

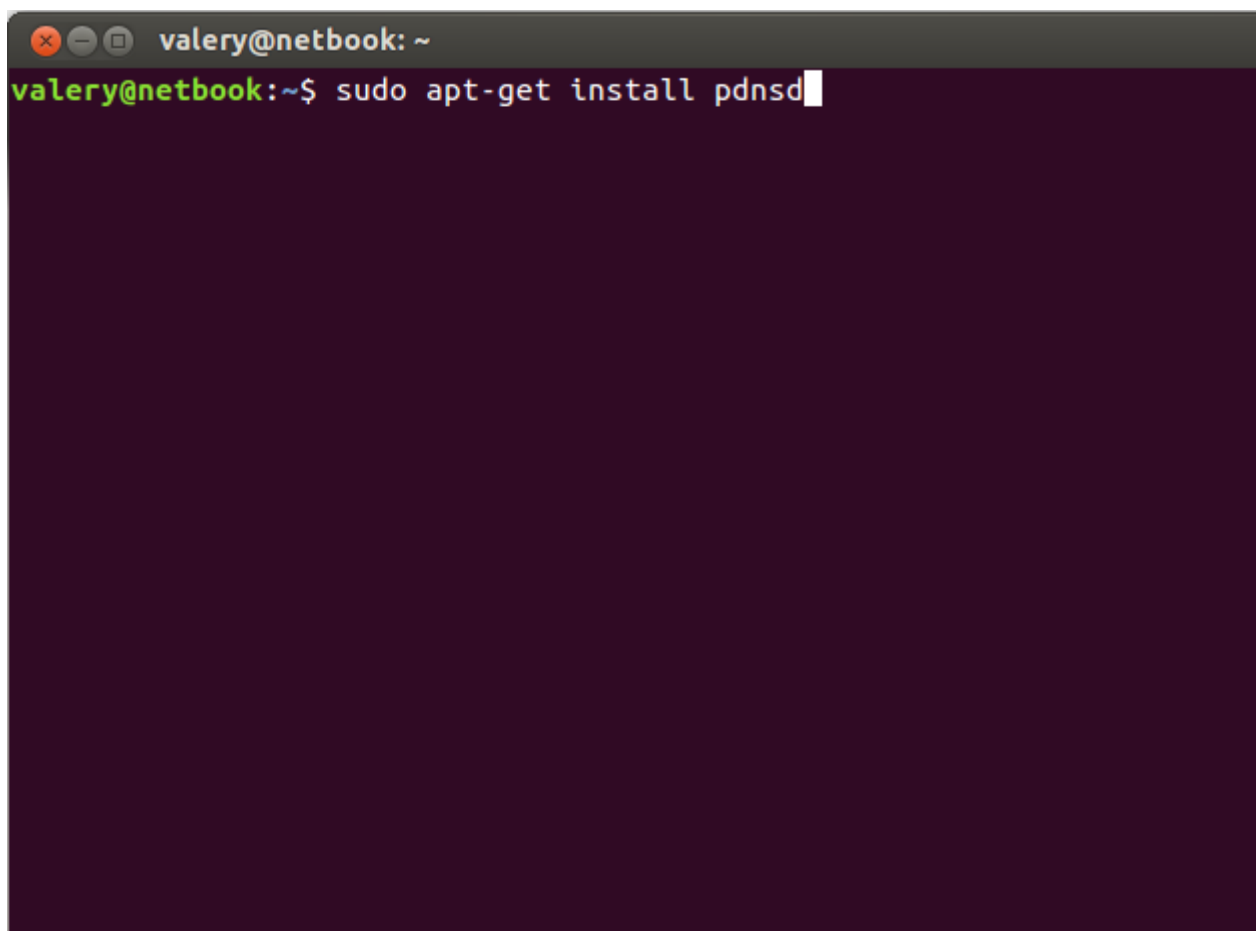
Ускоряем Интернет с помощью локального кэширующего DNS-сервера - PDNSD.

Не секрет, что скорость открытия web-страниц в браузере зависит от ширины Вашего Интернет-канала.

Однако не меньшее влияние на этот процесс имеет работа DNS-серверов. Если предоставляемый Вам провайдером DNS-сервер сбоит или отвечает с большой задержкой - то никакой сверхширокий канал не улучшит ситуацию. Решить проблему можно установив свой локальный кэширующий DNS-сервер.

В качестве такого сервера мы будем использовать лёгкий и быстрый PDNSD. Установим его командой в терминале:

```
sudo apt-get install pdnsd
```

A terminal window with a dark purple background. The title bar shows a window icon, a close button, and the text 'valery@netbook: ~'. The terminal prompt is 'valery@netbook:~\$' followed by the command 'sudo apt-get install pdnsd' and a white cursor at the end of the line.

Перед Вами должно появиться окно с запросом о способе настройки pdnsd. Если такое окно не появилось - то скорее всего Вы уже устанавливали пакет pdnsd и удалили его через `sudo apt-get remove pdnsd` (при этом конфигурационные файлы пакета остались). В таком случае необходимо запустить перенастройку пакета командой в терминале:

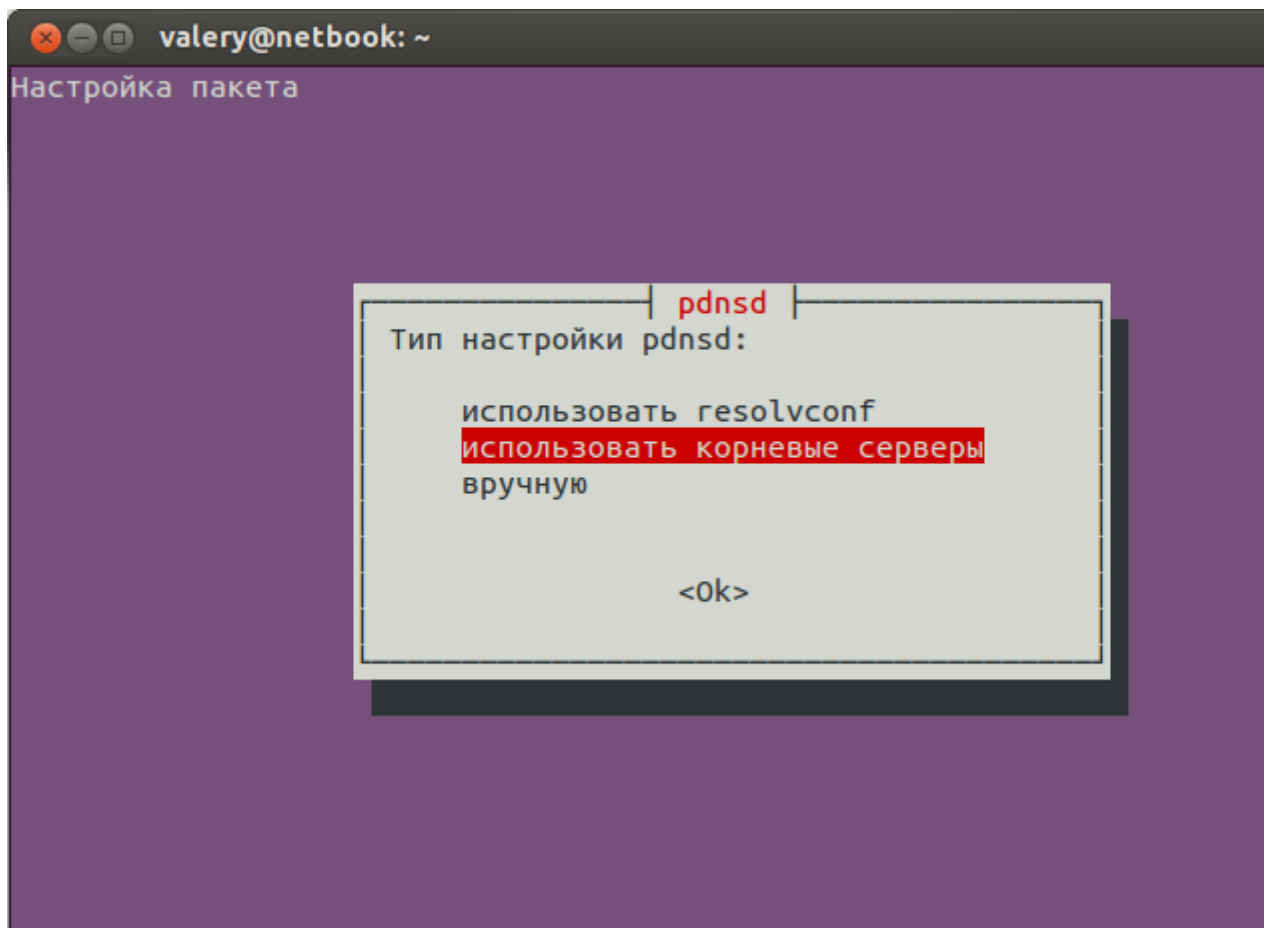
```
sudo dpkg-reconfigure pdnsd
```

```
valery@netbook: ~  
valery@netbook:~$ sudo dpkg-reconfigure pdnsd
```

Итак, перед нами окно с запросом о способе настройки pdnsd.

```
valery@netbook: ~  
Настройка пакета  
pdnsd  
Выберите способ настройки pdnsd, который больше всего вам подходит  
- использовать resolvconf      : использовать данные,  
                                предоставляемые resolvconf.  
- использовать корневые серверы: сделать из pdnsd кэширующий,  
                                рекурсивный DNS сервер.  
- вручную                      : настройка вручную. Демон pdnsd  
                                не запустится, пока вы не  
отредактируете                  файлы /etc/pdnsd.conf и  
/etc/default/pdnsd.  
Замечание: Если у вас уже есть DNS сервер, принимающий запросы с  
<Ok>
```

Т.к. мы хотим получить быстрое, стабильное и независимое от DNS-провайдеров решение - нам нужно использование корневых серверов. Перемещаемся клавишей <Tab> на пункт <Ok> и жмём <Enter>.



На следующем окне выбираем выбираем пункт "использовать корневые серверы" и затем клавишей <Tab> перемещаемся на пункт <Ok> и жмём <Enter>.

Проверим настройки запуска pdnsd. Набираем в терминале:

```
cat /etc/default/pdnsd
```

```
valery@netbook: ~  
valery@netbook:~$ cat /etc/default/pdnsd
```

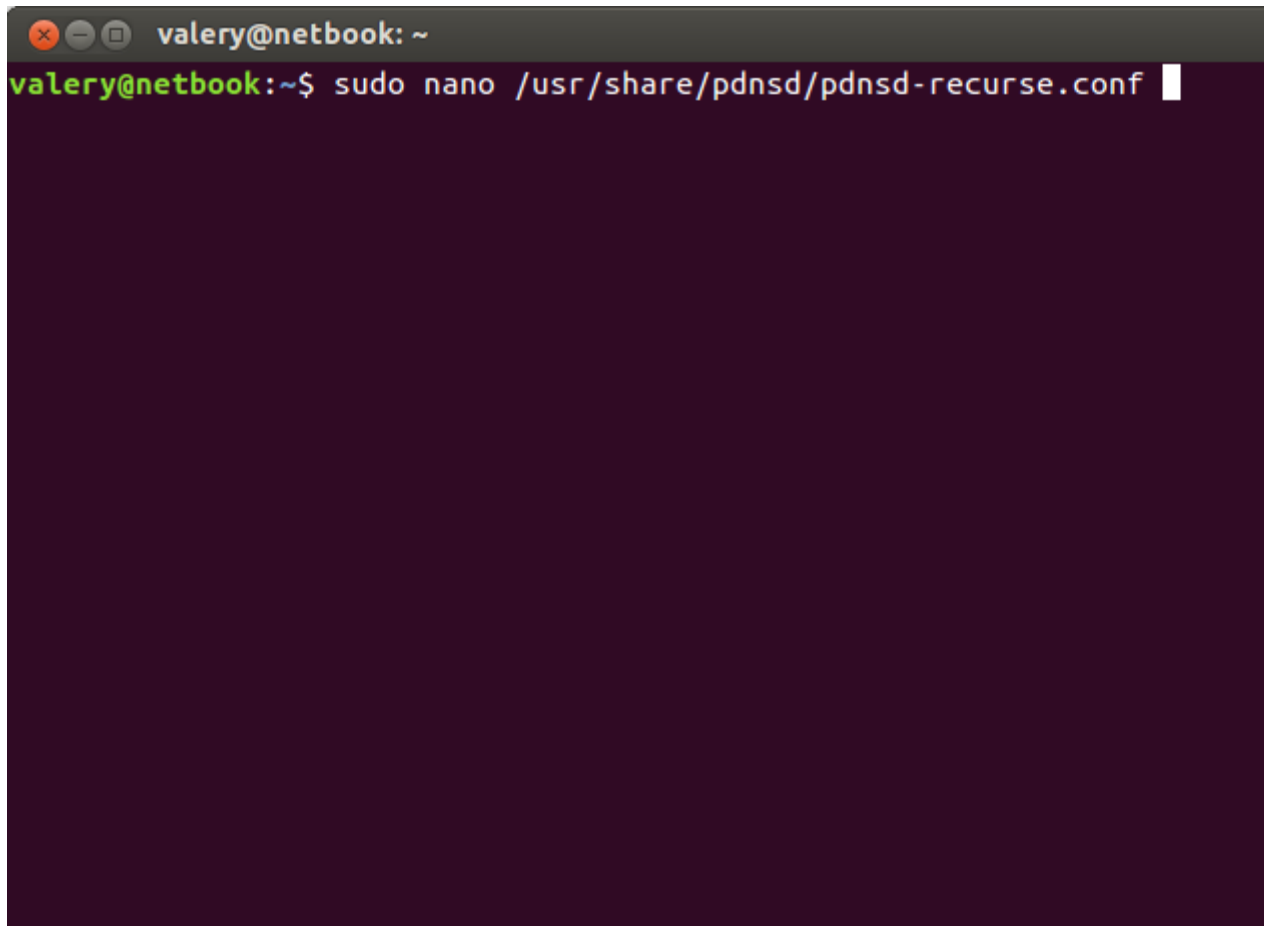
Проверяем наличие опций START_DAEMON=yes и AUTO_MODE=recurse.

```
valery@netbook: ~  
valery@netbook:~$ cat /etc/default/pdnsd  
# do we start pdnsd ?  
START_DAEMON=yes  
# auto-mode, overrides /etc/pdsnd.conf if set [see /usr/share/pdnsd/]  
AUTO_MODE=recurse  
# optional CLI options to pass to pdnsd(8)  
START_OPTIONS=  
valery@netbook:~$
```

В данном режиме пакет pdnsd будет брать настройки не из файла /etc/pdnsd.conf, а из файла /usr/share/pdnsd/pdnsd-recurse.conf

Откроем файл настроек pdnsd командой в терминале:

```
sudo nano /usr/share/pdnsd/pdnsd-recurse.conf
```

A terminal window with a dark background. The title bar shows window control icons and the text 'valery@netbook: ~'. The prompt is 'valery@netbook:~\$' and the command entered is 'sudo nano /usr/share/pdnsd/pdnsd-recurse.conf'. A white cursor is at the end of the command line.

```
valery@netbook:~$ sudo nano /usr/share/pdnsd/pdnsd-recurse.conf
```

Мы увидим следующее:

```
valery@netbook: ~
GNU nano 2.2.6      Файл: /usr/share/pdnsc/pdnsc-recurse.conf

/* Debian specific configuration to work as a recursive resolver */

global {
    perm_cache = 2048;
    cache_dir = "/var/cache/pdnsc";
    run_as = "pdnsc";
    server_ip = 127.0.0.1; // Use eth0 here if you want to allow other
                          // machines on your network to query pdnsc

    status_ctl = on;
    paranoid = on;
    min_ttl = 15m;        // Retain cached entries at least 15 minutes
    max_ttl = 1w;         // One week.
    timeout = 10;         // Global timeout option (10 seconds).

    delegation_only = "com","net";
}

server {
    label = "recurse";

```

Прочитана 91 строка

^G	Помощь	^O	Записать	^R	ЧитФайл	^Y	ПредСтр	^K	Вырезать	^C	Тер
^X	Выход	^J	Выровнять	^W	Поиск	^V	СледСтр	^U	ОтмВырезк	^T	Сл

Параметр `perm_cache` определяет сколько памяти в килобайтах будет отводиться под кеш (по умолчанию 2048).

Если Ваша система обслуживает целую сеть - рекомендуется увеличить это значение, например до 65536.

Ещё один важный для нас параметр `server_ip` (по умолчанию 127.0.0.1) - он определяет интерфейс на котором `pdnsc` будет принимать запросы. Для одиночного ПК можно оставить 127.0.0.1, но если Вы планируете быть быстрым DNS-сервером для своей сети - нужно указать Ваш сетевой интерфейс, например `eth0`.

```
valery@netbook: ~
GNU nano 2.2.6      Файл: /usr/share/pdnsd/pdnsd-recurse.conf

/* Debian specific configuration to work as a recursive resolver */

global {
    perm_cache = 65536;
    cache_dir = "/var/cache/pdnsd";
    run_as = "pdnsd";
    server_ip = eth0; // Use eth0 here if you want to allow other
                      // machines on your network to query pdnsd

    status_ctl = on;
    paranoid = on;
    min_ttl = 15m; // Retain cached entries at least 15 minutes
    max_ttl = 1w; // One week.
    timeout = 10; // Global timeout option (10 seconds).

    delegation_only = "com","net";
}

server {
    label = "recurse";
}
```

^G Помощь ^O Записать ^R ЧитФайл ^Y ПредСтр ^K Вырезать ^C Те
^X Выход ^J Выворнять ^W Поиск ^V СледСтр ^U ОтмВырезк ^T Сл

Сохраняем настройки нажав <Ctrl> + <o> и выходим нажав <Ctrl> + <x>.

По-умолчанию в Ubuntu применяется DNS-сервер dnsmasq. Его отличия от pdnsd в том, что он использует dns-сервера из файла /etc/resolv.conf и кэш держит только в оперативной памяти (при перезагрузке кэш теряется). Перед запуском pdnsd нам необходимо удалить dnsmasq, т.к. они оба по-умолчанию используют порт 53. Для удаления dnsmasq выполним в терминале:

```
sudo apt-get remove dnsmasq
```

```
valery@netbook: ~  
valery@netbook:~$ sudo apt-get remove dnsmasq
```

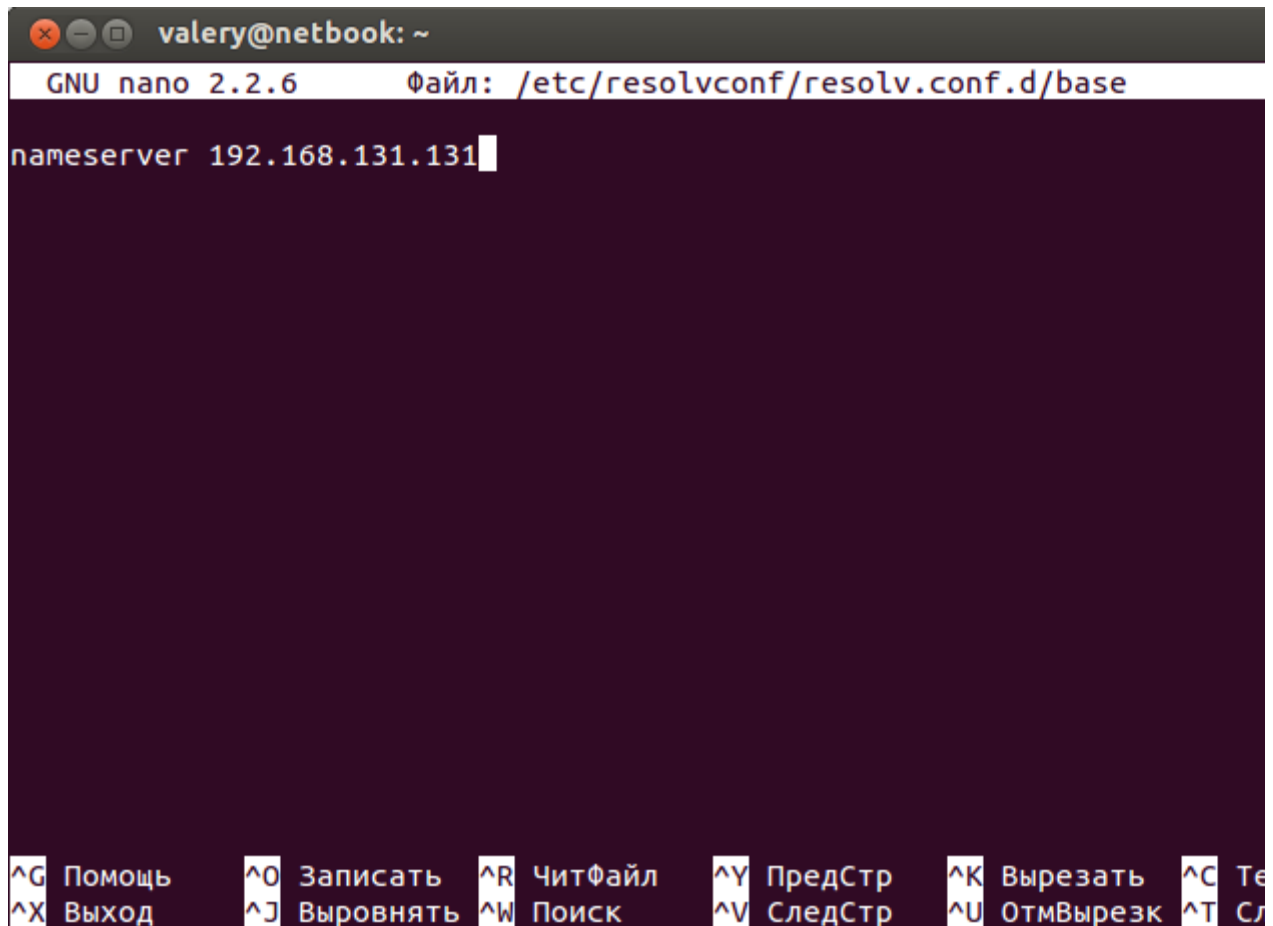
Теперь нам нужно указать пакету resolvconf (именно он формирует файл /etc/resolv.conf) наш новый быстрый DNS-сервер. Для этого выполним в терминале:

sudo nano /etc/resolvconf/resolv.conf.d/base

```
valery@netbook: ~  
valery@netbook:~$ sudo nano /etc/resolvconf/resolv.conf.d/base
```

Вписываем nameserver и IP-адрес (если Вы указывали server_ip=eth0 в файле

/usr/share/pdnsd/pdnsd-recurse.conf) или 127.0.0.1 (если Вы указывали server_ip=127.0.0.1 в файле /usr/share/pdnsd/pdnsd-recurse.conf).



```
valery@netbook: ~  
GNU nano 2.2.6      Файл: /etc/resolvconf/resolv.conf.d/base  
nameserver 192.168.131.131  
  
^G Помощь      ^O Записать    ^R ЧитФайл    ^Y ПредСтр    ^K Вырезать    ^C Те  
^X Выход      ^J Выворнять  ^W Поиск      ^V СледСтр    ^U ОтмВырезк  ^T Сл
```

Сохраняем настройки нажав <Ctrl> + <o> и выходим нажав <Ctrl> + <x>.

Обратите внимание:

- в файле настроек сетевых интерфейсов /etc/network/interfaces не должно быть строк с параметром dns-nameservers;
- в описываемой статье подразумевается использование статического IP-адреса на Вашем ПК.

Для чистоты эксперимента рекомендуется полностью перезагрузить ПК.

После перезагрузки убеждаемся, что в /etc/resolv.conf указан именно наш новый DNS-сервер. Выполним в терминале:

```
cat /etc/resolv.conf
```

```
valery@netbook: ~  
valery@netbook:~$ cat /etc/resolv.conf
```

Должен быть только наш DNS-сервер:

```
valery@netbook: ~  
valery@netbook:~$ cat /etc/resolv.conf  
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf  
#      DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN  
nameserver 192.168.131.131  
valery@netbook:~$
```

С помощью команды `dig` проанализируем ускорение dns-запросов. Выполним в терминале:

dig mail.ru

```
valery@netbook: ~  
valery@netbook:~$ dig mail.ru
```

Получим вывод:

```
valery@netbook: ~  
mail.ru.          900      IN       NS       ns3.mail.ru.  
mail.ru.          900      IN       NS       ns1.mail.ru.  
mail.ru.          900      IN       NS       ns5.mail.ru.  
  
;; ADDITIONAL SECTION:  
ns2.mail.ru.      900      IN       A        94.100.186.189  
ns2.mail.ru.      900      IN       AAAA     2a00:1148:1:1322::1:a6  
ns.mail.ru.       900      IN       A        217.69.129.230  
ns.mail.ru.       900      IN       AAAA     2a00:1148:1:1322::1:c0  
ns4.mail.ru.      900      IN       A        94.100.178.100  
ns4.mail.ru.      900      IN       AAAA     2a00:1148:1:1310::1:b6  
ns3.mail.ru.      900      IN       A        94.100.179.93  
ns3.mail.ru.      900      IN       AAAA     2a00:1148:1:1311::1:a6  
ns1.mail.ru.      900      IN       A        94.100.179.159  
ns1.mail.ru.      900      IN       AAAA     2a00:1148:1:1311::1:a6  
ns5.mail.ru.      900      IN       A        217.69.129.241  
ns5.mail.ru.      900      IN       AAAA     2a00:1148:1:1322::1:a6  
  
;; Query time: 191 msec  
;; SERVER: 192.168.131.131#53(192.168.131.131)  
;; WHEN: Mon Feb 24 11:18:48 2014  
;; MSG SIZE  rcvd: 460  
  
valery@netbook:~$
```

Затем ещё раз выполним тот же запрос:

```
dig mail.ru
```

И увидим другой результат:

```
valery@netbook: ~  
mail.ru.      835    IN      NS      ns3.mail.ru.  
mail.ru.      835    IN      NS      ns1.mail.ru.  
mail.ru.      835    IN      NS      ns5.mail.ru.  
  
;; ADDITIONAL SECTION:  
ns2.mail.ru.  835    IN      A       94.100.186.189  
ns2.mail.ru.  835    IN      AAAA    2a00:1148:1:1322::1:a6  
ns.mail.ru.   835    IN      A       217.69.129.230  
ns.mail.ru.   835    IN      AAAA    2a00:1148:1:1322::1:c6  
ns4.mail.ru.  835    IN      A       94.100.178.100  
ns4.mail.ru.  835    IN      AAAA    2a00:1148:1:1310::1:b6  
ns3.mail.ru.  835    IN      A       94.100.179.93  
ns3.mail.ru.  835    IN      AAAA    2a00:1148:1:1311::1:a6  
ns1.mail.ru.  835    IN      A       94.100.179.159  
ns1.mail.ru.  835    IN      AAAA    2a00:1148:1:1311::1:a6  
ns5.mail.ru.  835    IN      A       217.69.129.241  
ns5.mail.ru.  835    IN      AAAA    2a00:1148:1:1322::1:a6  
  
;; Query time: 6 msec  
;; SERVER: 192.168.131.131#53(192.168.131.131)  
;; WHEN: Mon Feb 24 11:19:53 2014  
;; MSG SIZE  rcvd: 460  
  
valery@netbook:~$
```

Время запроса значительно сократилось. Аналогично можно попробовать потестировать обращения к другим доменам. После попадания запроса в кэш - ответ получается гораздо быстрее.

Текущее состояние и заполненность кэша pdnsd можно узнать выполнив в терминале:

```
sudo pdnsd-ctl status
```

```
valery@netbook: ~  
valery@netbook:~$ sudo pdnsd-ctl status
```

Поднявшись в начало вывода можно увидеть данные о заполненности кэша:

```
valery@netbook:~$ sudo pdnsd-ctl status  
Opening socket /var/cache/pdnsd/pdnsd.status  
pdnsd-1.2.8-par running on netbook.localdomain.  
  
Cache status:  
=====
```

65536 kB maximum disk cache size.
16514 of 67119104 bytes (0.0246%) memory cache used in 62 entries.

```
  
Thread status:  
=====
```

server status thread is running.
pdnsd control thread is running.
tcp server thread is running.
udp server thread is running.
32 query threads spawned in total (0 queries dropped).
0 running query threads (0 active, 0 queued).

```
  
Configuration:  
=====
```

Global:

Cache size: 65536 kB

В приведённом примере это всего 62 записи, занявшие 0.0246% от выделенных нами под кэш 65536Кб.

Содержимое кэша pdnsd можно увидеть выполнив в терминале:

sudo pdnsd-ctl dump

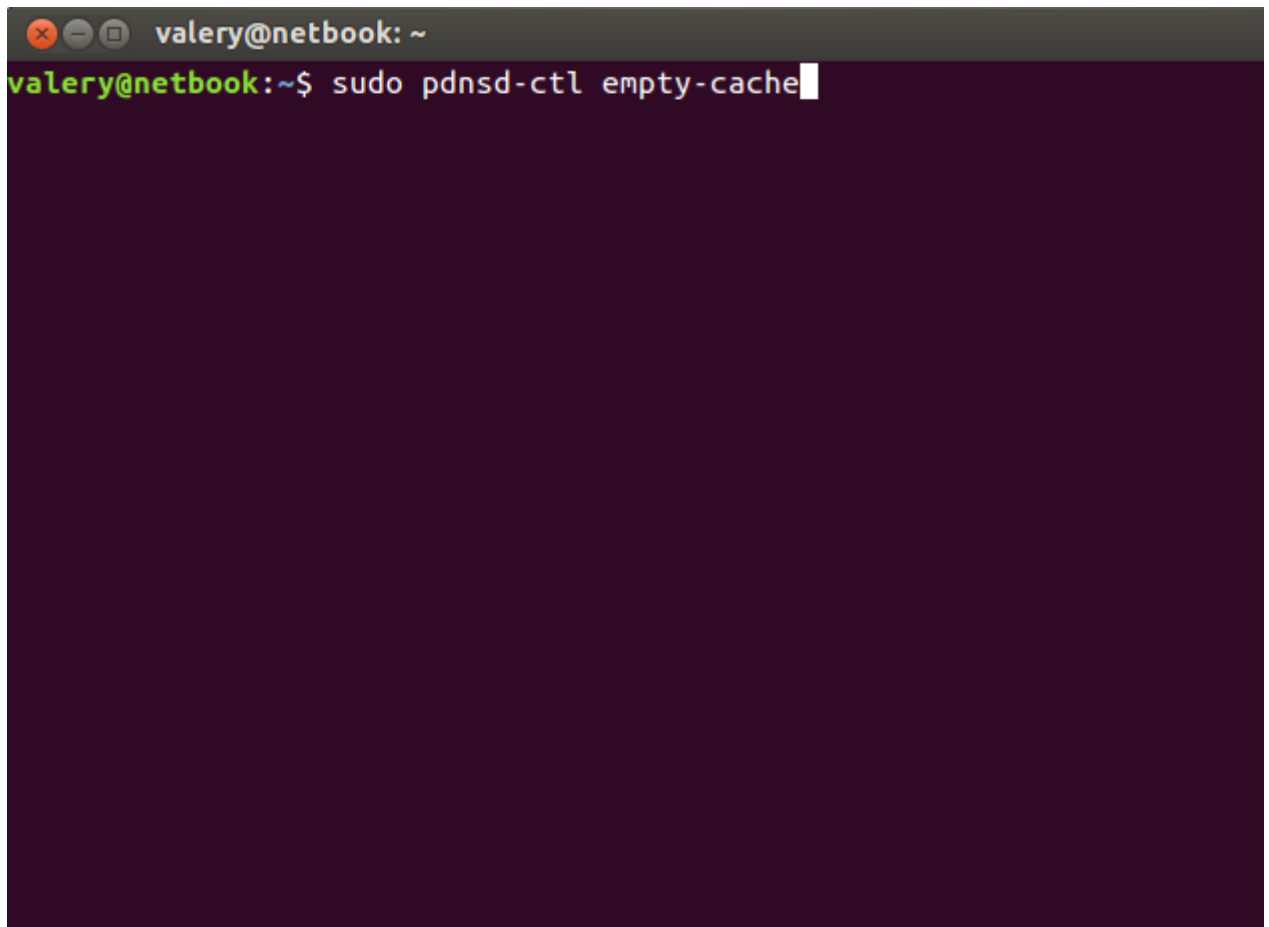
```
valery@netbook: ~  
valery@netbook:~$ sudo pdnsd-ctl dump
```

Получим что-то вида:

```
valery@netbook: ~  
02/24 11:22:21 NS ns2.yandex.ru.  
02/24 11:22:21 NS ns1.yandex.ru.  
ns6.nic.ru.  
02/24 11:22:56 A 31.177.74.100  
02/24 11:22:56 AAAA 2a02:2090:ec00:9040:31:177:74:100  
ns7.nic.ru.  
02/24 11:22:56 A 31.177.71.100  
02/24 11:22:56 AAAA 2a02:2090:ec00:9000:31:177:71:100  
vk.ru.  
02/24 11:22:56 A 89.111.176.202  
02/24 11:22:56 NS ns3-l2.nic.ru.  
02/24 11:22:56 NS ns4-l2.nic.ru.  
02/24 11:22:56 NS ns8-l2.nic.ru.  
02/24 11:22:56 NS ns8-cloud.nic.ru.  
02/24 11:22:56 NS ns4-cloud.nic.ru.  
www.odnoklassniki.ru.  
02/24 11:23:11 A 217.20.147.94  
Succeeded  
valery@netbook:~$
```

Сбросить (очистить) кэш можно командой в терминале:

```
sudo pdnsd-ctl empty-cache
```

A terminal window with a dark purple background. The title bar at the top shows a red close button, a grey minimize button, and a grey maximize button, followed by the text 'valery@netbook: ~'. The terminal content shows the prompt 'valery@netbook:~\$' in green, followed by the command 'sudo pdnsd-ctl empty-cache' in white, with a white cursor at the end of the command.

Но делать это просто так не стоит, т.к. именно накопленный кэш - ускоряет наши dns-запросы.