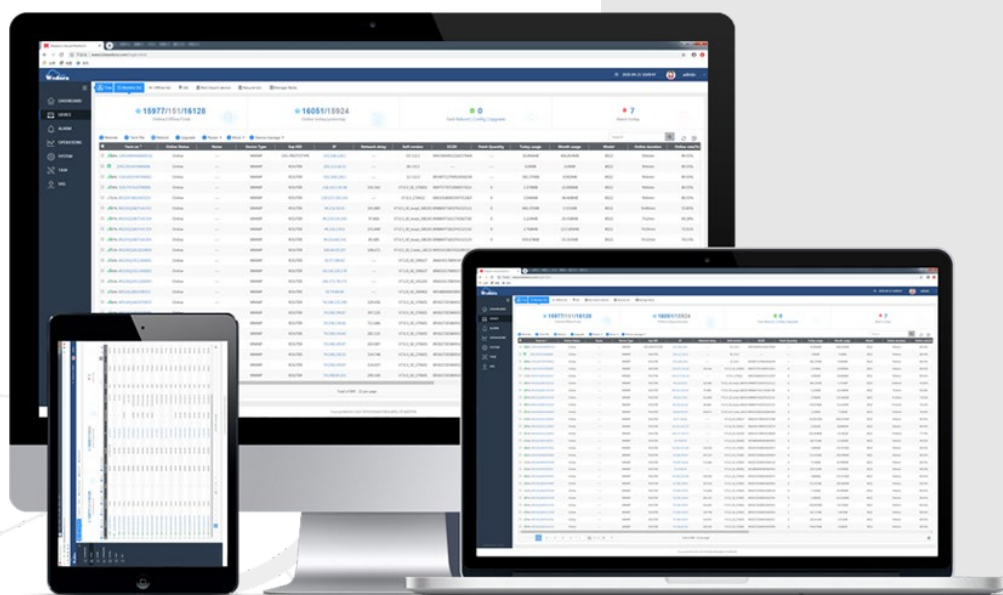




Installation instructions for Linux

Wedora Management platform V1.8.0



Contents

1 Installation Instructions	2
1.1 Server configuration requirements	2
1.1.1 Server preparation for software deployment and operation	2
1.1.2 Dependent environment preparation for software deployment and operation	2
1.1.3 Directory plan and configuration files for the required installation services.....	3
1.2 Platform service installation	3
1.2.1 System installation requirements	3
1.2.2 One-click installation package installation	4
1.3 one button start and stop service	5
2 FAQ for deployment operations	6
2.1 frequently asked questions	6
2.1.1 Open platform port.....	6
2.1.2 Clicking on the user group will cause a stuck phenomenon.....	7
2.1.3 After stopping the platform service, it cannot be started again	8
2.1.4 The device cannot register the platform after the platform is started (the device cannot be seen).....	8
2.1.5 Platform failed to upgrade remotely	9
2.2 platform configuration information	10
2.2.1 Default password settings	10
2.2.2 The Configuration Of Default Port	10
2.2.3 Configuration of Front End Service Wedora_sever.....	11
2.2.4 Configuration of the front-end machine Fep_mqtt	12
2.2.5 Configuration of Statistical component wse	13
2.2.6 Configuration of Mysql database.....	13
2.2.7 Configuration of Mongo database	14
2.2.8 Configuration of Redis database	14
2.2.9 The configuration of ipfix service	15
2.2.10 The Configuration of Mqtt server	15
2.3 commonly used methods for restarting services	16
2.3.1 Mysql	16
2.3.2 Mongo.....	16
2.3.3 Redis	17
2.3.4 Mqtt.....	17

2.3.5 Wedora_server	18
2.3.6 Wse.....	19
2.3.7 Fep.wmmp	19
2.3.8 lpfix	20

1 Installation Instructions

This chapter details the detailed process of platform one-button deployment and the related parameters and port introduction of the platform. Because the jdk, mongo, mqtt, node, redis, wedora_server, jar package, etc.

are integrated in the one-click installation package, before installing the platform, you need to manually install gcc gcc-g++ and some dependencies of mysql (depending on the local system status), after the basic environment is installed, you can use it with one-click installation.

1.1 Server configuration requirements

1.1.1 Server preparation for software deployment and operation

Because it involves massive data storage, it is recommended that it be deployed and adopted by the customer's production system:

Name	Use of the server	Operating system	Configuration requirements		
			CPU	RAM	hard disk
Public network: 1) Gigabit NIC is recommended; 2) Bandwidth is recommended to be no less than 20M/s					
server	Deploy mysql(directory /usr/bin), jdk1.7, redis, MongoDB, ndoe, mqtt (/home/wedora/server/tripartite), Deploy platform server (/home/wedora/server/), platform components:fep_mqtt, fep_mqtt, wedora_server, wse;	Centos 6.5/7.5	Dual core / 2.5GHz and above	8G+	>=500GB, where the Home directory is not less than 200GB

PS: During the installation of the operating system, the home directory is recommended to be as large as possible, so that users can store platform-related services and data.

1.1.2 Dependent environment preparation for software deployment and operation

Serial number	Service module	name of software	version number	Description
1	OS system	operating system	Centos 6.5/7.5	Standard version
2	Third party service	Database Mysql	Mysql5.5+ (低于 5.7)	Recommended version
3		JAVA environment	Jdk 1.8	Provided by the platform

4		Node JS	V10.6.0	Provided by the platform
5		Redis	V2.3	Provided by the platform
6		Mqtt server	V2.3	Provided by the platform
7		Database mongodb	v3.2.1	Provided by the platform
8		ipfix		Provided by the platform
8	Platform service	Wedora_server		Provided by the platform
9		wse		Provided by the platform
10		Front-endmachine wmp		Provided by the platform
11		Front-end machine mqtt		Provided by the platform

1.1.3 Directory plan and configuration files for the required installation services

The default service installation path for the platform: **/home/wedora/server**

The default installation path for third-party services: **/home/wedora/server/tripartite**

The configuration file path of the platform Redis and Mongo: **/home/wedora/server/conf/**

Serial number	service	Directory location	Configuration file (path)
Platform-side service			
1	Wedora_server	../wedora_server	../wedora_server/config/config.json
2	Front-end machine wmp	../fep_wmp	../fep_wmp/apps.xml
3	Wse statistics component	../wse	../wse/config/application.yml
Third party service			
1	Mysql database	/usr/bin/mysqld	/etc/my.cnf
2	Jdk	../java	.bashrc
3	Redis	../redis	../redis.conf
4	Mongodb database	../mongo	../mongodb.conf
5	Node	../node-v10.6.0	.bashrc
6	Mqtt server	../mqtt	../mqtt/etc/emq.conf
7	ipfix	/home/wedora/server/ipfix	./config.ini

1.2 Platform service installation

1.2.1 System installation requirements

1) Centos 6.5/7.5 The Centos 6.5/7.5 system is the prerequisite for the installation and operation of the service. Please install the centos version as required below! (avoid not being able to use)

The installed version of the system is : [CentOS-6.5-x86_64-bin-DVD1.iso](#)

[CentOS-7.5-x86_64-bin-DVD1.iso](#)

The download address of the Centos image is:<http://vault.centos.org/>

[Note: Due to the centos7.5 system itself, you need to download and install additional dependencies, so you need an external network connection! !]

2) Before proceeding to the next step, due to system problems with centos7, you need to install update gcc to version 4.8 and above in advance.

```
#yum install gcc gcc-g++
#gcc --version
[root@localhost ~]# gcc --version
gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-28)
Copyright © 2015 Free Software Foundation, Inc.
本程序是自由软件：请参看源代码的版权声明。本软件没有任何担保：
包括没有适销性和某一专用目的下的适用性担保。
```

3) Check if the necessary dependencies for mysql (such as perl and network-tools) are installed.

Use the following command to install:

```
#yum install -y perl
#yum install -y network-tools
```

1.2.2 One-click installation package installation

Get the installation package Wedora_V1.8.0_EN_20181228.iso,upload the installation package to the /root directory on the server to install it. The process is as follows:

1) Mount the installation package

```
#mkdir /mnt/cdrom (It is recommended to mount to /mnt/cdrom)
#mount -o loop wedora_V1.8.0_20181228.iso /mnt/cdrom/
```

2) One-click installation

```
#cd /mnt/cdrom
# sh start-install.sh
```

Select the installation directory, if you do not enter the directory name and directly press Enter, it will be installed in the default directory /home/wedora/server.

3) Environment variable preservation

```
#source /etc/profile
```

4) View the installation directory

```
#ll //Directory structure after installation is complete
```

```
[root@localhost wedora]# pwd
/usr/local/wedora
[root@localhost wedora]# ll
total 4
drwxrwxrwx. 2 root root 186 Nov 13 09:10 bin
drwxrwxrwx. 2 root root 44 Nov 13 09:10 conf
drwxrwxrwx. 2 root root 172 Nov 13 09:04 ext
drwxrwxrwx. 4 root root 205 Nov 13 09:11 fep_wmmp
drwxrwxrwx. 8 root root 87 Nov 13 09:10 tripartite
drwxrwxrwx. 9 root root 223 Nov 13 09:11 wedora_server
-rwxrwxrwx. 1 root root 15 Nov 13 09:04 WEDORA_VERSION
drwxrwxrwx. 4 root root 81 Nov 13 09:17 wse
```

1.3 one button start and stop service

```
#cd /home/wedora/server/bin
```

```
#ll
```

```
-rwxrwxrwx. 1 wedora hongdian 558 Jul 26 14:11 shutdown_wedora.sh
-rwxrwxrwx. 1 wedora hongdian 1142 Jul 26 14:11 startup_wedora.sh
```

```
#sh startup_wedora.sh //Start the wedora service
```

Start the platform:

Access platform address in the browser:

http://*server IP*:50107 (default address)

Default username and password:admin/123456

The default port of the platform is 50107.You can see the "50107" port in the config.json configuration under /home/wedora/server/wedora_server/config. You can modify the corresponding port number as needed.

```
#sh shutdown_wedora.sh // Stop the wedora service
```

```
[wedora@localhost bin]$ ls
install-mysql.sh install-wedora.sh logs nohup.out open_port.sh redis.sh shutdown_wedora.sh startup_wedora.sh
[wedora@localhost bin]$ ./shutdown_wedora.sh
shoutdown mongo
shoutdown redis
Stopping ...
Redis stopped
shoutdown mqtt and fep.mqtt
shoutdown fep.wmmp
shoutdown wedora server engine
shoutdown wedora web server
[PM2] Applying action stopProcessId on app [wedora](ids: 0)
[PM2] [wedora](0) ✓
```

App name	id	mode	pid	status	restart	uptime	cpu	mem	user	watching
wedora	0	fork	0	stopped	0	0	0%	0 B	wedora	disabled

Use `pm2 show <id|name>` to get more details about an app

```
[wedora@localhost bin]$
```

```
#sh startup_wedora.sh
```

```

Use `pm2 show <id|name>` to get more details about an app
[wedora@localhost bin]$ ./startup_wedora.sh
start mongo
about to fork child process, waiting until server is ready for connections.
forked process: 666
child process started successfully, parent exiting
start redis
Starting Redis server...
start mqtt
emqttd 2.3 is started successfully!
start fep.wmmp
start fep.mqtt
start wedora server engine
start wedora web server
nohup: appending output to `nohup.out`
nohup: appending output to `nohup.out`
nohup: appending output to `nohup.out`
[PM2] Applying action restartProcessId on app [wedora](ids: 0)
[PM2] [wedora](0) ✓
[PM2] Process successfully started

```

App name	id	mode	pid	status	restart	uptime	cpu	mem	user	watching
wedora	0	fork	1249	online	0	0s	0%	5.8 MB	wedora	disabled

```

Use `pm2 show <id|name>` to get more details about an app
start wedora success.
[wedora@localhost bin]$

```

In addition, the mysql service is not started in the script. After the mysql is installed, it will start automatically and does not need to be restarted. If a single service fails to start, you can start it manually. For details, see the last chapter of the document.

2 FAQ for deployment operations

2.1 frequently asked questions

2.1.1 Open platform port

The platform needs to be shut down in the actual use process, and the platform port needs to be released:

- execute the script to open the platform port
- `#!/home/wedora/server/bin/open_port.sh`
- Enter the number of open ports and specific ports
- Such as: open port 1833 to access
- Enter the number of ports and port number.**


```
[root@localhost ~]# /home/wedora/server/bin/open_port.sh
希望开放的端口数量: 1
输入一个想开放的端口: 1883
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Flushing firewall rules: [ OK ]
iptables: Unloading modules: [ OK ]
iptables: Applying firewall rules: [ OK ]
```

It is recommended to only open port 50107 for accessing the platform address and port for device access and upgrade (default is 41880, 41882)

```
[root@localhost ~]# firewall-cmd --add-port=1883/tcp --permanent
success
```

2.1.2 Clicking on the user group will cause a stuck phenomenon

Mysql database has a case-sensitive configuration, so edit the mysql configuration file and set it to be case insensitive.

```
#vim /etc/my.cnf
```

[Add the following content under [mysqld] and restart:

```
lower_case_table_names=1
```

```
[mysqld]
basedir=/usr/local/mysql
datadir=/usr/local/mysql/data
port=3306
sql_mode=STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent accidental data loss, especially if connection is from source machine where
symbolic-links=0
lower_case_table_names=1
[mysqld_safe]
log-error=/var/log/mysql.log
```

```

[mysqld]
basedir=/usr/local/mysql
datadir=/usr/local/mysql/data
port=3306
sql_mode=STRICT_TRANS_TABLES,NO_ZERO_IN_DATE
UTION
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to
symbolic-links=0
lower_case_table_names=1
[mysqld_safe]
log-error=/var/log/mysqld.log

```

2.1.3 After stopping the platform service, it cannot be started again

Check whether the service corresponding to the port is in the startup state and check whether the port is occupied.

```
# ps -ef|grep redis
```

```

[wedora@localhost bin]$ ps -ef|grep redis
wedora 16258 1 0 Jul26 ? 00:01:24 /home/wedora/server/tripa
wedora 30388 58215 0 12:49 pts/3 00:00:00 grep redis
[wedora@localhost bin]$

```

Restart after shutting down the service (see the last chapter of the document for details)

2.1.4 The device cannot register the platform after the platform is started (the device cannot be seen)

```
#cd /home/wedora/server/fep_wmmp/
```

```
#vi apps.xml
```

Change the address of the UDP protocol to the IP of the server:

In the example below:

```

<property-list>
<property name="IS_SEND_HEART">1</property>
<property name="TRAP_COUNT">5000</property>
  <property name="HTTP_ADDR">http://127.0.0.1:61503/wedora/</property>
  <property name="TERM_SERVER_ADDRESS">UDP://127.0.0.1:5002</property>
  <property name="TERM_UPGRADE_ADDRESS_LAN">TCP://127.0.0.1:5001</property>
  <property name="TERM_UPGRADE_ADDRESS_WAN">TCP://127.0.0.1:5001</property>

  <property name="HTTP_ADDR_CMD">http://127.0.0.1:61503/wedora/</property>

```

```
<!-- 云管理前置机组件 -->
<comp name="fep.wmmp1" type="exe" version="1.0">
  <property-list>
    <property name="IS_SEND_HEART">1</property>
    <property name="TRAP_COUNT">5000</property>
    <property name="HTTP_ADDR">http://127.0.0.1:61503/wedora/</property>
    <property name="TERM_SERVER_ADDRESS">UDP://172.16.22.38:5002</property>
    <property name="TERM_UPGRADE_ADDRESS_LAN">TCP://127.0.0.1:5001</property>
    <property name="TERM_UPGRADE_ADDRESS_WAN">TCP://127.0.0.1:5001</property>
  </property-list>
  <property name="HTTP_ADDR_CMD">http://127.0.0.1:61503/wedora/</property>
</comp>
```

:wq (Save and exit)

Restart the wmmp service

2.1.5 Platform failed to upgrade remotely

When the platform fails to upgrade through remote, you can check the problem by viewing the wmmp log:

1. If the device returns to the upgrade state, it is a failure, which proves that the platform command was successfully sent to the device, and the device does not support the upgrade of the installation package;
2. If the device does not return to the upgrade state, check whether the WAN port information configured in the apps.xml of wmmp is correct. The configuration of the external network address and the external mapping port may be incorrect.

2.2 platform configuration information

2.2.1 Default password settings

Under the premise that the customer has no special requirements:

name	account /password
MySQL	root/sa wedora/123456
Mongo	wedora/123456
redis	/
The Landing of the platform	admin/123456

2.2.2 The Configuration Of Default Port

For external network access, the external network access rights of http (50107) and fep_wmmp (41880, 41882) ports must be enabled by default.

name	The port number	The path of Configuration	Remarks
Internal port			
mysql	3306	/etc/my.cnf	No need to modify
mongo	27017	../mongodb.conf	No need to modify
redis	6379	../redis.conf	No need to modify
External port (external mapping required)			
http-port	50107	../wedora_server/config/config.json	Front end access port of http
mqt	1883	../mqt/etc/emq.conf	Access to IPC type devices needs to be mapped
Fep_mqt	1883	../fep_mqt/app.xml	The access port of IPC device
Fep_wmmp	41880	../fep_wmmp/app.xml	The access port of Router and DTU
	41882		The port that the device uses to upgrade data.
	5001		Required port for lperf function
ipfix	41803	../ipfix/config.ini	Default port for ipfix

2.2.3 Configuration of Front End Service Wedora_sever

```
#cd /home/wedora/server/wedora_server/config
```

```
#ll
```

```
[wedora@localhost config]$ ll
total 4
-rwxrwxrwx. 1 wedora hongdian 683 Jul 26 14:09 config.json
```

```
#cat config.json
```

```
[wedora@localhost config]$ cat config.json
```

```
{
  "app_name": "monitor",
  "http_port": 50107,
  "internet_ip": "127.0.0.1",
  "session_key": "JSSESSION",
  "language": "zh_CN",
  "redis_nodes": [
    {
      "address": "127.0.0.1",
      "port": "6379"
    }
  ],
  "login_timeout": 60,
  "mysql": {
    "db": "wedora",
    "user_name": "wedora",
    "password": "123456",
    "host": "127.0.0.1",
    "port": "3306"
  },
  "mongodb": {
    "uri": "mongodb://wedora:123456@127.0.0.1:27017/$db?authMechanism=SCRAM-SHA-1&authSource=admin",
    "db": "wedora",
    "collections": {
      "dev_info": "dev"
    }
  },
  "mqtt": {
    "url": "mqtt://127.0.0.1:1883"
  },
}
```

The configuration instructions of Config.json :

internet_ip: The front-end access IP of wedora_server needs to be filled in to the external network IP, and the port needs to be mapped.

Redis_nodes, mysql, mongodb, and mqtt are the default configurations. They must be consistent with the corresponding component ports and user passwords.

2.2.4 Configuration of the front-end machine Fep_mqtt

Check the apps.xml file

```
#cd /home/wedora/server/fep_wmmp
```

```
[wedora@localhost fep_wmmp]$ ll
total 13628
-rwxrwxrwx. 1 wedora hongdian    321 Jul 26 14:08 README.txt
-rwxrwxrwx. 1 wedora hongdian     6 Jul 26 14:08 VERSION
-rwxrwxrwx. 1 wedora hongdian  2076 Jul 26 17:07 apps.xml
-rwxrwxrwx. 1 wedora hongdian   398 Jul 26 14:08 cmd_example.txt
-rwxrwxrwx. 1 wedora hongdian    54 Jul 26 14:08 deviceSn.txt
-rwxrwxrwx. 1 wedora hongdian 13915602 Jul 26 14:08 fep_wmmp
-rwxrwxrwx. 1 wedora hongdian   826 Jul 26 14:08 fep_wmmp_startup.sh
-rwxrwxrwx. 1 wedora hongdian   318 Jul 26 14:08 fep_wmmp_stop.sh
drwxr-xr-x. 3 wedora hongdian   4096 Jul 26 14:24 file
drwxr-xr-x. 2 wedora hongdian   4096 Jul 26 17:20 log
```

```
#cat apps.xml (Check the apps.xml configuration file)
```

```
<!-- 云管理前置机组件 -->
name="fep.wmmp" type="exe" version="1.0">
  <property-list>
    <property name="TERM_SERVER_ADDRESS">UDP://0.0.0.0:41880</property>
    <property name="TERM_UPGRADE_ADDRESS_LAN">TCP://127.0.0.1:41882</property>
    <property name="TERM_UPGRADE_ADDRESS_WAN">TCP://127.0.0.1:41882</property>
    <property name="MQTT_SERVER_ADDR">127.0.0.1</property>
    <property name="MQTT_SERVER_PORT">1883</property>
    <property name="MQTT_USERNAME"></property>
    <property name="MQTT_USERPWD"></property>
    <property name="MQTT_QOS">0</property>
    <property name="MQTT_KEEPALIVE">60</property>
    <property name="MQTT_AUTHFILE"></property>
    <property name="DEVICE_TIMEOUT">90</property>
    <property name="CMD_TIMEOUT">30</property>
    <property name="RESEND_TIME">3</property>
    <property name="UPGRADE_TIMEOUT">180</property>
    <property name="TERM_MODEL_ID">1</property>
    <property name="TERM_TYPE_ID">1</property>
    <property name="UPGRADE_PATH"></property>
  </property-list>
```

Introduction to the apps.xml configuration:

The TERM_SERVER_ADDRESS and HTTP_ADDR_CMD parameters in the file are the local monitoring IP plus port (used to receive data reported by the device. After the service starts, it starts listening and waiting for device data reporting. It is also called registration port).

The TERM_UPGRADE_ADDRESS_LAN parameter in the file is the port used to upgrade the device, and also used for local monitoring.

The TERM_UPGRADE_ADDRESS_WAN parameter in the file is the device upgrade port, which is used to notify the device and let the device connect to the server for upgrade (it needs to be modified to be an external network interface to ensure that the device can be connected, otherwise the upgrade problem cannot be transmitted).

Note: Public network addresses in different environments need to be configured with public network ip and external mapping ports in TERM_UPGRADE_ADDRESS_WAN. The default external mapping port is 41882.

2.2.5 Configuration of Statistical component wse

```
#cd /home/wedora/server/wse/config  
# cat application.yml //View the configuration information of the statistics component
```

```
[root@localhost config]# cat application.yml  
#系统配置参数  
# mongo数据库连接配置  
mongo :  
  datasource :  
    host : 127.0.0.1  
    port : 27017  
    username : root  
    password : 123456  
    authSource : admin  
    database : wedora  
  
# DATABASE CONFIG  
mysql:  
  server: 127.0.0.1:3306  
  username: wedora  
  password: 123456  
  database: wedora  
  
mqtt :  
  server : 127.0.0.1:1883  
  
redis:  
  server : 127.0.0.1
```

2.2.6 Configuration of Mysql database

```
#cat /etc/my.cnf
```

```
[root@localhost ~]# cat /etc/my.cnf  
[mysqld]  
basedir=/usr/local/mysql  
datadir=/usr/local/mysql/data  
port=3306  
sql_mode=STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_F  
UTION  
socket=/var/lib/mysql/mysql.sock  
user=mysql  
# Disabling symbolic-links is recommended to prevent assorted secu  
symbolic-links=0  
  
[mysqld_safe]  
log-error=/var/log/mysql.log  
pid-file=/var/run/mysqld/mysqld.pid  
[root@localhost ~]#
```

2.2.7 Configuration of Mongo database

```
#cd /home/wedora/server/conf
```

```
#ll (check the list of files)
```

```
[wedora@localhost conf]$ ll
total 12
-rwxrwxrwx. 1 wedora hongdian  151 Jul 26 14:11 conf.ini
-rwxrwxrwx. 1 wedora hongdian   416 Jul 26 14:19 mongodb.conf
-rwxrwxrwx. 1 wedora hongdian 1271 Jul 26 14:19 redis.conf
[wedora@localhost conf]$
```

```
#cat mongondb.conf //View mongondb the information of configuration and port
```

```
[wedora@localhost conf]$ cat mongondb.conf
dbpath=/home/wedora/server/tripartite/mongo/data #数据文件存放目录
logpath=/home/wedora/server/tripartite/mongo/logs/mongondb.log #日志文件存放目录
port=27017 #端口
fork=true #以守护程序的方式启用，即在后台运行
nohttpinterface=true
logappend=true
directoryperdb=true
storageEngine=wiredTiger
wiredTigerCacheSizeGB=1
unixSocketPrefix=/home/wedora/server/tripartite/mongo/[wedora@localhost conf]$
```

2.2.8 Configuration of Redis database

```
#cd /home/wedora/server/conf
```

```
#cat redis.conf // View the configuration of redis
```

```
[wedora@localhost conf]$ cat redis.conf
daemonize yes
pidfile /home/wedora/server/tripartite/redis/pid/redis_6379.pid
port 6379
bind 0.0.0.0
timeout 1800
logfile /home/wedora/server/tripartite/redis/log/redis_6379.log
dbfilename dump.rdb
dir /home/wedora/server/tripartite/redis/6379
maxclients 10000

tcp-backlog 511
tcp-keepalive 0
loglevel notice
databases 16
save 900 1
save 300 10
```


2.2.9 The configuration of ipfix service

```
#cd /home/wedora/server/ipfix
#ll
```

```
[root@localhost ipfix]# ll
total 22128
drwxr-xr-x. 2 root root    67 Dec 26 11:06 __pycache__
-rwxrwxrwx. 1 root root  2682 Dec 24 09:28 cli.py
-rwxrwxrwx. 1 root root   216 Dec 24 09:28 config.ini
-rwxrwxrwx. 1 root root  1203 Dec 24 09:28 constants.py
-rwxrwxrwx. 1 root root   200 Dec 24 09:28 ipfix_shutdown.sh
-rwxrwxrwx. 1 root root   499 Dec 24 09:28 ipfix_startup.sh
-rwxrwxrwx. 1 root root  2728 Dec 24 09:28 ipfixserver.py
drwxr-xr-x. 2 root root    21 Dec 26 11:06 log
-rwxrwxrwx. 1 root root   563 Dec 24 09:28 logging.conf
-rwxrwxrwx. 1 root root    96 Dec 26 11:06 nohup.out
-rwxrwxrwx. 1 root root 22580749 Dec 24 09:28 python3.tgz
-rwxrwxrwx. 1 root root   300 Dec 24 09:28 readme.txt
-rwxrwxrwx. 1 root root    46 Dec 24 09:28 requirements.txt
-rwxrwxrwx. 1 root root  20156 Dec 24 09:28 rfc5103a.iespec
-rwxrwxrwx. 1 root root   8094 Dec 24 09:28 server.py
-rwxrwxrwx. 1 root root   1241 Dec 24 09:28 setup.sh
-rwxrwxrwx. 1 root root   266 Dec 24 09:28 test.py
[root@localhost ipfix]#
```

```
#cat config.ini
```

```
[root@localhost ipfix]# cat config.ini
[mongodbg]
uri=mongodb://wedora:123456@127.0.0.1:27017/?authSource=admin&authMechanism=SCRAM-SHA-1
database=wedora

[udpconfig]
host=0.0.0.0
port=41803
hassn=True

[server]
port=8888

[logging]
level=INFO[root@localhost ipfix]#
```

Native mongo database

External port corresponding to the service

2.2.10 The Configuration of Mqtt server

```
# cd /home/wedora/server/tripartite/mqtt/etc
# ll
```

```
[wedora@localhost etc]$ ll
total 32
-rwxrwxrwx. 1 wedora hongdian  801 Aug 13 2017 acl.conf
drwxrwxrwx. 2 wedora hongdian 4096 Aug 13 2017 certs
-rwxrwxrwx. 1 wedora hongdian 16305 Aug 13 2017 emq.conf
drwxrwxrwx. 2 wedora hongdian 4096 Aug 13 2017 lwml2m_xml
drwxrwxrwx. 2 wedora hongdian 4096 Aug 13 2017 plugins
```

cat emq.conf //The port used by the service is the same as that configured by wedora_server and fep_wmmp_fep_mqtt.

```
## MQTT Listeners
##-----
##-----
## External TCP Listener

## External TCP Listener: 1883, 127.0.0.1:1883, ::1:1883
listener.tcp.external = 0.0.0.0:1883

## Size of acceptor pool
listener.tcp.external.acceptors = 16

## Maximum number of concurrent clients
listener.tcp.external.max_clients = 102400
```

2.3 commonly used methods for restarting services

2.3.1 Mysql

Out of service:

```
#service mysql stop
```

Open the service:

```
#service mysql start
```

Restart the service

```
#service mysql restart
```

```
[root@localhost bin]# service mysqld stop
Stopping mysqld: [ OK ]
[root@localhost bin]# service mysqld start
Starting mysqld: [ OK ]
[root@localhost bin]#
```

PS:the mysql user uses the super user root to perform the restart, and other services need to be switched to the wedora user to execute

2.3.2 Mongo

Out of service:

```
#ps -ef|grep mongo
```

```
#kill -15 {PID}
```

```
[wedora@localhost ~]$ ps -ef|grep mongo
wedora  31925      1  1 11:08 ?          00:00:10 /home/wedora/server/tripartite/mongo/bin/mongod
er/conf/mongodb.conf --auth
wedora  57336 57198  0 11:19 pts/6      00:00:00 grep mongo
[wedora@localhost ~]$ kill -9 31925
```

Open the service:

```
# /home/wedora/server/tripartite/mongo/bin/mongod --config
/home/wedora/server/conf/mongodb.conf --auth
#ps -ef|grep mongo
```

```
[wedora@localhost ~]$ kill -9 31925
[wedora@localhost ~]$ ps -ef|grep mongo
wedora  57847 57198  0 11:21 pts/6      00:00:00 grep mongo
[wedora@localhost ~]$ /home/wedora/server/tripartite/mongo/bin/mongod --config /home/wedora/server/
h
about to fork child process, waiting until server is ready for connections.
forked process: 57874
child process started successfully, parent exiting
[wedora@localhost ~]$
```

2.3.3 Redis

```
#ps -ef|grep redis
Out of service:
#service redis stop
Open the service:
#service redis start
```

```
[wedora@localhost root]$ ps -ef|grep redis
wedora  4968      1  0 15:52 ?          00:00:07 /home/wedora/server/tripartite/redis

wedora  43108 43046  0 17:09 pts/6      00:00:00 grep redis
[wedora@localhost root]$ kill -9 4968
[wedora@localhost root]$ ps -ef|grep redis
wedora  43351 43046  0 17:09 pts/6      00:00:00 grep redis
[wedora@localhost root]$
```

2.3.4 Mqtt

```
Out of service:
#ps -ef|grep mqtt
#kill -9 {PID}
#ps -ef|grep mqtt
```

```

[wedora@localhost ~]$ ps -ef|grep mqtt
wedora 32025 1 0 11:08 ? 00:00:00 /home/wedora/server/tripartite/mqtt/erts-9.0/bin/epmd -
wedora 32032 1 0 11:08 ? 00:00:00 /home/wedora/server/tripartite/mqtt/erts-9.0/bin/run_erl
ra_erl_pipes/emq@127.0.0.1/ /home/wedora/server/tripartite/mqtt/log exec "/home/wedora/server/tripartite
"console"
wedora 32034 32032 0 11:08 pts/5 00:00:03 /home/wedora/server/tripartite/mqtt/erts-9.0/bin/beam.s
Q 65536 -P 256000 -A 32 -K true -zdbbl 32768 -- -root /home/wedora/server/tripartite/mqtt -prognam
partite/mqtt/bin/emqttd -- -home /home/wedora -- -boot /home/wedora/server/tripartite/mqtt/releases/2.3
ded -boot_var ERTS_LIB_DIR /home/wedora/server/tripartite/mqtt/erts-9.0/./lib -mnesia dir "/home/wedor
/mqtt/data/mnesia/emq@127.0.0.1" -config /home/wedora/server/tripartite/mqtt/data/configs/app.2018.07.3
kernel net_ticktime 60 -smp auto -setcookie emqsecretcookie -name emq@127.0.0.1 -vm_args /home/wedora/s
tt/data/configs/vm.2018.07.31.11.08.28.args -- console
wedora 32348 32299 0 11:08 ? 00:00:00 /home/wedora/server/tripartite/mqtt/lib/os_mon-2.4.2/pr
wedora 32349 32299 0 11:08 ? 00:00:00 /home/wedora/server/tripartite/mqtt/lib/os_mon-2.4.2/pr
wedora 32382 1 0 11:08 pts/4 00:00:00 /bin/sh /home/wedora/server/fep_mqtt/fep_mqtt_startup.s
wedora 32455 1 3 11:08 ? 00:00:25 /home/wedora/server/fep_mqtt/fep_mqtt fep -n fep.mqtt
wedora 58125 57198 0 11:22 pts/6 00:00:00 grep mqtt
[wedora@localhost ~]$ kill -9 32025
[wedora@localhost ~]$ kill -9 32032

```

You can see that the mqtt process is gone, but there may still be programs waiting for port 1883, so you need to wait for a while, port 1883 will be completely released, mqtt can be restarted successfully.

Open the service:

```
#cd /home/wedora/server/tripartite/mqtt/bin
```

```
./emqttd start
```

```
#ps -ef|grep mqtt
```

```

[wedora@localhost ~]$ cd /home/wedora/server/tripartite/mqtt/bin
[wedora@localhost bin]$ ./emqttd start
emqttd 2.3 is started successfully!
[wedora@localhost bin]$ ps -ef|grep mqtt
wedora 32382 1 0 11:08 pts/4 00:00:00 /bin/sh /home/wedora/server/fep_mqtt/fep_mqtt_sta
wedora 32455 1 2 11:08 ? 00:00:26 /home/wedora/server/fep_mqtt/fep_mqtt fep -n fep.
wedora 59061 1 0 11:24 ? 00:00:00 /home/wedora/server/tripartite/mqtt/erts-9.0/bin/
wedora 59068 1 0 11:24 ? 00:00:00 /home/wedora/server/tripartite/mqtt/erts-9.0/bin/
ra_erl_pipes/emq@127.0.0.1/ /home/wedora/server/tripartite/mqtt/log exec "/home/wedora/server/tri
"console"
wedora 59071 59068 17 11:24 pts/5 00:00:02 /home/wedora/server/tripartite/mqtt/erts-9.0/bin/
Q 65536 -P 256000 -A 32 -K true -zdbbl 32768 -- -root /home/wedora/server/tripartite/mqtt -prognam
partite/mqtt/bin/emqttd -- -home /home/wedora -- -boot /home/wedora/server/tripartite/mqtt/releas
ded -boot_var ERTS_LIB_DIR /home/wedora/server/tripartite/mqtt/erts-9.0/./lib -mnesia dir "/home
/mqtt/data/mnesia/emq@127.0.0.1" -config /home/wedora/server/tripartite/mqtt/data/configs/app.201
kernel net_ticktime 60 -smp auto -setcookie emqsecretcookie -name emq@127.0.0.1 -vm_args /home/we
tt/data/configs/vm.2018.07.31.11.24.59.args -- console
wedora 59387 59358 0 11:25 ? 00:00:00 /home/wedora/server/tripartite/mqtt/lib/os_mon-2.
wedora 59388 59358 0 11:25 ? 00:00:00 /home/wedora/server/tripartite/mqtt/lib/os_mon-2.
wedora 59460 57198 0 11:25 pts/6 00:00:00 grep mqtt
[wedora@localhost bin]$

```

At this point, check that the mqtt process has started, and the service is successfully started.

2.3.5 Wedora_server

Out of the the service:

```
#pm2 stop wedora
```

```
[wedora@localhost bin]$ pm2 stop wedora
[PM2] Applying action stopProcessId on app [wedora](ids: 0)
[PM2] [wedora](0) ✓
```

App name	id	mode	pid	status	restart	uptime	cpu	mem	user	watching
wedora	0	fork	0	stopped	0	0	0%	0 B	wedora	disabled

```
Use `pm2 show <id|name>` to get more details about an app
[wedora@localhost bin]$
```

Open the service:

#pm2 start wedora

```
wedora 30388 58215 0 12:49 pts/3 00:00:00 grep redis
[wedora@localhost bin]$ pm2 start wedora
[PM2] Applying action restartProcessId on app [wedora](ids: 0)
[PM2] [wedora](0) ✓
[PM2] Process successfully started
```

App name	id	mode	pid	status	restart	uptime	cpu	mem	user	watching
wedora	0	fork	30400	online	0	0s	0%	6.2 MB	wedora	disabled

```
Use `pm2 show <id|name>` to get more details about an app
[wedora@localhost bin]$
```

View related files:

#pm2 list

```
[wedora@localhost bin]$ pm2 list
```

App name	id	mode	pid	status	restart	uptime	cpu	mem	user	watching
wedora	0	fork	30435	online	0	1s	114.8%	105.8 MB	wedora	disabled

```
Use `pm2 show <id|name>` to get more details about an app
[wedora@localhost bin]$
```

2.3.6 Wse

#ps -ef | grep java

```
[root@localhost ~]# ps -ef | grep java
root 15622 1 0 Nov13 ? 00:07:06 java -jar -Xms2048m -Xmx4096m /usr/local/wedora/wse/wedora-server-engine.jar --spring.config.location=/usr/local/wedora/wse/config/application.yml
root 50267 49491 0 10:50 pts/0 00:00:00 grep --color=auto java
```

out of the service:

#kill -9 {pid}

Open the service:

#cd /home/wedora/server/wse/

#nohup java -jar -Xms2048m -Xmx4096m /home/wedora/server/wse/wedora-server-engine.jar

--spring.config.location=/home/wedora/server/wse/config/application.yml &

2.3.7 Fep.wmmp

Out of the service:

#ps -ef | grep fep.wmmp

#kill -9 {PID1} (In the legend, the 102653 script process is stopped first.)

#kiil -9 {PID2} (Then stop the 104920 service)

#ps -ef | grep fep.wmmp

```
[wedora@localhost fep_wmmp]$ ps -ef|grep fep.wmmp
wedora 61827 1 0 11:29 pts/4 00:00:00 /bin/sh /home/wedora/server/fep_wmmp/fep_wmmp_startup.sh
wedora 61882 1 0 11:29 ? 00:00:00 /home/wedora/server/fep_wmmp/fep.wmmp fep -n fep.wmmp
wedora 62855 57198 0 11:32 pts/6 00:00:00 grep fep.wmmp
[wedora@localhost fep_wmmp]$ kill -9 61827
[wedora@localhost fep_wmmp]$ kill -9 61882
[wedora@localhost fep_wmmp]$ ps -ef|grep fep.wmmp
wedora 63190 57198 0 11:33 pts/6 00:00:00 grep fep.wmmp
[wedora@localhost fep_wmmp]$
```

The related process of Fep.wmmp is gone, and it proves that it has stopped successfully.

open the service:

```
#cd /home/wedora/server/fep_wmmp
```

```
#. /fep_wmmp_startup.sh
```

```
#ps -ef|grep fep.wmmp
```

```
[wedora@localhost fep_wmmp]$ ./fep_wmmp_startup.sh
/home/wedora/server/fep_wmmp
fep.wmmp: no process killed
^C
[wedora@localhost fep_wmmp]$ ps -ef|grep fep.wmmp
wedora 63385 1 0 11:34 ? 00:00:00 /home/wedora/server/fep_wmmp/fep.wmmp fep -n fep.wmmp
wedora 63442 57198 0 11:34 pts/6 00:00:00 grep fep.wmmp
[wedora@localhost fep_wmmp]$
```

2.3.8 Ipfix

Out of the service:

```
#ps -ef | grep ipfix
```

```
#kill -9 {pid1} //2486 script process as shown
```

```
#kill -9 {pid2} //2535 service process as shown
```

```
[root@localhost ipfix]# ps -ef | grep ipfix
root 2486 1 0 Dec26 ? 00:00:35 /bin/sh /home/wedora/server/ipfix/ipfix_startup.sh
root 2535 2486 0 Dec26 ? 00:01:16 python3 /home/wedora/server/ipfix/ipfix_server.py
root 55098 18767 0 11:20 pts/1 00:00:00 grep --color=auto ipfix
[root@localhost ipfix]#
```

Use a script to stop:

```
#cd /home/wedora/server/ipfix
```

```
#sh ipfix_shutdown.sh
```

Open the service:

```
#cd /home/wedora/server/ipfix
```

```
#sh ipfix_startup.sh
```



Connecting Things

Contact US

Hongdian Corporation

Add: Tower A, Hongdian Building, 100 Huabao Road, Pinghu, Longgang District, Shenzhen, China

Tel: +86-755-88864288-2

Fax: +86-755-83404677

E-mail: Sales@hongdian.com