

**KBEngine**

**WebConsole Guide**

目录

[环境配置 3](#_Toc469995937)

[1. 运行环境 3](#_Toc469995938)

[2. 使用方法 3](#_Toc469995939)

[运行步骤 3](#_Toc469995940)

[1. 启动服务器 3](#_Toc469995941)

[2. 登录系统 4](#_Toc469995942)

[功能板块说明 5](#_Toc469995943)

[1. 用户管理界面 5](#_Toc469995944)

[2. 集群管理 7](#_Toc469995945)

[3. 性能分析 9](#_Toc469995946)

[4. 日志查看 12](#_Toc469995947)

[5. 状态图 13](#_Toc469995948)

[6. Python控制台 14](#_Toc469995949)

[7. Watcher 15](#_Toc469995950)

[8. SpaceViewer 16](#_Toc469995951)

## 环境配置

### 运行环境

Python版本：Python3.X (建议3.3或3.3以上版本）

Django版本：1.8.9

### 使用方法

* + 1. 请自行为你所使用的python安装与之匹配的django模块，如果不想安装django，有另外两个选择:
* 如果使用的是python2.6.6，可进入“kbe/tools/server/django\_packages”目录，并在该目录下解压Django-1.6.11.tar.gz文件（解压到当前目录）；
* 如果使用的是python2.7或以上，可进入“kbe/tools/server/django\_packages”目录，并在该目录下解压Django-1.8.9.tar.gz文件（解压到当前目录）。
  + 1. 修改sync\_db.bat、sync\_db.sh以及run\_server.bat、run\_server.sh，使其正确指向你所使用的python的路径，第一次使用，需要初始化数据：
* windows下python3.3 + django 1.8.9，运行“sync\_db.bat”命令；
* linux下python3.3 + django 1.8.9，运行“sync\_db.sh”命令；
* linux下python2.6 + django 1.6.11，运行“sync\_db\_dj-1.6.sh”命令；
* windows下python2.6 + django 1.6.11，请参考“sync\_db\_dj-1.6.sh”自行创建.bat文件。

## 运行步骤

### 启动服务器

* Linux：执行run\_server.sh脚本，或者部署到nginx上，浏览器打开地址 “http://xxx.xxx.xxx.xxx:8000/wc/”进行访问，其中“xxx.xxx.xxx.xxx”为运行webconsole的机器IP；
* Windows：运行run\_server.bat(需将run\_server.bat中的python文件夹改为自己的python文件夹)，之后在浏览器打开地址 “http://xxx.xxx.xxx.xxx:8000/wc/”进行访问，其中“xxx.xxx.xxx.xxx”为运行webconsole的机器IP。

### 登录系统

初次登陆需用默认用户名密码登入用户管理界面，并新建管理用户。

注意事项：

* 第一次使用此控制台时，默认的登录账号为“Admin”，默认密码为“123456”，此账号也是后台唯一的管理账号，登录后请及时修改密码；
* 第一次使用Admin进入后台，需要根据自己启动服务器的用户账号名和用户uid创建新的控制用户，创建完成后，需要退出Admin使用新用户登录才能进行实际的后台操作；
* 此后台的开发环境为python3.3 + django 1.8.9，在linux下使用python2.6.6 + django-1.6.11测试通过；
* 此工具所有的操作都源于Machine，因此，想要使用这个工具的功能，必须确保machine进程正确运行；
* 有任何使用问题，请在KBEngine官方平台上提出。



2-1-1登录界面

## 功能板块说明

### 用户管理界面

#### 账号管理

在这个页面可对使用webconsole的用户进行管理，也可对管理员账户进行管理。



3-1-1-1账号管理界面

#### 新建管理用户

* 账号：登陆用的账号；
* 昵称：登陆后显示的名称；
* 登录密码：任意字母数字组合；
* 确认密码：再输入一遍密码；
* 操作系统用户：Linux系统的用户名称，Windows下开发调试请无视；
* 操作系统用户uid：Linux系统用户uid，Windows下开发需与被管理的KBE的uid相同，否则无法管理该KBE；
* KBE\_ROOT：被管理的KBE的KBE\_ROOT目录，默认为当前webconsole所在的KBE的KBE\_ROOT路径(可为空)；
* KBE\_RES\_PATH：被管理的KBE的KBE\_RES\_PATH目录，默认为当前webconsole所在的KBE的KBE\_RES\_PATH路径(可为空)；
* KBE\_BIN\_PATH：被管理的KBE的KBE\_BIN\_PATH目录，默认为当前webconsole所在的KBE的KBE\_BIN\_PATH路径(可为空)。

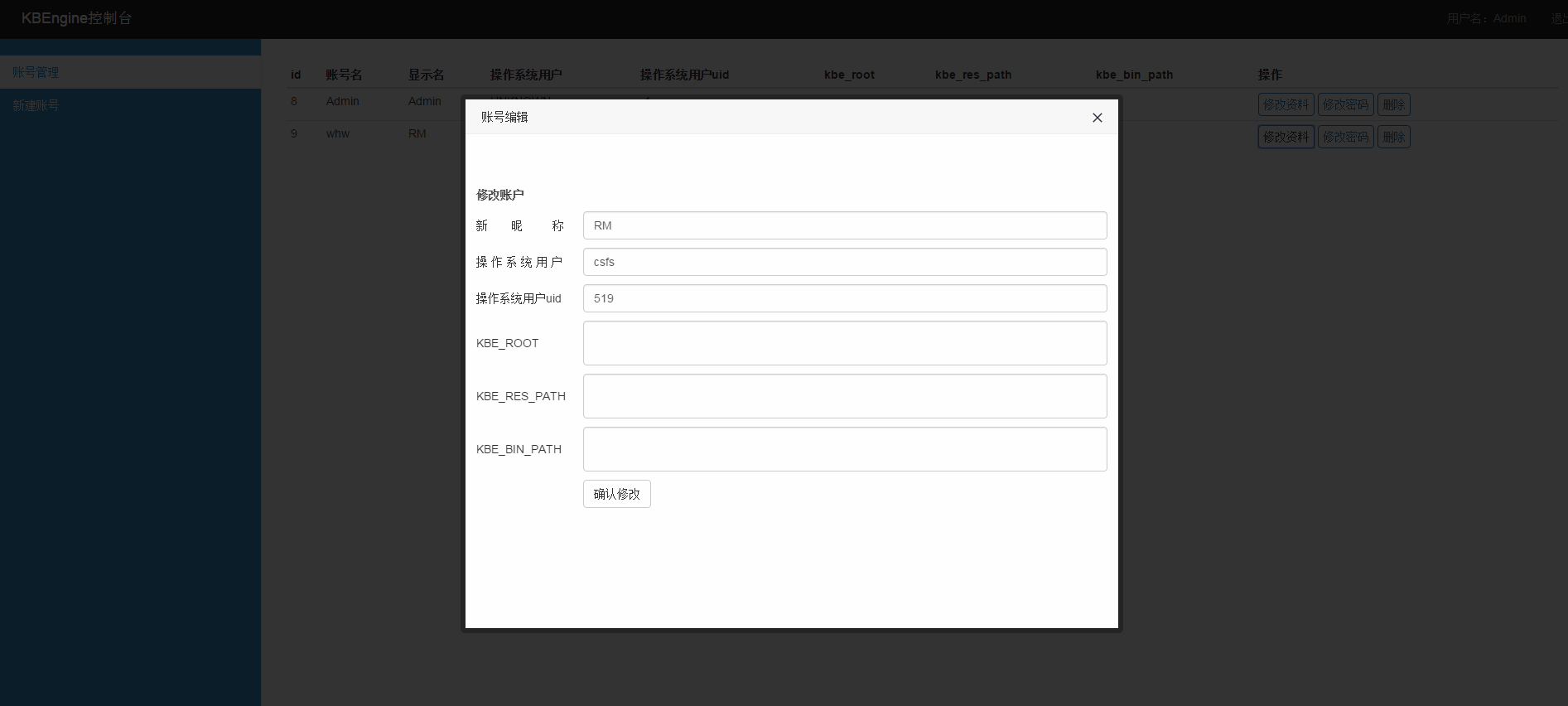
注：因一台服务器上可同时存在多个KBE，故每个被管理的KBE的系统用户、UID、KBE\_ROOT、KBE\_RES\_PATH、KBE\_BIN\_PATH都不能与其他用户的参数重复。



3-1-2-1新建账号

#### 修改资料

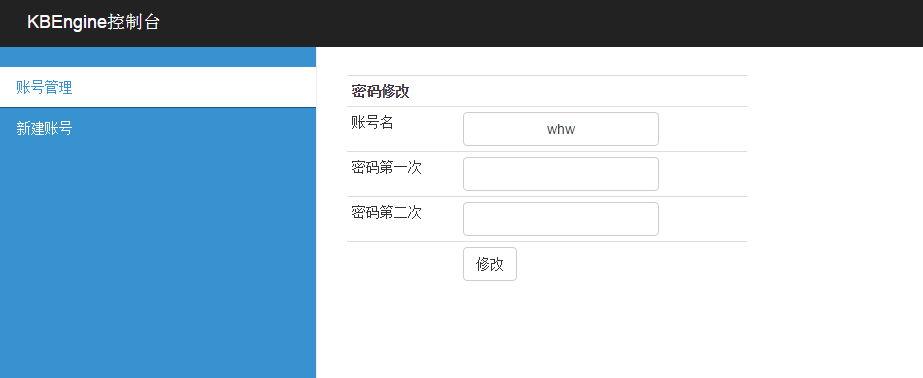
输入新的用户参数，点击确认修改即可。



3-1-3-1账号编辑

#### 修改密码

输入新的密码两次并点击修改。



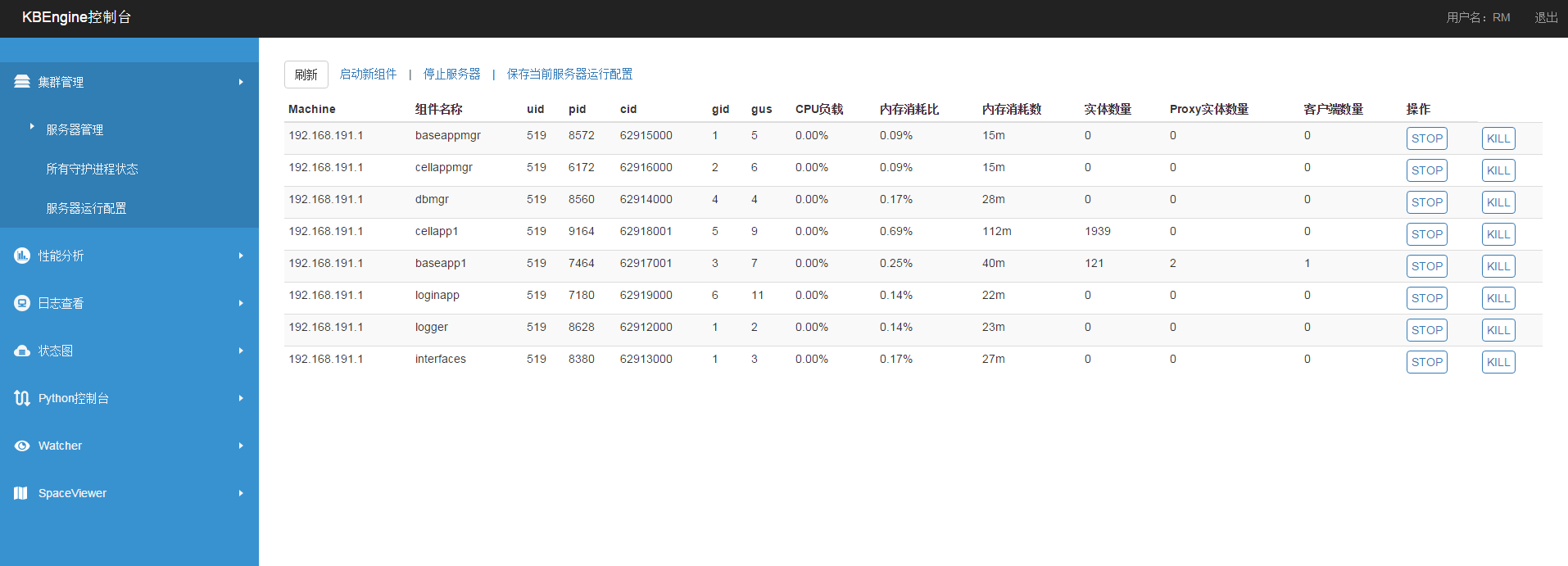
3-1-4-1密码修改

### 集群管理

#### 服务器管理

在“服务器管理”页面，你可以对已启动的KBE进程进行管理和资源消耗查看。

* STOP操作：停止当前进程；
* KILL操作：杀死当前进程；
* 启动新组件：启动新的KBE服务组件；
* 停止服务器：停止当前KBE服务器所有进程；
* 保存当前服务器运行配置：保存当前KBE服务器进程的数量配置信息。



3-2-1-1服务器管理界面

#### 启动新组件

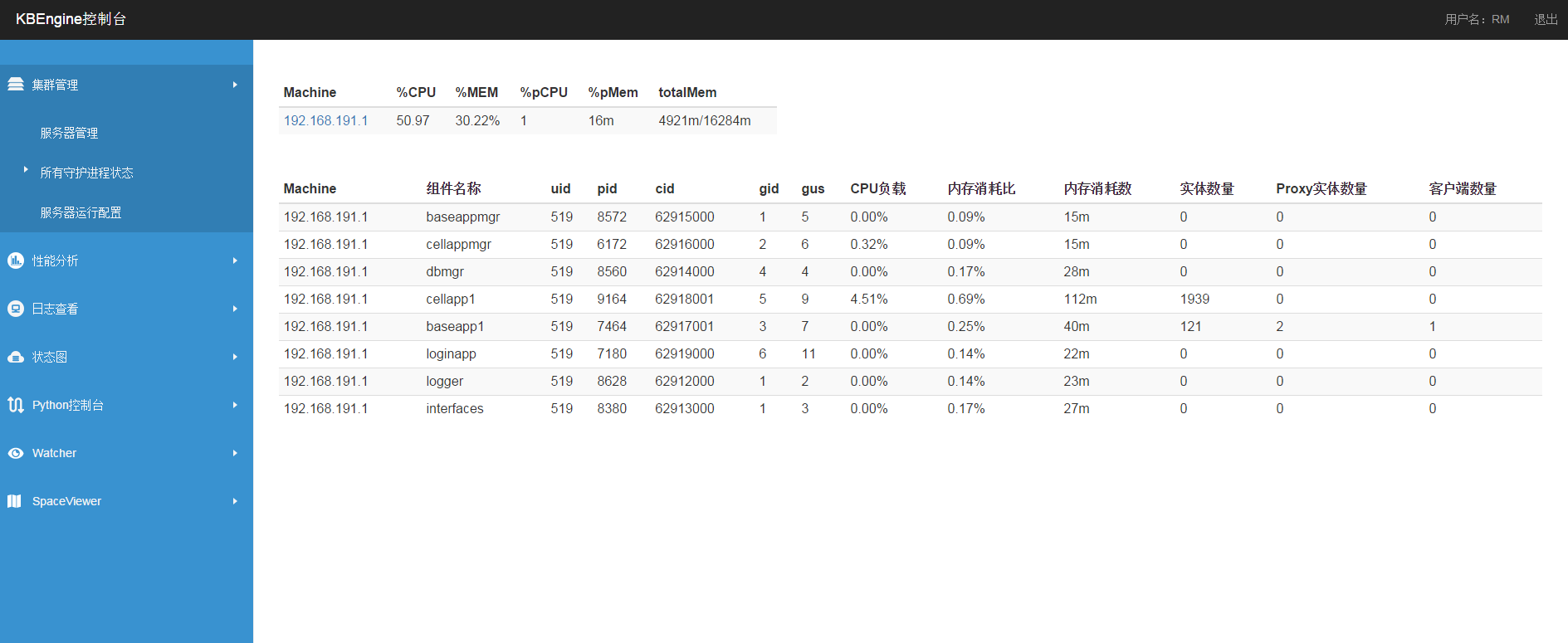
在该页面，你可以在服务器群内新建任意数量的组件进程。



3-2-2-1启动新组件界面

#### 所有守护进程状态

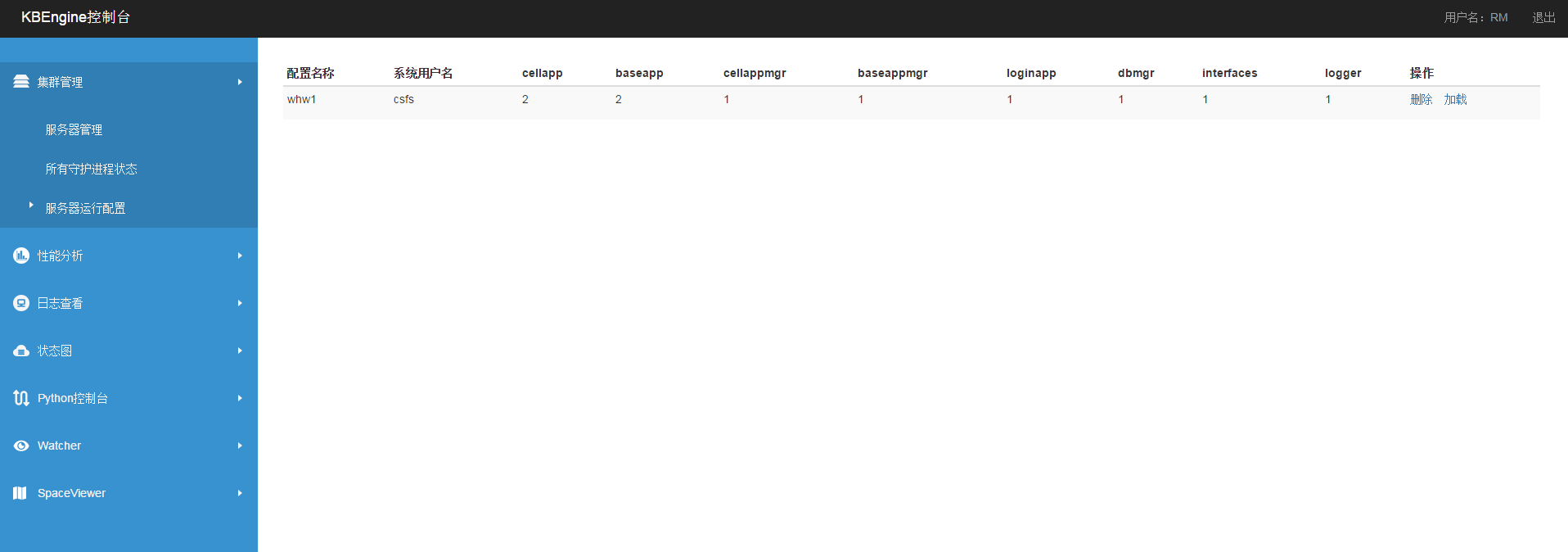
在当前界面，你可以查看当前服务器群内所有KBE的Machine信息和资源占用情况。



3-2-3-1所有守护进程界面

#### 服务器运行配置

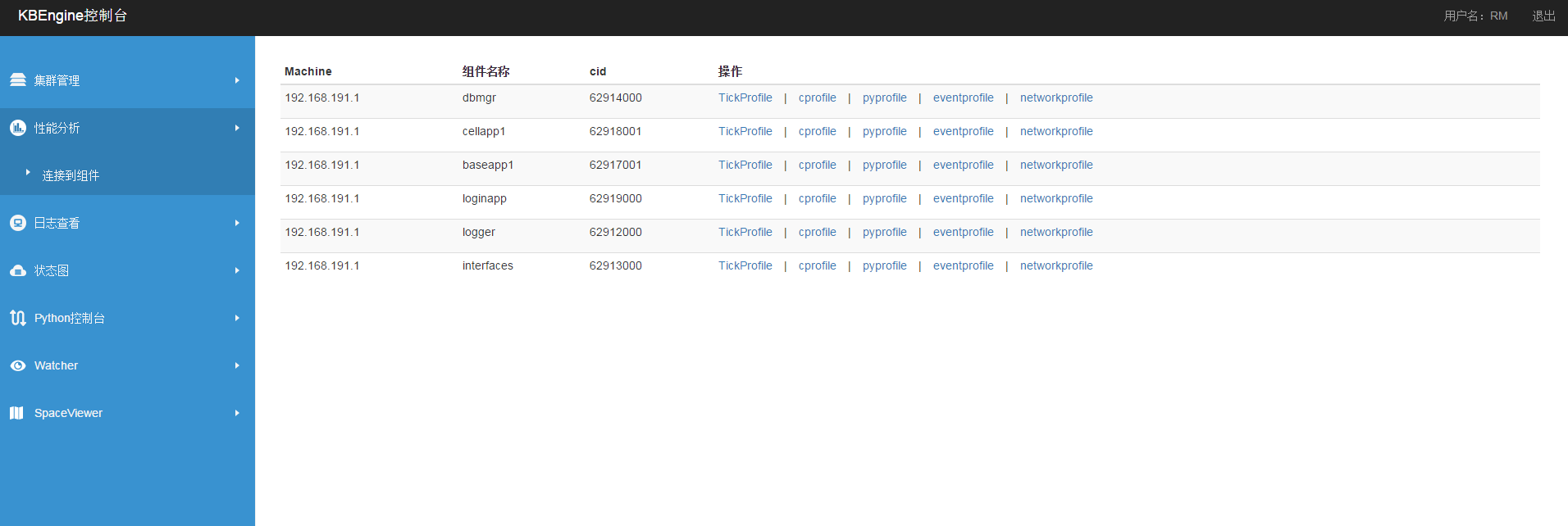
在这个页面，你可以对之前保存的服务器配置进行加载和删除操作。



3-2-4-1服务器运行配置

### 性能分析

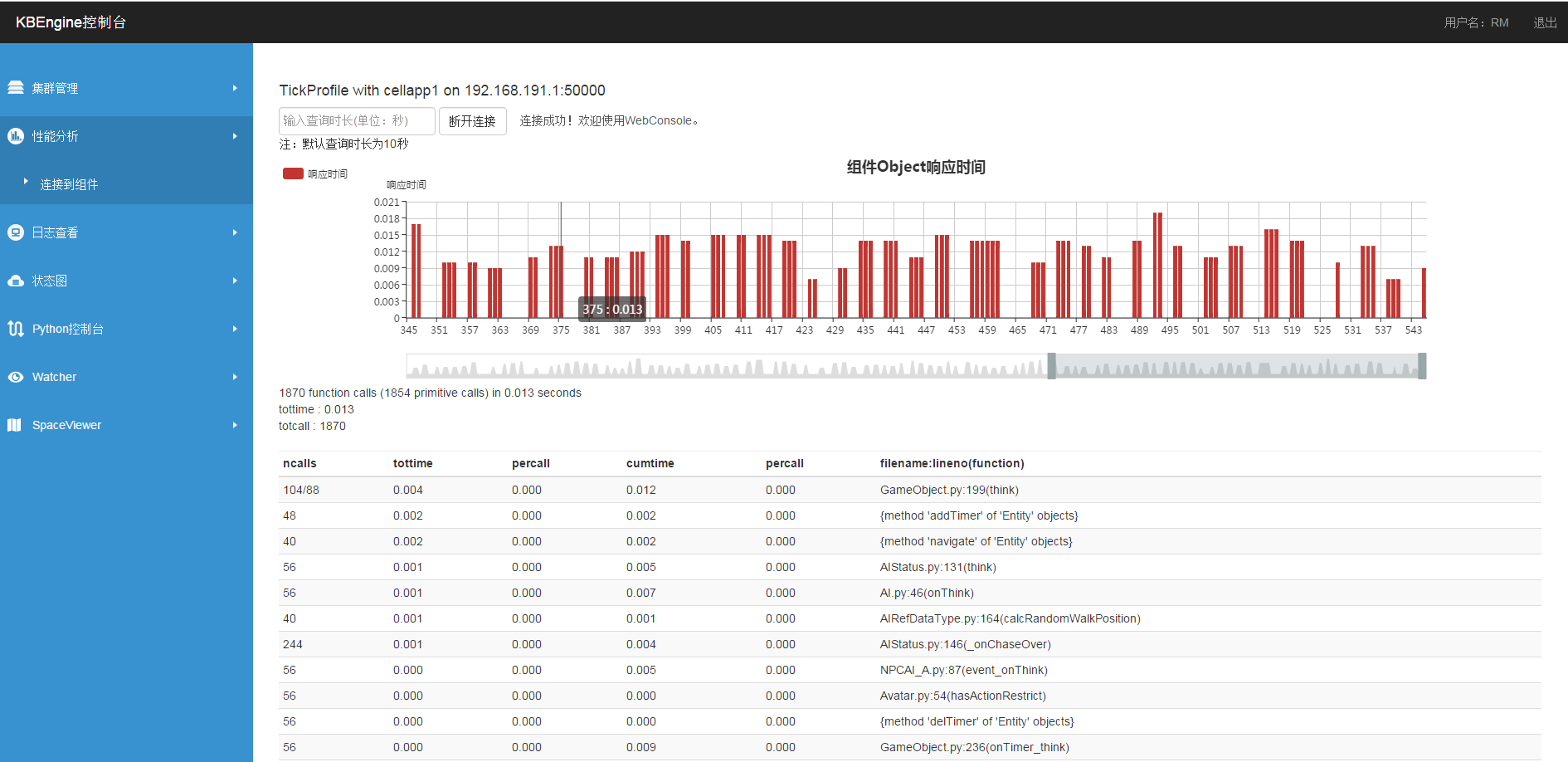
在这个页面，选择你想进行分析查看的操作



3-3-0-1组件进程选择界面

#### TickProfile

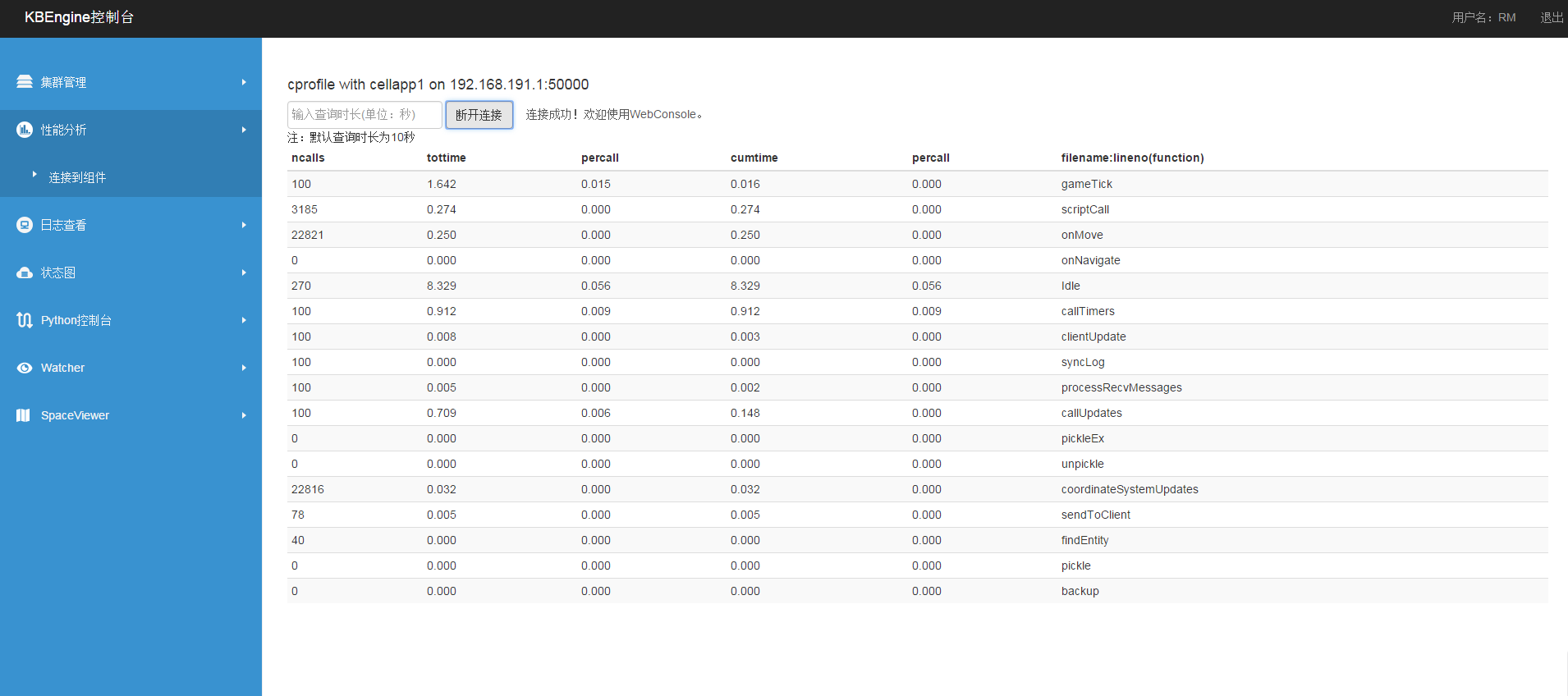
在这个页面，你可以进行TickProfile的查询操作，查询时长为空，即默认查询时长为10秒，通过点击柱条可查看相关详细信息



3-3-1-1 TickProfile界面

#### Cprofile

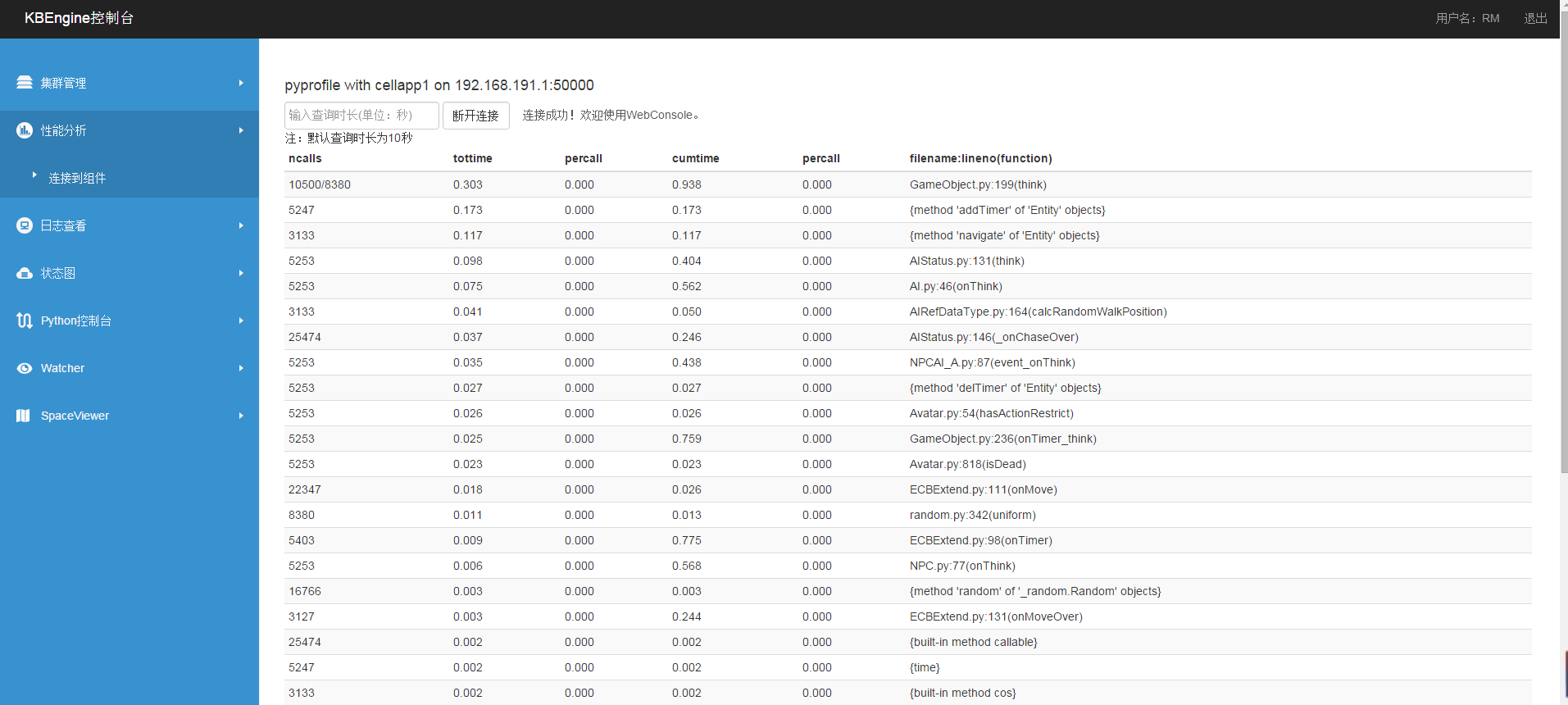
在该界面，你可以进行cprofile的性能分析查询，查询时长为空，即默认查询时长为10秒。



3-3-2-1 CProfile界面

#### Pyprofile

在该界面，你可以进行pyprofile的信息查询分析，查询时长为空，即默认查询时长为10秒。



3-3-3-1 PyProfile界面

#### Eventprofile

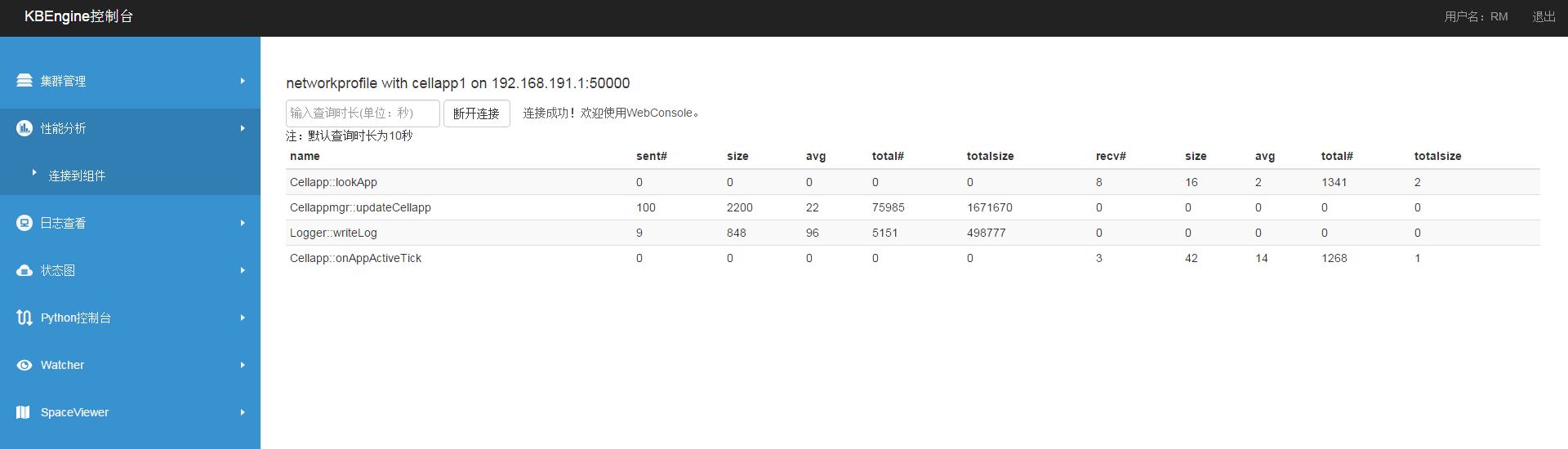
在该界面，你可以进行eventprofile的信息查询分析，查询时长为空，即默认查询时长为10秒。



3-3-4-1 EventProfile界面

#### Networkprofile

在该界面，你可以进行网络性能的信息查询分析，查询时长为空，即默认查询时长为10秒。

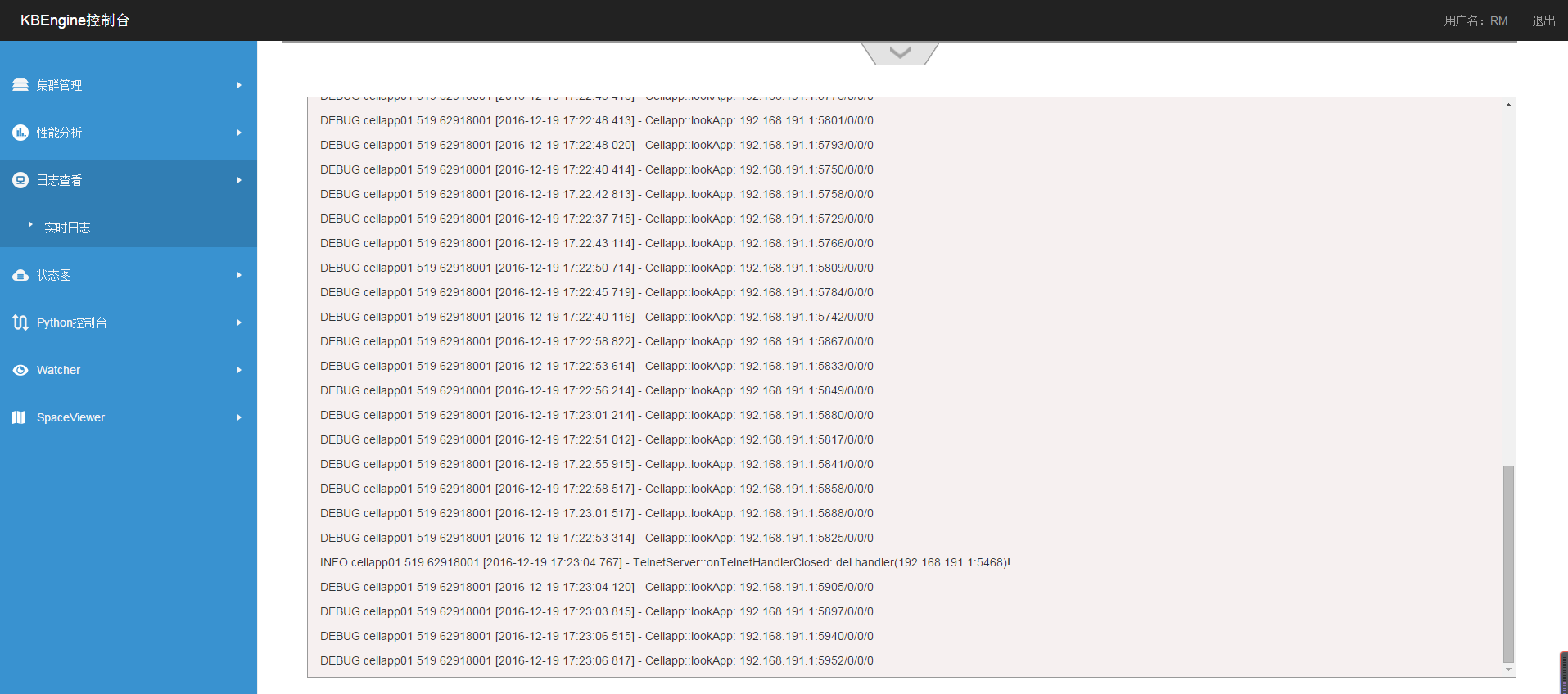


3-3-4-1 NetWorkProfile界面

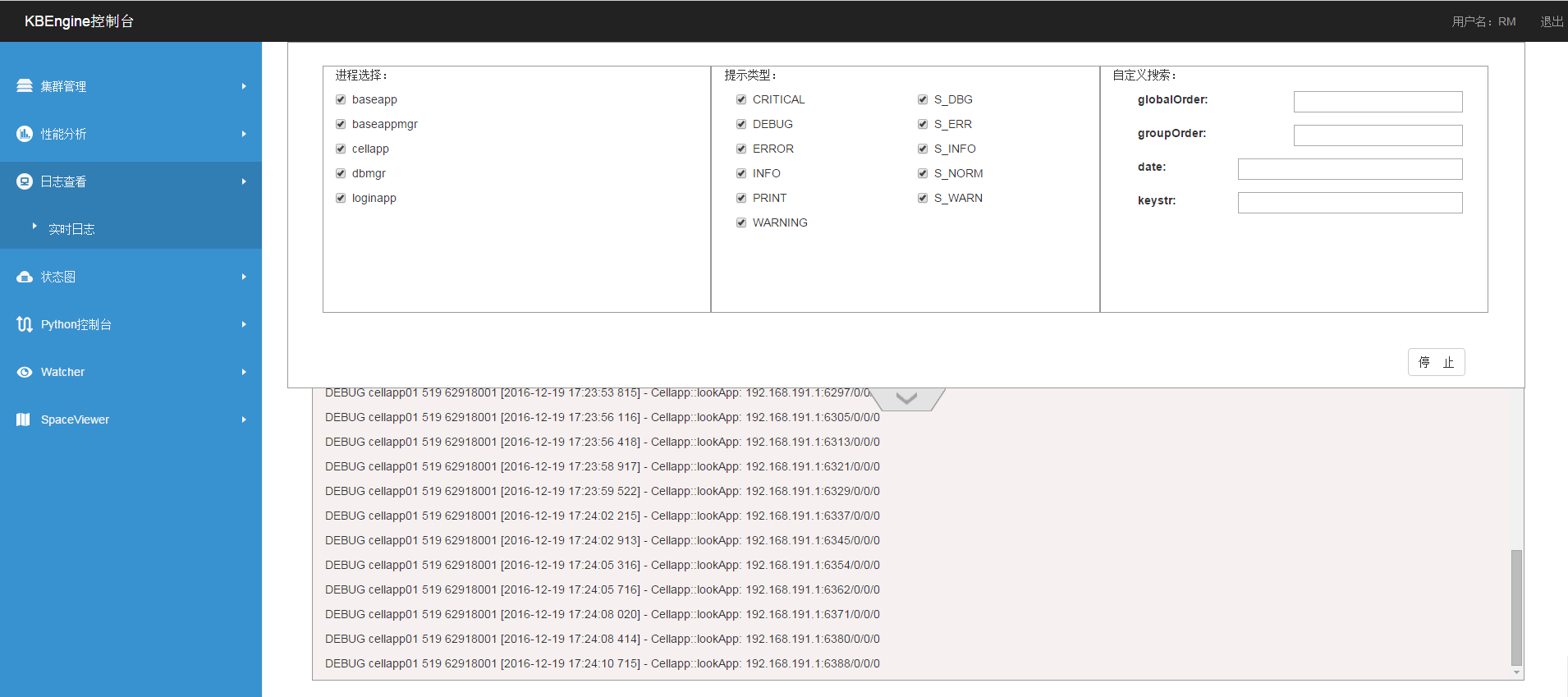
### 日志查看

#### 实时日志

该页面提供实时日志的查看和筛选，点击上方的箭头可下拉筛选菜单，默认为查看所有进程和所有类型的日志数据。



3-4-1-1 实时日志界面



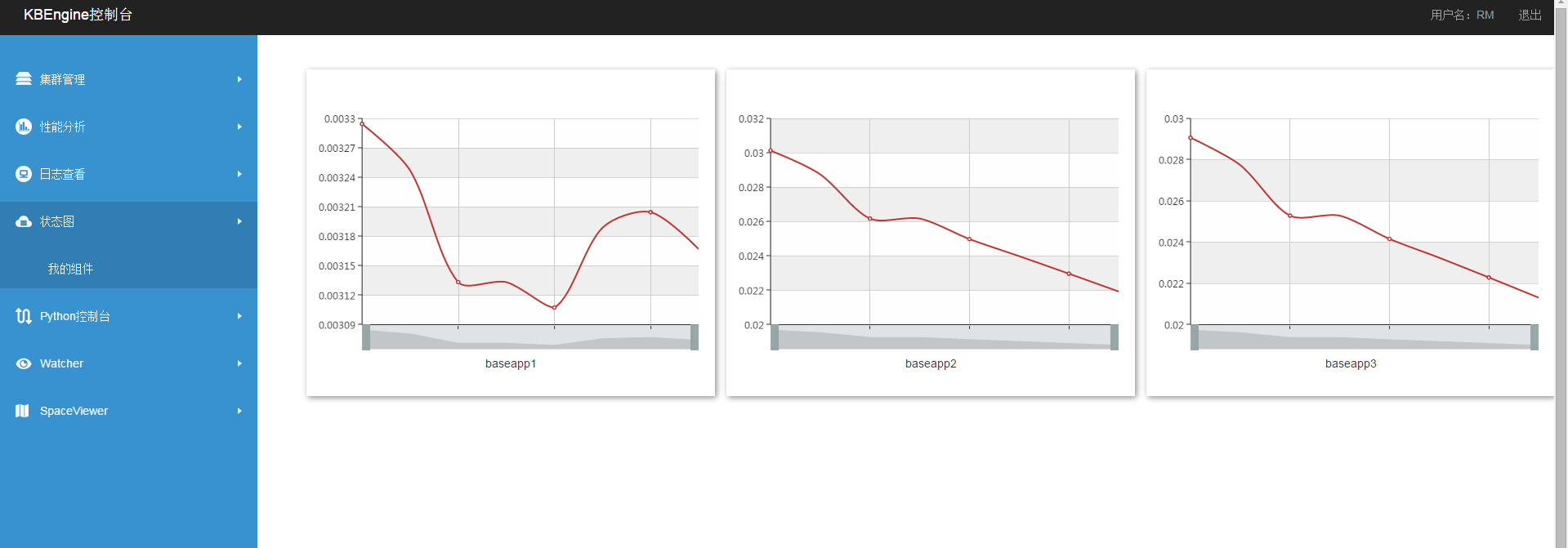
3-4-1-2 实时日志筛选

### 状态图

状态图页面提供给开发者查看当前cellapp和baseapp的运行状态线性图表，点击名称可查看单个组件进程的状态线形图。



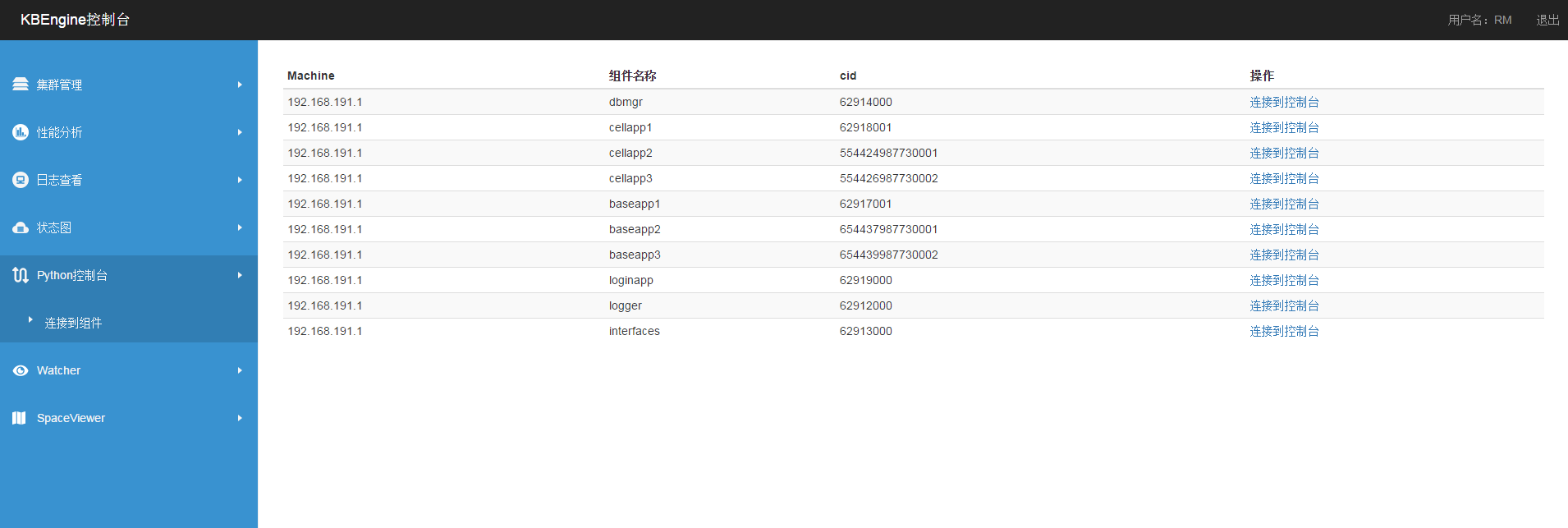
3-5-1-1 状态图组件总览页面



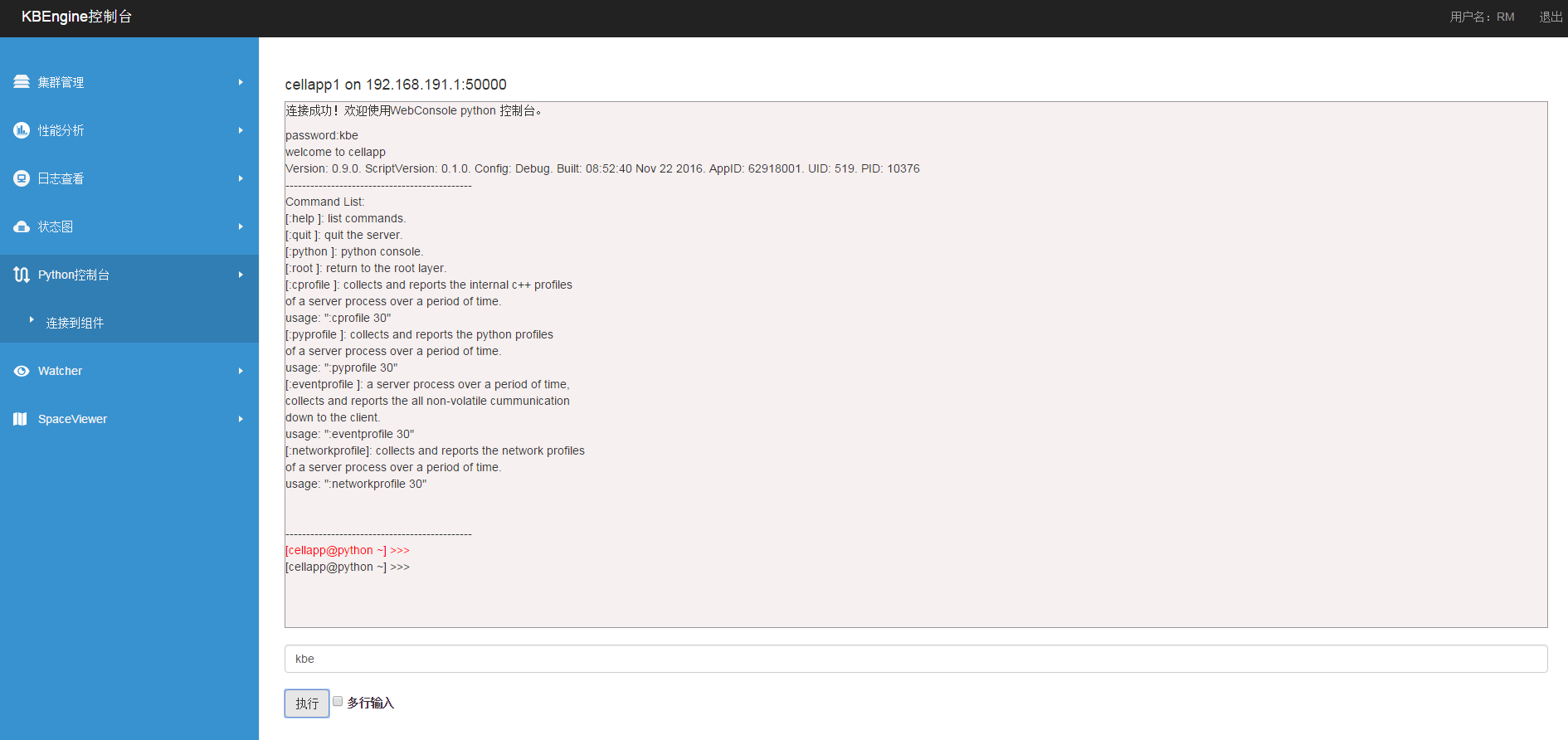
3-5-1-2 状态图组件进程页面

### Python控制台

在该功能下，你可以选择对单个进程使用控制台功能（Telnet形式），点击“连接到控制台”进入该进程控制台，在下方输入框输入命令，点击执行。



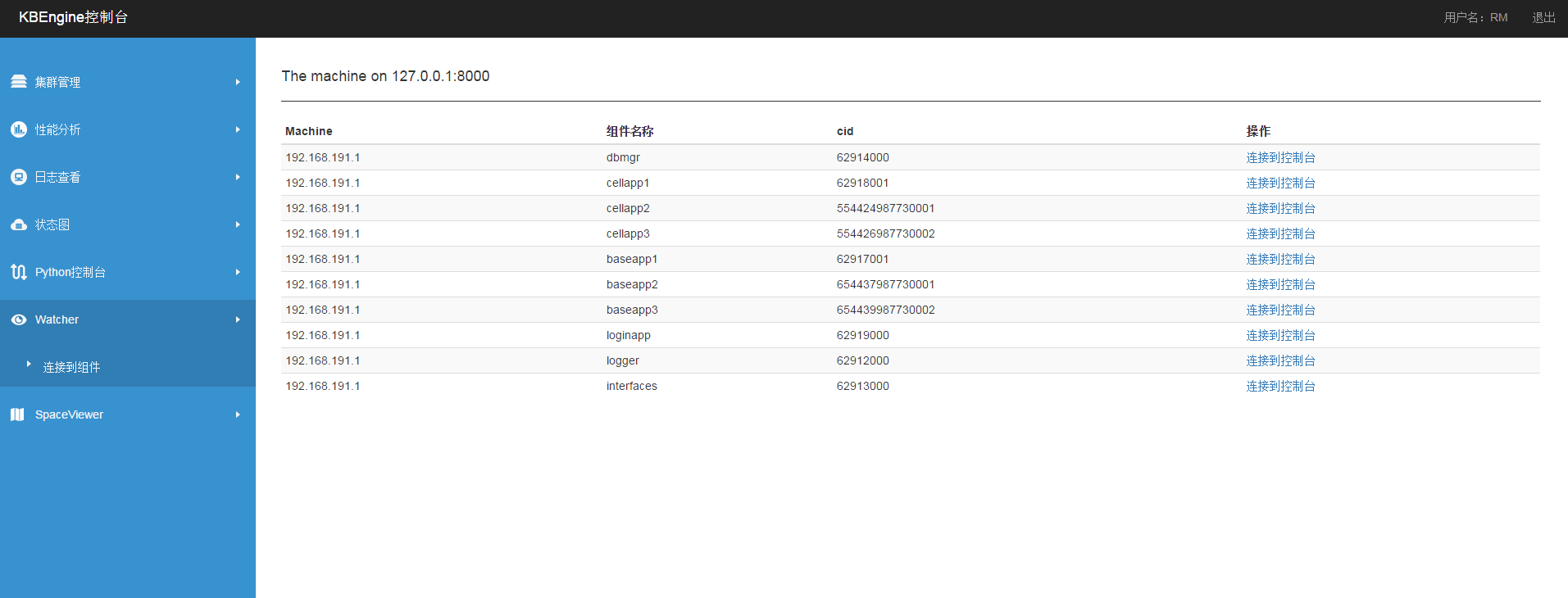
3-6-1-1 Python控制台组件进程选择页面



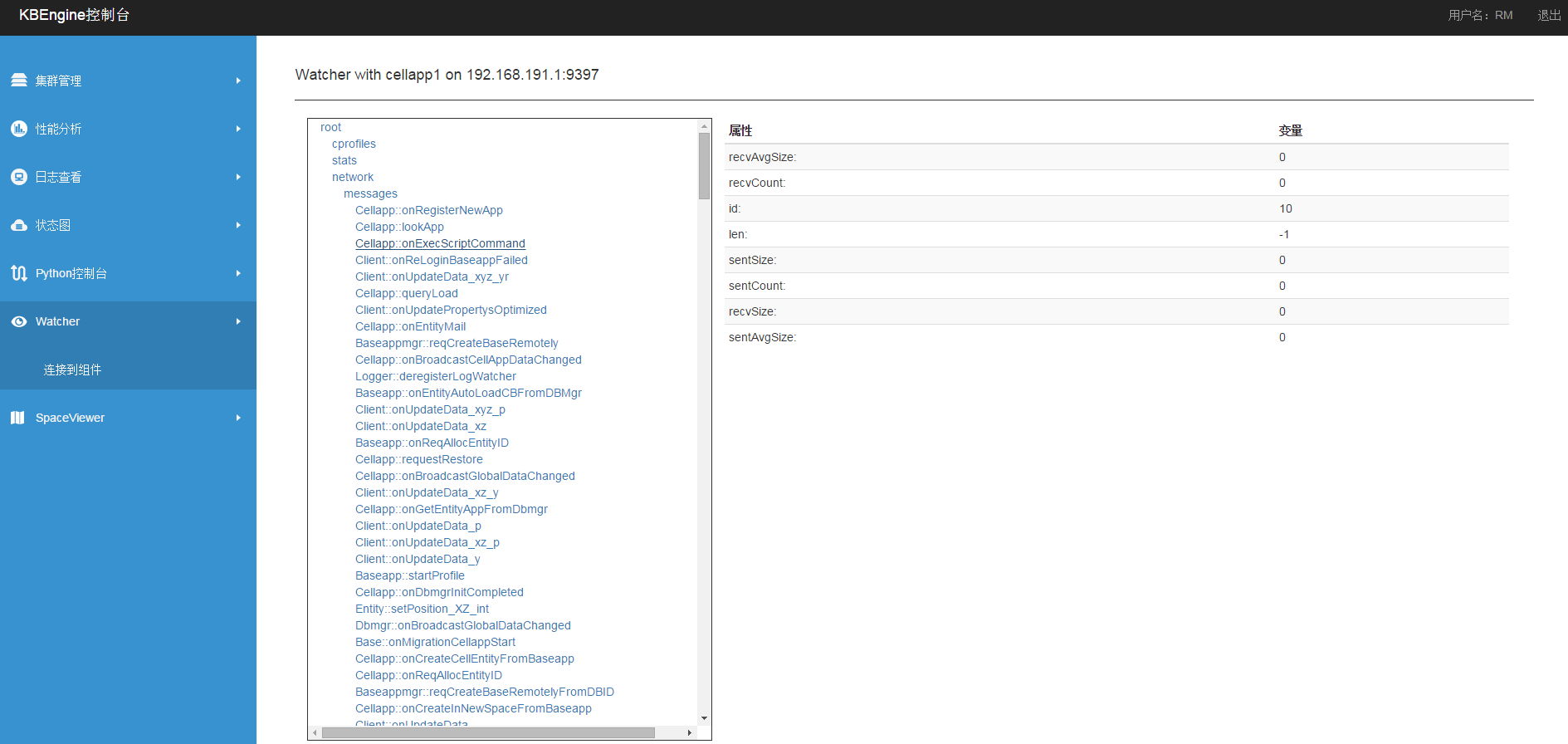
3-6-1-2 Python控制台页面

### Watcher

Watcher功能提供给开发者一个查看单个组件进程的所有状态信息（网络信息、响应速度等），并实时更新数据到进程Watcher页面。



3-7-1-1 Watcher组件进程选择页面

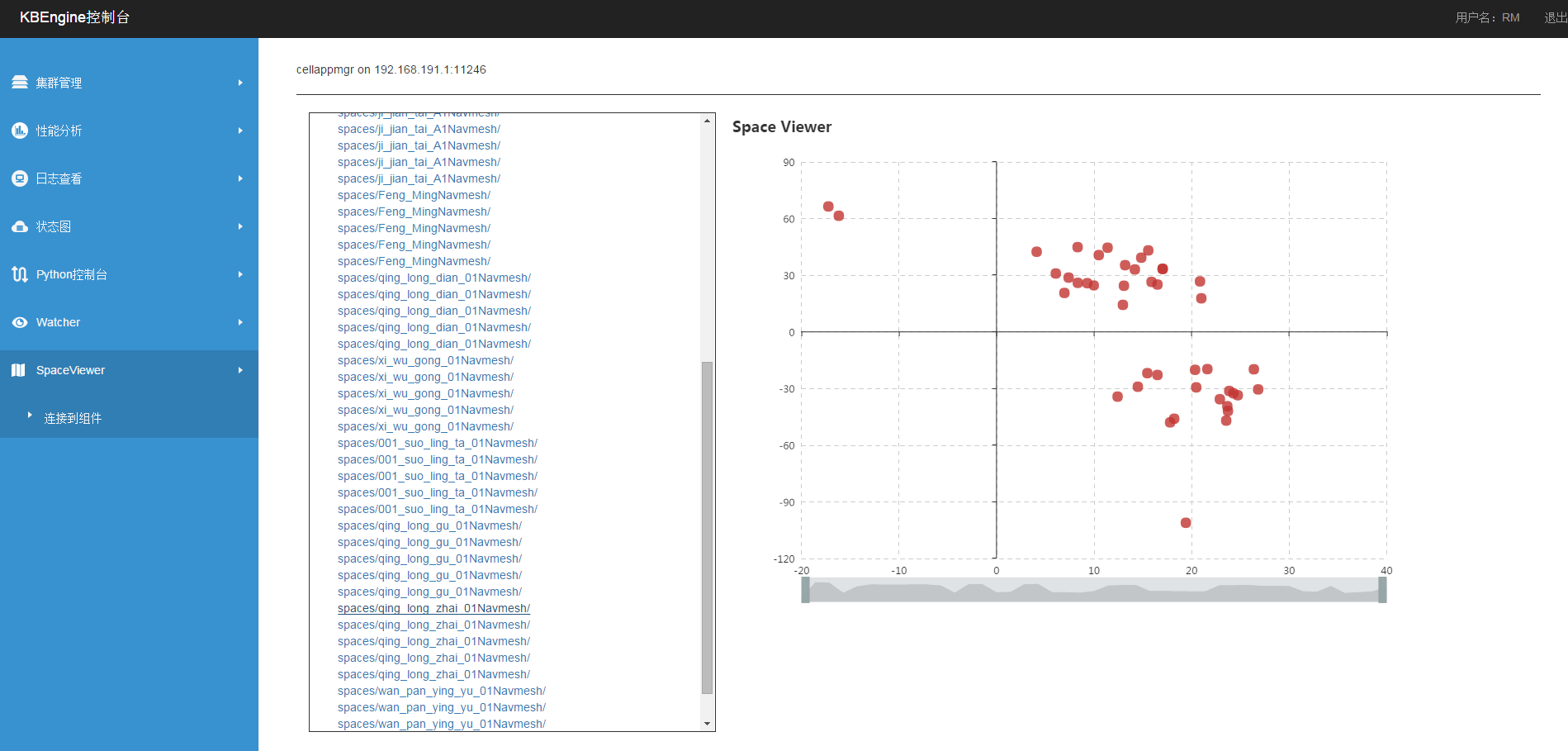


3-7-1-2 Watcher操作页面

### SpaceViewer

在SpaceViewer中，你可以查看游戏当中所有Space中的entity的分布情况，左边的space列表会随着space的产生而更新，右边实时显示该space的entity的运动情况和分布情况。

注：因暂时无法获取space地图的大小，所以该SpaceViewer的XY轴的值为所有entity中最大X值和最大Y值。



3-8-1-1 SpaceView页面