

]

Recent developments in nginx.conf scripting

Recent *developments* in **nginx.conf** scripting

☺agentzh@gmail.com☺

章亦春 (agentzh)

[2010.6](#)

♥ Background reading

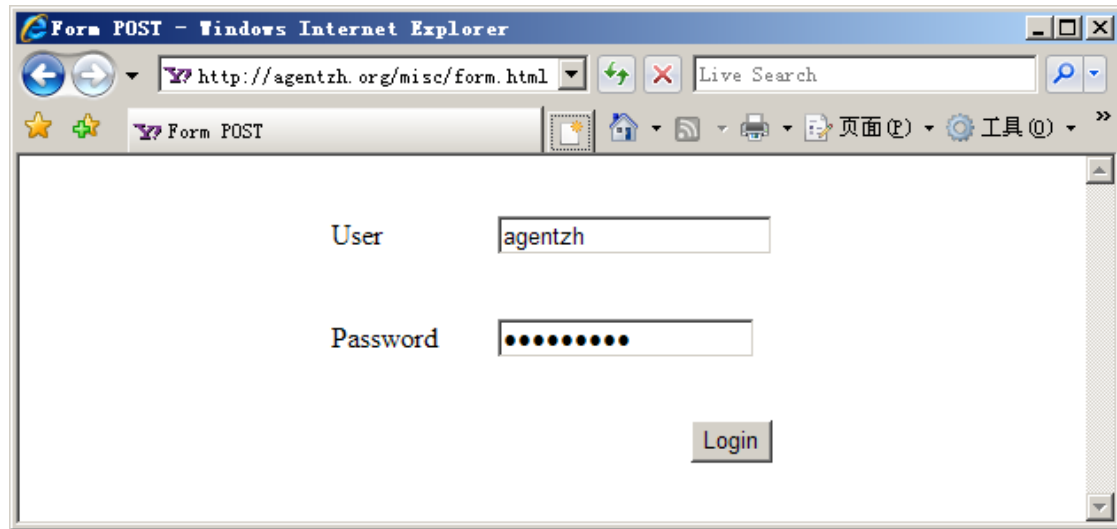
<http://agentzh.org/misc/slides/nginx-conf-scripting/>



♥ Handle *POST request bodies* painlessly
in nginx.conf by means of `ngx_form_input`!

Thanks our new teammate, Calio Zhi!

<http://github.com/calio/form-input-nginx-module>



```
<form method="post" action="/login">  
  <input name="name" type="text" />  
  <input name="pass" type="password" />  
  <input type="submit" value="Login" />  
</form>
```

```
location = /login {  
  set_form_input $user 'user';  
  set_form_input $password 'pass';  
  
  if ($user != 'agentzh') {  
    return 403;  
    break;  
  }  
  if ($password != '123456') {  
    return 403;  
    break;  
  }  
  echo "You're logged in!";  
}
```

♥ *ngx_postgres* has **already** landed.

Thanks Piotr Sikora!

http://github.com/FRiCKLE/ngx_postgres



configure the PostgreSQL upstream backend

upstream my_pg_backend {

postgres_server 10.62.136.3:5432 dbname=test

user=someone password=123456;

}

```
location /cats {  
  postgres_query 'select * from cats';  
  postgres_pass my_pg_backend;  
  rds_json on;  
}
```

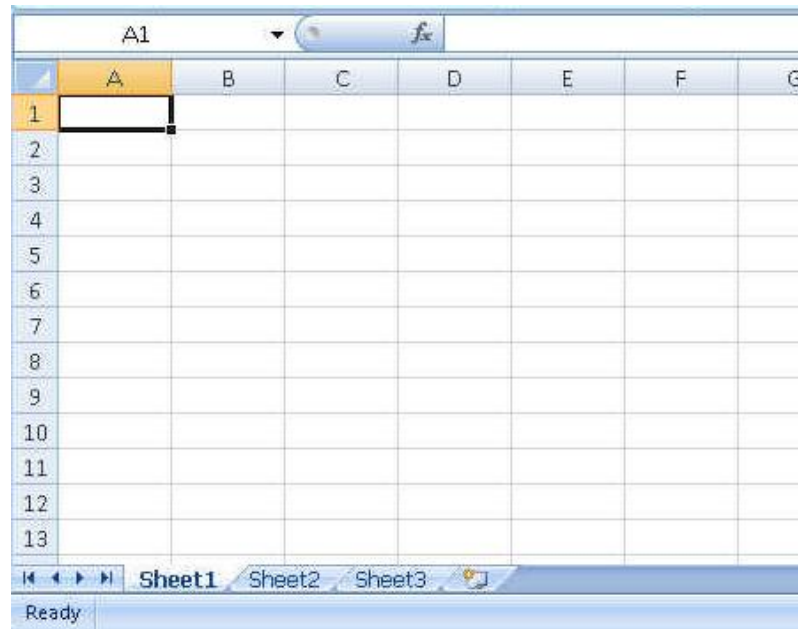
```
$ curl 'localhost/cats'
```

```
[{"name":"Marry","age":32},{"name":"Bob","age":12}]
```

♥ Everything is also *non-blocking* as ngx_drizzle.

Thanks to libpq's nonblocking API!

☺ Let's extract a specific *cell* in the result set!



How many cats are there?

location /count {

default_type 'text/plain';

postgres_query 'select count(*) from cats';

postgres_pass my_pg_backend;

extract the cell in the result set

in the first column, first row.

postgres_output value 0 0;

}

```
$ curl 'localhost/count'  
2
```

☺ or even *save* a cell in the result set
into an nginx **variable**


```
# read a value from Pg and save it into
# a cookie
location /foo {
    postgres_query "select signature
        from users where user='foo'";
    postgres_pass my_pg_backend;

    # extract the cell in the result set
    # in the first column, first row.
    postgres_set $signature 0 0;

    add_header Set-Cookie "sig=$signature; path=/";
}
```

♥ Construct fully *RESTful* queries
in a **single** location

```
location ~ '^/cat/(\d+)' {  
    set $id $1;  
    set_form_input $name;  
    set_quote_sql_str $quoted_name $name;  
  
    postgres_query GET "select * from cats where id=$id";  
    postgres_query DELETE "delete from cats where id=$id";  
    postgres_query POST "insert into cats (id, name) values($id, $quoted_name)";  
  
    postgres_pass my_pg_backend;  
}
```

♥ Qunar.com is running *ngx_postgres* + *ngx_rds_json*
in **production**.

Thanks Liseen Wan's promotion!

北京侏家酒店_北京侏家酒店预订/价格查询-去哪儿Qunar.com - Windows Internet Explorer

http://hotel.qunar.com/city/beijing_city/q-%E5%84%B2%E5%AE%B6#fromDate=2010-06-20&t...

文件(F) 编辑(E) 查看(V) 收藏夹(A) 工具(T) 帮助(H)

北京侏家酒店_北京侏家酒店预订/价格查询-去...

¥500以上
 中关村、上地
 [更多商圈](#)
 五星级/豪华
 新开业/装修
 会议室

93家酒店满足条件：评分从高到低 | 侏家 [清空所有条件](#)

Qunar城市百科：酒店品牌推荐>如家

【大众住宿领导者】国内排名第一的经济型连锁酒店，中国经济型酒店的先行者与领导者。08年第一家和颐酒店（上海漕宝路店）在上海开业，如家开始进军中高端商务酒店领域。作为上海的本土酒店品牌，如家依托上海世博会的辐射效应，在上海和长三角周边地区都取得了非常高的入住率。同时还推出了世博绿色通道，世博门票、住宿一站搞定。

酒店名/地址	用户点评	最低报价
1 如家快捷北京天坛店 连锁经济型 商旅推荐 位于崇文区天坛东路，地处南二环以里，毗邻享誉世界的中国古代建筑明珠—天坛公园。网友评价说“小吃也挺多，价格比较合理，装潢很好”。	4.5分/209条评论 在该城市排39位	¥159起 查看详情
2 如家快捷酒店北京西客站店 连锁经济型 商旅推荐 位于北京西站北广场，毗邻军事博物馆、中央电视塔、玉渊潭公园、北京茶叶一条街、白云观、中华世纪坛等旅游景点。网友评价说“尤其是环境卫生，周边小吃多”。	4.5分/44条评论 在该城市排39位	¥261起 查看详情
3 如家快捷酒店北京景泰桥珐琅厂店 连锁经济型	4.5分/113条评论	¥169起

大图模式 随屏幕浮动 随地图缩放

POWERED BY Google 地图数据 ©2010 M...

http://hotel.qunar.com/city/beijing_city/dt-3828/#fromDate=2010-06-20&toDate=2010-06-23

Internet 100%



城市: 北京 关键词: rujia 日期: 从 2010-06-20 到 后天 搜索



排序方式: 类别 (综合, 评分), 价格 (低-高, 高-低), 档次 (低-高, 高-低)

价格: ¥200以下, ¥200-¥300, ¥300-¥500, ¥500以上

商圈: 全部, 燕莎国贸, 天安门王府井, 西直门, 北京南站商圈, 崇文门商圈, 中关村、上地, 更多商圈

品牌: 餐厅, 商务中心, 健身房, 会议室, 如家

93家酒店满足条件: 评分从高到低 | 儒家 清空所有条件

Qunar城市百科: 酒店品牌推荐>如家

♡ *Caching* database responses using **memcached**
via ngx_srcache and ngx_memc.

<http://github.com/agentzh/srcache-nginx-module>

enable the ngx_srcache and other

modules in your nginx build

**\$./configure --prefix=/opt/nginx **

**--add-module=/path/to/srcache-nginx-module **

**--add-module=/path/to/rds-json-nginx-module **

**--add-module=/path/to/memc-nginx-module **

--add-module=/path/to/drizzle-nginx-module

♥ It's very important to put `ngx_srcache` *before* `ngx_rds_json` during nginx configure so that we cache the final JSON rather than RDS.

configure the mysql upstream backend

upstream mysql_backend {

drizzle_server 127.0.0.1:3306 dbname=test

password=some_pass user=monty protocol=mysql;

}

configure the cache storage location

location /memc {

internal;

set \$memc_key \$query_string;

set \$memc_exptime 300;

memc_pass 127.0.0.1:11211;

}

```
location /cats {  
    srcache_fetch GET /memc $uri;  
    srcache_store PUT /memc $uri;  
  
    default_type application/json;  
  
    drizzle_pass mysql_backend;  
    drizzle_query 'select * from cats';  
  
    rds_json on;  
}
```

```
$ curl 'localhost/cats'
```

```
[{"name":"Marry","age":32},{"name":"Bob","age":12}]
```

if it is a cache miss

\$ memcached -vvv -p 11211

...

<10 new client connection

<10 get /cats

> NOT FOUND /cats

>10 END

<10 connection closed.

<10 new client connection

<10 set /cats 0 300 44

> NOT FOUND /cats

>10 STORED

<10 connection closed.

if it is a cache hit

\$ memcached -vvv -p 11211

...

<10 new client connection

<10 get /cats

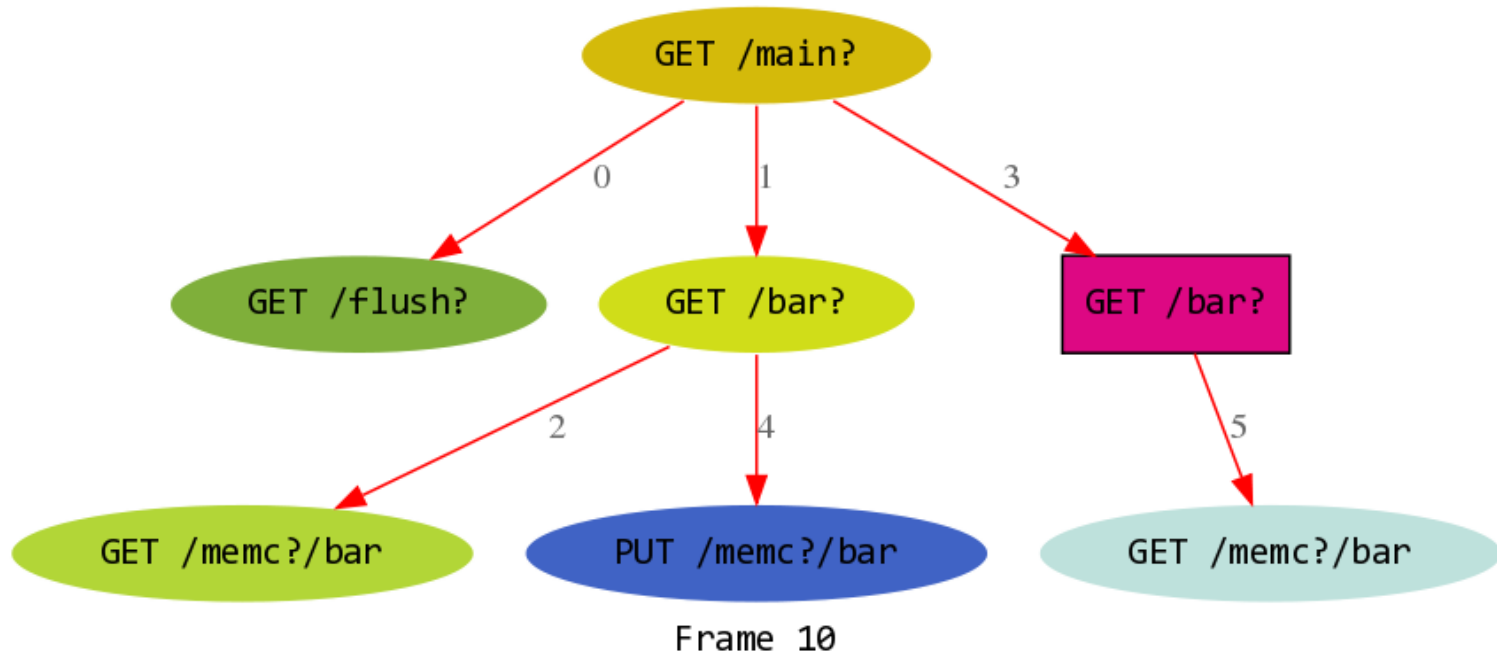
> FOUND KEY /cats

>10 sending key /cats

>10 END

<10 connection closed.

♥ You can even cache *subrequests*' responses
out of the box!



```
location = /main {  
    echo -n "[\n "  
    echo_location /cats dir=asc;  
    echo -n ",\n "  
    echo_location /cats dir=desc;  
    echo "\n]";  
}
```

```
location = /cats {  
  internal;  
  set $key "$uri-$arg_dir";  
  srcache_fetch GET /memc $key;  
  srcache_store PUT /memc $key;  
  
  default_type application/json;  
  drizzle_query 'select * from cats order by id $arg_dir';  
  drizzle_pass backend;  
  rds_json on;  
}
```

```
$ curl 'localhost/main'
```

```
[
```

```
 [{"id":2,"name":null}, {"id":3,"name":"bob"}],
```

```
 [{"id":3,"name":"bob"}, {"id":2,"name":null}]
```

```
]
```

♥ Theoretically speaking, you can cache *arbitrary* nginx locations using ngx_srcache!

```
location /blah {  
    set $key "$uri-$arg_foo-$arg_bar";  
  
    srcache_fetch GET /memc $key;  
    srcache_store PUT /memc $key;  
  
    # proxy_pass/fastcgi_pass/postgres_pass/echo/...  
}
```

♥ *Access control* using `ngx_encrypted_session`

<http://github.com/agentzh/encrypted-session-nginx-module>

private key used in AES (with MAC) encryption/decryption:
encrypted_session_key "abcdefghijklmnopqrstuvwxyz123456";

init vector setting:
encrypted_session_iv "12345678";

expiration time for generated sessions
(in seconds by default):
encrypted_session_expires 1d; # here we specify 1 day


```
location = /login {
    eval_subrequest_in_memory off;
    eval_override_content_type text/plain;
    eval $uid {
        postgres_query "select id
            from users
            where name=$arg_name and pass=$arg_pass";
        postgres_pass pg_backend;
        postgres_output value 0 0;
    }
    if ($uid !~ '^\d+$') {
        rewrite ^ /relogin redirect; break;
    }
    set_encrypt_session $session $uid;
    set_encode_base32 $session;
    add_header Set-Cookie "session=$session; path=/";
    rewrite ^ /main redirect; break;
}
```

```
location = /some_api {  
    include conf/access.conf;
```

```
    # your normal content handler settings like
```

```
    # fastcgi_pass/postgres_pass/proxy_pass
```

```
}
```

```
# conf/access.conf
if ($arg_session = "") {
    rds_json_ret 100 "Login required"; break;
}
```

```
set_unescape_uri $session $arg_session;
```

```
set_decode_base32 $session;
if ($session = "") {
    rds_json_ret 100 "Login required"; break;
}
```

```
set_decode_lz_session $uid $session;
if ($uid !~ '^\\d+$') {
    rds_json_ret 100 "Login required"; break;
}
```

♥ ngx_lua is quite usable *now*!

<http://github.com/chaoslawful/lua-nginx-module>



♥ chaoslawful is *crazy!*



first install lua (or even luajit) into your system...

enable the ngx_lua module in your nginx build

```
$ ./configure --prefix=/opt/nginx \  
    --add-module=/path/to/nginx_devel_kit \  
    --add-module=/path/to/echo-nginx-module \  
    --add-module=/path/to/lua-nginx-module
```

```
location = /adder {  
    set_by_lua $res  
        "local a = tonumber(ngx.arg[1])  
        local b = tonumber(ngx.arg[2])  
        return a + b" $arg_a $arg_b;  
  
    echo $res;  
}
```

```
$ curl 'localhost/adder?a=25&b=75'  
100
```



```
location = /fib {  
  set_by_lua $res "  
    function fib(n)  
      if n > 2 then  
        return fib(n-1) + fib(n-2)  
      else  
        return 1  
      end  
    end  
    local num = tonumber(ngx.arg[1])  
    return fib(num)  
  " $arg_n;  
  
  echo $res;  
}
```

```
$ curl 'localhost/fib?n=10'
```

```
55
```

♥ or use *external* Lua script file...

```
location = /fib {  
    set_by_lua_file $res "conf/fib.lua" $arg_n;  
  
    echo $res;  
}
```

-- conf/fib.lua file

function fib(n)

if n > 2 then

return fib(n-1) + fib(n-2)

else

return 1

end

end

local num = tonumber(ngx.arg[1])

return fib(num)

\$ *Complex* database cluster hashing
can also be done in Lua

```
http {  
    upstream A {  
        drizzle_server ...;  
    }  
    upstream B {  
        drizzle_server ...;  
    }  
    upstream C {  
        drizzle_server ...;  
    }  
    ...  
}
```

```
location ~ '^/user/(\d+)' {  
    set $uid $1;  
    set_by_lua_file $backend "conf/hash.lua" $uid;  
    if ($backend = "") {  
        return 400; break;  
    }  
  
    drizzle_query "select * from users  
        where uid=$uid";  
    drizzle_pass $backend;  
    rds_json on;  
}
```


-- hash.lua

function hash(uid)

if uid > 0 and uid <= 1200 then return 'A' end

if uid > 1200 and uid <= 5300 then return 'B' end

if uid > 5300 and uid <= 7100 then return 'C' end

return ""

end

return hash(tonumber(ngx.arg[1]))

♥ Use **Lua** to code up nginx *content handler* directly

```
location = /lua {  
    content_by_lua 'ngx.say("Hello, Lua!");  
}
```

```
$ curl 'localhost/lua'
```

```
Hello, Lua!
```

♡ ...and we can read arbitrary *nginx variables* from within our Lua **content** handler!

```
location = /hello {  
  content_by_lua 'local who = ngx.var.arg_who  
    ngx.say("Hello, ", who, "!");  
}
```

```
$ curl 'localhost/hello?who=agentzh'
```

```
Hello, agentzh!
```

♥ We can also put Lua code into *external* .lua **file** to eliminate escaping nightmare.


```
location /foo {
```

```
...
```

```
content_by_lua_file /path/to/your/lua-file.lua;
```

```
}
```

♥ We can also issue nginx *subrequests* directly from within Lua content handler *now!*

```
location /other {  
    echo "hello, world";  
}
```

transparent non-blocking I/O in Lua

```
location /lua {  
    content_by_lua '  
        local res = ngx.location.capture("/other")  
        if res.status == 200 then  
            ngx.print(res.body)  
        end';  
}
```

```
$ curl 'localhost/lua'
```

```
hello, world
```

♥ We'd call this whole set of nginx modules *ngx_openresty*
and our work is heavily **funded** by *Taobao.com*.

淘宝网

♥ It is already **powering** lz.taobao.com.



淘宝量子店铺统计V3.0 - Windows Internet Explorer

http://lz.taobao.com/ Live Search

文件(F) 编辑(E) 查看(V) 收藏夹(A) 工具(T) 帮助(H)

★ 淘宝量子店铺统计V3.0

论坛 帮助 ★收藏量子

量子统计 测试版

lz.taobao.com

首页 产品介绍 最新动态 客户展示



了解并提高流量、转化率，直接提高销售额

了解上万广告费，到底效果如何

探索流行趋势，推广人气宝贝

对您的访客了若指掌，看看他们到底在干嘛？

.....

活动介绍

量子统计登录

用户名:

密码: [忘记密码?](#)

登录

使用淘宝账号登录，即可免费试用。

使用心得

- 量子店铺统计宝典
- 挖掘报表数据，指导经营
- 2个月上4星，量子是最大的功臣
- 量子店铺统计功能，增强我的信心

我们一直在进步

- 淘宝量子店铺统计V3.0于5月17日全新升级，诚邀免费体验!
- 量子店铺统计3月16日火爆升级!
- 量子服务器扩容
- 详细来源功能升级了!

我们的客户

已经有**416,950**人使用过量子统计...

 柠檬绿茶	 妖精 emma 5ikoudai.com	 易酷生活	 朵朵云
---	--	---	--

关于淘宝 广告服务 合作伙伴 帮助中心 诚征英才 联系我们 网站地图 热门品牌 版权说明 加入商城 淘宝客 买卖安全

Internet 100%

您好, ! 退出

淘宝网首页 | 我的淘宝 | 收藏量子 | 搜索 | 网站导航 | 帮助



最新公告: 淘宝量子店铺统计V3.0测试版震撼登场, 诚邀免费试用!

跟淘宝说几句 去论坛提问

店铺基本

店铺概况

- 实时客户访问
- 按小时流量分析
- 按天流量分析
- 宝贝被访排行
- 分类页被访排行
- 店内搜索关键词

营销效果

- 流量来源构成
- 淘宝搜索关键词

客户分析

- 访客地区分析

百宝箱

- 个性化统计图标

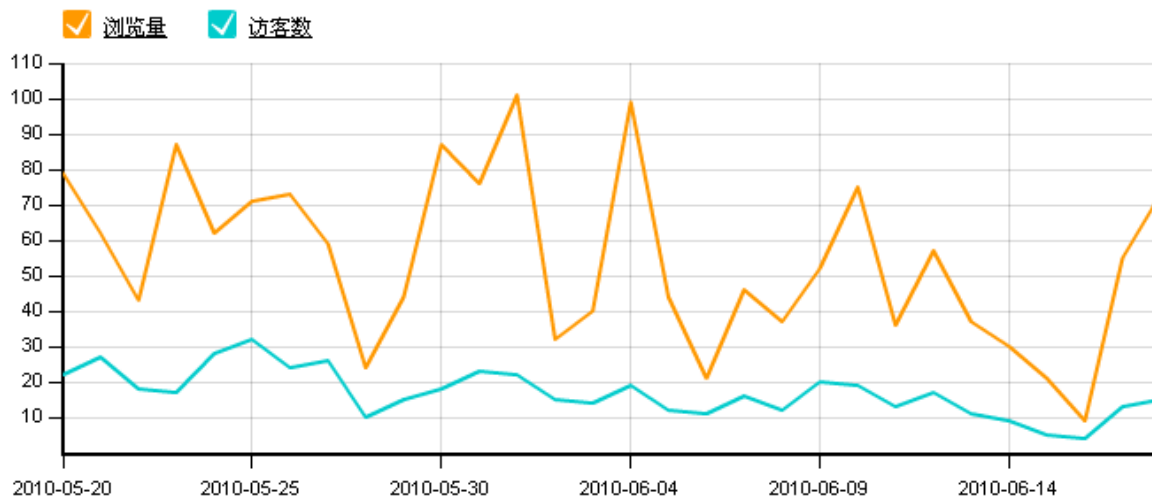
+ 添加功能模块

管理功能模块

流量概况

按天 按小时

详细信息 >>



数据汇总

浏览量 (PV)

访客数 (UV)

人均访问页面数

人均店内停留时间 (秒)

浏览回头率

完成



Internet

100%



最新公告: 淘宝量子店铺统计V3.0测试版震撼登场, 诚邀免费试用! [跟淘宝说几句](#) [去论坛提问](#)

- 店铺基本
 - 店铺概况
 - 实时客户访问
 - 按小时流量分析
 - 按天流量分析
 - 宝贝被访排行**
 - 分类页被访排行
 - 店内搜索关键词
 - 营销效果
 - 流量来源构成
 - 淘宝搜索关键词
 - 客户分析
 - 访客地区分析
 - 百宝箱
 - 个性化统计图标
- + 添加功能模块
- 管理功能模块

宝贝被访排行 宝贝被访详情

选择时间段: 2010-06-18 到 2010-06-18 查看 快速选择: 今天 昨天 前天 最近7天 最近30天 本周 上周 本月 上月

宝贝数据汇总

全店宝贝总浏览量: 2	平均每个宝贝查看时间: 15秒	平均每个宝贝查看人次: 1
全店宝贝查看总人次: 2	平均每个宝贝入店人次: 1	平均每个宝贝出店人次: 1
全店宝贝平均跳失率: -		

宝贝访客数排行TOP10

查看类型: 访客数 浏览量 关注度

- 正品邦贝宜/贝贝怡纯棉/婴儿服/宝宝衣服绑带长袖... 50%
- 邦贝宜100纯棉环保婴儿内衣四色长袖连体爬服哈衣... 50%



淘宝量子店铺统计V3.0 - 流量来源构成 - Windows Internet Explorer

http://lz.taobao.com/#srccomp/1/1

文件(F) 编辑(E) 查看(V) 收藏夹(A) 工具(T) 帮助(H)

淘宝量子店铺统计V3.0 - 流量来源构成

您好, [退出](#) [淘宝网首页](#) | [我的淘宝](#) | [收藏量子](#) | [搜索](#) | [网站导航](#) | [帮助](#)

量子统计 测试版
lz.taobao.com

最新公告: 淘宝量子店铺统计V3.0测试版震撼登场, 诚邀免费试用! [跟淘宝说几句](#) [去论坛提问](#)

选择时间段: 到 [查看](#) 快速选择: [今天](#) [昨天](#) [前天](#) [最近7天](#) [最近30天](#) [本周](#) [上周](#) [本月](#) [上月](#)

今天来源构成报表 [下载](#) [打印](#)

来源	详细	到达页浏览量	百分比	趋势
搜索引擎	114	1	1.89%	查看
	合计	1	1.89%	
淘宝站内	淘宝站内其他	18	33.96%	查看
	淘宝店铺搜索	6	11.32%	查看
	淘宝其他店铺	6	11.32%	查看
	淘宝信用评价	3	5.66%	查看
	淘宝搜索	2	3.77%	查看
	淘宝收藏	1	1.89%	查看
	淘宝管理后台	1	1.89%	查看
	合计	37	69.81%	
直接访问	合计	10	18.87%	查看
其他	dr02.taobao.fenxi001.com	2	3.77%	查看
	club.baby.sina.com.cn	1	1.89%	查看
	sheetzoho.com	1	1.89%	查看
	blog.bbcare.cn	1	1.89%	查看
	合计	5	9.43%	

店铺基本

- 店铺概况
- 实时客户访问
- 按小时流量分析
- 按天流量分析
- 宝贝被访排行
- 分类页被访排行
- 店内搜索关键词

营销效果

- 流量来源构成**
- 淘宝搜索关键词

客户分析

- 访客地区分析

百宝箱

- 个性化统计图标

[+ 添加功能模块](#)

[管理功能模块](#)

Internet 100%

♥ *Join* us at the **OpenResty** Google Group

<http://groups.google.com/group/OpenResty>

and the **nginx-devel** mailing list

<http://nginx.org/mailman/listinfo/nginx-devel>

♥ or just *catch* us on IRC:

irc.freenode.net **#nginx #openresty**

😊 *Any questions?* 😊

