ОБНОВЛЕНО 16 МАР

Оптимизация производительности платформы

Скрипт оптимизации сетевой карты

Для улучшения производительности сетевых карт важно настроить их параметры таким образом, чтобы они могли работать с максимальной пропускной способностью и минимальными задержками. Включение агрессивной компрессии (CQE) и отключение расслабленного порядка PCI-операций помогает ускорить передачу данных и оптимизировать взаимодействие с памятью. Включение DevX увеличивает пропускную способность и снижает задержки.

Для оптимизации работы сетевой карты запустить следующий скрипт:

```
#!/bin/sh

# Enable aggressive CQE Zipping.
cqe_compression=1

# Disable relaxed ordering for PCI operations.
pci_wr_ordering=0

# Enable DevX.
uctx_en=1


devs=$(ls -d /sys/class/net/*/device/infiniband_verbs/uverbs* | cut -d / -f 5)

for dev in $devs; do
    pci_addr=$(grep PCI_SLOT_NAME /sys/class/net/$dev/device/uevent | tail -c
+15)

    echo Device: $dev $pci_addr

    mstconfig -y -d $pci_addr s \
        CQE_COMPRESSION=$cqe_compression \
        PCI_WR_ORDERING=$pci_wr_ordering \
        UCTX_EN=$uctx_en

    # Switch link type to Ethernet if supported
    mstconfig -y -d $pci_addr s LINK_TYPE_P1=2
```

```
    mstconfig -y -d $pci_addr s LINK_TYPE_P2=2
done
```

Режим pass-through для IOMMU

Для оптимальной работы виртуализированных систем важно обеспечить прямой доступ к аппаратным ресурсам для виртуальных машин. Это возможно при включении режима pass-through для IOMMU. Такой подход значительно повышает производительность, поскольку исключает вмешательство гипервизора, что позволяет виртуальным машинам работать с максимально возможной пропускной способностью и низкими задержками.

- Открыть файл **/etc/default/grub**
- Найти строку

```
GRUB_CMDLINE_LINUX=""
```

и заменить на:

```
GRUB_CMDLINE_LINUX="iommu=pt"
```

- Обновить параметры загрузчика с помощью команды:

```
update-grub
```

Скрипты оптимизации для сетевых карт Mellanox

Для повышения производительности сетевых карт Mellanox важно оптимизировать распределение прерываний и настроить привязку их к процессорным ядрам NUMA. Это помогает улучшить балансировку нагрузки, снизить задержки и обеспечить правильную работу системы при высокой нагрузке. Отключение службы irqbalance важно, чтобы избежать возможных конфликтов в распределении прерываний. Этот процесс помогает обеспечить стабильную работу и высокую производительность при использовании сетевых карт Mellanox в высоконагруженных системах.

- Загрузить архив скриптов оптимизации Mellanox
- Определить номера процессорных ядер для NUMA с помощью команды:

```
lscpu | grep "NUMA node"
```

- Остановить и отключить автозапуск сервиса *irqbalance*:

```
systemctl stop irqbalance.service && systemctl disable irqbalance.service
```

- Запустить загруженный скрипт *set_irq_affinity_cpulist.sh* с указанием нужных номеров ядер NUMA и названия интерфейса. В качестве параметров указать номера ядер нужной NUMA, а так же название интерфейса. Пример команды: `/set_irq_affinity_cpulist.sh 0-19,40-59 enp216s0f0np0`

Режим максимальной производительности для CPU

Применение режима performance позволяет процессорам работать на максимальной тактовой частоте, что особенно важно при выполнении ресурсоемких операций. Установка этого режима помогает избежать потери производительности из-за энергосберегающих функций процессора, гарантируя, что система будет работать на максимуме своих вычислительных возможностей.

- Проверить текущий режим работы процессора командой:

```
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

- Проверить доступные режимы работы процессора:

```
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_available_governors
```

- Если среди доступных режимов есть `performance`, применить его командой:

```
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
```

Установка и настройка драйверов для сетевых карт Silicom Bypass

1. Сборка драйвера ixgbe (версия 5.19.6)

Рекомендованное ядро:

- `6.5` для Ubuntu
- `6.1.67-un-def` для Альт СП

Необходимые пакеты:

- `image` и `kernel-headers-modules` соответствующей версии.
- Компилятор `gcc`.

Установка компилятора:

```
apt-get install gcc
```

Загрузка исходных файлов:

```
wget --http-user=[ваш логин] --http-password=[ваш пароль] https://public-repo.svcp.io/PxGxBP_VSD40f.zip
```

Сборка и установка драйвера для Ubuntu:

```
unzip PxGxBP_VSD40f.zip
```

```
cd PxGxBP_VSD40f/Linux/Network_Drivers/PE210GxBPi9/ixgbe-5.19.6/
```

```
tar -xzvf ixgbe-5.19.6ms7.2.tar.gz
```

```
cd ixgbe-5.19.6ms7.2/src/
```

```
uname -r
```

```
sudo make -C /lib/modules/$(uname -r)/build M=$(pwd) modules
```

```
sudo make install
```

```
modprobe ixgbe
```

Сборка и установка драйвера для Альт СП:

```
unzip PxGxBP_VSD40f.zip
```

```
cd PxGxBP_VSD40f/Linux/Network_Drivers/PE210GxBPi9/ixgbe-5.19.6/
```

```
tar -xzvf ixgbe-5.19.6ms7.2.tar.gz
```

```
cd ixgbe-5.19.6ms7.2/src/
```

```
make KSRC=/usr/src/linux-6.1.67-un-def-alt0.c10f.1
```

```
make KSRC=/usr/src/linux-6.1.67-un-def-alt0.c10f.1 install
```

```
modprobe ixgbe
```

Проверка загрузки драйвера и версии:

```
lsmod | grep ixgbe
```

```
modinfo ixgbe
```

2. Сборка bypass-драйвера

Сборка и установка для Ubuntu:

```
cd ~/PxGxBP_VSD40f/Linux/Bypass/Libs/Kernel/
```

```
tar -xzvf bypass-9.0.8.tar.gz
```

```
cd bypass-9.0.8/lib
```

```
uname -r
```

```
make -C /lib/modules/$(uname -r)/build M=$(pwd) modules
```

```
sudo make -C /lib/modules/$(uname -r)/build M=$(pwd) modules_install
```

```
sudo depmod -a
```

```
sudo modprobe bypass
```

Сборка и установка для Альт СП:

```
cd ~/PxGxBP_VSD40f/Linux/Bypass/Libs/Kernel/
```

```
tar -xzvf bypass-9.0.8.tar.gz
```

```
cd bypass-9.0.8/lib
```

```
make KSRC=/usr/src/linux-6.1.67-un-def-alt0.c10f.1
```

```
make KSRC=/usr/src/linux-6.1.67-un-def-alt0.c10f.1 install
```

```
modprobe bypass
```

Проверка загрузки драйвера и версии:

```
lsmod | grep bypass
```

```
modinfo bypass
```

3. Сборка и проверка утилиты bp_ctl

Сборка утилиты Ubuntu:

```
cd ~/PxGxBP_VSD40f/Linux/Bypass/BP_Control/
```

```
tar -xzvf bp_ctl-5.2.0.49.tar.gz
```

```
cd bp_ctl-5.2.0.49/
```

```
uname -r
```

```
make -C /lib/modules/$(uname -r)/build M=$(pwd) install
```

```
bpctl_start
```

Сборка утилиты для Альт СП:

```
cd ~/PxGxBP_VSD40f/Linux/Bypass/BP_Control/
```

```
tar -xzvf bp_ctl-5.2.0.49.tar.gz
```

```
cd bp_ctl-5.2.0.49/
```

```
make KSRC=/usr/src/linux-6.1.67-un-def-alt0.c10f.1 install
```

```
bpctl_start
```

Проверка работы на интерфейсе:

```
bpctl_util enp94s0f0 get_bypass_pwoff
```

При выключенном bypass ожидается следующий вывод:

```
The interface is in the Bypass mode at power off state.
```

4. Настройка автоматического запуска bpctl

Если в системе уже установлен DosGate, важно, чтобы XDP от DosGate применялся после инициализации bypass-драйвера и запуска утилиты bpctl.

Для этого создаем systemd-сервис:

```
nano /etc/systemd/system/bpctl_autostart.service
```

Добавляем содержимое:

```
[Unit]
Description=Bypass Service
After=network.target

[Service]
Type=oneshot
ExecStartPre=/bin/bash -c 'CHECK_BPCTL=$(bpctl_util info); if echo $CHECK_BPCTL
| grep -q "Bypass-SD"; then echo "bpctl is already running"; exit 1; fi'
ExecStart=/bin/bpctl_start
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

Активируем сервис:

```
sudo systemctl daemon-reload
```

```
systemctl enable --now bpctl_autostart
```

```
systemctl status bpctl_autostart
```

Перезагрузить сервер и убедиться, что интерфейсы не отключаются при совместной работе DosGate и bpctl.

Проверка успешного запуска bpctl:

```
bpctl_util
```

Если утилита запустилась корректно, она выведет список доступных команд.

Проверка работы XDP на интерфейсах:

```
ip link | grep xdp
```

Если xdp присутствует на интерфейсах — DosGate запущен и работает корректно.

Изменение максимального размера XDP map

XDP map — это таблица «ключ–значение», к которой обращаются BPF-программы. В качестве ключа могут использоваться IP-адрес источника, параметры соединения (l3_proto — IPv4 или IPv6, l4_proto — TCP или UDP, адреса и порты источника и назначения) или любые другие идентификаторы, по которым ядро должно быстро находить запись. Значение XDP map содержит служебные данные: счётчики пакетов или байт, временные метки, флаги состояния или другие параметры, которые используют правила DosGate.

XDP maps применяются для хранения IP-адресов и соединений, поддержки счётчиков, а также для TCP-авторизации — в этом случае таблица фиксирует текущий шаг и состояние отправителя, чтобы на каждом пакете понимать, на каком этапе проверки находится отправитель.

XDP maps являются частью жизненного цикла объектов BPF и размещаются в файловой системе **bpffs** по пути:

```
/sys/fs/bpf
```

Для DosGate используется отдельное поддерево:

```
/sys/fs/bpf/dosgate
```

В системе доступны два уровня XDP maps. Глобальные карты едины для всей системы и находятся по пути:

```
/sys/fs/bpf/dosgate/base/maps
```

Локальные карты относятся к конкретной арене и размещаются здесь:

```
/sys/fs/bpf/dosgate/contexts/<name>/maps
```

Доступные для изменения XDP map

Используйте команду `dosgate -m` чтобы вывести общий список XDP map доступных к изменению в CLI.

Список XDP map

```
Map "daemon_stats_map", owned by daemon
    Description: Daemon Statistics
    Pin as: daemon_stats_map
    Type percpu_array (6), flags 0x0 ()
    Key size 4b, value size 16b
    Default max entries 6
    Features: BTF-, L2-
    Tunables: max entries-
```

```
Map "daemon_xsk_map", owned by daemon
    Description: AF_XDP redirect to upper half
    Pin as: daemon_xsk_map
    Type xskmap (17), flags 0x0 ()
    Key size 4b, value size 4b
    Default max entries 1000000
    Features: BTF-, L2-
    Tunables: max entries-

Map "daemon_log_entry_map", owned by daemon
    Description: Logging export ring buffer
    Pin as: not pinned
    Type ringbuf (27), flags 0x0 ()
    Key size 0b, value size 0b
    Default max entries 1048576
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 512, max 67108864
    Suffix unit: 1024

Map "daemon_log_map", owned by daemon
    Description: Logging export ring buffer collection
    Pin as: daemon_log_map
    Type array_of_maps (12), flags 0x0 ()
    Key size 4b, value size 4b
    Default max entries 0, effective 4
    Features: BTF-, L2+
    Tunables: max entries-
    L2 map: daemon_log_entry_map

Map "daemon_flow_index_map", owned by daemon
    Description: Logging export flow collector index
    Pin as: daemon_flow_index_map
    Type percpu_array (6), flags 0x0 ()
    Key size 4b, value size 144b
    Default max entries 65536
    Features: BTF-, L2-
    Tunables: max entries-

Map "daemon_flow_cache_map", owned by daemon
    Description: Logginig export flow collector cache
    Pin as: daemon_flow_cache_map
    Type hash (1), flags 0x0 ()
    Key size 136b, value size 24b
    Default max entries 0, effective 262148
    Features: BTF-, L2-
    Tunables: max entries-

Map "interfaces_map", owned by daemon
    Description: Interfaces
    Pin as: interfaces_map
    Type devmap (14), flags 0x0 ()
    Key size 4b, value size 4b
```

```
    Default max entries 100
    Features: BTF-, L2-
    Tunables: max entries-

Map "geoip_map", owned by daemon
    Description: GeoIP database
    Pin as: geoip_map
    Type lpm_trie (11), flags 0x1 (BPF_F_NO_PREALLOC)
    Key size 32b, value size 8b
    Default max entries 10000000
    Features: BTF-, L2-
    Tunables: max entries-

Map "ctx_map", owned by daemon
    Description: Context
    Pin as: ctx_map
    Type percpu_array (6), flags 0x0 ()
    Key size 4b, value size 3336b
    Default max entries 1
    Features: BTF-, L2-
    Tunables: max entries-

Map "prog_map", owned by arena
    Description: Program
    Pin as: prog_map
    Type prog_array (3), flags 0x0 ()
    Key size 4b, value size 4b
    Default max entries 1100
    Features: BTF-, L2-
    Tunables: max entries-

Map "tree_ipv4_map", owned by arena
    Description: IPv4 Router
    Pin as: tree_ipv4_map
    Type lpm_trie (11), flags 0x1 (BPF_F_NO_PREALLOC)
    Key size 8b, value size 8b
    Default max entries 1000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "tree_ipv6_map", owned by arena
    Description: IPv6 Router
    Pin as: tree_ipv6_map
    Type lpm_trie (11), flags 0x1 (BPF_F_NO_PREALLOC)
    Key size 24b, value size 8b
    Default max entries 1000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000
```

```
Map "stats_map", owned by arena
    Description: Statistics
    Pin as: stats_map
    Type percpu_array (6), flags 0x0 ()
    Key size 4b, value size 16b
    Default max entries 256012
    Features: BTF-, L2-
    Tunables: max entries-

Map "lock_map", owned by arena
    Description: Locking
    Pin as: lock_map
    Type array (2), flags 0x0 ()
    Key size 4b, value size 4b
    Default max entries 1048576
    Features: BTF+, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "hmark_inet_map", owned by arena
    Description: Host Mark IPv4
    Pin as: hmark_inet_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 22b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "hmark_inet6_map", owned by arena
    Description: Host Mark IPv6
    Pin as: hmark_inet6_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 34b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "sdhmark_inet_map", owned by arena
    Description: Source-Destination Host Mark IPv4
    Pin as: sdhmark_inet_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 26b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000
```

```
Map "sdhmark_inet6_map", owned by arena
    Description: Source-Destination Host Mark IPv6
    Pin as: sdhmark_inet6_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 50b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "dmark_inet_map", owned by arena
    Description: Destination Host Mark IPv4
    Pin as: dmark_inet_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 22b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "dmark_inet6_map", owned by arena
    Description: Destination Host Mark IPv6
    Pin as: dmark_inet6_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 34b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "connmark_map", owned by arena
    Description: Connection Mark IPv4 and IPv6
    Pin as: connmark_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 124b, value size 16b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "ratelimit_map", owned by arena
    Description: RateLimit IPv4 and IPv6
    Pin as: ratelimit_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 124b, value size 32b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
```

```
    Suffix unit: 1000

Map "sample_map", owned by arena
    Description: Sampling IPv4 and IPv6
    Pin as: sample_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 124b, value size 8b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "tcpauth_map", owned by arena
    Description: TCP Authentication temporary data
    Pin as: tcpauth_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 34b, value size 48b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "prefixset_map", owned by arena
    Description: Prefixset storage
    Pin as: prefixset_map
    Type lpm_trie (11), flags 0x1 (BPF_F_NO_PREALLOC)
    Key size 32b, value size 8b
    Default max entries 10000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "rate_map", owned by arena
    Description: Rate estimation data
    Pin as: rate_map
    Type lru_hash (9), flags 0x2 (BPF_F_NO_COMMON_LRU)
    Key size 124b, value size 104b
    Default max entries 1000000, effective 2000000
    Features: BTF-, L2-
    Tunables: max entries+
    Max entries min 100, max 100000000
    Suffix unit: 1000

Map "pstats_map", owned by passthrough
    Description: Passthrough statistics
    Pin as: pstats_map
    Type percpu_array (6), flags 0x0 ()
    Key size 4b, value size 16b
    Default max entries 6
```

```
    Features: BTF-, L2-
    Tunables: max entries-
```

Настройка размера XDP map

Перед изменением конфигурации необходимо остановить работу сервиса:

```
sudo service dosgate stop
```

Очистить данные во всех XDP map командой:

```
dgadm --clear=sa -y
```

Для изменения размера конкретной XDP map выполнить следующие шаги:

Определите размер ключа (`Key size`) и значение (`Value size`) в байтах для требуемой XDP map.

Установите новый максимальный размер XDP map (например: 2 000 000 записей для TCP-авторизации).

Рассчитайте требуемый объем оперативной памяти по формуле:

Объем памяти = `Key size` * Новый размер XDP map

> **Примечание**
> Если у XDP map `Key size` и `Value size` равны 0b, используйте обозначения K (килобайты), M (мегабайты), G (гигабайты).

- Изменение размера локальной XDP map

Откройте файл **/etc/dosgate.conf** и добавьте параметр maps с новым значением:

```
arenas:
  - name: first
    id: 1
    maps:
      hmark_inet_map: 50000000
```

- Изменение размера глобальной XDP map

```
daemon:
  maps:
    daemon_log_entry_map: 1M

arenas:
  - name: first
    id: 1
```

Запуск сервиса DosGate:

```
service dosgate start
```

Убедитесь, что сервис работает корректно и ошибки отсутствуют:

```
service dosgate status
```