Atitit 常用微服务实现

目录

[1. 健康检测 max\_fails与fail\_timeout参 数 1](#_Toc13005)

[1.1. spring cloud dubbo实现 2](#_Toc14089)

[1.2. 、nginx多个tomcat负载均衡的实例 3](#_Toc17272)

[2. Rmi实现 4](#_Toc23924)

[3. ref 4](#_Toc29081)

# 健康检测 ****max\_fails****与****fail\_timeout****参 数

1. 弄清楚HTTP Upstream 模块中Server指令的**max\_fails**与**fail\_timeout**参 数的关系、

提出server指令后面的参数部分，以下摘抄nginx wiki 内容

语法：**server** **name** [**parameters**]

**parameters**包 含：

·**weight** = NUMBER - 设置服务器权重，默认为1。

·**max\_fails** = NUMBER - 在一定时间内（这个时间在fail\_timeout参数中设置）检查这个服务器是否可用时产生的最多失败请求数，默认为1，将其设置为0可以关闭检查，这 些错误在proxy\_next\_upstream或fastcgi\_next\_upstream（404错误不会使max\_fails增加）中定义。

·**fail\_timeout** = TIME - 在这个时间内产生了max\_fails所设置大小的失败尝试连接请求后这个服务器可能不可用，同样它指定了服务器不可用的时间（在下一次尝试连接请求发起 之前），默认为10秒，fail\_timeout与前端响应时间没有直接关系，不过可以使用proxy\_connect\_timeout和 proxy\_read\_timeout来控制。

·**down** - 标记服务器处于离线状态，通常和ip\_hash一起使用。

**关于max\_fails 参数的理解：根 据上面的解释，max\_fails默认为1，fail\_timeout默 认为10秒，也就是说，默认情况下后端服务器在10秒钟之内可以容许有一次的失 败，如果超过1次则视为该服务器有问题，将该服务器标记为不可用。等待10秒后再 将请求发给该服务器，以此类推进行后端服务器的健康检查。但如果我将max\_fails设置为0， 则代表不对后端服务器进行健康检查，这样一来fail\_timeout参数也就没什么意义了。那若后端服务器真的出现 问题怎么办呢？上文也说了，可以借助proxy\_connect\_timeout和proxy\_read\_timeout进 行控制。**

下面介绍http proxy模块中的相关指令：

**proxy\_next\_upstream**  
语法： proxy\_next\_upstream [error|timeout|invalid\_header|http\_500|http\_502|http\_503|http\_504|http\_404|off]  
确定在何种情况下请求将转发到下一个服务器。转发请求只发生在没有数据传递到客户端的过程中。

**proxy\_connect\_timeout**  
后端服务器连接的超时时间\_发起握手等候响应超时时间

**proxy\_read\_timeout**  
连接成功后\_等候后端服务器响应时间\_其实已经进入后端的排队之中等候处理（也可以说是后端服务器处理请求的时间）

**proxy\_send\_timeout**  
后端服务器数据回传时间\_就是在规定时间之内后端服务器必须传完所有的数据

**proxy\_pass**  
这个指令设置被代理服务器的地址和被映射的URI

**结论：**

**（1）**proxy\_next\_upstream参数很有用，他可以避免很多 错误  
**（2）**max\_fails 参数在繁忙的大型系统中建议设置为3，如果没有几个后端服务器的话保持默认即可。  
**（3）**proxy\_read\_timeout要根据自身程序而定，不要过大，也不 要太小。如果是php程序，请参照php.ini中的max\_execution\_time选项值。

## spring cloud dubbo实现

spring cloud 为开发人员提供了快速构建分布式系统的一些工具，包括配置管理、服务发现、断路器、路由、微代理、事件总线、全局锁、决策竞选、分布式会话等等

\*eureka是一个高可用的组件，它没有后端缓存，每一个实例注册之后需要向注册中心发送心跳（因此可以在内存中完成）

当client向server注册时，它会提供一些元数据，例如主机和端口，URL，主页等。Eureka server 从每个client实例接收心跳消息。 如果心跳超时，则通常将该实例从注册server中删除。

## 、nginx多个tomcat负载均衡的实例

这里假定有3台Tomcat服务器，分别开放不同的端口，地址如下：

[?](http://www.yuyublog.com/1/yuyublog/article/94.html)

|  |  |
| --- | --- |
| 1  2  3 | 192.168.12.131:8000  192.168.12.132:8080  192.168.12.133:8090 |

Nginx的相关配置代码如下：

[?](http://www.yuyublog.com/1/yuyublog/article/94.html)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | upstream mytomcats {        server 192.168.12.131:8000;        server 192.168.12.132:8080;        server 192.168.12.133:8090;  }    server {        listen 80;        server\_name www.ixdba.net;    location ~\* \.(jpg|gif|png|swf|flv|wma|wmv|asf|mp3|mmf|zip|rar)$ {         root /web/www/html/;  }    location / {            proxy\_pass http://mytomcats;            proxy\_redirect off;            proxy\_set\_header Host $host;            proxy\_set\_header X-Real-IP $remote\_addr;            proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;            client\_max\_body\_size 10m;            client\_body\_buffer\_size 128k;            proxy\_connect\_timeout 90;            proxy\_send\_timeout 90;            proxy\_read\_timeout 90;            proxy\_buffer\_size 4k;            proxy\_buffers 4 32k;            proxy\_busy\_buffers\_size 64k;            proxy\_temp\_file\_write\_size 64k; |

# Rmi实现

注册，，客户端启动点时候，指明服务端注册中心与要注册的ip port

访问的时候检测健康。。如果fail了，，则从redis移出错误client

# ref

Nginx之负载均衡 - jimisun - 博客园.html

Nginx HTTP负载均衡\_反向代理的相关参数测试 - 小炒花生米 - 博客园.html