



Сегодня мы расскажем как использовать трафик генератор Ostinato в стендах UNetLab. Ostinato это многопоточный генератор трафика, предназначенный для тестирования служб, обеспечивающих работу сети на различных уровнях стека сетевых протоколов.

К поддерживаемым протоколам относятся: Ethernet/802.3/LLC SNAP; VLAN (с поддержкой QinQ); ARP, IPv4, IPv6, IP-in-IP a.k.a IP Tunnelling (6over4, 4over6, 4over4, 6over6); TCP, UDP, ICMPv4, ICMPv6, IGMP, MLD; HTTP, SIP, RTSP, NNTP и многие другие.

Пользователю предоставляется возможность создавать пакеты данных произвольного содержания, определяя как заголовок пакета, так и содержимое всех его полей. Кроме содержимого пакетов можно выбрать интерфейс и частоту генерации трафика.

Ostinato обладает хорошо структурированным интерфейсом с наглядностью представления процесса работы.

Проект Ostinato предоставляет виртуальную машину Dron - собственно генератор, которую можно импортировать в GNS3 или UnetLab. Управление осуществляется через отдельный графический интерфейс, устанавливаемый под Windows

Дрон VM для QEMU можно скачать здесь
<http://www.bernhard-ehlers.de/projects/ostinato4gns3/ost-drone-0.7-v1.qcow2>

А GUI для Windows можно взять с официального сайта проекта
<http://ostinato.org/downloads.html>

На текущий момент это версия 0.7.1. Версия GUI должна совпадать с версией дрона.

На клиентской Windows машине выполняете обычную инсталляцию. В процесс будет также установлен и дрон под Windows, но мы не будем его использовать. Нам нужен только GUI интерфейс

Развёртывание Ostinato в стенде

Из меню Actions –Nodes добавьте с стенд объект Ostinato и обязательно выберите как минимум два интерфейса при создании.



Один интерфейс e0 подключите через облако к pnet0. Это управляющий интерфейс, через который будет управляться трафик генератор с вашего клиентского PC.

Второй интерфейс e1 – подключите к стенду, например к коммутатору. Это интерфейс, который будет посылать генерированные пакеты. Если предполагается, что пакеты будут dot1Q, то интерфейс коммутатора, к которому подключен e1, переведите в транк, в противном случае – оставьте access.

Запускаем Ostinato

```
UNL-VM | SW1 | Ostinato x
Loading /boot/vmlinuz.....
Loading /boot/core.gz.....
Loading /boot/ost-extension.gz.....ready.
Booting Core 6.3
Running Linux kernel 3.16.6-tinycore.
Checking boot options... Done.
Starting udev daemon for hotplug support... Done.
Scanning hard disk partitions to create /etc/fstab
Setting Language to C Done.
Possible swap partition(s) enabled.
Loading extensions... Done.
Setting keymap to us Done.
Restoring backup files from /mnt/sda1//mydata.tgz /
Done.
Skipping DHCP broadcast/network detection as requested on boot commandline.
Setting hostname to box Done.

Core Linux
box login: █
```

В приглашении ввести логин и пароль вводим tc без пароля и попадаем в linux-shell

```
box login: tc
C°-
//\  Core is distributed with ABSOLUTELY NO WARRANTY.
v_/_  www.tinycorelinux.com

tc@box:~$ █
```

Выполним команду **ifconfig -a**

```
tc@box:~$ ifconfig -a
cntl    Link encap:Ethernet  HWaddr 50:00:00:02:00:00
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:7 errors:0 dropped:0 overruns:0 frame:0
         TX packets:24 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:794 (794.0 B)  TX bytes:8208 (8.0 kib)

dummy0   Link encap:Ethernet  HWaddr 72:E9:96:6A:02:A9
         BROADCAST NOARP  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:0
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

eth0     Link encap:Ethernet  HWaddr 50:00:00:02:00:01
         UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Нас интересует интерфейс с именем **cntl**. Это интерфейс управления дроном. Дрон после запуска посылает DHCP запросы на данном интерфейсе на получения IP адреса . Если вы запускаете VM UNL под VmWare Workstation, с большой долей вероятности, дрон получит IP адрес автоматически, так как его интерфейс находится в той же сети, что и управляющий интерфейс UNL VM.

Но в случае, если это не произойдет, или ваша UNL VM работает под ESXi , то IP адрес на управляющий интерфейс дрона нужно назначить вручную. IP адрес должен быть из той же подсети,

что и управляющий интерфейс UNL VM.

```
sudo ifconfig cntl 172.16.1.90 netmask 255.255.255.0
```

Выполним еще раз команду **ifconfig cntl**

```
tc@box:~$ sudo ifconfig cntl 172.16.1.90 netmask 255.255.255.0
tc@box:~$
tc@box:~$ ifconfig cntl
cntl
Link encap:Ethernet  Hwaddr 50:00:00:02:00:00
inet addr:172.16.1.90  Bcast:172.16.1.255  Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:20 errors:0 dropped:0 overruns:0 frame:0
TX packets:72 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1574 (1.5 KiB)  TX bytes:24624 (24.0 KiB)

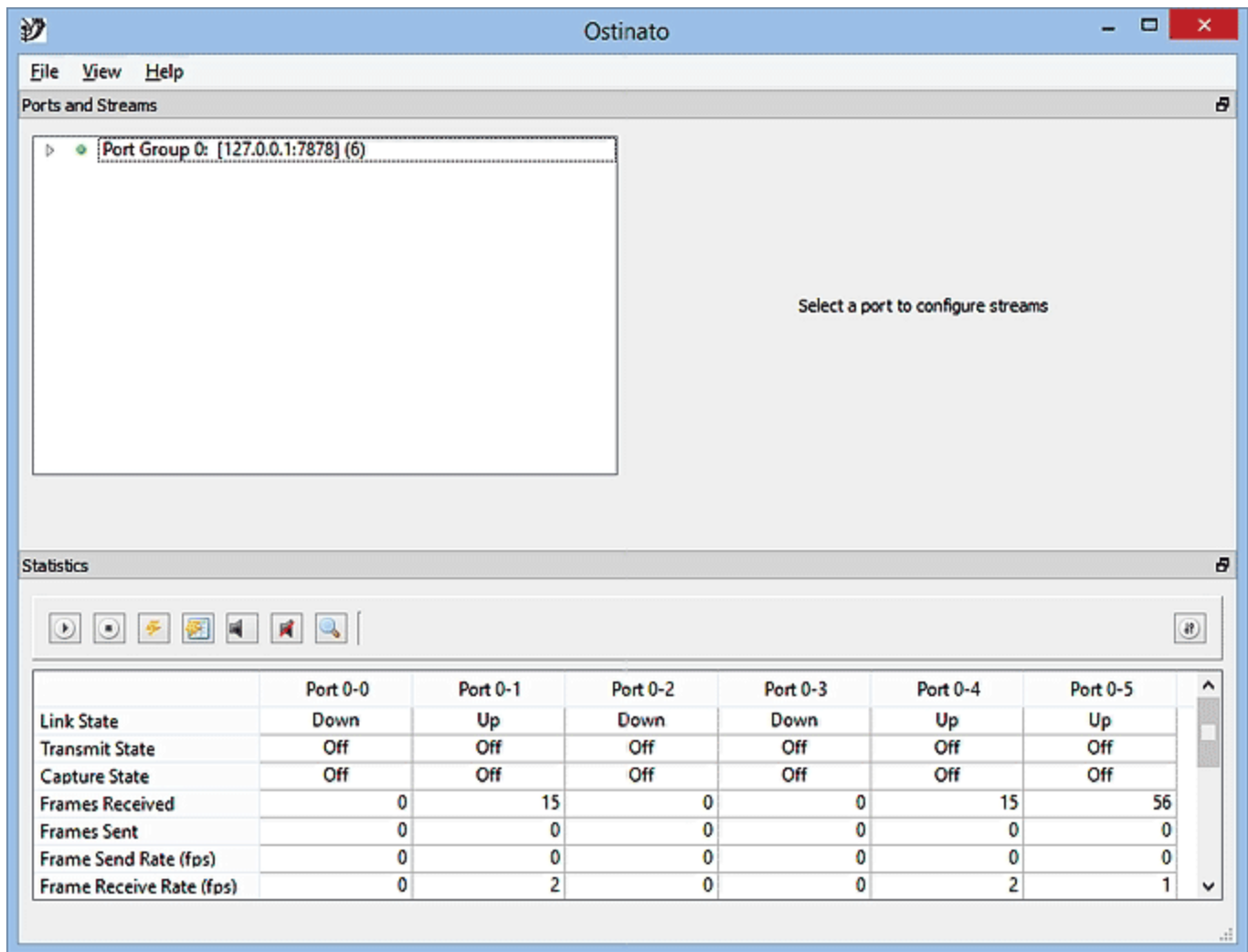
tc@box:~$
tc@box:~$ ping 172.16.1.91
PING 172.16.1.91 (172.16.1.91): 56 data bytes
64 bytes from 172.16.1.91: seq=0 ttl=64 time=0.969 ms
64 bytes from 172.16.1.91: seq=1 ttl=64 time=0.629 ms
64 bytes from 172.16.1.91: seq=2 ttl=64 time=0.535 ms
^C
--- 172.16.1.91 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.535/0.711/0.969 ms
```

Мы назначили IP адрес на интерфейс **cntl** и проверили связность управляющим интерфейсом VM UNL. Теперь дрон готов к управлению.

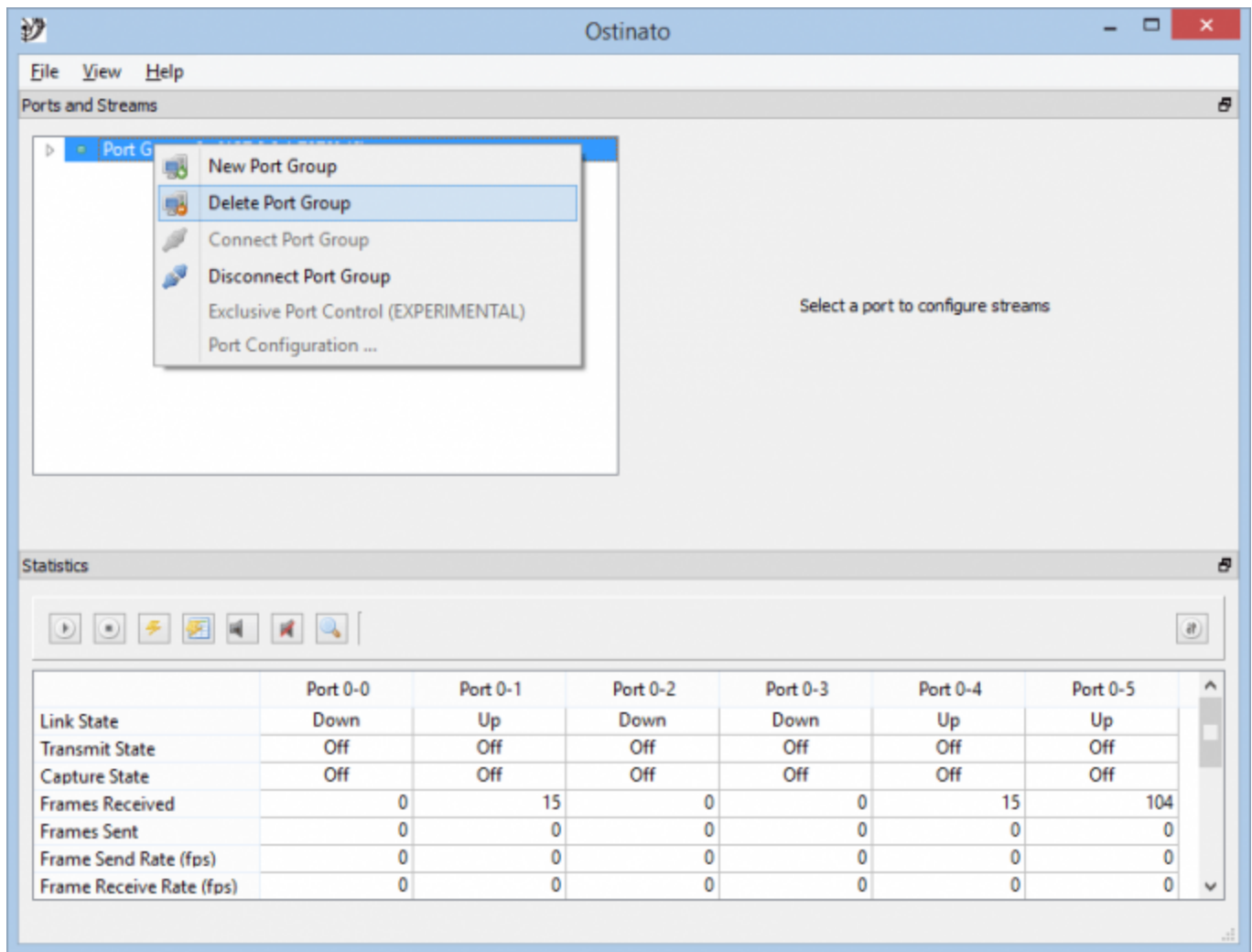
Второй интерфейса **eth0** внутри VM **Ostinato** виден как **e1** в UNL – это порт который будет генерировать пакеты в соответствии требованиям поступающим через интерфейс управления.

Настройка графического интерфейса управления **Ostinato**.

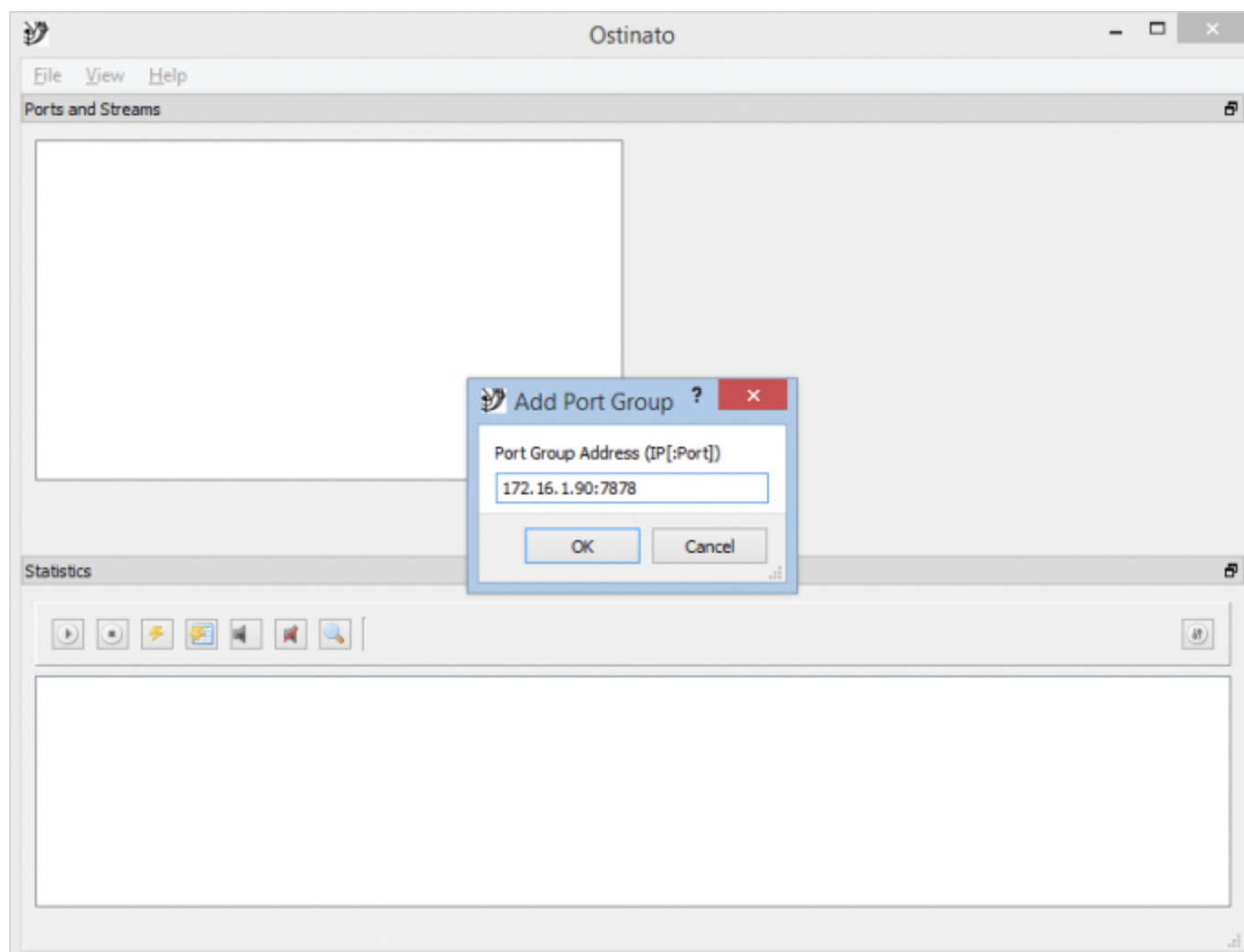
На клиентском Windows PC запускаем файл *ostinato.exe* - это графического интерфейса управления GUI **Ostinato**.



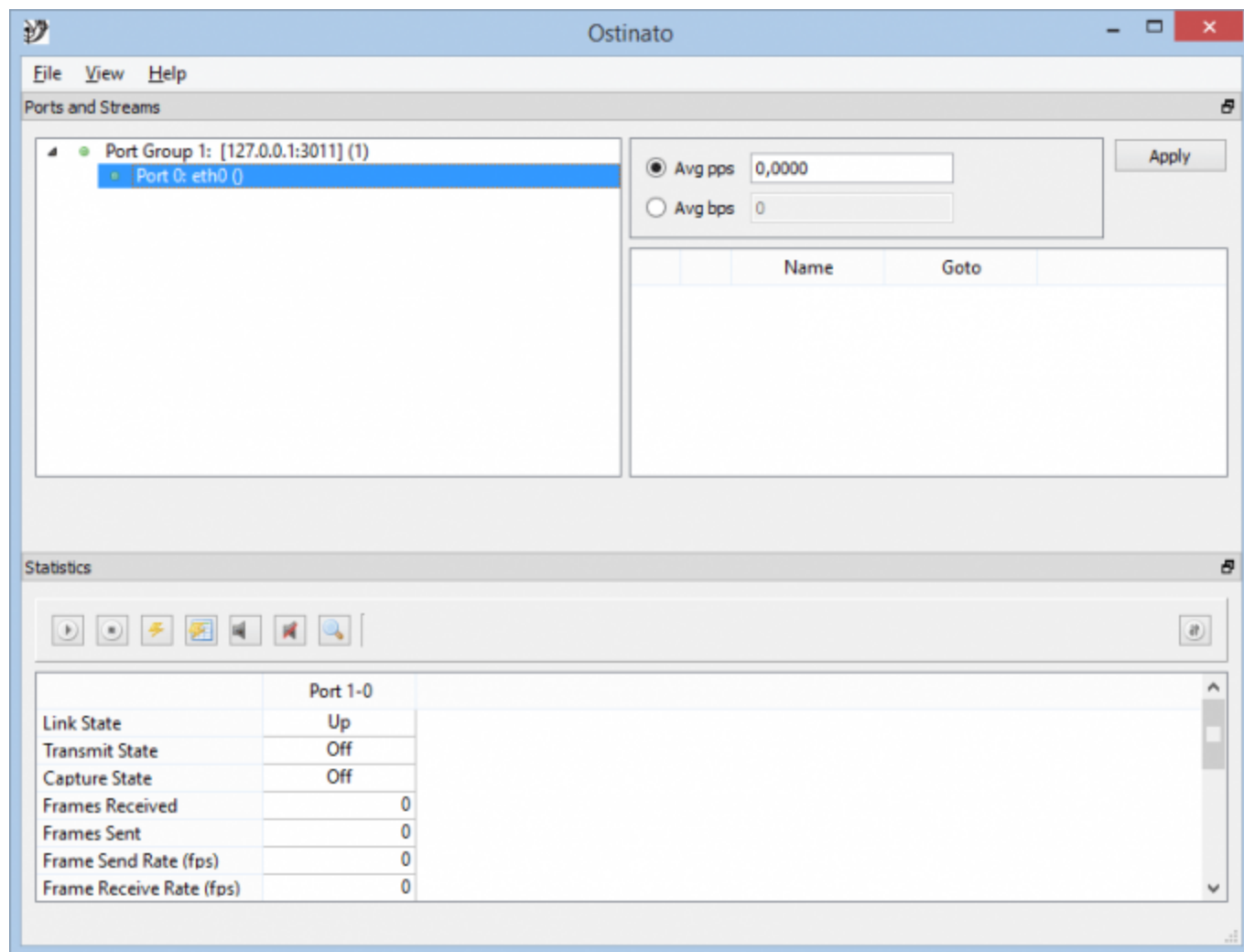
После запуска, в верхнем левом окне мы видим портовую группу 0 от локального дрона. Нам она не нужна, так как мы будем использовать дрона в UNL. Поэтому просто удаляем ее.



Далее выбираем меню File – New Port Group. В появившемся окне вводим IP адрес управляющего интерфейса дрона и порт **7878**.



Нажимаем ОК. Если все прошло хорошо в левом верхнем окне появится Port Group 1, которая содержит один интерфейс дрона eth0.

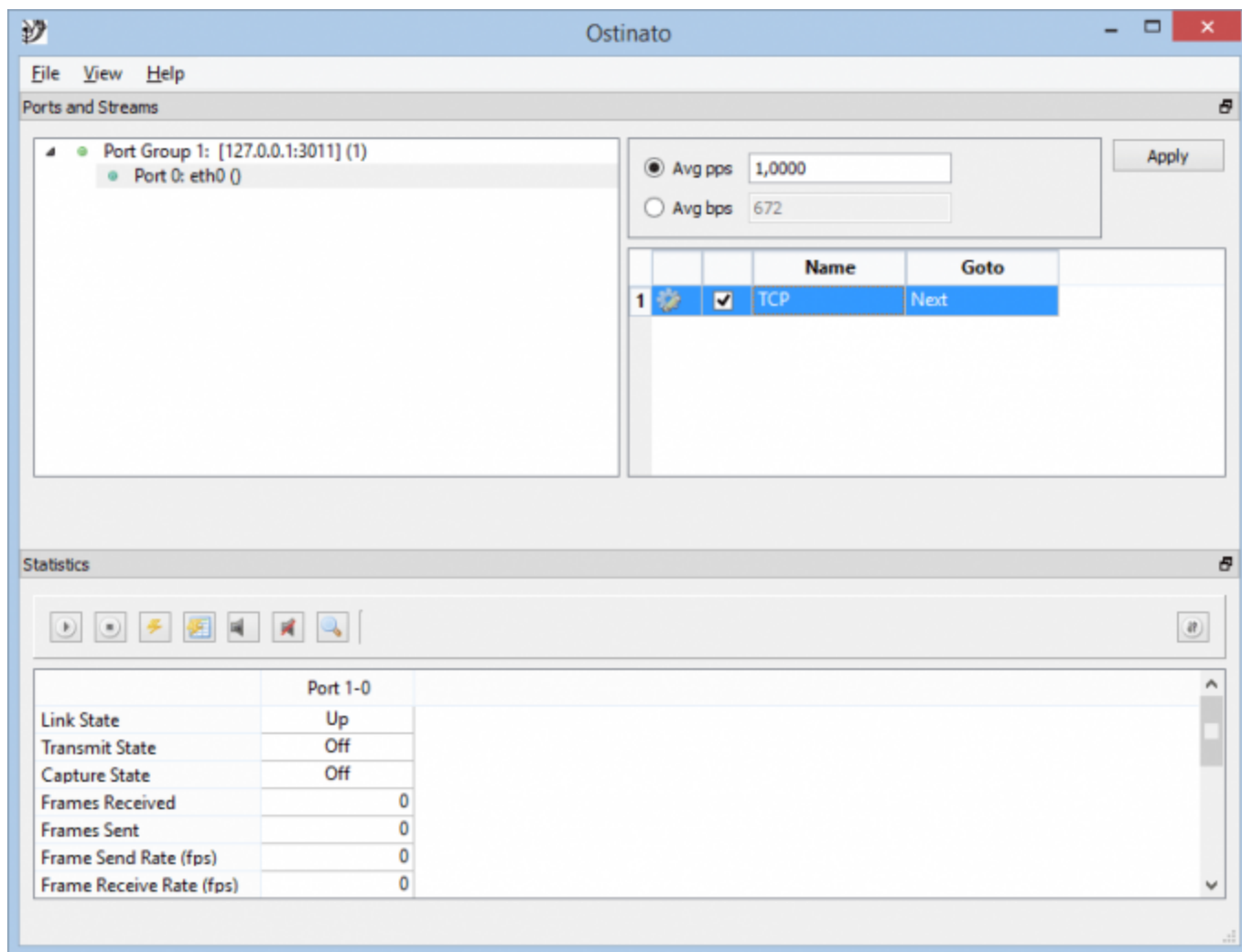


Зеленый индикатор состояния рядом с портовой группой показывает, что подключение к дрону прошло успешно и можно конфигурировать пакеты.

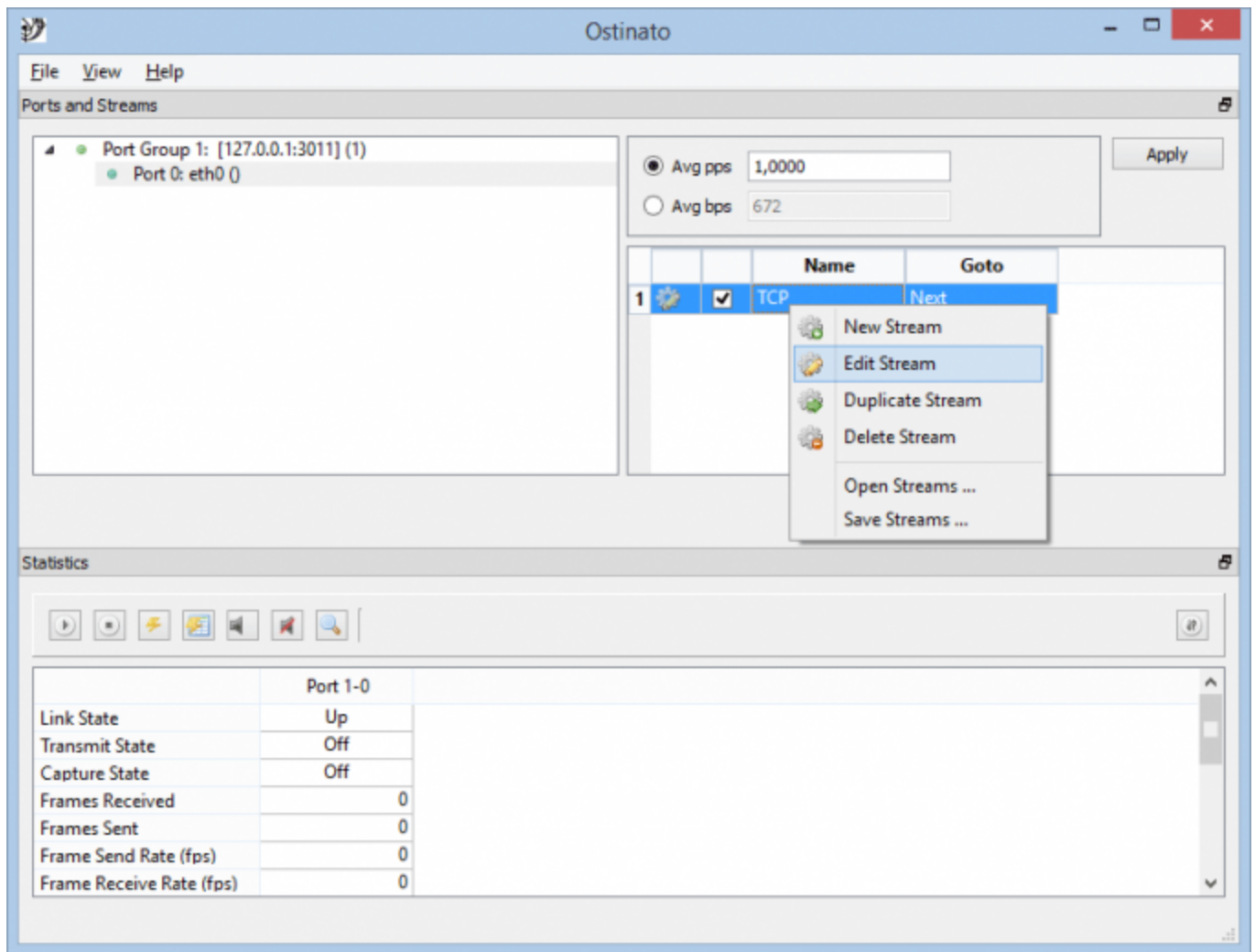
Создание потока

В качестве примера создадим простой TCP SYN пакет. Наш поток будет состоять из 100 пакетов со скоростью 1 пакет в секунду.

Чтобы создать поток (Stream) сначала выберите соответствующий порт в портовой группе и затем выберите File – New Stream. В окне справа появится новый поток. Изменим его имя на TCP.



Далее необходимо настроить все параметры нашего потока. Правкой кнопкой мыши нажимаем на имя потока и выбираем Edit Stream из контекстного меню



На этой странице (Protocol Selection) вы можете настроить длину фрейма. Можно установить как фиксированное значение, так и переменное. Мы будем использовать фиксированное по умолчанию.

Edit Stream

Protocol Selection | Protocol Data | Variable Fields | Stream Control | Packet View

Frame Length (including FCS)

Fixed Min 64 Max 1518

Simple

L1: ☐ None ☒ Mac ☐ Other

VLAN: ☒ Untagged ☐ Tagged ☐ Stacked

L2: ☐ None ☒ Ethernet II ☐ 802.3 Raw ☐ 802.3 LLC ☐ 802.3 LLC SNAP ☐ Other

L3: ☐ None ☐ ARP ☒ IPv4 ☐ IPv6 ☐ IP 6over4 ☐ IP 4over6 ☐ IP 4over4 ☐ IP 6over6 ☐ Other

L4: ☐ None ☐ ICMP ☐ IGMP ☐ MLD ☒ TCP ☐ UDP ☐ Other

L5: ☒ None ☐ Text ☐ Other

Payload: ☒ None ☐ Pattern ☐ Hex Dump ☐ Other

Advanced


OK Cancel

Далее настраиваем структуру пакета, составляя его из нужным нам протоколов. В качестве L3 выбираем IPv4, а в качестве L4 – TCP. Так как пакет TCP SYN, то для протокола L5 – выбираем None, и Payload – None.

Во вкладке Protocol Data настраиваем соответствующие заголовки. В Media Access Protocol настраиваем MAC адреса. В качестве SRC MAC поставим 5000.0002.00FF, а в качестве MAC DST - 5000.0002.0001.

	Address	Mode	Count	Step
Destination	50 00 00 02 00 01	Fixed	16	1
Source	50 00 00 02 00 FF	Fixed	16	1

В разделе IPv4 указываем IP адреса источника и приемника. Также можно указать фиксированный адрес и переменный. В случае переменного – IP адрес будет меняться в каждом посланном пакете.


Edit Stream
?
✕

Protocol Selection
Protocol Data
Variable Fields
Stream Control
Packet View

Media Access Protocol
Ethernet II
Internet Protocol ver 4

☐ Override Version
☐ Override Header Length (x4)
TOS/DSCP
☐ Override Length
Identification

Fragment Offset (x8)
☐ Don't Fragment
☐ More Fragments
Time To Live (TTL)
☐ Override Protocol
☐ Override Checksum

	Mode	Count	Mask
Source	Fixed	16	255.255.255.0
Destination	Fixed	16	255.255.255.0

Options

Transmission Control Protocol

OK

Cancel

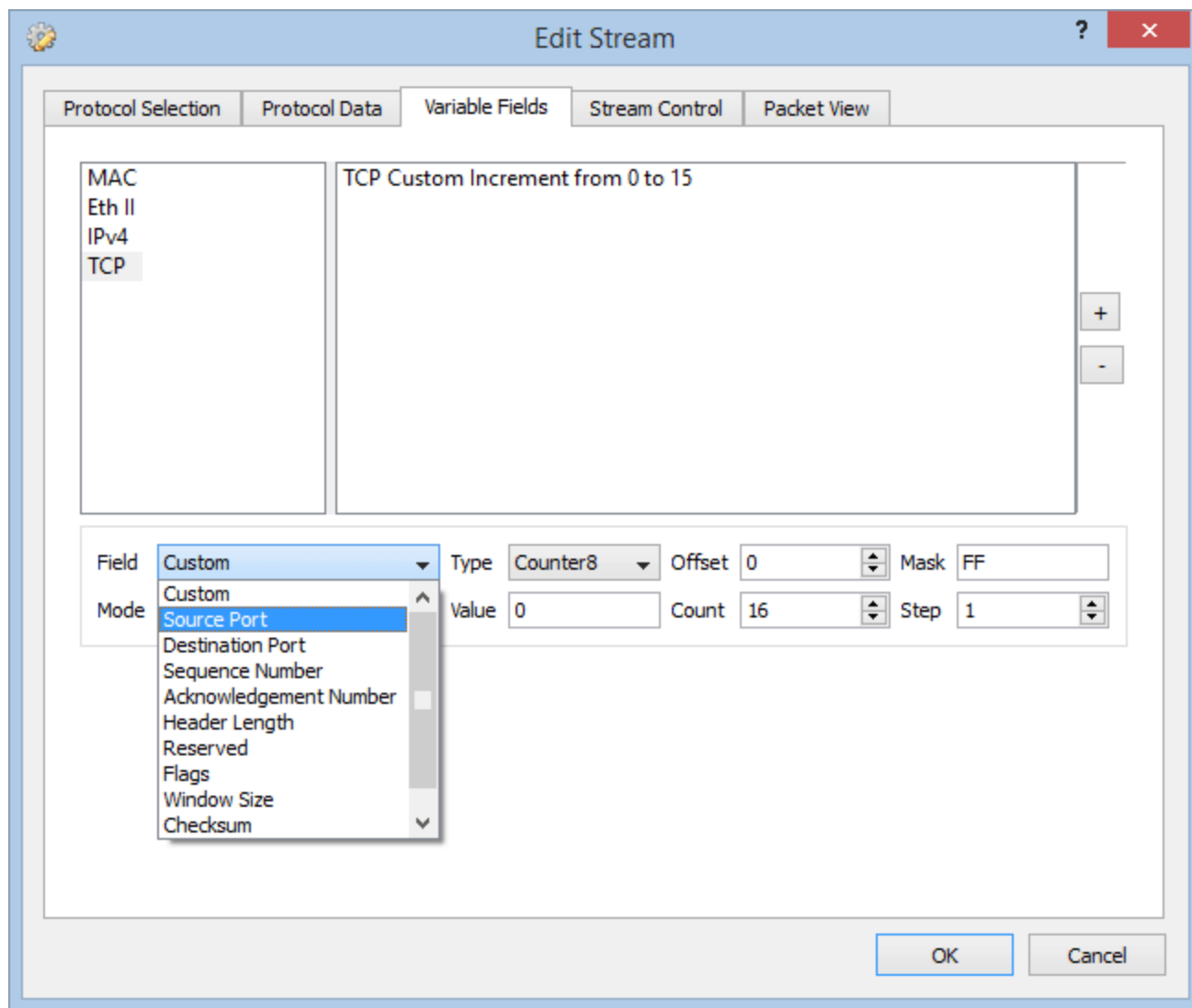
В разделе Transmission Control Protocol указываем порты и флаги. Так как у нас TCP SYN пакет, выбираем соответствующий флажок.

The 'Edit Stream' dialog box is shown with the 'Protocol Data' tab selected. The protocol stack consists of Media Access Protocol, Ethernet II, Internet Protocol ver 4, and Transmission Control Protocol. The TCP options section includes fields for Source Port (1024), Destination Port (80), Sequence Number (129018), Acknowledgement Number (0), Header Length (5), and Window (1024). It also features checkboxes for 'Override Source Port' and 'Override Destination Port', both of which are checked. The 'Override Checksum' field contains 'B3 E9', and the 'Urgent Pointer' is set to 0. The 'Flags' section includes checkboxes for URG, ACK, PSH, RST, SYN (checked), and FIN. The 'OK' and 'Cancel' buttons are at the bottom right.

Protocol Selection		Protocol Data		Variable Fields		Stream Control		Packet View	
Media Access Protocol									
Ethernet II									
Internet Protocol ver 4									
Transmission Control Protocol									
<input checked="" type="checkbox"/>	Override Source Port	1024		<input type="checkbox"/>	Override Checksum	B3 E9			
<input checked="" type="checkbox"/>	Override Destination Port	80		Urgent Pointer		0			
	Sequence Number	129018		Flags					
	Acknowledgement Number	0		<input type="checkbox"/>	URG	<input type="checkbox"/>	ACK	<input type="checkbox"/>	PSH
<input type="checkbox"/>	Override Header Length (x4)	5		<input type="checkbox"/>	RST	<input checked="" type="checkbox"/>	SYN	<input type="checkbox"/>	FIN
	Window	1024							

OK Cancel

Во вкладке Variable Fields можно указать дополнительные поля в пакете, который будут принимать переменные значения.



Во вкладке Stream Control указываем скорость генерации пакетов и их количество. Выберем генерировать 100 пакетов и скорость в 1 PPS.

Edit Stream ? X

Protocol Selection Protocol Data Variable Fields **Stream Control** Packet View

Send
☒ Packets
☐ Bursts

Numbers
Number of Packets: 100
Number of Bursts: 1
Packets per Burst: 10

Rate
☒ Packets/Sec: 1,0000
☐ Bursts/Sec: 1,0000
☐ Bits/Sec: 672

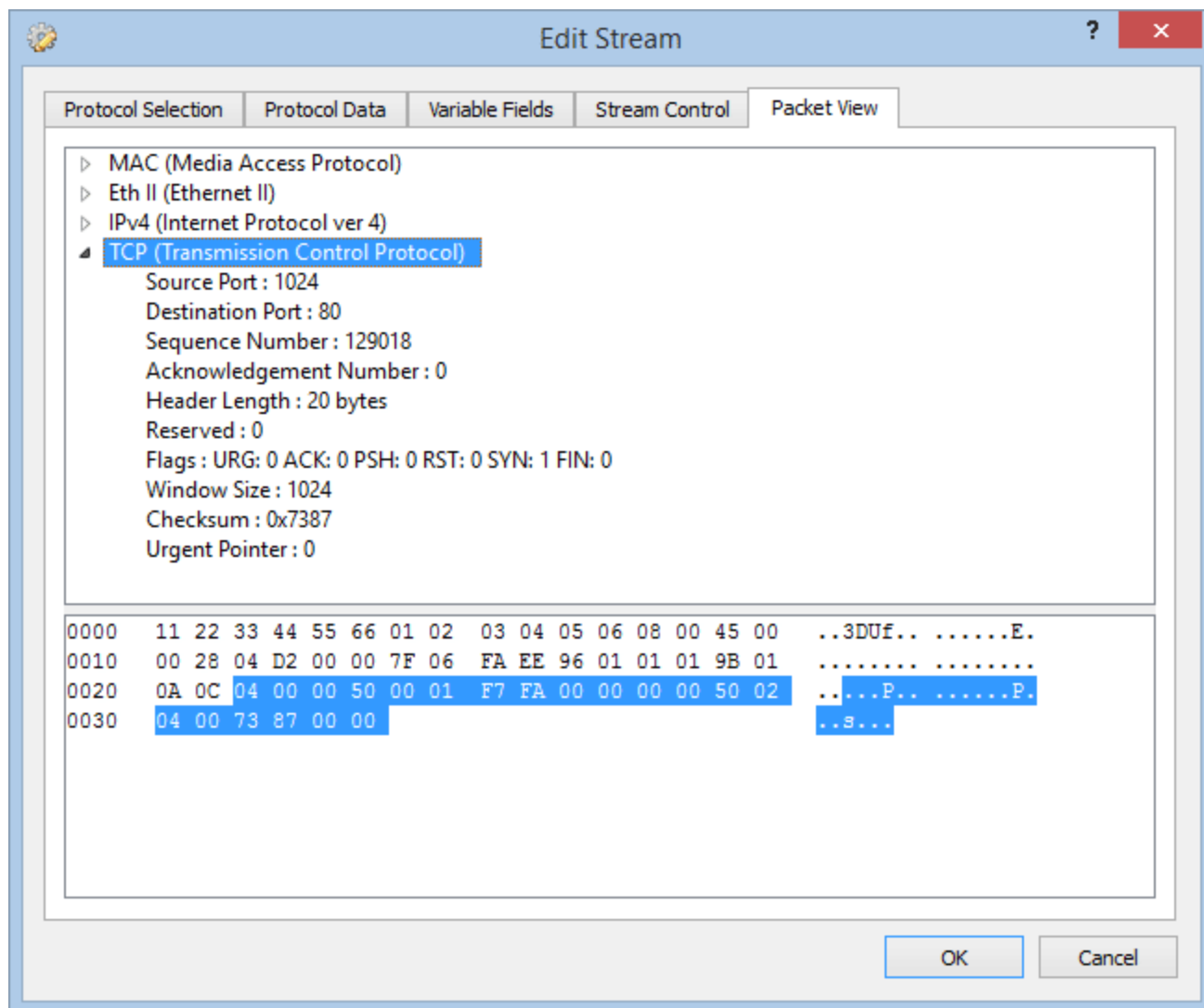
After this stream
☐ Stop
☒ Goto Next Stream
☐ Goto First

Mode
☒ Fixed
☐ Continuous

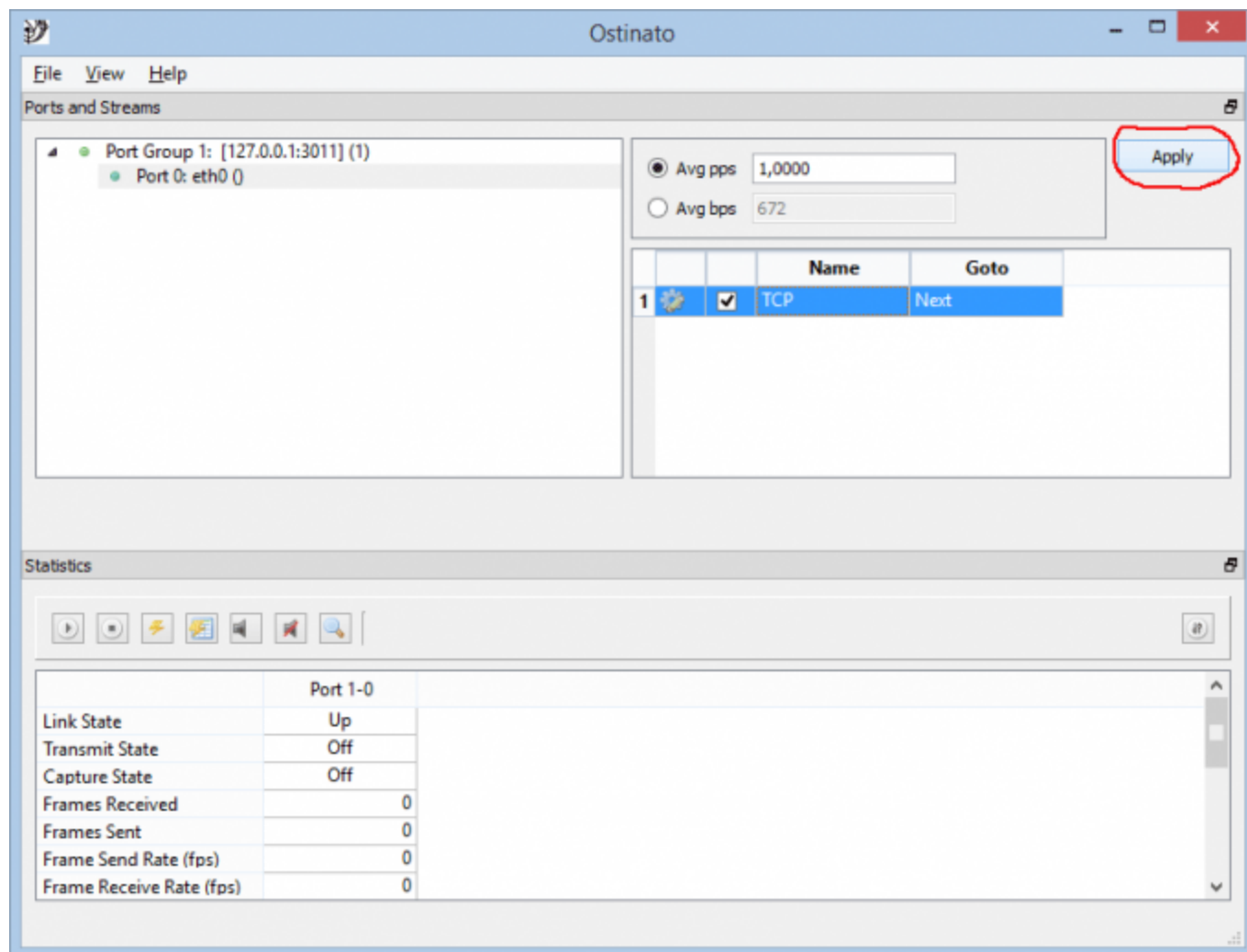
Gaps (in seconds)
ISG PKT 1 IPG PKT 2 ... PKT N IBG PKT 1 ...
ISG: 0.0 IBG: 0.0 IPG: 1,000000000

OK Cancel

Во вкладке Packet View – можно посмотреть результирующий пакет.

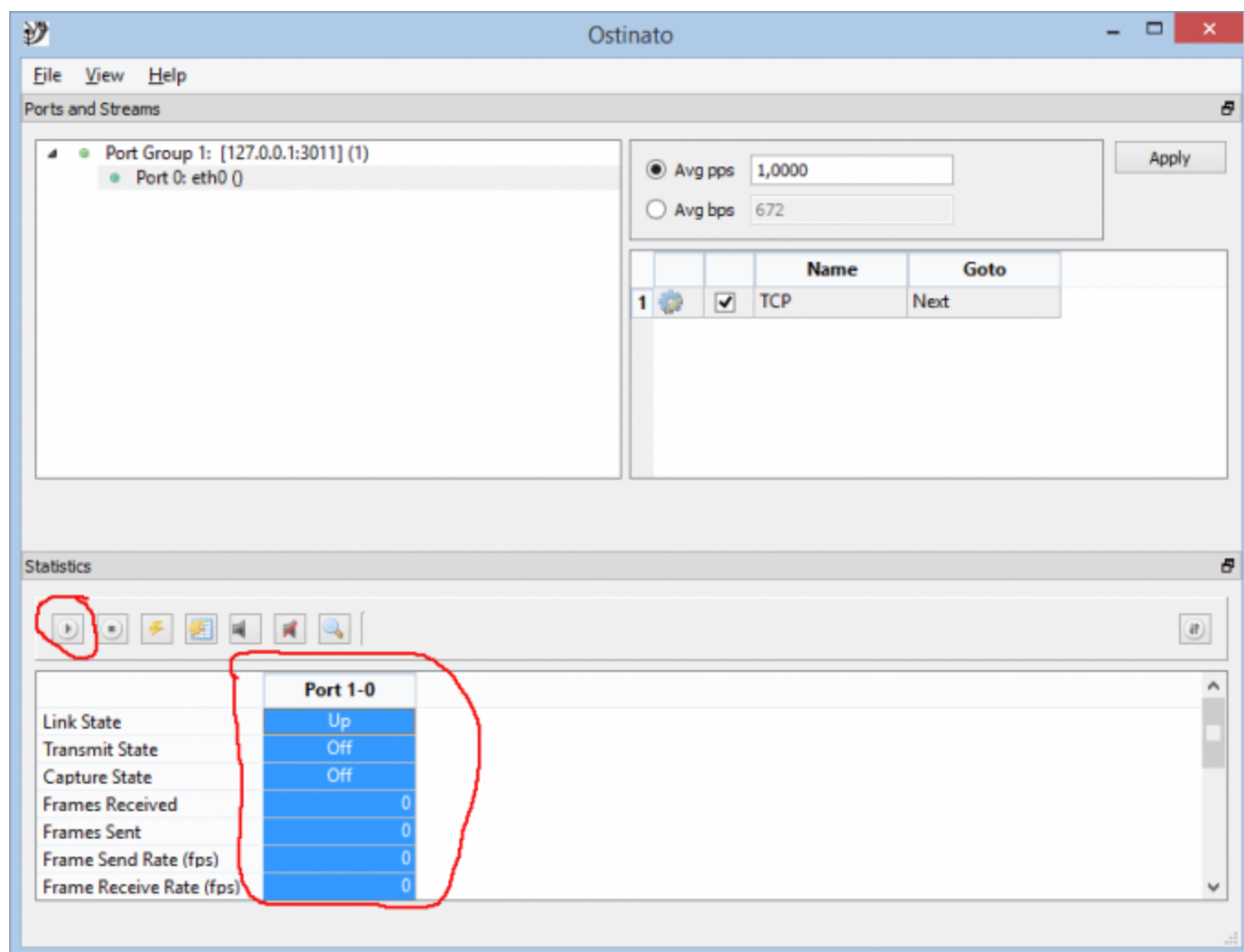


Нажмите ОК и далее обязательно Apply.

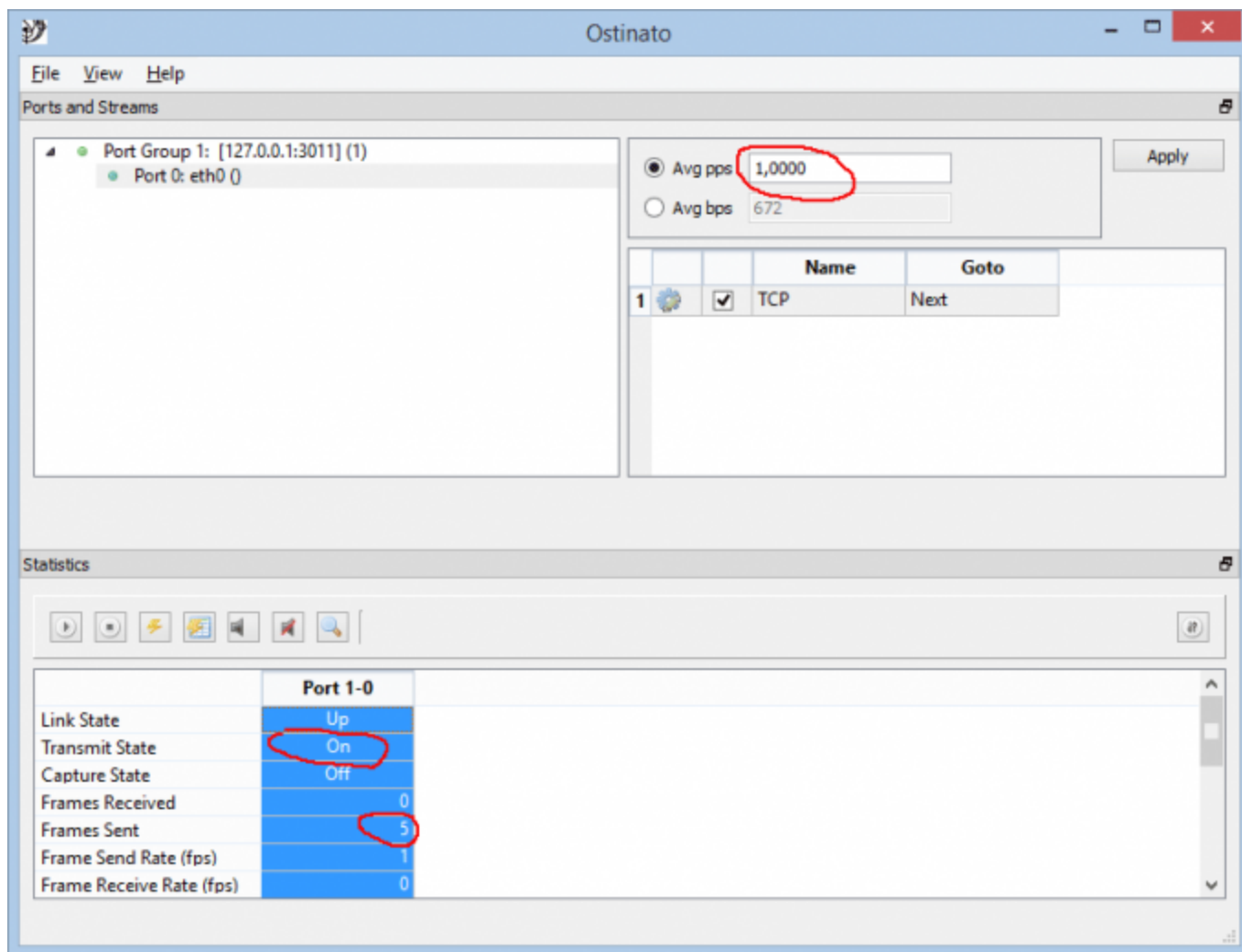


После любого изменения, необходимо нажимать Apply. При этом результирующая конфигурация передается на дрон.

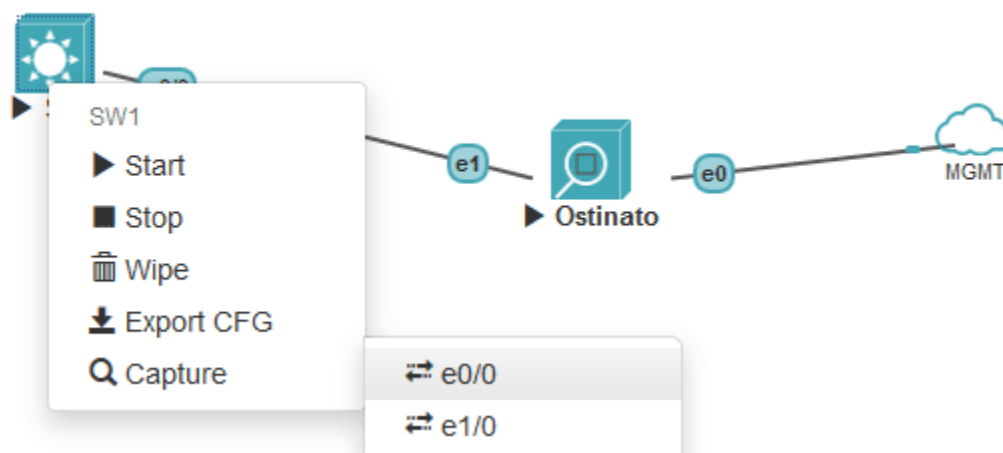
Для того, чтобы запустить поток, выделите ВСЮ колонку порта в окне статистики (Statistic) и нажмите кнопку Start Tx в верхней части окна статистики.



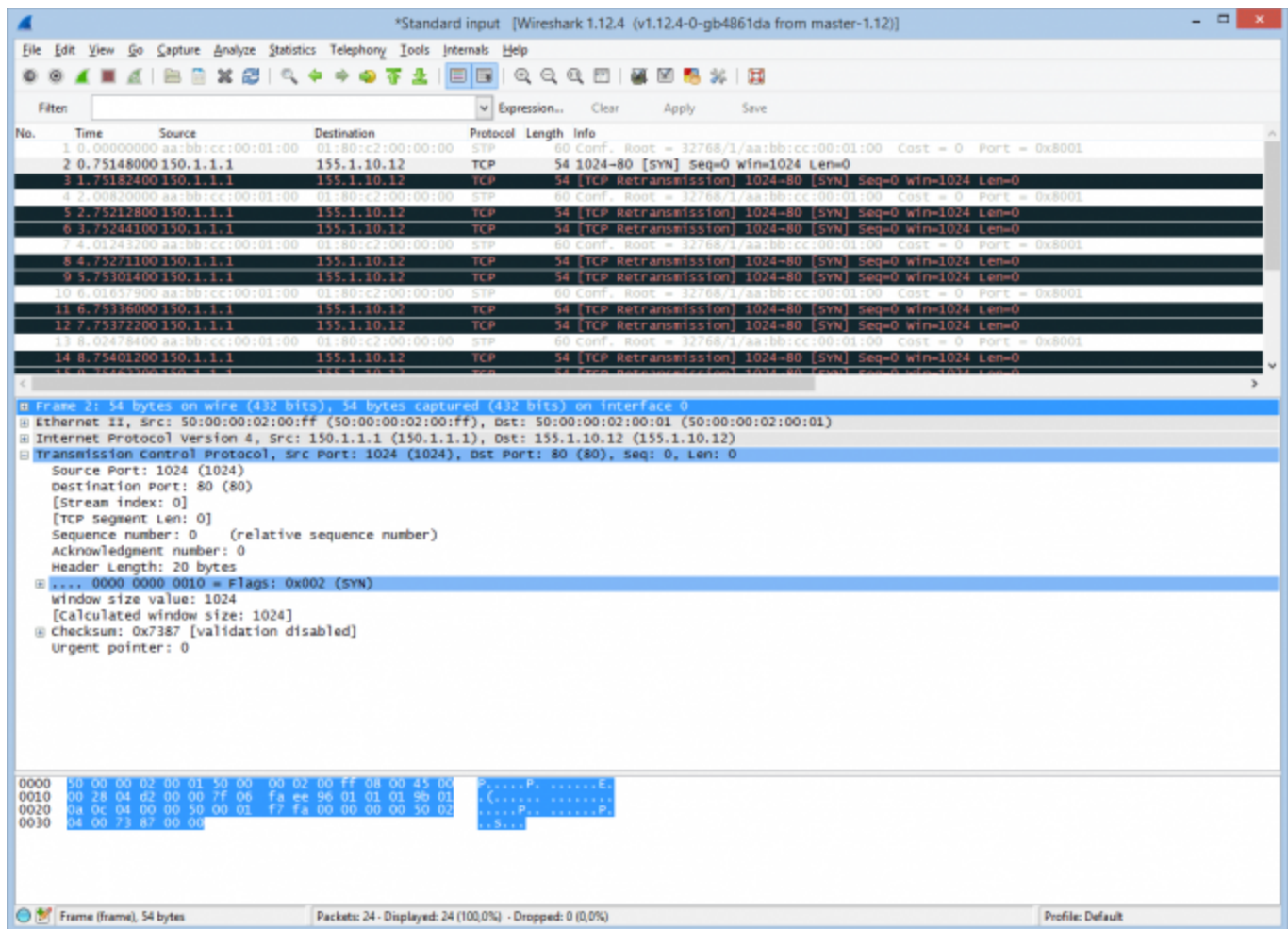
Если запуск прошел успешно, то Transmit State перейдет в состоянии On и Frames Sent будет увеличиваться.



Теперь можно посмотреть как трафик попадает на коммутатор. UNetLab имеет возможность снимать дамп трафика в любой точке стенда и отправлять его в WireShark запущенный на локальном PC.



Выберем правой кнопкой мыши свитч SW1 и из контекстного меню выберем пункт Capture – e0/0 . При этом на локальном PC запустится WireShark



Здесь мы можем увидеть наши сконструированные TCP SYN пакеты, которые поступают на порт E0/0 коммутатора SW1. Запустив на коммутаторе команду **show mac address-table**, видим, что SRC MAC адрес из нашего пакета изучен на порту коммутатора.

```
SW1#show mac address-table
Mac Address Table
-----
Vlan    Mac Address      Type      Ports
-----
1       5000.0002.00ff   DYNAMIC   Et0/0
Total Mac Addresses for this criterion: 1
SW1#
```

Другие новости по теме:

- [Как импортировать образы vIOS из Cisco VIRL в UNetLAB](#)
- [Authentication Proxy - Перехватывающая аутентификация на Cisco роутерах](#)
- [Как импортировать образы QEMU в UNetLAB - Cisco ASA и ASAv.](#)
- [GNS3 - Графический Сетевой Симулятор](#)
- [UNetLab - Строим стенд для подготовки к CCIE](#)

[Вернуться](#) 20 943 0

Категория: [Лаборатория](#)