Guide for Setting Up Postgresql

12/2/2017

Contents

[1. 硬件参数 2](#_Toc480202453)

[2. 信息收集 2](#_Toc480202454)

[3. 开始之前 2](#_Toc480202455)

[3.1. 所需文件 2](#_Toc480202456)

[3.2. 所需工具 2](#_Toc480202457)

[4. 安装前置 3](#_Toc480202458)

[4.1. 更新Hostname 3](#_Toc480202459)

[5. 安装Postgresql 9.5 3](#_Toc480202460)

[5.1. 使用ubuntu的apt工具进行安装 3](#_Toc480202461)

[6. 建立防火墙 3](#_Toc480202462)

[6.1. 更新防火墙配置 3](#_Toc480202463)

[6.2. 检查防火墙状态 4](#_Toc480202464)

[7. 配置Postgresql 4](#_Toc480202465)

[7.1. 修改默认账号postgres用户的密码 4](#_Toc480202466)

[7.2. 修改postgres用户的密码 4](#_Toc480202467)

[7.3. 修改postgresql.conf配置文件 5](#_Toc480202468)

[7.4. 修改pg\_hba.conf配置文件 5](#_Toc480202469)

[7.5. 添加用户及授权 5](#_Toc480202470)

[7.6. 启停服务 5](#_Toc480202471)

[8. 验证Postgresql安装 5](#_Toc480202472)

[9. 部署 6](#_Toc480202473)

# 硬件参数

|  |  |  |
| --- | --- | --- |
| **Item** | **Specification** | **Comments** |
| RAM |  |  |
| CPU |  |  |
| OS Disk |  |  |
| Log Disk |  | /var/log/postgresql/ |
| Data Disk |  | /var/lib/postgresql/9.5/main |
| Swap Disk |  |  |

# 信息收集

Fill the form below

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Key** | **Value** | **Comments** |
| Postgresql Node | IP Address |  |  |
| App Server 1 | APP1 IP |  |  |
| App Server 2 | APP2 IP |  |  |

# 开始之前

## 所需文件

请确保你获得以下文件:

* ubuntu-16.04.1-server-amd64.iso

## 所需工具

请准备好以下工具:

* WinSCP (<http://winscp.net/> )
* PuTTY (<http://www.putty.org/>)
* Notepad ++ (可选, <https://notepad-plus-plus.org/>) (WinSCP和Notepad++ 的集成可参见 <https://winscp.net/eng/docs/integration_editor>)

# 安装前置

本说明以以下假设为前提: 服务器满足硬件参数条件; Linux HyperV虚拟机集成服务或者VmWare虚拟机的VmWare Tools已安装.

关于Linux集成服务的更多信息请参阅:

<http://blogs.technet.com/b/virtualization/archive/2015/05/01/linux-integration-services-4-0-announcement.aspx>

## 更新Hostname

hostnamectl set-hostname **PostgresqlNodeName**

例如:

hostnamectl set-hostname postgres

# 安装Postgresql 9.5

## 使用ubuntu的apt工具进行安装

保证服务器可以访问外网

sudo apt-get update

sudo apt-get install postgresql-9.5

检查postgresql版本来验证是否安装完成

psql --version

# 建立防火墙

## 更新防火墙配置

sudo apt install firewalld

sudo firewall-cmd --permanent --zone=internal --add-port=5432/tcp

sudo firewall-cmd --permanent --zone=internal --add-source=**app1\_ip(172.16.1.1)**

**注意：外部使用GUI工具访问Postgres需要先将其IP添加在防火墙白名单，或者关闭防火墙（sudo service firewalld stop）**

sudo firewall-cmd –reload

## 检查防火墙状态

sudo firewall-cmd --zone=internal --list-all

输出应该如下:

internal

interfaces:

sources: ***ip\_addresses***

services: dhcpv6-client ipp-client mdns samba-client ssh

ports: **5432/tcp**

masquerade: no

forward-ports:

icmp-blocks:

rich rules:

# 配置Postgresql

## 修改默认账号postgres用户的密码

sudo -u postgres psql

如果你使用的是root账号，请忽略该错误could not change directory to "/root": Permission denied

## 修改postgres用户的密码

alter user postgres with encrypted password '!guomao1';

Grant Privileges on Table

You can grant users various privileges to tables. These permissions can be any combination of SELECT, INSERT, UPDATE< DELETE, INDEX, CREATE, ALTER, DROP, GRANT OPTION or ALL:

CREATE USER bps\_user WITH PASSWORD '!honeywell1';

GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO bps\_user;

GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO bps\_user;

Similarly, you can create other user **bps\_securityuser** for **bpssecurity** and **activiti** database.

输入\q退出postgresql命令行

\q

## 修改postgresql.conf配置文件

sudo vim /etc/postgresql/9.5/main/postgresql.conf

让server支持远程访问，找到listen\_addresses，删掉前面的#，并修改为：

listen\_addresses = '\*'

## 修改pg\_hba.conf配置文件

sudo vim /etc/postgresql/9.5/main/pg\_hba.conf

修改超级用户postgres的认证方式为密码访问，找到

local all postgres peer

修改为

local all postgres md5

添加远程访问的IP范围

host all all 192.168.5.1/16 md5

注：192.168.5.1/16表示以192.168开头的所有IP均可以访问该server,16表示只看前两个8位。

## 添加用户及授权

请访问https://acswiki.honeywell.com/display/BPSR200/PostgreSQL%2C+Redis+and+Cassandra+Authentication

## 启停服务

postgresql安装后，会自动在/etc/init.d/下建立postgresql启动文件，开机时自动启动，可通过service命令来对服务进行管理，例如：

检查状态  
sudo service postgresql status  
重新启动  
sudo service postgresql restart

# 验证Postgresql安装

reboot

shutdown -r now

系统重启后, 检查服务状态

sudo service postgresql status

输出的状态应为active.

● postgresql.service - PostgreSQL RDBMS

Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor prese

Active: active (exited) since Wed 2017-01-25 10:48:53 CST; 1 weeks 5 days ago

Main PID: 17403 (code=exited, status=0/SUCCESS)

Tasks: 0

Memory: 0B

CPU: 0

CGroup: /system.slice/postgresql.service

# 部署

**pgAdmin**连接数据库：

Name:172.16.5.1

Host：172.16.5.1

Port：5432

Maintenance:postgres

Username :postgres

Password:!honeywell1

使用**pgAdmin**连接数据库成功后，在bpssecurity库下执行下面脚本创建**bpsadmin**和**daqadmin**用户和 **role**以及**role\_feature**

--Feature

INSERT INTO feature(id, description, name, permissionstring) VALUES (1, '', 'UserBasic', 'user:basic');

INSERT INTO feature(id, description, name, permissionstring) VALUES (2, '', 'UserAdmin', 'user:admin');

INSERT INTO feature(id, description, name, permissionstring) VALUES (3, '', 'Alarm', 'alarm');

INSERT INTO feature(id, description, name, permissionstring) VALUES (4, '', 'DashBorardRead', 'dashboard:read');

INSERT INTO feature(id, description, name, permissionstring) VALUES (5, '', 'DashBorardWrite', 'dashboard:write');

INSERT INTO feature(id, description, name, permissionstring) VALUES (6, '', 'Configuration', 'configuration');

INSERT INTO feature(id, description, name, permissionstring) VALUES (7, '', 'PatternRead', 'pattern:read');

INSERT INTO feature(id, description, name, permissionstring) VALUES (8, '', 'PatternWrite', 'pattern:write');

INSERT INTO feature(id, description, name, permissionstring) VALUES (9, '', 'Subsystem', 'subsystem');

INSERT INTO feature(id, description, name, permissionstring) VALUES (10, '', 'WorkOrderBasic', 'workorder:basic');

INSERT INTO feature(id, description, name, permissionstring) VALUES (11, '', 'WorkOrderReassign', 'workorder:reassign');

INSERT INTO feature(id, description, name, permissionstring) VALUES (12, '', 'DAQ', 'daq');

--Role

INSERT INTO role (id, description, issystem, name) VALUES (1, '', false, 'System Operator');

INSERT INTO role (id, description, issystem, name) VALUES (2, '', false, 'Technician');

INSERT INTO role (id, description, issystem, name) VALUES (3, '', false, 'Technician Leader');

INSERT INTO role (id, description, issystem, name) VALUES (4, '', false, 'Field Engineer');

INSERT INTO role (id, description, issystem, name) VALUES (5, '', true, 'DAQ');

--F-R Map

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 1);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 2);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 3);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 4);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 7);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 9);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 10);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (1, 11);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (2, 1);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (2, 3);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (2, 4);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (2, 10);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 1);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 3);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 4);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 7);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 9);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 10);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (3, 11);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 1);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 2);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 3);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 4);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 5);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 6);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 7);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 8);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 9);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 10);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 11);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (4, 12);

INSERT INTO role\_feature(role\_id, feature\_id) VALUES (5, 12);

--User - Field Engineer

INSERT INTO bps\_user(

            createdatetime, devicetoken, displayid, email, isforcechangepwd,

            issystem, name, password,

            phonenum, salt, status, role\_id)

    VALUES (now(), '', 'BPSAdmin', '', true,

            false, 'BPS Administrator', '$shiro1$SHA-256$500000$dj5B/fkEeI/+YECIvxhl6Q==$ejbIISevylxjpW26zSc9EpyHyEFqsnACKy5Pu5zxMtI=',

            '', '', 1, 4);

--User - DAQ

INSERT INTO bps\_user(

            createdatetime, devicetoken, displayid, email, isforcechangepwd,

            issystem, name, password,

            phonenum, salt, status, role\_id)

    VALUES (now(), '', 'DAQAdmin', '', false,

            true, 'DAQ Administrator', '$shiro1$SHA-256$500000$dj5B/fkEeI/+YECIvxhl6Q==$ejbIISevylxjpW26zSc9EpyHyEFqsnACKy5Pu5zxMtI=',

            '', '', 1, 5);

--$shiro1$SHA-256$500000$dj5B/fkEeI/+YECIvxhl6Q==$ejbIISevylxjpW26zSc9EpyHyEFqsnACKy5Pu5zxMtI=

--$shiro1$SHA-256$500000$s2wKVocNqAw3amCHDke9PA==$S4EOViPuc8ttYziCLM9tUrUxHuomoB0zgoFRyzmwYak=

为组态工具创建site

INSERT INTO public."Site"( "EntityId", "BuildingStructure", "Enabled", "Name", "YearBuilt", "Buildingstorey")

VALUES (1, 7, true, 'name', 2017, 7);