managed by CAPsMAN
# channel: 2412/20-Ce/gn(18dBm), SSID: WI-FI, CAPsMAN forwarding
set wlan1 discover=no
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
/system logging action
set 1 disk-file-name=log
set 2 remember=yes
set 3 src-address=0.0.0.0
/ip neighbor discovery settings
set default=no
/interface wireless cap
set discovery-interfaces=ether1 enabled=yes interfaces=wlan1
/ip dhcp-client
add dhcp-options=hostname,clientid disabled=no interface=ether1
/ip ipsec policy
set (unknown) dst-address=0.0.0.0/0 src-address=0.0.0.0/0
/ip upnp
set allow-disable-external-interface=no
/snmp
set trap-community=public
/system identity
set name=AP2
/system routerboard settings
set cpu-frequency=400MHz
/tool mac-server
set [ find default=yes ] disabled=yes
add interface=ether1
/tool mac-server mac-winbox
set [ find default=yes ] disabled=yes
add interface=ether1
```

Конфигурирование RSC-файлом

Вариант 16
(отредактированный файл)

- 1) Настраиваем эталонный роутер
- 2) Делаем `/export file=config`
- 3) Удаляем всё лишнее и тестируем
- 4) Загружаем в новый роутер
- 5) При необходимости
«доработать напильником»



Конфигурирование RSC-файлом

Вариант 16

Сценарий конфигурации после удаления ненужного

```
/interface wireless
set [ find default-name=wlan1 ] adaptive-noise-immunity=ap-and-client-mode \
    antenna-gain=2 distance=indoors frequency-mode=\
    regulatory-domain hw-protection-mode=cts-to-self hw-protection-threshold=\
    300 radio-name=map-2n-2 \
    wireless-protocol=802.11

/ip neighbor discovery settings
set default=no

/interface wireless cap
set discovery-interfaces=ether1 enabled=yes interfaces=wlan1

/ip dhcp-client
add dhcp-options=hostname,clientid disabled=no interface=ether1

/system identity
set name=AP2
```

Конфигурирование RSC-файлом

Вариант 16



Удобства

- Меньше проблем при заливке на разные модели роутеров.



Неудобства

- Всё еще сложно вносить изменения если файл большого размера.

Конфигурирование RSC-файлом

Вариант 16

(отредактированный файл)

Пока конфигурация не параметризована (все устройства настроены идентично), её можно заливать используя FlashFig.

Конфигурирование RSC-файлом

Вариант 2 (параметризация)

- Объявим в начале файла переменные
`:local RouterIdentity "myRouter-1"`
- При желании можно менять значение переменных в процессе загрузки
`:set RouterIdentity "rt$RouterIdentity"`
- И выполним подстановку значений переменных в нужных местах конфигурации.

Конфигурирование RSC-файлом

Вариант 2 (параметризация)

Синтаксис подстановки переменных

- Простой. Знак доллара и имя переменной
`:system identity set name=$RouterIdentity`
- Если имя переменной содержит разделители
`:system identity set name="$Router_Identity"`

Конфигурирование RSC-файлом

Вариант 2

Синтаксис подстановки переменных

- Можно выполнять подстановку переменной внутри строки. Просто:

```
:log info "Configuration of $RouterIdentity loaded"
```

- Или в более сложных случаях

```
:interface wireless  
  set wlan1 ssid="$mySSID"2"  
  set wlan2 ssid="$mySSID"5"
```

В данном случае, вложенные кавычки не требуют экранирования, поскольку первым идёт специальный символ - знак доллара.

Конфигурирование RSC-файлом

Вариант 2

- 
- 1) В начале конфигурации устанавливаем переменные
 - 2) В нужных местах указываем подстановку переменных
 - 3) Выполняем тестирование и отладку
 - 4) Меняем переменные в соответствии с требованиями
 - 5) Загружаем в новый роутер

Импорт RSC-файла и отладка

1. Можно просто скопировать команды из текстового редактора и вставить в командную строку роутера.

- Это просто и удобно для небольших фрагментов конфигурации в 1 — 5 строк.
- Можно не заметить ошибки при вставке большого количества команд.
- Это вообще не работает для блоков конфигурации более 4 Кб.

Импорт RSC-файла и отладка

2. Можно отредактировать в файле переменные и залить в роутер, а потом загрузить его командой `:import`.

- Это удобно для больших и очень больших конфигураций.
- В случае ошибки импорт остановится на ошибочной команде.

Отладка RSC-файлов

Сообщение об ошибке без знания команды которая её вызвала, не информативно.

```
> :import demo.rsc  
failure: already have device with such name
```

Включение параметра `verbose=yes` выводит каждую команду перед её выполнением.

```
[mmv@MMV Home] > import demo.rsc verbose=yes  
#line 1  
/interface ethernet  
#line 2..3  
set [ find default-name=ether1 ] comment=Strela mac-address=00:C1:28:02:02:2D \  
    name=ether1-strela  
#line 4  
set [ find default-name=ether5 ] comment=2Com name=ether5-2com  
#line 5  
/interface eoip  
#line 6..8  
add keepalive=1s,5 local-address=172.17.252.2 mac-address=\  
    02:E0:B3:01:2A:B6 mtu=1500 name=eoip-guests-wl2 remote-address=\  
    172.17.252.49 tunnel-id=99  
failure: already have device with such name  
[mmv@MMV Home] > █
```

Отладка RSC-файлов

Используя параметр from-line= можно пропустить часть строк RSC файла.

```
[mmv@MMV Home] > import demo.rsc verbose=yes from-line=9
#line 1, skip
/interface ethernet
#line 2..3, skip
set [ find default-name=ether1 ] comment=Strela mac-address=00:C1:28:02:02:2D \
    name=ether1-strela
#line 4, skip
set [ find default-name=ether5 ] comment=2Com name=ether5-2com
#line 5, skip
/interface eoip
#line 6..8, skip
add keepalive=1s,5 local-address=172.17.252.2 mac-address=\
    02:E0:B3:01:2A:B6 mtu=1500 name=eoip-guests-wl2 remote-address=\
    172.17.252.49 tunnel-id=99
#line 9
:log info "RSC import finished"
[mmv@MMV Home] > █
```

До появления этих параметров в ROS 6.1 и 6.2 соответственно, отладка была более трудоёмка.

Конфигурирование RSC-файлом

Вариант 3 (модульный)

- Единый файл большого размера (300 строк и более).
- Содержит различные функции (WAN IP, LAN IP, DHCP-server, VPN server, динамическую маршрутизацию, ...).

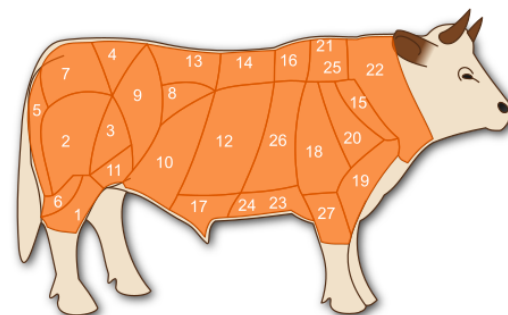


Конфигурирование RSC-файлом

Вариант 3 (модульный)

Разделяй и властвуй конфигурируй.

- Работает похоже на include.
- Каждый модуль в отдельном файле.



1. Geretto posteriore	10. Scalfo	19. Brione
2. rosa	11. Spinacino	20. Cappello del prete
3. Noce	12. Bianco costato di pancia	21. Reale
4. Codone	13. Controfiletto	22. Collo
5. Magatello o Girello	14. Costate	23. Punta di petto
6. Pesce o Piccione	15. Fusello	24. Fiocco
7. Fetta di mezzo	16. Coste della croce	25. Bianco costato di reale
8. Filetto	17. Pancia	26. Bianco costato della croce
9. Scamone	18. Fesone di spalla	27. Geretto anteriore

Конфигурирование RSC-файлом

Вариант 3 (модульный)

- Удобно подключать и отключать модули.
- Удобно использовать модуль отдельно от конфигурации.
- Параметры (потенциально приватные) отделены от кода модулей.
- Удобно компоновать новую конфигурацию из разных модулей.

Конфигурирование RSC-файлом

Вариант 3 (модульный)

- В основном файле, объявляются глобальные переменные и задаются их значения.
- Командами `:include` подключаются файлы модулей.
- В каждом модуле, объявляются используемые глобальные переменные и выполняются команды конфигурирования.

Конфигурирование RSC-файлом

Вариант 3 (модульный)

Пример главного файла

```
{
:global "interface-LAN" ether2-LAN
:global "interface-WAN" ether1-WAN

:global "WAN-IP" 192.0.2.206
:global "WAN-prefix" 30
:global "DefaultGW" 192.0.2.205

:import "upnp.rsc"
:import "WAN.rsc"
:import "sntp_ip_update.rsc" verbose=yes
#:import "ntp_ip_update.rsc" verbose=yes
}
```

* Подключать и отключать модули можно комментируя строки

Конфигурирование RSC-файлом

Вариант 3 (модульный)

Пример простого модуля — upnp.rsc

```
{
:global "interface-LAN"
:global "interface-WAN"

/ip upnp interfaces
add interface="$"interface-WAN"" type=external
add interface="$"interface-LAN"" type=internal

/ip upnp
set allow-disable-external-interface=no enabled=yes \
    show-dummy-rule=no
}
```

* Фигурные скобки нужны были чтобы переменные работали при вставке в командную строку в ранних версиях ROS.

Конфигурирование RSC-файлом

Вариант 3 (модульный)

Установка значений переменных по умолчанию. Модуль — sntp_ip_update.rsc

```
:global "primaryntp"
:global "secondaryntp"
:global "timezone"

:if ([ :len $timezone]=0) do={:set timezone "Europe/Moscow"}
:if ([ :len $primaryntp]=0) do={:set primaryntp "ru.pool.ntp.org"}
:if ([ :len $secondaryntp]=0) do={:set secondaryntp "2.ru.pool.ntp.org"}

# add parameter check and defaults

/system script
add name=sntp_client_check_update policy=ftp,read,write,test source="# tested on v5.24\r\
\n# work without ntp package\r\
\n# Resolve NTP server ip from name if time not synchronized\r\
\n#\r\
\n# Set ntp server names. Set to empty string to keep untouched\r\
\n:local primaryntp \"$\"primaryntp\"\"\r\
\n:local secondaryntp \"$\"secondaryntp\"\"\r\
\n\r\
...
```

Модуль с установкой переменных по умолчанию можно заливать отдельно, без установки переменных.

Конфигурирование RSC-файлом

Вариант 3 (модульный)

Поскольку RSC-скрипт по сути программа, нет ограничения на логику развертывания и проверку условий. Была бы возможность получить нужные данные.

Для этого нет строгой необходимости в разбиении на модули, но без этого сложные скрипты сложно сопровождать и разрабатывать.

Конфигурирование RSC-файлом

Вариант 3 (модульный)

«Гибкий» модуль — WAN.rsc

```
{
:global "interface-WAN"

:global "WAN-IP"
:global "WAN-prefix"
:global "DefaultGW"

:put "interface-WAN: $"interface-WAN" ["WAN-IP"/$"WAN-prefix]"

:if ([:len $"WAN-IP"] > 0) do={
  /ip address
    add interface=" $"interface-WAN" " address=" $"WAN-IP"/$"WAN-prefix" " comment="WAN"

  /ip route
    add comment="Gefault GW" gateway=" $"DefaultGW"

} else={
:put "No WAN-IP set. Use DHCP-client"
:if ([:len [:ip address find dynamic=no and interface="ether1-WAN"]] = 0) do={
  :put "Satatic IP set. Don't touch."
  :delay 3;
} else={
:put "No static IP set. Use DHCP-client."
/ip dhcp-client
:if ([:len [find where interface=" $"interface-WAN""]] = 0) do={
  add interface=" $"interface-WAN" " disabled=no
} else={
  :put "DHCP-client already configured";
  :delay 3;
}
}
}

/ip firewall nat
add action=masquerade chain=srcnat out-interface=" $"interface-WAN" "
}
```


Источники информации

- Описание синтаксиса языка сценариев
<http://wiki.mikrotik.com/wiki/Manual:Scripting>
- Полезные примеры использования скриптов
<http://wiki.mikrotik.com/wiki/Manual:Scripting-examples>
- Интересная статья в wiki о подходе к написанию сценариев конфигурации
http://wiki.mikrotik.com/wiki/How_to_Make_an_Automated_Configuration_and_Uninstall
- Утилита для обработки сценариев конфигурации
<http://wiki.mikrotik.com/wiki/Milliscript>
- Одна из веток с обсуждением сценариев в официальном форуме
<http://forum.mikrotik.com/viewtopic.php?t=40507>



СПЕЦИАЛЬНАЯ АКЦИЯ

MUM 2016 Moscow

RouterZ



- Посетите стенд ITElite на MUM
- Зарегистрируйтесь на тренинг *MTCNA Weekend*
в Москве 8-9 октября 2016
по специальной цене 9999 руб.!
Это всего 150USD

Спасибо за внимание!

Вопросы?
Дополнения?

tr@routerz.ru