

# Исследование размера окна приема ТСР в различных ОС

Дунайцев Р.А. (СПбГУТ)

[roman.dunaytsev@spbgut.ru](mailto:roman.dunaytsev@spbgut.ru)

# Краткая теоретическая справка

- Одной из многочисленных функций, реализуемых TCP, является управление потоком данных для защиты принимающей стороны от перегрузки. Иными словами, управление потоком представляет собой регулирование количества посылаемых отправителем данных в соответствии со свободным местом в буфере получателя для предотвращения переполнения буфера. Получатель осуществляет управление потоком данных, анонсируя в поле Window в виде двоичного числа количество байт данных (т.н. «окно»), которые он готов принять от отправителя. Длина этого поля составляет 16 бит, что ограничивает максимальный объем передаваемых за один раунд данных 65535 байтами. Опция масштабирования окна (**TCP Window Scale Option**) позволяет повысить производительность TCP, увеличив размер окна путем умножения его на специальный коэффициент масштабирования, который указывается на этапе установления TCP-соединения в сегментах SYN и SYN/ACK.
- <https://tools.ietf.org/html/rfc7323>
- <https://tools.ietf.org/html/rfc793>

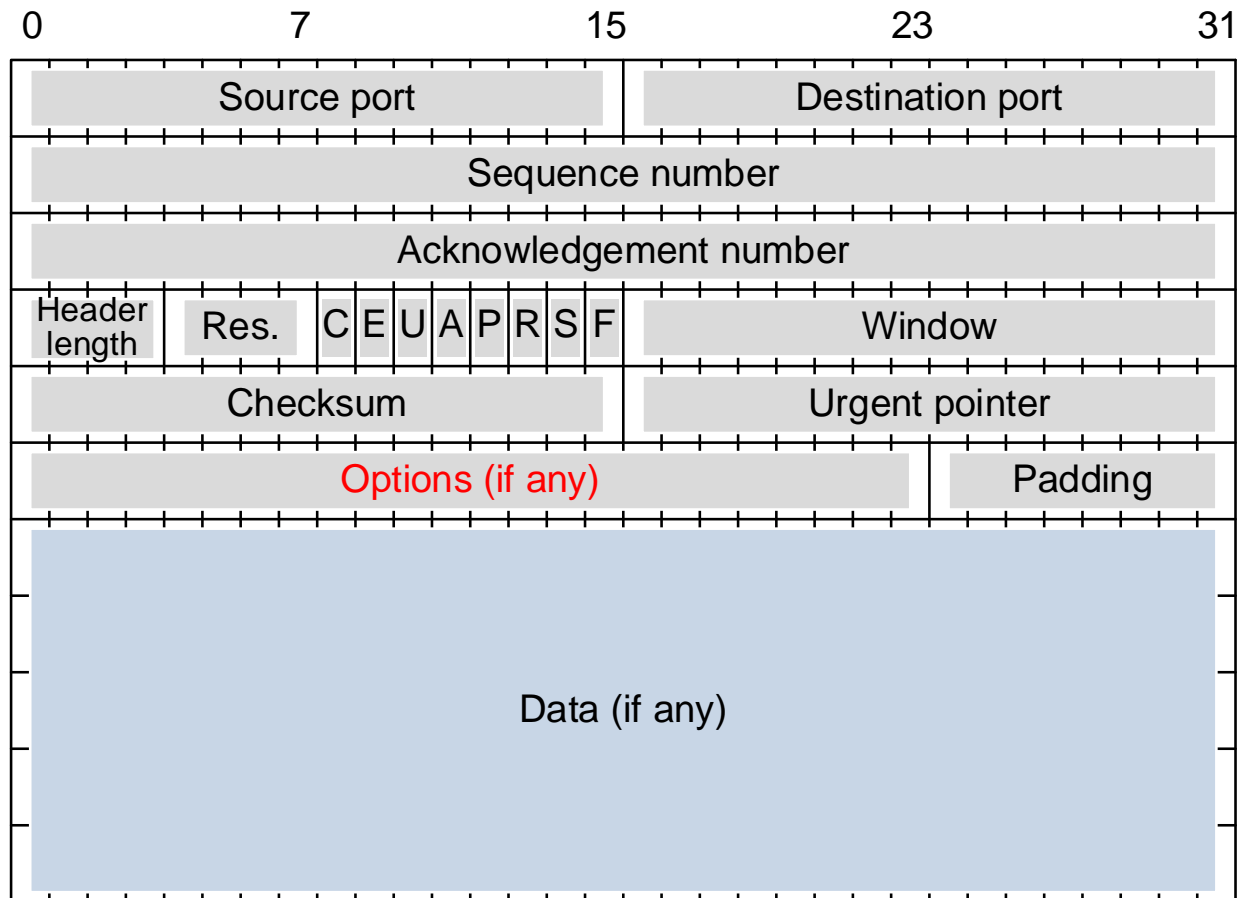
# Задание на дом

- 1) Используя ПО для создания и запуска виртуальных машин (VirtualBox, VMware и т.п.), установить не менее 5 операционных систем (ОС) разных лет выпуска и/или разных типов. Например:
  - Windows XP/Vista/8.0/8.1/10
  - Ubuntu 9/11/13/15/17, см. <http://old-releases.ubuntu.com/releases/>
  - Любые ОС на ваш выбор из списка самых популярных <https://distrowatch.com/dwres.php?resource=popularity>
  - Если места на жестком диске мало, то ОС лучше устанавливать по очереди, удаляя предыдущую
- 2) С помощью Wireshark определить, какие опции TCP используют данные ОС, а также максимальный размер окна (с учетом масштабирования, если такая опция используется)
- 3) Ознакомиться с назначением обнаруженных опций TCP и дать их развернутое описание
- 4) **К отчету приложить собранные Wireshark пакеты!**

# Оформление результатов

ОС	Год выпуска	Опции TCP	Макс. окно, байт

# Структура TCP-сегмента



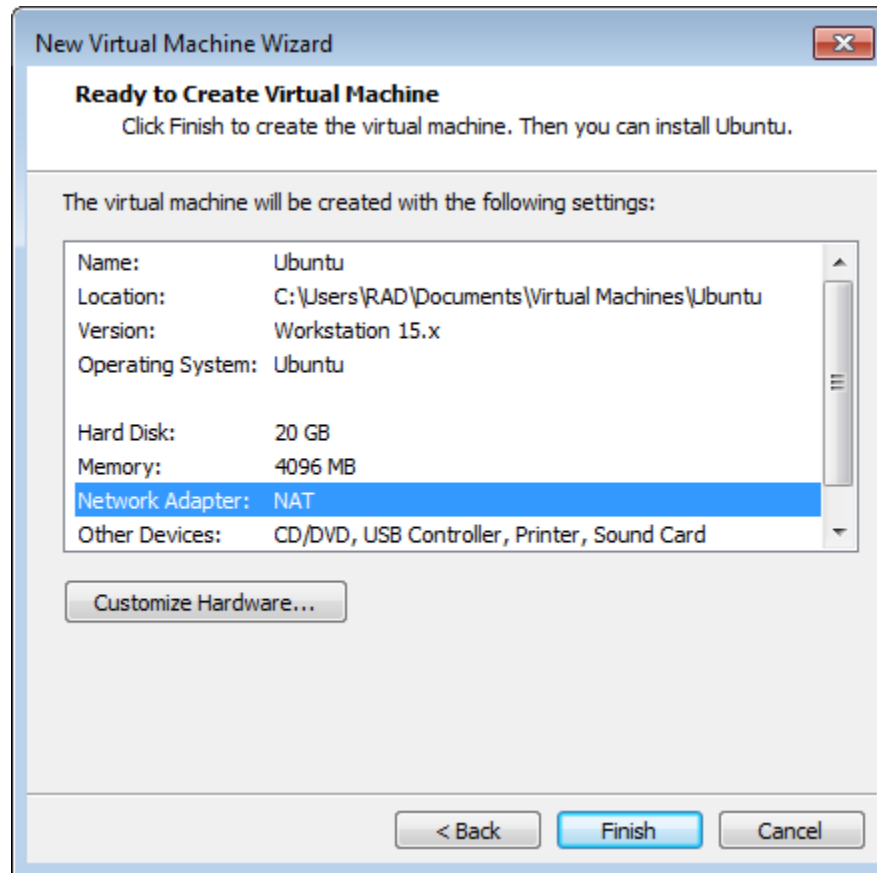
# Пример исследования ОС

- Основная ОС: Windows 7 Enterprise 64-bit
- Тестируемая ОС: Linux Mint 19.3 Tricia Cinnamon 32-bit
  - <https://www.linuxmint.com/edition.php?id=273>
- Используемое ПО: VMware Workstation 15 Player
  - <https://download3.vmware.com/software/player/file/VMware-player-15.5.2-15785246.exe>
- Для Windows **32-bit** используйте VMware Player 6
  - <https://download3.vmware.com/software/player/file/VMware-player-6.0.7-2844087.exe>
- либо VMware Workstation 10
  - <https://download3.vmware.com/software/wkst/file/VMware-workstation-full-10.0.7-2844087.exe>

# Player и Workstation бесплатны 😊

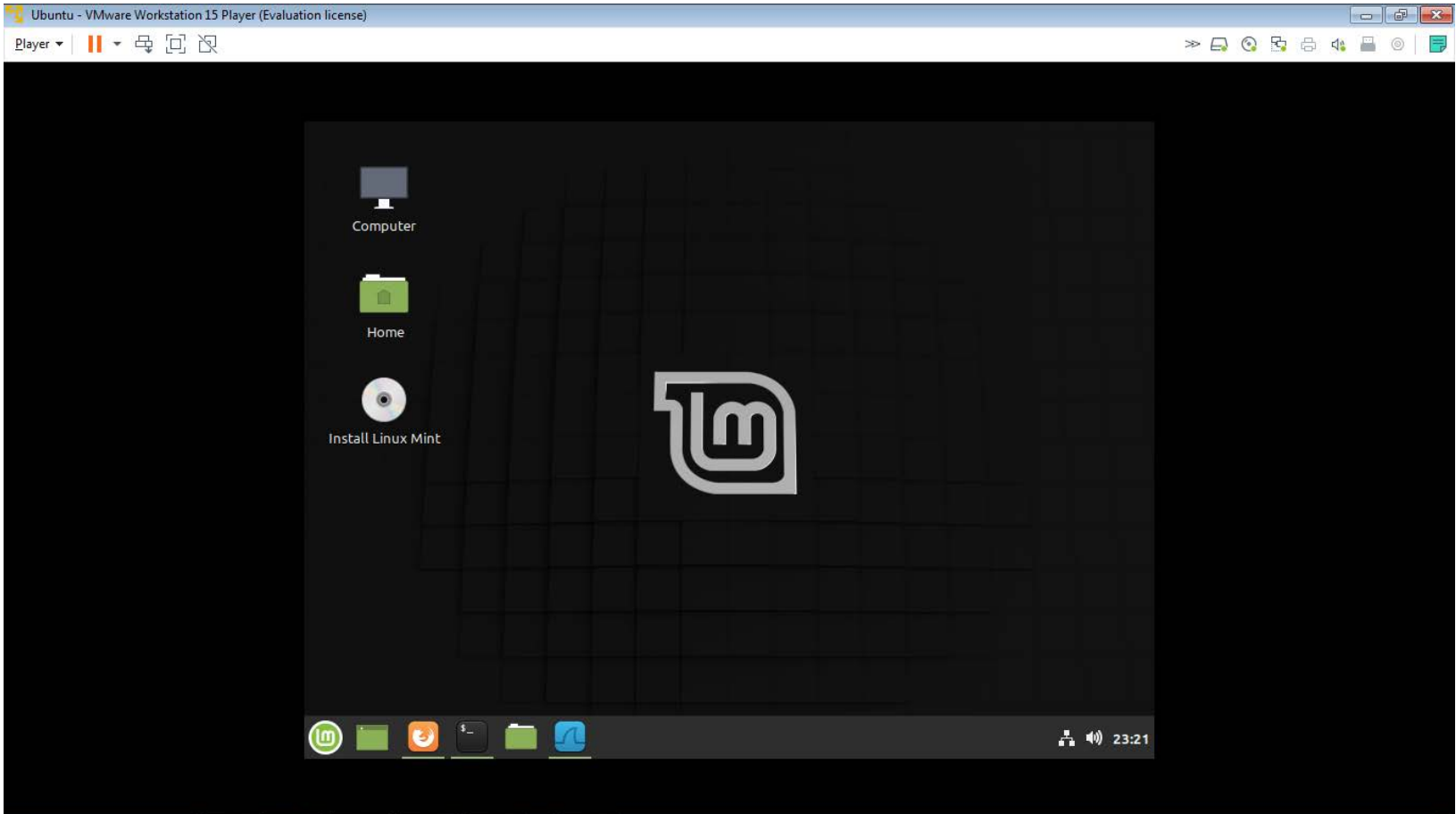


# В VMware используется NAT!





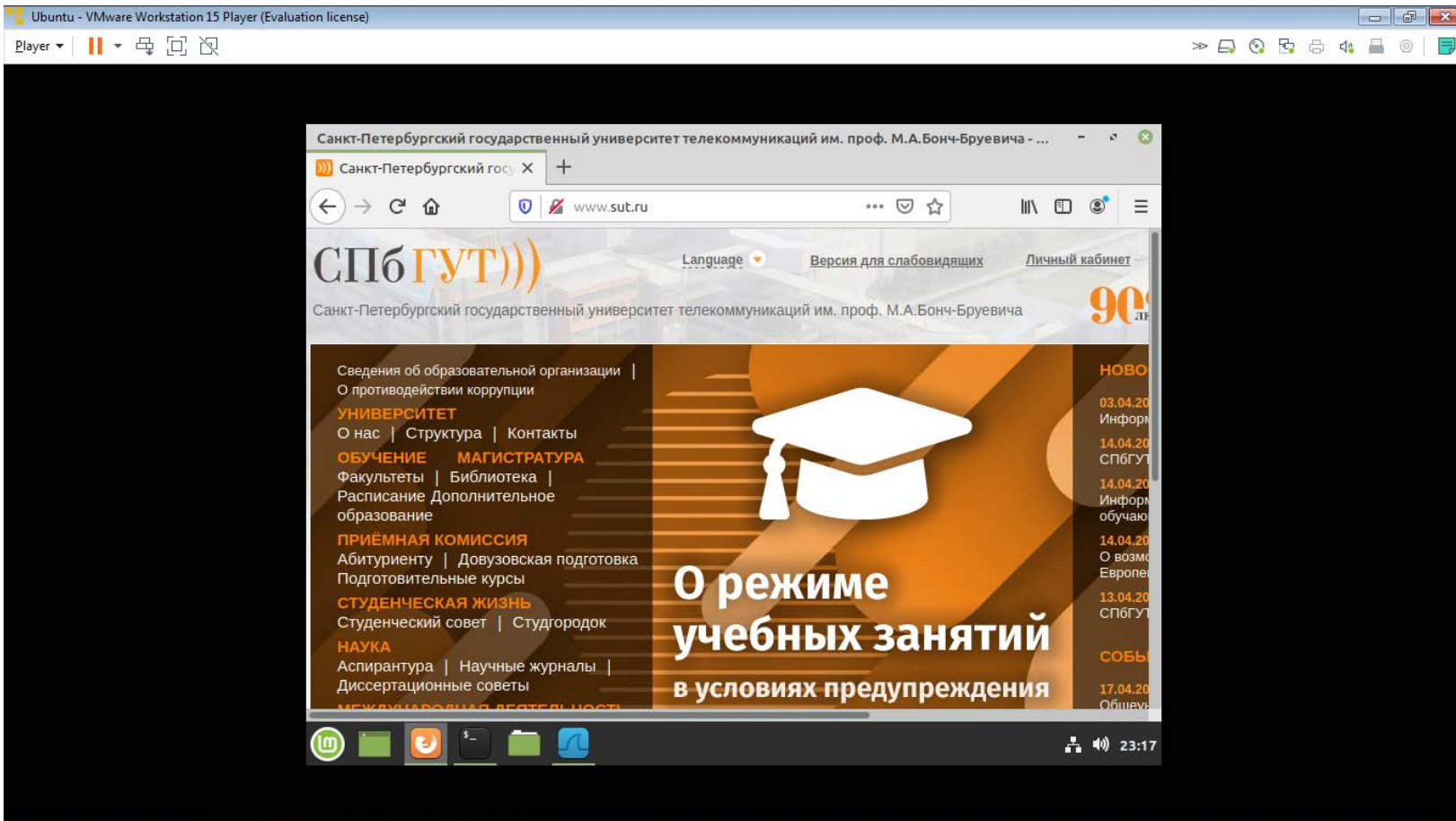
# Linux Mint 19.3 'Tricia' Cinnamon



# Установка Wireshark в Linux

- 1) Запускаем Terminal:  
`<Ctrl> + <Alt> + T`
  - 2) Обновляем пакеты:  
`sudo apt-get update`
  - 3) Устанавливаем Wireshark:  
`sudo apt-get install wireshark`
  - 4) Запускаем Wireshark для захвата трафика:  
`sudo wireshark`
- Кстати, в Ubuntu мне не удалось установить Wireshark. Может вам повезет больше... ☹️

# Открываем какой-нибудь сайт



# Пакеты внутри VMware

Ubuntu - VMware Workstation 15 Player (Evaluation license)

Player | [Icons]

\*ens33

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.stream eq 14 Expression...

No.	Time	Source	Destination	Protocol	Length	Info
59	12.057169553	192.168.72.128	91.238.230.94	TCP	74	43980 → 80 [SYN] Seq=0 W1
60	12.067157083	91.238.230.94	192.168.72.128	TCP	60	80 → 43980 [SYN, ACK] Seq=
61	12.067294815	192.168.72.128	91.238.230.94	TCP	54	43980 → 80 [ACK] Seq=1 Ac
62	12.067691177	192.168.72.128	91.238.230.94	HTTP	370	GET / HTTP/1.1
63	12.068416284	91.238.230.94	192.168.72.128	TCP	60	80 → 43980 [ACK] Seq=1 Ac
64	12.708027811	91.238.230.94	192.168.72.128	TCP	1514	80 → 43980 [ACK] Seq=1 Ac
65	12.708066329	192.168.72.128	91.238.230.94	TCP	54	43980 → 80 [ACK] Seq=317
66	12.708100333	91.238.230.94	192.168.72.128	TCP	1514	80 → 43980 [ACK] Seq=1464

[Checksum Status: Unverified]  
Urgent pointer: 0

Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale

- ▶ TCP Option - Maximum segment size: 1460 bytes
- ▶ TCP Option - SACK permitted
- ▶ TCP Option - Timestamps: TSval 2366413569, TSecr 0
- ▶ TCP Option - No-Operation (NOP)
- ▶ TCP Option - Window scale: 7 (multiply by 128)

```
0000 00 50 56 e5 d9 6f 00 0c 29 dc b3 64 08 00 45 00 .PV.. )...d..E.
0010 00 3c 11 4a 40 00 40 06 dd fc c0 a8 48 80 5b ee .<.J.@. ...H.[.
0020 e6 5e ab cc 00 50 62 1a e9 8c 00 00 00 00 a0 02 .^...Pb. ....
0030 fa f0 e1 c7 00 00 02 04 05 b4 04 02 08 0a 8d 0c .....
0040 0b 01 00 00 00 01 03 03 07 .....

```

TCP Options (tcp.options), 20 bytes Packets: 4467 · Displayed: 29 (0.6%) · Dropped: 0 (0.0%) Profile: Default

[Icons] 00:00

# Те же пакеты в основной ОС

The screenshot shows a Wireshark capture of network traffic on a local area connection (tcp port http). The filter is set to `ip.addr==91.238.230.94 && tcp.port==80`. The packet list shows several packets, including a SYN packet (No. 4574) and several ACK packets (Nos. 4578-4972). The detailed view of the selected packet (No. 4972) shows the following information:

- Window size value: 64240  
[Calculated window size: 64240]
- Checksum: 0x1628 [unverified]
- [Checksum Status: Unverified]
- Urgent pointer: 0
- Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SACK permitted
  - ▷ TCP Option - Maximum segment size: 1460 bytes
  - ▷ TCP Option - No-Operation (NOP)
  - ▷ TCP Option - Window scale: 0 (multiply by 1)
  - ▷ TCP Option - No-Operation (NOP)
  - ▷ TCP Option - No-Operation (NOP)
  - ▷ TCP Option - SACK permitted
- [Timestamps]

The packet bytes pane shows the raw data in hexadecimal and ASCII. The first few bytes are `08 4f a9 67 df 7f 3c 97 0e 76 b1 d9 08 00 45 00`, which correspond to the ASCII string `..g-<..v...E-`. The status bar at the bottom indicates that 7385 packets were captured, 16 (0.2%) were displayed, and 0 (0.0%) were dropped.

# После NAT опции ТСР изменились

Внутри VMware Workstation 15 Player	В основной ОС после NAT
Maximum Segment Size: 1460 bytes	Maximum Segment Size: 1460 bytes
SACK permitted	No-Operation (NOP)
Timestamps	Window scale: 0 (multiply by 1)
No-Operation (NOP)	No-Operation (NOP)
Window scale: 7 (multiply by 128)	No-Operation (NOP)
	SACK permitted

# Выводы

- При обращении из виртуальной машины к сайтам в сети Интернет используется **NAT (Network Address Translation)**
- Как показал проведенный эксперимент, использование NAT меняет состав и содержимое опций TCP
- В частности, отключилась опция масштабирования окна, так как вместо исходного коэффициента 128 умножение будет происходить на 1 и, следовательно, величина окна не изменится
- Также исчезла опция временной метки
- Таким образом, для получения объективной информации захват пакетов с помощью Wireshark следует производить внутри виртуальной машины, а не снаружи!

# Итоговая таблица с одной ОС

ОС	Год выпуска	Опции TCP	Макс. окно, байт
linuxmint-19.3-cinnamon-32bit	December 18, 2019	MSS, SACK, Timestamps, NOP, Window scale	$65535 \times 128 = 8.388.480$