

永洪 BI 系统 安装环境检查手册

目录

1. 确认 OS 版本	3
2. 确认已创建好如下文件系统	3
3. 服务器系统时间需一致	4
4. 设置 ulimit 参数	4
5. 防火墙设置	4
6. 创建用户、组	5
7. 检查语言环境	6
8. 操作系统字体	7
9. jdk 安装	8
10. 中文字体安装	8
11. 服务器 chrome 浏览器安装	10
12. tomcat 最大连接数	10

1. 确认 OS 版本

查看OS版本 #cat /proc/version

```
Last login: Sun Jul 12 10:37:32 2020
[root@localhost ~]# cat /proc/version
Linux version 3.10.0-693.el7.x86_64 (builder@builder.dev.centos.org) (gcc version 4.8.5 20150623 (Red Hat 4.8.5-16) (GCC) ) #1 SMP Tue Aug 22 21:09:27 UTC 2017
```

注：由于定制组件导出需要调用 chrome 浏览器插件，服务器上需要安装 chrome 浏览器，推荐服务器 linux 内核 3.10 及以上。

2. 确认已创建好如下文件系统

2.1 查看文件系统类型

df -T

```
[root@localhost ~]# df -T
文件系统          类型          1K-块    已用    可用  已用%  挂载点
/dev/mapper/centos-root xfs    17811456 17357992 453464   98% /
devtmpfs          devtmpfs     488792      0 488792    0% /dev
tmpfs             tmpfs       499848      0 499848    0% /dev/shm
tmpfs             tmpfs       499848    6896 492952    2% /run
tmpfs             tmpfs       499848      0 499848    0% /sys/fs/cgroup
/dev/sda1         xfs    1038336 127452 910884   13% /boot
tmpfs             tmpfs       99972      0 99972    0% /run/user/0
```

注：搭建集群环境，需要确保安装永洪 BI 的盘符文件系统类型一致

2.2 查看文件系统名称及大小 (Size)

新环境搭建要看文件系统利用率 (Used%)，不超过1%确保环境上不存在其它文件,建议永洪安装在磁盘空间最大的盘符上，推荐1T以上。

#df -h

```
[root@localhost ~]# df -h
文件系统          容量  已用  可用  已用%  挂载点
/dev/mapper/centos-root 17G   17G   443M   98% /
devtmpfs          478M    0   478M    0% /dev
tmpfs             489M    0   489M    0% /dev/shm
tmpfs             489M  6.8M   482M    2% /run
tmpfs             489M    0   489M    0% /sys/fs/cgroup
/dev/sda1         1014M 125M   890M   13% /boot
tmpfs             98M    0    98M    0% /run/user/0
```

2.3 查看分区节点编号使用情况，

新环境搭建要看分区节点编号使用情况，确保安装路径的 inode 编号利用率不超过 1%。

#df -i

```
[root@localhost ~]# df -i
文件系统          Inode 已用(I) 可用(I) 已用(I)% 挂载点
/dev/mapper/centos-root 1038120 130859 907261    13% /
devtmpfs          122198    387 121811    1% /dev
tmpfs             124962     1 124961    1% /dev/shm
tmpfs             124962    507 124455    1% /run
tmpfs             124962     16 124946    1% /sys/fs/cgroup
/dev/sda1         524288    327 523961    1% /boot
tmpfs             124962     1 124961    1% /run/user/0
```

3. 服务器系统时间需一致

查看系统时间#date

```
[root@localhost ~]# date
2020年 07月 12日 星期日 11:21:29 CST
```

注：搭建集群环境，需要确保每台服务器系统时间一致，若是数据库系统还得跟数据库服务器保持一致。

4. 设置 ulimit 参数

1. 查看文件 `more /etc/security/limits.conf` 是否存在下面参数，如果没有则增加

```
#vi /etc/security/limits.conf
*          soft  nofile   65536
*          hard   nofile   65536
```

2. 退出当前账户，重新登录，改参数服务器重启生效。

5. 防火墙设置

1. 查看防火墙状态（截图以 centos7 为例，系统不一样可能命令不一样）

```
#systemctl status firewalld.service    #查看防火墙状态
```

#systemctl stop firewalld.service #关闭防火墙

#systemctl disable firewalld.service #禁止 firewall 开机启动

```
[root@localhost ~]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
   Active: inactive (dead)
     Docs: man:firewalld(1)
```

2.不关防火墙则需要开通相应的端口：

包含 Tomcat 端口 (2 个, 默认 8080,8005)、Zookeeper 端口 (3 个,2181,2888,3888)、集群通信端口 (2 个, 默认 5066 5083)

例 (以 centos7 为例)：

开放端口

```
firewall-cmd --permanent --add-port=8080/tcp
firewall-cmd --permanent --add-port=8005/tcp
firewall-cmd --permanent --add-port=2181/tcp
firewall-cmd --permanent --add-port=2888/tcp
firewall-cmd --permanent --add-port=3888/tcp
firewall-cmd --permanent --add-port=5066/tcp
firewall-cmd --permanent --add-port=5083/tcp
```

#重启防火墙(修改配置后要重启防火墙)

```
firewall-cmd --reload
```

6. 创建用户、组

创建组：groupadd -g 1200 yhuser

创建用户：useradd -u 1200 -g yhuser -d /yhuser yhuser

修改用户密码：passwd yhuser

```
[root@localhost ~]# groupadd -g 1200 yhuser
[root@localhost ~]# useradd -u 1200 -g yhuser -d /yhuser yhuser
[root@localhost ~]# passwd yhuser
更改用户 yhuser 的密码。
新的 密码：
无效的密码： 密码少于 8 个字符
重新输入新的 密码：
passwd: 所有的身份验证令牌已经成功更新。
```

(注: 查询组 #cat /etc/group

查询用户 #cat /etc/passwd

删除组 #groupdel yhuser

删除用户 #userdel yhuser)

设置目录权限: chmod 777 /yhuser

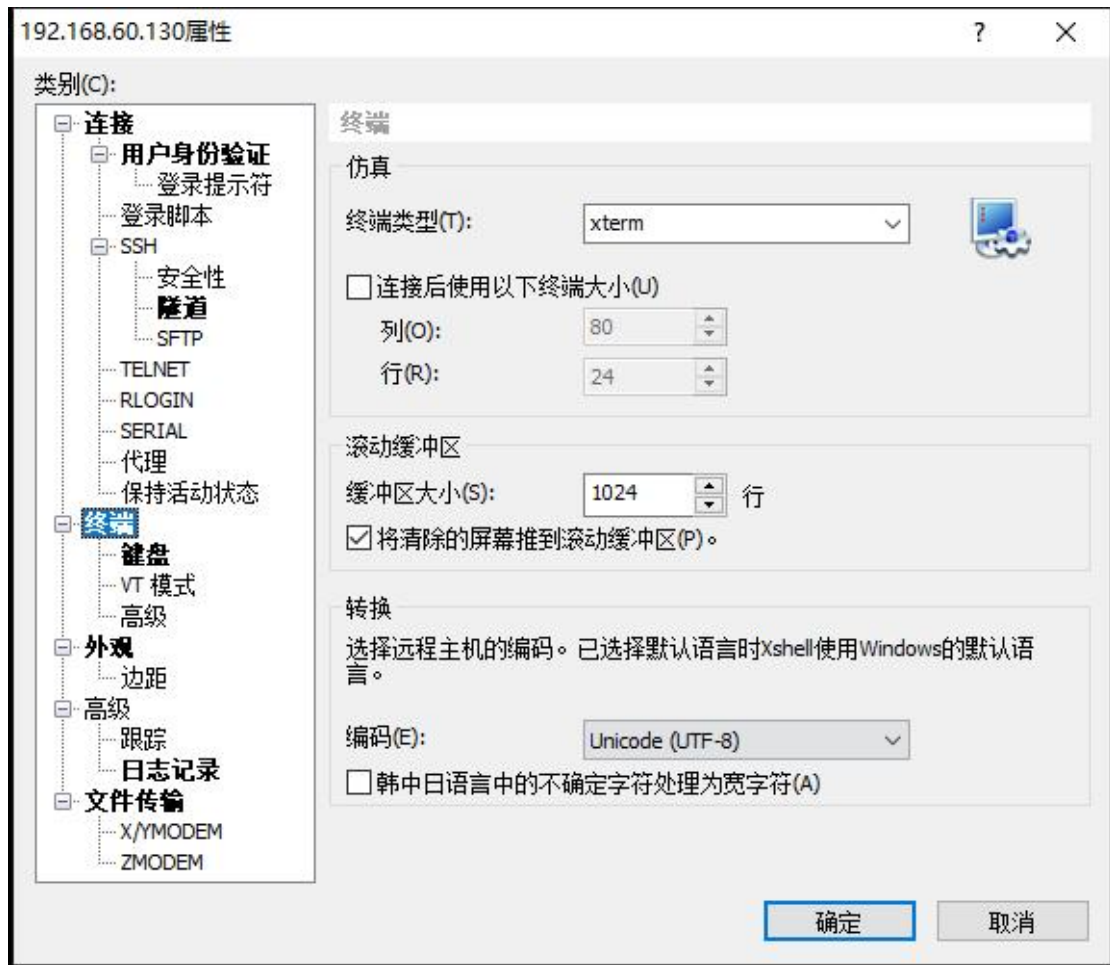
如果搭建集群, 集群中所有主机上的用户 UID, GID 都要相同。

7. 检查语言环境

```
#echo $LANG
```

```
#locale
```

```
[root@localhost ~]# echo $LANG
zh_CN.UTF-8
[root@localhost ~]# locale
LANG=zh_CN.UTF-8
LC_CTYPE="zh_CN.UTF-8"
LC_NUMERIC="zh_CN.UTF-8"
LC_TIME="zh_CN.UTF-8"
LC_COLLATE="zh_CN.UTF-8"
LC_MONETARY="zh_CN.UTF-8"
LC_MESSAGES="zh_CN.UTF-8"
LC_PAPER="zh_CN.UTF-8"
LC_NAME="zh_CN.UTF-8"
LC_ADDRESS="zh_CN.UTF-8"
LC_TELEPHONE="zh_CN.UTF-8"
LC_MEASUREMENT="zh_CN.UTF-8"
LC_IDENTIFICATION="zh_CN.UTF-8"
LC_ALL=
```



注:若连接工具修改编码后文件仍显示乱码,则需注意检查系统语言。

8. 操作系统字体

1. 检查操作系统是否安装字体

```
#cd /usr/share
```

```
#ls
```

```
[root@localhost share]# pwd
/usr/share
[root@localhost share]# ls
aclocal      backgrounds  doc          games        glib-2.0    hwdata      licenses    omf           redhat-release  tuned       zsh
alsa         bash-completion  egl         gcc-4.8.2   glynd       i18n        locale     open-vm-tools  selinux        vulkan
anaconda    centos-logos    emacs       gcc-4.8.5   gnome       icons       lua         os-prober      sounds        wallpapers
appdata     centos-release  empty       GConf        gnome-background-properties  idl         magic       pll-kit       systemd        XML
application-registry  cracklib     file        gdb          gnome-control-center  info       nan         pixmaps       systemtap     xml
applications  dbus-1         firewallld  GeoIP        gnupg       kde4        mime       pkgconfig     tabset        xsession
augeas      defaults       firstboot   gettext      groff       kdump      mime-info  pki           terminfo      yum-cli
authconfig  desktop-directories  fontconfig  gettext-0.19.8  grub       libdrm     misc       plymouth      themes        yum-plugins
awk         dict           fonts       ghostscript  gtk-3.0    libthai    mysql     polkit-1     thumbnails    zoneinfo
```

2. 安装服务器字体

(1)#yum install fontconfig(centOS)

(2)#sudo apt-get -y install fontconfig(ubuntu)

9. jdk 安装

1. 解压 tar 包到 yhuser

```
tar xzfjdk-11.0.2_linux-x64_bin.tar.gz -C /yhuser
```

```
[yhuser@localhost yhuser]$ cd package/  
[yhuser@localhost package]$ ls  
jdk-11.0.2_linux-x64_bin.tar.gz  Yonghong Z-Suite V8.5.sh  
[yhuser@localhost package]$ tar xzfjdk-11.0.2_linux-x64_bin.tar.gz -C /yhuser
```

2. 配置环境变量并使环境变量生效

vi /etc/profile, 添加变量, source /etc/profile 使jdk 环境变量生效

```
[root@localhost 10.1]# vim /etc/profile  
[root@localhost 10.1]# source /etc/profile
```

```
export JAVA_HOME=/home/jdk-11.0.7  
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar  
export PATH=$JAVA_HOME/bin:$PATH
```

3. 查看 jdk 版本, 输入 java -version

```
[root@localhost 10.1]# java -version  
java version "11.0.7" 2020-04-14 LTS  
Java(TM) SE Runtime Environment 18.9 (build 11.0.7+8-LTS)  
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.7+8-LTS, mixed mode)
```

注: 版本要求jdk9 以上, 优先推荐jdk11, 服务器系统最好不要使用openjdk (使用openjdk

有可能会因为服务器缺少字体库而导致功能异常)

10. 中文字体安装

1. 服务器/usr/share/fonts 目录下要放入中文字体 (截图中这里放的是微软雅黑字体)。


```
[root@localhost fonts]# cd /usr/share/fonts/
[root@localhost fonts]# pwd
/usr/share/fonts
[root@localhost fonts]# ll
总用量 99492
drwxr-xr-x. 2 root root    4096 1月  9 2020 1
-rw-r--r--. 1 root root  1613668 1月  9 2020 calibril.ttf
-rw-r--r--. 1 root root  1196672 1月  9 2020 calibril.ttf
-rw-r--r--. 1 root root  1061248 1月  9 2020 calibril.ttf
-rw-r--r--. 1 root root  1438808 1月  9 2020 calibril.ttf
-rw-r--r--. 1 root root  1648120 1月  9 2020 calibril.ttf
-rw-r--r--. 1 root root  1159032 1月  9 2020 calibril.ttf
-rw-r--r--. 1 root root   397896 1月  9 2020 consolab.ttf
-rw-r--r--. 1 root root   467584 1月  9 2020 consolai.ttf
-rw-r--r--. 1 root root   459180 1月  9 2020 consola.ttf
-rw-r--r--. 1 root root   407956 1月  9 2020 consolaz.ttf
-rw-r--r--. 1 root root   451824 1月  9 2020 constanb.ttf
-rw-r--r--. 1 root root   448748 1月  9 2020 constani.ttf
-rw-r--r--. 1 root root   448376 1月  9 2020 constan.ttf
-rw-r--r--. 1 root root   455564 1月  9 2020 constanz.ttf
drwxr-xr-x. 2 root root    4096 7月 18 2019 dejavu
drwxr-xr-x. 2 root root    4096 4月  8 12:59 liberation
-rw-r--r--. 1 root root  21663376 7月 12 12:07 msjh.ttf
-rw-r--r--. 1 root root  21767952 7月 12 12:07 msyh1.ttf
-rw-r--r--. 1 root root  14602860 7月 12 12:07 msyhbd.ttf
-rw-r--r--. 1 root root  15067744 7月 12 12:07 msyh.ttf
-rw-r--r--. 1 root root  17064180 1月  9 2020 simsunb.ttf
[root@localhost fonts]#
```

2.永洪所使用的的 jdk 路径/lib/fonts 目录下也要放入中文字体，没有就新建该目录。

```
[root@localhost fonts]# cd /usr/java/jdk-11.0.2/lib/fonts/
[root@localhost fonts]# ll
总用量 71396
-rw-r--r--. 1 root root  21663376 7月 12 12:11 msjh.ttf
-rw-r--r--. 1 root root  21767952 7月 12 12:11 msyh1.ttf
-rw-r--r--. 1 root root  14602860 7月 12 12:11 msyhbd.ttf
-rw-r--r--. 1 root root  15067744 7月 12 12:11 msyh.ttf
[root@localhost fonts]#
```

3.检查环境是否安装中文字体。

#fc-list :lang=zh

```
[root@localhost fonts]# fc-list :lang=zh
/usr/share/fonts/msyh1.ttf: Microsoft YaHei:style=Regular,Normal
/usr/share/fonts/msyhbd.ttf: Microsoft YaHei:style=Bold,Félkövér
/usr/share/fonts/msyh.ttf: Microsoft YaHei:style=Regular
/usr/share/fonts/msjh.ttf: Microsoft JhengHei,微软正黑體:style=Regular
[root@localhost fonts]#
```

11.服务器 chrome 浏览器安装

root 用户执行 `yum install`

https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm 安装最新的

chrome。

```
[root@localhost fontz]# yum install https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm
已加载插件: fastestmirror
google-chrome-stable_current_x86_64.rpm
正在检查 /var/tmp/yum-root-85301j/google-chrome-stable_current_x86_64.rpm: google-chrome-stable-83.0.4103.116-1.x86_64
/var/tmp/yum-root-85301j/google-chrome-stable_current_x86_64.rpm 将作为 google-chrome-stable-81.0.4044.92-1.x86_64 的更新
正在解决依赖关系
--> 正在检查事务
--> 软件包 google-chrome-stable-83.0.4103.116-1 将被 升级
--> 软件包 google-chrome-stable-81.0.4044.92-1 将被 更新
--> 解决依赖关系完成

依赖关系解决

-----
Package                架构          版本          源                大小
-----
正在更新:
google-chrome-stable   x86_64        83.0.4103.116-1 /google-chrome-stable_current_x86_64 225 M
-----
事务概要
-----
升级 1 软件包
总计: 225 M
Is this ok [y/d/n]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
 正在更新 : google-chrome-stable-83.0.4103.116-1.x86_64 1/2
Redirecting to /bin/systemctl start atd.service
Failed to start atd.service: Unit not found.
 清理     : google-chrome-stable-81.0.4044.92-1.x86_64 2/2
 验证中   : google-chrome-stable-83.0.4103.116-1.x86_64 1/2
 验证中   : google-chrome-stable-81.0.4044.92-1.x86_64 2/2
```

注: 安装 chrome 不一定只能用这个命令, 别的方式可以安装上也是可以的, 比如 `yum install google-chrome-stable --nogpgcheck` 或者其他方式。

12.tomcat 最大连接数(产品安装后)

`tmcat/conf/server.xml` 这个文件, `acceptCount` 和 `maxThreads` 默认值是 200 和 100, 需要改大。

```
<Connector port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" acceptCount="500" maxThreads="400" />
```