

Модуль ngx_http_map_module

[Пример конфигурации](#)

[Директивы](#)

[map](#)

[map_hash_bucket_size](#)

[map_hash_max_size](#)

Модуль `ngx_http_map_module` создаёт переменные, значения которых зависят от значений других переменных.

Пример конфигурации

```
map $http_host $name {
    hostnames;

    default      0;

    example.com  1;
    *.example.com 1;
    example.org  2;
    *.example.org 2;
    .example.net 3;
    wap.*        4;
}

map $http_user_agent $mobile {
    default      0;
    "~Opera Mini" 1;
}
```

Директивы

Синтаксис: **map** строка \$переменная { ... }

Умолчение: —

Контекст: http

Создаёт новую переменную, значение которой зависит от значений одной или более исходных переменных, указанных в первом параметре.



NGINX
SPONSORED BY F5

[english](#)

[русский](#)

[новости](#) [en]

[об nginx](#)

[скачать](#)

[безопасность](#) [en]

[документация](#)

[faq](#)

[книги](#) [en]

[сообщество](#)

[компания](#)

[x.com](#)

[blog](#)

[njs](#)

[ingress controller](#)

[gateway fabric](#)

До версии 0.9.0 в качестве первого параметра можно было указать только одну переменную.

Поскольку переменные вычисляются только в момент использования, само по себе наличие даже большого числа объявлений переменных “map” не влечёт за собой никаких дополнительных расходов на обработку запросов.

Параметры внутри блока `map` задают соответствие между исходными и результирующими значениями.

Исходные значения задаются строками или регулярными выражениями (0.9.6).

Строки проверяются без учёта регистра.

Перед регулярным выражением ставится символ “~”, если при сравнении следует учитывать регистр символов, либо символы “~*” (1.0.4), если регистр символов учитывать не нужно. Регулярное выражение может содержать именованные и позиционные выделения, которые могут затем использоваться в других директивах совместно с результирующей переменной.

Если исходное значение совпадает с именем одного из специальных параметров, описанных ниже, перед ним следует поставить символ “\”.

В качестве результирующего значения можно указать текст, переменную (0.9.0) и их комбинации (1.11.0).

Также поддерживаются следующие специальные параметры:

`default` *значение*

задаёт результирующее значение, если исходное значение не совпадает ни с одним из перечисленных. Если параметр `default` не указан, результирующим значением по умолчанию будет пустая строка.

`hostnames`

указывает, что в качестве исходных значений можно использовать маску для первой или последней части имени хоста, например,

```
*.example.com 1;  
example.*      1;
```

Вместо двух записей

```
example.com 1;  
*.example.com 1;
```

можно использовать одну:

```
.example.com 1;
```

Этот параметр следует указывать перед списком значений.

include *файл*

включает файл со значениями. Включений может быть несколько.

volatile

указывает, что переменная не кэшируется (1.11.7).

Поиск производится в следующем порядке приоритета и прекращается на первом подходящем варианте:

1. строковое значение без маски
2. самое длинное строковое значение с маской в начале, например “ *.example.com ”
3. самое длинное строковое значение с маской в конце, например “ mail.* ”
4. первое подходящее регулярное выражение (в порядке следования в конфигурационном файле)
5. значение по умолчанию (default)

Синтаксис: **map_hash_bucket_size** *размер*;

Умолчание: map_hash_bucket_size 32|64|128;

Контекст: http

Задаёт размер корзины в хэш-таблицах для переменных [map](#). Значение по умолчанию зависит от размера строки кэша процессора. Подробнее настройка хэш-таблиц обсуждается в отдельном [документе](#).

Синтаксис: **map_hash_max_size** *размер*;

Умолчание: map_hash_max_size 2048;

Контекст: http

Задаёт максимальный размер хэш-таблиц для переменных [map](#). Подробнее настройка хэш-таблиц обсуждается в отдельном [документе](#).