

Proxmox: HugePages для виртуальных машин

Источники:

- <https://dev.to/sergelogvinov/proxmox-hugepages-for-vms-1fh3>
- <https://telegra.ph/Proxmox-HugePages-dlya-virtualnyh-mashin-08-30>
- <https://www.kernel.org/doc/Documentation/vm/hugetlbpage.txt>
- https://pve.proxmox.com/wiki/Manual:_qm.conf

Когда приложению нужно получить значение какой-то переменной, оно просит процессор достать его из памяти по адресу этой переменной. Но этот адрес виртуальный, и процессору приходится переводить его в физический - то есть в реальное место на чипе памяти сервера. Память у процессора разбита на блоки, которые называются страницами, и размер страницы зависит от архитектуры CPU. Например, в системах x86_64 каждая страница обычно равна 4 КБ. Получается, что на сервере с 16 ГБ памяти будет примерно 4 миллиона страниц.

И вот здесь может начаться проблема: чем больше памяти у сервера, тем больше страниц приходится обслуживать процессору. А перевод виртуальных адресов в физические начинает замедляться. Чтобы ускорить работу, придумали HugePages.

HugePage - это страница, которая больше стандартной. В x86_64 их размер может быть либо 2 МБ, либо 1 ГБ. За счёт этого у процессора становится меньше страниц для управления, и всё работает быстрее.

Например, если вы создаёте виртуальную машину (VM) с 1 ГБ памяти, Linux по умолчанию разобьёт её на примерно 256 000 страниц. А если использовать HugePages размером 1 ГБ - будет всего одна страница. Количество работы для процессора резко уменьшается, и производительность растёт.

Как настроить HugePages

Сначала проверьте, какие размеры HugePages поддерживает ваша система. Для этого выполните команду:

```
ls /sys/devices/system/node/node0/hugepages/
```

Пример вывода для AMD EPYC:

```
hugepages - 1048576kB  hugepages - 2048kB
```

Пример вывода для ARM Ampere Altra:

```
hugepages - 1048576kB  hugepages - 2048kB  hugepages - 32768kB  hugepages - 64kB
```

Это значит, что на AMD доступны HugePages размером 4KB, 2MB и 1GB. В примере мы будем использовать HugePages по 1GB.

Чтобы их настроить, нужно отредактировать файл `/etc/default/grub` и добавить параметры `default_hugepagesz` и `hugepagesz` в переменную `GRUB_CMDLINE_LINUX`. В нашем случае выставим размер страницы 1GB и количество HugePages равным 16:

```
GRUB_CMDLINE_LINUX="default_hugepagesz=1G hugepagesz=1G hugepages=16"
```

Если у вас архитектура NUMA, то параметры `hugepagesz` и `hugepages` придётся задавать для каждого узла отдельно. Например, если у вас 2 узла, количество HugePages нужно разделить и указать в формате `$HUNODE:$AMOUNT,$HUNODE:$AMOUNT`:

```
GRUB_CMDLINE_LINUX="default_hugepagesz=1G hugepagesz=1G hugepages=0:8,1:8"
```

Ядро Linux резервирует под HugePages 16 ГБ памяти, но эта память используется только для HugePages и недоступна остальным приложениям. Поэтому количество HugePages нужно рассчитывать исходя из общей памяти сервера, оставив запас для самой системы и других приложений. Для операционной системы должно оставаться минимум 1 ГБ свободной памяти. Если вы используете ZFS или Ceph — им тоже нужно выделить дополнительную память.

После того как вы отредактировали `/etc/default/grub`, обновите конфигурацию `grub`:

```
update-grub
```

И не забудьте перезагрузить сервер, чтобы изменения вступили в силу.

Проверить HugePages можно командой:

```
cat /proc/meminfo | grep Huge
```

Пример вывода:

```
AnonHugePages:      0 kB
ShmemHugePages:     0 kB
FileHugePages:      0 kB
HugePages_Total:    448
HugePages_Free:      0
HugePages_Rsvd:      0
HugePages_Surp:      0
Hugepagesize:       1048576 kB
Hugetlb:            469762048 kB
```

Если у вас NUMA-архитектура, HugePages нужно проверять для каждого узла. Для этого выполните:

```
numastat -cm | egrep "Huge|Node|Mem"
```

Пример вывода:

	Node 0	Node 1	Node 2	Node 3	Total
MemTotal	128655	129020	129020	128977	515672
MemFree	12654	12933	12643	13139	51369

MemUsed	116001	116087	116377	115838	464303
AnonHugePages	0	0	0	0	0
ShmemHugePages	0	0	0	0	0
HugePages_Total	114688	114688	114688	114688	458752
HugePages_Free	0	0	0	0	0
HugePages_Surp	0	0	0	0	0

В этом примере у нас 4 узла, на каждом по 128 ГБ памяти, и на каждом выделено по 112 HugePages ($112 \times 1024 = 114\,688$ MB).

Как настроить HugePages для виртуальных машин

Большинство параметров HugePages недоступны из интерфейса Proxmox. Их нужно задавать через командную строку или API.

В примере будем использовать HugePages размером 1 ГБ. Количество HugePages зависит от объёма памяти, выделенной VM. Для VM с 8192 MB памяти это будет 8 HugePages.

Пример конфигурации VM `/etc/pve/qemu-server/$ID.conf`:

```
# VM config /etc/pve/qemu-server/$ID.conf
memory: 8192
hugepages: 1024
keephugepages: 1
```

HugePages - это выделенные страницы памяти, закреплённые за конкретной VM. Другие виртуалки не могут использовать те же страницы. Параметр `keephugepages` позволяет оставлять HugePages зарезервированными даже после остановки VM. Такая схема не только ускоряет работу, но и повышает безопасность VM.

Некоторые приложения, например базы данных, умеют использовать HugePages для хранения данных в памяти. Это ускоряет доступ к данным. Чтобы разрешить использование HugePages внутри VM, добавьте флаг `CPU +pdpre1gb` в конфигурацию:

```
# VM config /etc/pve/qemu-server/$ID.conf
cpu: host,flags=+pdpre1gb;
```

[proxmox](#), [HugePages](#)

From:

<https://wiki.rtzra.ru/> - RTzRa's hive

Permanent link:

<https://wiki.rtzra.ru/software/proxmox/proxmox-hugepages>

Last update: **2025/08/31 16:25**

