汉水大数据操作系统企业版 7.0 快速入门

本文宗旨: 本文为帮助新手从零开始在虚拟机中安装部署伪分布式 hadoop 大数据集群, 便于学习大数据开发的相关技术。不适合在企业级环境中应用,如需要部署分布式企业级 hadoop 集群请跟产品销售联系,联系方式: 021-52682238。

版权申明:本文档采用开放文档授权协议进行发布,任何个人和组织都可以自由修改和传播这个文档,如果您在这个文档的基础上进行了更新或者有修订请求,请联系文档的维护 人员:allen@hanwater.com,便于从软件产品网站上更新发布。

本文档的官方发布站点为 http://hanwate.com/docs

文档更新记录:

文档版本	时间	作者/修订者
1.0	2018-01-26	张一峰/上海汉水信息技术服务有限公司
2.0	2018-03-23	张一峰/上海汉水信息技术服务有限公司

一、VMware 虚拟机的安装准备

请进入电脑开机主板 BIOS 设置页面,激活 CPU 虚拟化技术。

二、Hanwate 操作系统在 VMware 虚拟机中的安装

一、虚拟机的创建及安装前准备

1. 新建虚拟机

点击"创建新的虚拟机"选项进行新虚拟机的创建

		WORKSTAT	ION ⁻ 12 PRO	6	
	会議業新的成初	17开 虚拟机	(1) 连接近程服务器	遙絶到 VMware vCloud Air	
		5.击"创建新的虚拟机" ¹⁵ 项			
vmware					

在"新建虚拟机向导"中选择"自定义"选项(通过自定义可以选择 VMware 版本),然后点击"下一步"



2.选择版本

保持默认配置就好,无需改动,点击"下一步"继续

显拟机硬件兼容性 硬件兼容性(出):	Workstation	12.0 -	
兼容:	ESX Serve	r(<u>S</u>)	
兼容产品:		限制:	
Fusion 8.x Workstation 12.0	*	64 GB 内存 16 个处理器 10 个网络适配器 8 TB 磁盘大小	*
	-	•	Ŧ

3. 安装客户机操作系统

选择"稍后安装操作系统",安装一块空白硬盘,并点击"下一步"继续

装来源:	
○安装程序光盘(□):	
无可用驱动器	*
◎ 安装程序光盘映像文1 F:\HanWate\Hanwa	牛(iso)(<u>M</u>): ate_Bigdata_OS_7_Firefly-x86_64.iso → 浏览(<u>R</u>)
② 安装程序光盘映像文付 F:\HanWate\Hanwa	牛(iso)(<u>M</u>): ate_Bigdata_OS_7_Firefly-x86_64.iso ▼ 浏览(R)
 ○ 安装程序光盘映像文付 F:\HanWate\Hanwa 第:后安装操作系统(S) 	(牛(iso)(<u>M</u>): ate_Bigdata_OS_7_Firefly-x86_64.iso ▼ 浏览(R)
 ● 安装程序光盘映像文件 F:\HanWate\Hanwa ● 稍后安装操作系统(Si ●创建的虚拟机将包含 	件(iso)(<u>M</u>): ate_Bigdata_OS_7_Firefly-x86_64.iso ▼ 浏览(R))。 一个空白硬盘。

4. 选择客户机操作系统

在此配置项中选择"Linux"以及"CentOS 64 位"选项,并点击"下一步"继续

所建虚拟机向导	1		0	X
选择客户机操作系统 此虚拟机中将安装	哪种操作系统?			
客户机操作系统 ○ Microsoft Windows(<u>W</u> ④ Linux(<u>L</u>) ○ Novell NetWare(<u>E</u>) ○ Solaris(<u>S</u>) ○ VMware ESX(<u>X</u>) ○ 其他(<u>O</u>)	0	选择Lin 类型	ux操作系统	
版本(⊻)				
CentOS 64 位				•
	▲ 在版本中 继续	选择″Cent(OS 64位″并点	∶击″下一步″
帮助	<	上一步(B)	下一步(N) >	取消

5. 命名虚拟机

为虚拟机设置名称并设置虚拟机的物理位置,任意设置即可 由于虚拟机占用一定空间,最好是不要按照默认安装在 C 盘: 设置完毕后点击"下一步"即可

命名虚拟机	公告的现在分词 优美的支
您要为此虚拟机使用什么名称?	结虚拟机起名子,任息即可
虚拟机名称(⊻):	
Hanwate Linux	
位置(L):	
D:\Documents\Virtual Machines\Hanwate	Linux 浏览(R)
	\mathbf{A}
	点击"浏览"修改虚拟机(物理位置,任意修改即可 并"点击"下一步继续
	点击"浏览"修改虚拟机的 物理位置,任意修改即可 并"点击"下一步继续
	点击"浏览"修改虚拟机组物理位置,任意修改即可并"点击"下一步继续

6. 处理器配置

保持默认配置就好,无需改动,点击"下一步"继续

从1田駅東1里	1.0
为此虚拟机指定处理器数	敗重。
处理器	
处理器数量(P):	1
每个处理器的核心数量(C):	1
总处理器核心数里:	1
	\ 尤特殊要求,保持默认就好,并点击" 下一步"继续
	\ 尤特殊要求,保持默认就好,并点击" 下一步"继续
	、尤特殊要求,保持默认就好,并点击" 下一步"继续
	、 尤特殊要求,保持默认就好,并点击" 下一步"继续

7. 虚拟机内存设置

给虚拟机设置内存大小,一般保持默认设置的"1024MB"即"1G"即可,也可按需进行调整, 设置完毕后点击"下一步"继续

	-	
此虚拟机的内 。 您要为此。	存 虚拟机使用多少内存?	
指定分配给此虚拟	机的内存里。内存大小必须	5为 4 MB 的倍数。
64 GB - 32 GB -	此虚拟机的内存(凹):	1024 💌 MB
16 GB	■ 最大推荐内存: 4768 MB	1
1 GB - C 4 512 MB - 4 256 MB -	a] ■ 推荐内存: 1024 MB	内存设置,一般1024MB,周
128 MB - 64 MB - 32 MB -	客户机操作系统最低 512 MB	1G内存,也可根据需要自 链荐内存:调整内存大小
16 MB - 8 MB - 4 MB -		
	点	(击"下一步"继续 、
±684		

8. 网络设置

选择"使用网络地址转换(NAT)"模式,这样,设置完毕后虚拟机就可以使用本机的网络,而 无需再次进行设置,设置完毕后点击"下一步"继续

要添加哪类网络?			
网络连接			
使用桥接网络(R) 为客户机操作系统提供直括有自己的 IP 地址。	妾访问外部以太	网网络的权限。客户板	1在外部网络上必须
使用网络地址转换(NAT)(E) 为客户机操作系统提供使所 权限。) 月主机 IP 地址;	5问主机拨号连接或外	部以太网网络连接的
使用仅主机模式网络(出) 将客户机操作系统连接到:	主机上的专用虚	似网络。	
◎不使用网络连接(工)		/ 选择NAT模式,并 继续	并点击"下一步"
≢βBh	Cz H-	-#(B) T-#(N)	T III III III III III III III III III I

9.选择 I/O 控制器类型

保持默认配置就好,无需改动,点击"下一步"继续

10 控制器尖型	
SCSI 控制器:	
BusLogic(U)	(不适用于 64 位客户机)
ISI Logic(L)	(推荐)
C LSI Logic SAS	<u>s)</u>
	保持默认配置,点击"下一步"即可

10.选择硬盘类型

保持默认配置就好,无需改动,点击"下一步"继续



11.选择磁盘

选择"创建新虚拟磁盘"并点击"下一步"继续



12. 指定磁盘容量

设置磁盘空间大小,一般保持默认设置的"20.0G"磁盘空间就可以,也可按需修改,设置完 毕后点击"下一步"继续

最大磁盘大小(GB)(S):	20.0	A	
针对 CentOS 64 位 的建议大	ː/ʃv: 20 GB 🔍	┛ _ ── 维持默认≯	推荐配置的20
立即分配所有磁盘空间(A))。	盘空间不到	5,并点击"下
分配所有容重可以提高性 所有空间,虚拟磁盘的空	能,但要求所有物J 间最初很小,会随著	理磁盘空间立即可用 着您向其中添加数据	。如果不立即分酉 而不断变大。
◎ 将虚拟磁盘存储为单个文(<u>‡(0</u>)		
◎ 将虚拟磁盘拆分成多个文化	 ‡(<u>M</u>)		
拆分磁盘后,可以更轻松 性能。	地在计算机之间移i	动虚拟机,但可能会	降低大容望磁盘的

- 13. 指定磁盘文件
- 保持默认配置就好,无需改动,点击"下一步"继续

建虚拟机向导 指定磁盘文件 您要在何处存储磁	盘文件?	0	X
磁盘文件 将为每2GB容里的虚拟。 名称将根据此处所提供的	磁盘创建一个磁盘 文件名称自动生成	文件。除第一个文件之外,每 。	环文件的文件
Hanwate Linux.vmdk			浏览(<u>R</u>)
	保持续	默认配置,并点击"下-	一步"继
帮助	<	:一步(B) 下一步(N) >	取消

14. 已准备好创建虚拟机

查看之前已经配置好的虚拟机信息,若无差错则点击"完成"完成虚拟机的创建工作

各使用下列设置创	建虚拟机:
名称:	Hanwate Linux
位击:	D:\Documents\Virtual Machines\Hanwate Linux
版本:	Workstation 12.0
操作系统:	CentOS 64 位
硬盘:	20 GB, 拆分
内存:	1024 MB
网络适配器:	NAT
其他设备:	CD/DVD, USB 控制器, 打印机, 声卡
自定义硬件	(C)… 查看之前配置的虚拟机信息,若无 题则点击"完成"

15. Hanwate 大数据操作系统镜像的选择 点击"CD/DVD",并进行镜像文件的选择



在弹出的设置窗口中,在"连接"中选择"使用 ISO 映像文件",并点击"浏览"找到镜像文件 的位置,并选择(保证"设备状态"中的"启动时连接"选项被勾选),确认无误后点击"确定"完成 配置

件选项		
设备	摘要	设备状态
■ 内存 □ 处理器	1 GB 1	□ 已连接(C) ☑ 启动时连接(Q)
CD/DVD (IDE)	自动检测	连接
國网络适配器	NAT	○ 使用物理驱动器(P):
♂ USB 控制器	存在	自动检测
● 声卡 ■ tTED机	目动检测	● / 使用 ISO 映像文件(M):
■显示器	自动检测	Cillen Metellingunte Bindi
		F:\Hanvvate\Hanvvate_Bigu, ▼ 300(B)
		人 选择"使用ISO映像文件",并点击"浏览"选择 Hanwate操作系统.iso文件的物理位置,并选 择
	⑦ 添加(A) 移除(B)	

16. 虚拟机的启动

虚拟机设置完毕,点击"开启此虚拟机"开启 Hanwate 操作系统并进行 Hanwate Linux 操作系统的安装和配置



二、Hanwate 操作系统配置

1. 操作系统启动成功后,会出现下面的界面

这里选择"Install Hanwate Bigdata OS 7 (Firefly)"选项,即"安装 Hanwate Bigdata OS 7"系 统,回车确定



2. 进入系统配置选项界面,按照顺序对其依次进行设置

Vanuate [®]	安装信息摘要				HANWATE BIGDA 🖾 cn	TA OS 7 安装 帮助!
- AND	本地化					
	Θ	日期和时间(T) 亚洲/上海 时区	E	键盘(K) 汉语		
	á	语言支持(L) 简体中文 (中国)				
	软件					
	0	安装源(I) 本地介质	1	软件选择(S) <i>未选择</i>		
	系统					
	<u>S</u>	安装位置(D) 没有选择磁盘	Q	KDUMP 已启用 Kdump		
	¢→	网络和主机名(N) 未连接		SECURITY POLICY No profile selected		
à						
				<i></i>	退出(Q)	开始安装(B)

3. 日期和时间的配置

选择"亚洲--上海"时区,并点击完成



4. 键盘布局设置

保持默认的汉语即可.也可按需进行修改,设置完毕后点击"完成"

键盘布局 完成(D)			HANWATE BIGDATA OS 7 安装
	您要在系统中使用哪种键盘布局?可以将任意布局移动到列表顶部将其选为	默认布局。	
k	汉语	测试以下布局配置(T):	
			<i>未配置布局切换。</i> 遗项(O)
	+ - ^ > =		

5. 语言设置

保持默认的"简体中文"即可.也可按需进行修改,设置完毕后点击"完成"

			E co	
释要安装的其他语言支持 :				
Komana	Komanian	Ø 简休由文 (由国)		
Русский	Russian	 ■ 繁體中文(台灣) 		
සිහල	Sinhala	■ 繁體中文(中華人民共和國香港特別行政區)		
Slovenčina	Slovak	🔲 简体中文 (新加坡)		
Slovenščina	Slovenian			
Shqip	Albanian			
Српски	Serbian			
Svenska	Swedish			
தமிழ்	Tamil			
తెలుగు	Telugu			
Точики	Tajik			
ไทย	Thai			
Türkçe	Turkish			
Українська	Ukrainian			
اردو	Urdu			
Tiếng Việt	Vietnamese			
中文	Chinese >			
lsiZulu	Zulu			

6. 安装源设置

保持默认配置即可,无需改动,点击"完成"

ŧ.w			
成(D)			em en 帮助
使用哪个安装源?			
)自动检测到的安装介质(A):			
设备: srO 标签: Hanwate_Bigdata_OS_7.	.Firefly 验证(V)		
) 在网络上(0):			
and the second sec			
最近的镜像 ▼ 不要安装最新的软件更新(U)。3	2装以上安装源内提供的默认版?		代理设置(P)
最近的現像 ▼ 「不要安装最新的软件更新(U)。3 软件仓库	z装以上安装源内提供的默认版*		】 【代理设置(P)
#近的現像 * ///////////////////////////////////	2装以上安装源内提供的默认版 名称(N):		(代理设置(P)
#21的現象 / / / / / / / / / / / / / / / / / / /	2装以上安装源内提供的默认版# 名称(N): http:// *)) 代理设置(P)
國立印貨幣 不要安装最新的软件更新(U), 3 软件仓库 58用 名称	2歳以上安装漂内提供的默认版 名称(N): http:// ▼	▲ 〕 该 URL 指向一个镜像列表(L),) 代理设置(P)
國立印貨幣 》 不要安装最新的软件更新(U), 3 软件仓库 約約月	2歳以上安装漂内提供的默认版 名称(N): http:// ▼ 代理 URL (X):	● 读 URL 指向一个镇像列表(L)。) 代理设置(P)
■过的現像 ▼ 7 不要安装最新的软件更新(U), 3 软件仓库 3息用 名称	2歳以上安装漂内提供的默认版 名称(N): http:// ▼ 代理 URL (X): 用户名(S):	● 读 URL 指向一个镇像列表(L)。) 代理设置(P)
能力的現象 * 「 「 「 不要安装最新的软件更新(い)」 9 - - - - - - - - - -	2装以上安装源内提供的默认版# 名称(N): http:// ▼ 代理 URL(X): 用户名(S): 密码(W):	▶] 该 URL 指向一个镜像列表(L)。	

7. 软件选择

选择"汉水大数据操作系统",附加选项无需选择,点击"完成"即可

<mark>软件选择</mark> 完成(D)	HANWATE BIGDATA OS 7 安装 四 cn 帮助 !
 <u>スペ大変獲換作系統</u> <u>スペ大変獲換作系統</u> <u>スペ大変運換作系統</u> <u>スペ大変運換作系統</u> <u>スペ大変運動作系統</u> <u></u> <u></u> <u></u>	 □ Kerberos 安全論证 Kerberos 母份验证系统 □ LDAP録→以证功能
*	

8. 安装目标位置设置

选择配置虚拟机时分配的 20G 磁盘空间,选择"自动分配分区",点击"完成"即可

	安装目标位置 ^{完成(D)}		HANWATE BIGDATA OS 7 安装 岡 cn 帮助!(F1)
iy	备选择 选择要在其中安装系统的设备。点击 本地标准磁盘	主菜单中的"开始安装"按钮前不会对该设备进行任何操作。	
	20 GiB VMware, VMware Virtual S sda / 20 GiB 空闲	1. 选择安装时配置的200标准磁盘空间	
	专用磁盘 & 网络磁盘		不会对未在此处选择的磁盘进行任何操作。
•	☑ 添加磁盘(A)		
其 [它存储选项 分区 • 自动配置分区(U), ○ 我要配置分区(I) 我想让额外空间可用(M)。 如密 加密我的数据(E)。然后设置密码。	2. 选择自动配置分区	不会对未在此处选择的磁盘进行任何操作。
売	整磁盘摘要以及引导程序(F)		已选择 1 个磁盘;容量 20 GB; 20 GB 空闲 - <u>剧新(P)</u>

9. KDUMP 设置

保持默认配置,无需改动,点击 " 完成 " 即可

	HANWATE BIGDATA OS 7 安装 ^{帮助} !
p 将捕获系统信息,这对于诊断崩溃的原因非常有用。注意,kdump 需要预留一部分系统内存,且这部分内存对于其他	用户是不可用的。
~ 根据所需内存大小决定是否启用Kdump	
	p 将捕获系统信息,这对于诊断崩溃的原因非常有用。注意,kdump 需要预留一部分系统内存,且这部分内存对于其他J 根据所需内存大小决定是否启用Kdump

10. 网络和主机名配置

网络和主机名配置会在安装过程结束之后进入系统之后进行配置,这里保持默认配置, 无需改动,点击 " 完成 " 即可

	网络和主机名(_N) 完成(D)		HANWATE BIGDATA OS 7 安装 囲 cn 帮助!
h		WARM (ens32) E時用達提 G内地址 00:00:29:22:10:59 J速 1000 Mb/s	EB CH (1993 -) 关闭 配置(0)
	主机名 (H) : localhost.localdomain 应用(A)		当前主机名: localhost.localdomain

11. SECURITY POLICY 配置

保持默认配置,无需改动,点击 " 完成 " 即可

SECURITY 完成(D)	POLICY	HANWATE BIGDATA OS 7 安装 圖 cn 帮助!
[Change content Apply security policy: #8	
	Choose profile below: Default The implicit XCCDF profile. Usually, the default contains no rules. Standard System Security Profile This profile contains rules to ensure standard security baseline of CentOS Linux 7 system.	
	This post descent to Baseline for CentOS Linux 7 This is a *draft* profile for PCI-DSS v3 C25 For CentOS Linux 7 This profile demonstrates compliance against the U.S. Government Commercial Cloud Services (C2S) baseline. This baseline was inspired by the Center for Internet Security (CIS) CentOS Linux 7 Benchmark, v1.1.0 - 04-02-2015. For the SCAP Security Guide project to remain in compliance with CIS 'terms and conditions, specifically Restrictions(8), note there is no representation or claim that the C2S profile will ensure a system is in compliance or consistency with the CIS	
ŧ	Select profile Changes that were done or need to be done: No profile selected	

12. 安装

至此,所有基本配置已全部完成,点击"安装"则系统进入安装界面(如下图)

Vanuate®	配置					HANWATE BIGDAT/ Cn	A OS 7 安装 帮助!
150K		用户设置					
		ଫୁ	ROOT 密码 <i>Root 密码未设置</i>	-	创建用户(U) 不会创建任何用户		
<i>\$</i>							
		⑦ 正在 /dev/ma	oper/hbo-root 上创建文件系统 ext4				
			汉水企业级Linux				
			上 · 善 · 若 · 水 ② 服务热线 021-52682238 http://400linux.com				

13. 设置 root 用户密码

安装成功后默认的存在用户只有 root,我们需要密码进入安装好的系统,所以我们需要给 root 用户设置一个登录系统的密码

在安装界面点击 "ROOT 密码 "进入密码设置界面来进行密码设置

若密码过于简单,则系统会提示,此时无需担心,强制保存弱密码即可(连续点击两次"完成"即可)

ROOT 密码 完成(D)		HANWATE BIGDATA OS 7 安装 國 cn
*	root 靴户用于管理系统。为root 用户输入密码。 Root 密码(R): ••••• 请以(C): •••••	

14. 安装成功 此时系统已安装完成,点击 "重启 "即可

Wanwate®	配量	HANWATE BIGDATA OS 7 安装 圖 cn
	用户设置	
	ROOT 密码 已经设置 root 密码	创建用户(U) 将创建用户 hadoop
	<u></u> ΞΕΡΣ !	Hanwate Bigdata OS 已成功安装并可以使用!
	安装完毕,点击"重启"[可	即 重高然后使用吧! 重启(R)

15.登录 此时系统已经重启完毕,直接使用账号和密码登录就可以直接使用

Hanwate BigData OS 7 (Firefly) Kernel 3.10.0-693.5.2.el7.x86_64 on an x86_64

localhost login: _

三、Hadoop 伪分布式集群搭建

Hadoop 本身是基于分布式的系统应用,但很多时候如果只是进行简单的测试,没有必要做许多的机器集群。所谓的伪分布式,本质上指的就是进行单机版的 hadoop 配置。

注意事项:

1. hadoop 集群管理上要求不要对 IP 进行变更,也就是说必须要保证从项目的开发以 及最终的运行状态,都要求 IP 地址是同一个,如果变更了,那么就意味着所有的组件都要 重新配置,很容易发生不可预知的后果。

2. 在整个 hadoop 的处理过程中,都是利用 ssh 实现通讯的,就算是在本机(伪分布式),也必须在 Linux 上配置 ssh 免登陆处理。

搭建步骤如下:

- 一、配置 IP 地址
- 1. 使用 root 账户登录系统



2. 输入 vim /etc/sysconfig/network-scripts/ifcfg-ens32 命令来修改 IP 地址配置





- 3. 在文件中修改并加入如下配置项,保存退出
 - BOOTPROTO=static IPADDR=191.168.X.100 NETMASK=255.255.255.0 GATEWAY=192.168.X.1



4. 输入 systemctl restart network 命令来重启网络服务,重启成功后使用 ip addr 命令 查看网络配置是否成功生效



- 二、修改主机名
- 1. 在 root 用户下,输入 vim /etc/hostname 命令来编辑修改/etc/hostname 文件



2. 将里面的 localhost.localdomain 修改为 hadoop,保存退出即可



三、修改主机的映射配置

1. 在 root 用户下,输入 vim /etc/hosts 命令来编辑修改/etc/hosts 文件

[root@localhost ~]# vim /etc/hosts_

2. 在原内容的基础上,添加 ip 地址和主机名的映射,保存退出即可



3.为了让刚才的配置生效,在 root 用户下,输入 reboot 重启系统



4. 重启后效果如下,配置生效



- 四、配置 SSH 免密登录
- 1. 使用 root 账号登录系统



2. 输入 ssh-keygen -t rsa - P "命令来生成公钥,所有的出现的确认的配置信息都使用 默认的方式进行处理(都按"回车")

如下图所示



文件夹生成成功

[root@hadoop ~]# 11 -a total 56 dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -rw 1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rw 1 root root 1234 Jan 10 09:18 .bash_history -rw 1 root root 1234 Jan 10 09:18 .bash_logout -rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashcc -rw-r-r 1 root root 176 Dec 29 2013 .bashcc -rw-r-r 1 root root 100 Dec 29 2013 .bashcc drwxr-xr-x 2 root root 4096 Jan 10 09:03 input -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 1 root root 129 Dec 29 2013 .tcshcc -rw 1 root root 129 Dec 29 2013 .tcshcc -rw 1 root root 5949 Jan 10 09:03 .viminfo Cdt+toot 000 7 1#	Iroot@hadoop ~]# 11 -a total 56 dr-xr-xr-x18 root root 4096 Jan 9 16:12 -rw1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rw1 root root 1234 Jan 10 09:18 .bash_history -rw1 root root 176 Dec 29 2013 .bash_logout -rw1 root root 176 Dec 29 2013 .bash_profile -rw-rr1 root root 176 Dec 29 2013 .bashc -rw-rr1 root root 176 Dec 29 2013 .bashc -rw-rr1 root root 176 Dec 29 2013 .bashc -rw-r-r1 root root 160 Dec 29 2013 .bashc -rw-r-r1 root root 160 Dec 29 2013 .cshc drwxr-xr-x 2 root root 162 Jan 9 23:17 original-ks.cfg -rw1 1 root root 162 Jan 9 23:17 original-ks.cfg -rw1 1 root root 129 Dec 29 2013 .tcshc -rw1 1 root root 129 Dec 29 2013 .tcshc -rw1 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_ 文件夫创建成功										
total 56 dr-xr-x 4 root root 4096 Jan 10 09:21 dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -rw 1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rw 1 root root 1234 Jan 10 09:18 .bash_listory -rw-r-r 1 root root 180 Dec 29 2013 .bash_logout -rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashrc drwxr-xr-x 2 root root 100 Dec 29 2013 .bashrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 .cshrc drwxr-xr- 1 root root 120 Dec 29 2013 .tcshrc -rw 1 root root 120 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Croot@hadoop ~1# 文件夫创建成功	total 56 dr-xr-x 4 root root 4096 Jan 10 09:21 dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -rw 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 176 Dec 29 2013 .cshc drwxr-xr-x 2 root root 4096 Jan 10 09:03 .pml -rw-r-r 1 root root 162 Jan 9 23:17 original-ks.cfg drwxr 2 root root 4096 Jan 10 09:21 -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:83 .viminfo Iroot@hadoop ~]#_	[root@hadou	op '	~]# 1	1 -a						
dr-xr-x 4 root root 4096 Jan 10 09:21 dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -rw 1 root root 1234 Jan 9 23:17 anaconda-ks.cfg -rw-r-r 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 18 Dec 29 2013 .bash_Dogout -rw-r-r 1 root root 176 Dec 29 2013 .bashcc -rw-r-r 1 root root 176 Dec 29 2013 .bashcc -rw-r-r 1 root root 100 Dec 29 2013 .bashcc -rw-r-r 1 root root 100 Dec 29 2013 .bashcc -rw-r-r 1 root root 162 Jan 10 09:03 .cshrc drwxr-xr- 2 root root 4096 Jan 10 09:21 .mul -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~1#	dr-xr-xr 4 root root 4096 Jan 10 09:21 dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -rw 1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rw-r-r 1 root root 1234 Jan 10 09:18 .bash_logout -rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 180 Dec 29 2013 .bashc -rw-r-r 1 root root 180 Dec 29 2013 .bashc -rw-r-r 1 root root 180 Dec 29 2013 .bashc -rw-r-r 1 root root 160 Dec 29 2013 .bashc -rw-r-r 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 -rw 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	total 56									
dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -rw 1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rwr 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 1234 Jan 10 09:18 .bash_logout -rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 176 Dec 29 2013 .bashrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 .nput -rw 1 root root 122 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:03 .viminfo -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~1# 文件夫创建成功	dr-xr-xr-x 18 root root 4096 Jan 9 16:12 -w 1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rw-r 1 root root 1234 Jan 10 09:18 .bash_history -rw-rr 1 root root 1234 Jan 10 09:18 .bash_logout -rw-rr 1 root root 176 Dec 29 2013 .bash_profile -rw-rr 1 root root 176 Dec 29 2013 .bashc drwxr-xr-x 2 root root 100 Dec 29 2013 .bashc drwxr-xr-x 2 root root 4096 Jan 10 09:03 . -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwxr 2 root root 4096 Jan 10 09:21 . -rw 1 root root 129 Dec 29 2013 .tshrc -rw 1 root root 129 Dec 29 2013 .tshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	dr-×r-×	4	root	root	4096	Jan	10	09:21		
-rw 1 root root 1854 Jan 9 23:17 anaconda-ks.cfg -rw 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 18 Dec 29 2013 .bash_logout -rw-r-r 1 root root 176 Dec 29 2013 .bash_logout -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 180 Dec 29 2013 .bashc drwxr-xr-x 2 root root 180 Dec 29 2013 .bashc drwxr-xr-x 2 root root 180 Pie3 input -rw 1 root root 160 Pie3 input -rw 1 root root 162 Jan 92:13 .tcshrc drwxrr 2 root root 160 Pie3 21 pie1	-rw 1 root root 1054 Jan 9 23:17 anaconda-ks.cfg -rw 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 120 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 176 Dec 29 2013 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 input -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwxr 2 root root 4096 Jan 10 09:03 .cshrc -rw 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	dr-xr-xr-x	18	root	root	4096	Jan	9	16:12		
-rw 1 root root 1234 Jan 10 09:18 .bash_history -rw-rr 1 root root 18 Dec 29 2013 .bash_logout -rw-rr 1 root root 176 Dec 29 2013 .bash_profile -rw-rr 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 100 Dec 29 2013 .cshrc druxr-xr-x 2 root root 4096 Jan 10 09:03 instance arw 1 root root 162 Jan 9 23:17 original-ks.cfg druxr 2 root root 4096 Jan 10 09:21 instance -rw-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo CreteNadoop ~]#_ 文件夫创建成功	-rw 1 root root 1234 Jan 10 09:18 .bash_history -rw-r-r 1 root root 18 Dec 29 2013 .bash_logout -rw-rr 1 root root 176 Dec 29 2013 .bash_profile -rw-rr 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 100 Dec 29 2013 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:83 .mul -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwxr 2 root root 4096 Jan 10 09:21	-rw	1	root	root	1054	Jan	9	23:17	anaconda-ks.cfg	
-rw-rr 1 root root 18 Dec 29 2013 .bash_logout -rw-rr 1 root root 176 Dec 29 2013 .bash_profile -rw-rr 1 root root 176 Dec 29 2013 .bashc -rw-rr 1 root root 180 Dec 29 2013 .bashc drwxr-xr-X 2 root root 4096 Jan 10 09:03 .cshrc -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 .tcshrc -rw 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~1#_	-rw-rr 1 root root 18 Dec 29 2013 .bash_logout -rw-rr 1 root root 176 Dec 29 2013 .bash_profile -rw-rr 1 root root 176 Dec 29 2013 .bashrc -rw-rr 1 root root 100 Dec 29 2013 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 input -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 anh -rw-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	-rw	1	root	root	1234	Jan	10	09:18	.bash_history	
-rw-r-r 1 root root 176 Dec 29 2013 .bash_profile -rw-r-r 1 root root 176 Dec 29 2013 .bashc -rw-r-r 1 root root 100 Dec 29 2013 .cshc drwxr-xr-x 2 root root 4096 Jan 10 09:03 input -rw-r 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 ash -rw-r-r 1 root root 129 Dec 29 2013 .cshc -rw-r-r 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~1#_	-rw-rr 1 root root 176 Dec 29 2013 .bash_profile -rw-rr 1 root root 176 Dec 29 2013 .bashrc -rw-rr 1 root root 176 Dec 29 2013 .bashrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 .cshrc drwxr 2 root root 4096 Jan 10 09:21 .cshrc -rw-r 1 root root 129 Dec 29 2013 .cshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	-rw-rr	1	root	root	18	Dec	29	2013	.bash_logout	
-rw-rr 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 100 Dec 29 2013 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 input -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	-ru-r-r 1 root root 176 Dec 29 2013 .bashrc -rw-r-r 1 root root 100 Dec 29 2013 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:83 inmit -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 met -rw-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	-rw-rr	1	root	root	176	Dec	29	2013	.bash_profile	
-rw-rr 1 root root 100 Dec 29 2013 .cshrc drwxr-xr-x 2 root root 4096 Jan 10 09:03 ingl -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 ingl -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]#_	-rw-rr 1 root root 100 Dec 29 2013 .cshrc drwxr-xr-X 2 root root 4096 Jan 10 09:03 inni -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 inni -rw-r-rr- 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]#_	-rw-rr	1	root	root	176	Dec	29	2013	.bashrc	
drwxr-xr-x 2 root root 4096 Jan 10 09:03 root -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 root -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]#_	drwxr-xr-x 2 root root 4096 Jan 10 09:03 Input -rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 sob -rw-r-r 1 root root 129 Dec 29 2013.tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]# _ 文件夫创建成功	-rw-rr	1	root	root	100	Dec	29	2013	.cshrc	
-rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 mph -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# 文件夫创建成功	-rw 1 root root 162 Jan 9 23:17 original-ks.cfg drwx 2 root root 4096 Jan 10 09:21 0511 -rw-r-r- 1 root root 129 Dec 29 2013.tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo Iroot@hadoop ~]# _ 文件夫创建成功	drw×r-×r-×	2	root	root	4096	Jan	10	09:03		
drwx 2 root root 4096 Jan 10 09:21 mmh -rw-r-r 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# _ 文件夫创建成功	drwx 2 root root 4096 Jan 10 09:21 meth -rw-rr- 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# _ 文件夹创建成功	-rw	1	root	root	162	Jan	9	23:17	original-ks.cfg	
-rw-rr 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# 文件夫创建成功	-rw-rr 1 root root 129 Dec 29 2013 .tcshrc -rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# _ 文件夹创建成功	drwx	2	root	root	4096	Jan	10	09:21		
-ru 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# _ 文件夹创建成功	-rw 1 root root 5949 Jan 10 09:03 .viminfo [root@hadoop ~]# 文件夫创建成功	-rw-rr	1	root	root	129	Dec	29	2013	.tcshrc	
[root@hadoop ~]# _ 文件夹创建成功	[root@hadoop ~]# _ 文件夫创建成功	-rw	1	root	root	5949	Jan	10	09:03	.viminfo	
		[root@hadou	op í	~]# _							文件夹创建成功

3. 输入命令 cd .ssh 进入.ssh 文件夹

输入命令 cat id_rsa.pub >> authorized_keys 将公钥信息保存在授权认证的文件中 如下图

[root@hadoop ~]# cd .ssh	
[root@hadoop .ssh]# 11	
total 8	
-rw 1 root root 1679 Jan 10 09:21 id_rsa	
-rw-rr 1 root root 393 Jan 10 09:21 id_rsa.pub	
[root@hadoop .ssh]# cat id_rsa.pub >> authorized_keys 👞	茲//组加/>到授权文
[root@hadoop .ssh]# 11	~ 准备 binn cange can
total 12	
-rw-rr 1 root root 393 Jan 10 09:24 authorized_keys	
-rw 1 root root 1679 Jan 10 09:21 id_rsa	
-rw-rr 1 root root 393 Jan 10 09:21 id_rsa.pub	
[root@hadoop .ssh]# cat authorized_keys	
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDB0sHjWV3nXTXiwAaRyn3	3pRm1cDBBS8.jGCmo∕sTnsa
Auffcw0caRvh0Cr4SUJv+E+oJCLBb12vreP2011G9VVjvDrEmMa8QcZ8kg	XBo9bugaia/g5PddTsh/4
oSmkI74jrlpvPPa0cSFrQMCjTSBKmHqf8TUH06e9A0KmUk1WpK1nVK8OyZ0	s0oG7IgK2a+nLaHdCBmIP
MWjSpDvRIKjf+E6w5CEWjCrHIJ3Jrg+aUYlGg2gByg3u3LitD/1b0F3gK11	zM4aOf jvwZgkwLrQ49LNe
8KJX8CJQeFEHY+igEn+/GoHEIZX i iF9w+FEd i5Krbs377cJ1avySpGEWa2r	nR root@hadoop
[root@hadoop .ssh]#	
an sa ulan processa 🖬 🗉 - Andrewski =	
公钥信息	

4. 输入命令 chmod 600 authorized_keys 修改 authorized_keys 文件的权限为 600(rw-----)



5. SSH 免密登录配置成功

使用"ssh hadoop"来登录 hadoop 账户 使用"exit"退出登录



五、配置环境变量 JAVA_HOME

1. 输入命令 vim /etc/profile 来配置环境变量



- export CLASSPATH=.:\$CLASSPATH:\$JAVA_HOME/lib export PATH=\$PATH:\$JAVA_HOME/bin
- 3. 修改后的 profile 文件,如下图



4. 输入命令 source /etc/profile 使得配置信息生效



5. 检验 JAVA_HOME 是否配置成功(输入命令 Java -version 检测)



- 六、配置环境变量 HADOOP_HOME
- 1. 输入命令 vim /etc/profile



- 将如下内容添加到文件内容末尾,保存退出即可 export HADOOP_HOME=/usr/hdp/2.6.3.0-235/hadoop export PATH=\$PATH:\$HADOOP_HOME/bin
- 3. 修改后的 profile 文件,如下图

else		
umask B22		
f <u>i</u>		
<pre>for i in /etc/profile.d/*.sh ; do if [-r "\$1"]; then if ["\$(-#*i)" != "\$-"]; then</pre>		
done		
unset i	HADOOP_HOME	
unset -f pathmunge		
export mESUS_HUME=/usr/local/mesos		
EXPORT CLHOOFHINE, SOULDOFFHIN, SOUNDERVIEW		
export HHDOUR_HURL=/USr/nap/2.0.3.0-235/nadoop	LAND HIDER ALL CHARGE	Linking of
export rHin-grain.gataua_nune/sbin.gataua_nune/bin.ga	HOH_HUNC/BIN SHADOO	
1)n Beelen (2011) Maaroon (2011) Maaroon (2011)		
将HADOOP_HOME加到P	PATH中	
	83,0-1	Bot

4. 输入命令 source /etc/profile 使得配置信息生效



5. 查看配置是否成功



- 七、配置 " hadoop-env.sh " 文件
- 输入命令 cd \$HADOOP_HOME/etc/hadoop
 输入命令 vim hadoop-env.sh 来修改 hadoop-env.sh 文件

[root@hadoop hadoop]# vim hadoop-env.sh _

 将如下内容添加到文件内容末尾,保存退出即可 export JAVA_HOME=/usr



- 八、配置 " core-site.xml " 文件
- 1. 输入命令 vim core-site.xml 来修改 core-site.xml 文件

[root@hadoop hadoop]# vim core-site.xml_

- 2. 将如下内容添加到文件内容中(configuration 标签中),保存退出即可 <property>
 - <name>hadoop.tmp.dir</name> <value>/hadoop/hadoop_tmp</value> </property> <property> <name>fs.defaultFS</name> <value>hdfs://hadoop:9000</value> </property>
- 3. 修改后的 core-site.xml 文件,如下图



 输入命令 mkdir -p /hadoop/hadoop_tmp 在"/"目录下创建临时文件目录 并输入命令 chmod -R 777 /hadoop 将临时文件目录及其子目录的权限都修改成



- 九、配置"hdfs-site.xml"文件
- 1. 输入命令 cd \$HADOOP_HOME/etc/hadoop

注 意 : 默 认 情 况 下 , 当 前 目 录 下 没 有 hdfs-site.xml 文 件 , 需 要 从 \$HADOOP_HOME/../etc/hadoop/conf.empty/目录下复制 hdfs-site.xml 文件

输入命令 cp \$HADOOP_HOME/../etc/hadoop/conf.empty/hdfs-site.xml ./ 复制 hdfs-site.xml 到当前目录下

输入命令 vim hdfs-site.xml 来修改 hdfs-site.xml 文件

[root@hadoop [root@hadoop -rw-rr 1 [root@hadoop	hadoop]# cp \$HADOOP_HOME//etc/hadoop/conf.emp hadoop]# 11 lgrep hdfs-site.xml root root 775 Mar 23 17:27 hdfs-site.xml hadoop]# vim hdfs-site.xml	uty/hdfs-site.xml ./
	编辑修改hdfs-site.xml文件	复制hdfs-site.xml到当 前目录下

2. 将如下内容添加到文件内容中(configuration 标签中),保存退出即可

```
<property>
```

<name>dfs.replication</name>

- <value>1</value>
- </property>

<property>

- <name>dfs.namenode.name.dir</name>
- <value>/hdfs/name</value>
- </property>
- <property>

<name>dfs.datanode.data.dir</name>

<value>/hdfs/data</value>

</property>

3. 修改后的 hdfs-site.xml 文件,如下图

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, e	ither express or implied.
See the License for the specific language gover	ning permissions and
limitations under the License. See accompanying	LICENSE file.
>	
Put site-specific property overrides in this</p	file>
<configuration></configuration>	
<property></property>	
<name>dfs.replication</name>	
<value>1</value>	
	添加如下配置信息
<property></property>	A STATE OF A
<name>dfs.namenode.name.dir<td>></td></name>	>
<value>/hdfs/name</value>	
<property></property>	
<name>dfs.datanode.data.dir<td>></td></name>	>
<value>/hdfs/data</value>	
	2E 0 1 Pot
	11.PI-1 DILL

4. 输入命令 mkdir -p /hdfs/{name,data}在根目录"/"下创建数据节点和名称节点的路 径

输入命令 chown -R hdfs:hadoop /hdfs 修改文件夹的属主以及属组



十、配置 "yarn-site.xml" 文件

注 意 : 默 认 情 况 下 , 当 前 目 录 下 同 样 没 有 yarn-site.xml 文 件 , 需 要 从 \$HADOOP_HOME/../etc/hadoop/conf.empty/目录下复制 yarn-site.xml 文件

输入命令 cp \$HADOOP_HOME/../etc/hadoop/conf.empty/yarn-site.xml ./ 复制 yarn-site.xml 到当前目录下

1. 输入命令 vim yarn-site.xml 来修改 yarn-site.xml 文件



2. 将如下内容添加到文件内容中(configuration 标签中),保存退出即可 <property>

```
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
<property>
<name>yarn.application.classpath</name>
<value>$HADOOP_CONF_DIR,
/usr/hdp/2.6.3.0-235/hadoop/*,
/usr/hdp/2.6.3.0-235/hadoop-lib/*,
/usr/hdp/2.6.3.0-235/hadoop-hdfs/*,
/usr/hdp/2.6.3.0-235/hadoop-hdfs/*,
```

```
/usr/hdp/2.6.3.0-235/hadoop-yarn/*,
```

/usr/hdp/2.6.3.0-235/hadoop-yarn/lib/*, /usr/hdp/2.6.3.0-235/hadoop-mapreduce/*, /usr/hdp/2.6.3.0-235/hadoop-mapreduce/lib/*, /usr/hdp/2.6.3.0-235/hadoop-httpfs/*, /usr/hdp/2.6.3.0-235/hadoop-httpfs/lib/* </value>

</property>

3. 修改后的 yarn-site.xml 文件,如下图

<property> <name> <value </value </name></property> <property> <name> <value </value </name></property>	<pre>yarn.nodemanager.aux-services >mapreduce_shuffle yarn.application.classpath >\$HADOOP_CONF_DIR, /usr/hdp/2.6.3.0-235/hadoop/lib/*, /usr/hdp/2.6.3.0-235/hadoop-hdfs/lib/*, /usr/hdp/2.6.3.0-235/hadoop-hdfs/lib/*, /usr/hdp/2.6.3.0-235/hadoop-yarn/*, /usr/hdp/2.6.3.0-235/hadoop-yarn/*, /usr/hdp/2.6.3.0-235/hadoop-mapreduce/*, /usr/hdp/2.6.3.0-235/hadoop-mapreduce/!lib/*, /usr/hdp/2.6.3.0-235/hadoop-httpfs/*, /usr/hdp/2.6.3.0-235/hadoop-httpfs/lib/* e></pre>
--	---

十一、配置"mapred-site.xml"文件

注意:默认情况下当前目录下没有 mapred-site.xml 文件,只有一份模板文件 mapred-site.xml.template,需要复制该模板文件到本地目录下并修改文件名为 mapred-site.xml

1. 输入命令 cp ./mapred-site.xml.template ./mapred-site.xml 复制模板文件到当前目 录并修改文件名为 mapred-site.xml



1. 输入命令 vim mapred-site.xml 来修改 mapred-site.xml 文件

[root@hadoop conf]# vim mapred-site.xml_

将如下内容添加到文件内容中(configuration标签中),保存退出即可
 configuration标签中),保存退出即可

<name>mapreduce.framework.name</name> <value>yarn</value>

</property>

<property>

<name>mapreduce.application.classpath</name>

<value>

/etc/hadoop/conf/*,

/usr/hdp/2.6.3.0-235/hadoop/*,

/usr/hdp/2.6.3.0-235/hadoop-hdfs/*,

/usr/hdp/2.6.3.0-235/hadoop-yarn/*,

/usr/hdp/2.6.3.0-235/hadoop-mapreduce/*,

/usr/hdp/2.6.3.0-235/hadoop/lib/*,

/usr/hdp/2.6.3.0-235/hadoop-hdfs/lib/*,

/usr/hdp/2.6.3.0-235/hadoop-yarn/lib/*,

/usr/hdp/2.6.3.0-235/hadoop-mapreduce/lib/*

</value>

</property>

3. 修改后的 mapred-site.xml 文件,如下图



十二、格式化 HDFS 文件系统

1. 输入命令 su - hdfs 切换到 hdfs 用户身份

FrootOhadoop ~]# su - hdfs Last login: Fri Jan 12 11:09:34 CST 2018 on tty1 -bash-4.2\$ _

2. 输入命令 hdfs namenode -format 进行文件系统的格式化

-bash-4.2\$ hdfs namenode -format_

3. 格式化成功反馈

18/01/08 16:13:45 INFO namenode.FSNamesystem: Retry cache on namenode is enabled
18/01/08 16:13:45 INFO namenode.FSNamesystem: Retry cache will use 0.03 of total
heap and retry cache entry expiry time is 600000 millis
18/01/08 16:13:45 INFO util.GSet: Computing capacity for map NameNodeRetryCache
18/01/08 16:13:45 INFO util.GSet: UM type = 64-bit
18/01/08 16:13:45 INFO util.GSet: 0.029999999329447746% max memory 966.7 MB = 29
7.0 KB
18/01/08 16:13:45 INFO util.GSet: cavacitu = 2^15 = 32768 entries
18/01/08 16:13:45 INFO namenode.FSImage: Allocated new BlockPoolId: BP-176379818
1-192.168.216.100-1515399225660
18/01/08 16:13:45 INFO common.Storage: Storage directory /hadoop/hdfs/name has b
cen successfully formatted.
18/01/08 16:13:45 INFO namenode.FSImageFormatProtobuf: Saving image file //hadoop
/hdfs/name/current/fsimage.ckpt 000000000000000000 using no compression
18/01/08 16:13:46 INFO namenode.FSImageFormatProtobuf: Image file /hadoop/hdfs/n
ame/current/fsimage.ckpt 00000000000000000 of size 321 butes saved in 0 second
S -
18/01/08 16:13:46 INFO namenode.NNStorageRetentionManager: Going to retain 1 ima
aes with txid >= 0
18/01/08 16:13:46 INFO util.ExitUtil: Exiting with status 0 成功格式化
18/01/08 16:13:46 INFO namenode.NameNode: SHUTDOWN MSG:
/**************************************
SHUTDOWN MSG: Shutting down NameNode at hadoon/192.168.216.100 退出状态为O证明格

[root0hadoon_hadoon]#

十三、启动 HDFS 服务

输入命令 systemctl start hadoop-hdfs-namenode 启动 namenode 服务

输入命令 systemctl start hadoop-hdfs-datanode 启动 datanode 服务

输入命令 systemctl start hadoop-hdfs-secondarynamenode 启动 secondarynamenode 服务

并输入 jps 查看是否启动成功



HDFS 服务启动成功后,通过 Web 浏览器访问 192.168.216.137(当前 IP 地址):50070 就可以查看到 HDFS 后台的情况,如下图所示

Started:	Mon Jan 08 16:42:43 CST 2018		
Version:	2.7.3.2.6.3.0-235, r45bfd33bba8acadfa0e6024c80981c023	b28d454	
Compiled:	2017-10-30T02:33Z by Jenkins from (detached from 45bf	133)	
Cluster ID:	CID-7dbc2c1a-2c54-41f0-90d7-0580b019fc95		
Block Pool ID:	BP-1763798181-192.168.216.100-1515399225660		
 Heap Memory used 37.8 Non Heap Memory used 	House – Local mesystem oujecus). I MB of 47.29 MB Heap Memory. Max Heap Memory is 966.69 MB. 43.61 MB of 44.38 MB Committed Non Heap Memory. Max Non Heap	ıp Memory is ≺unbonded>.	
Configured Capacity:		16.6 GB	
DFS Used:		28 KB (0%)	
Non DFS Used:		6.68 GB	
DFS Remaining:		9.06 GB (54.58%)	
Block Pool Used:		28 KB (0%)	ode tv Hk
DataNodes usages% (/lin/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%	

十四、启动 Yarn 服务

注意:默认的软件安装没有创建 yarn-env.sh 文件, 需手动创建此文件以保证 yarn 的服务可以正常启动!

输入命令 touch \$HADOOP_HOME/etc/hadoop/yarn-env.sh 创建启动 yarn 服务必须 的环境文件

输入命令 systemctl start hadoop-yarn-resourcemanager 启动 resourcemanager 服务 输入命令 systemctl start hadoop-yarn-nodemanager 启动 nodemanager 服务 并输入 jps 查看是否启动成功



Yarn 服务启动成功后,通过 Web 浏览器访问 192.168.216.137 (当前 IP 地址):8088 就可以查看到 Resourcemanager 任务管理器后台的情况,如下图所示

						All	Ap	pli	cati	ons						
Cluster M	etrics															
Apps Submittee	Apps Pendin	Apps g Running	Apps Completer	Container d Running	rs Mem Use	ory Memory d Total	Me	emory served	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decom	nissioned odes	Lost Nodes	Unhealth Nodes
0	0 0 0 0 0 0B 8GB 0B 0 8 0							0	1	Q		0	Q			
Schedule	Scheduler Metrics															
Canacity	Scheduler Type Scheduling Resource T							Minimum Allocation Maximum Allocation Maximum Allocation						location		
Channe 20	Copacity sufficiency [memory1] <memory1024, vcores:2=""> <memory5122, vcores:2=""> <memory512, td="" vcor<=""></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory512,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory5122,></memory1024,>															
SHOW 20	• enuies										Allenanderel	Allenanderal	04	N	search.	
ID User ▼ ≎	Name 0	Application Type 0	Queue Al	oplication S riority ©	tartTime ≎	FinishTime ¢	State ¢	FinalSt	atus Co	ontainers 0	CPU VCores ¢	Memory MB ©	% of Queue	% of Cluster 0	Progress ¢	Tracking UI 0
	No data available in table															
Showing 0	to 0 of 0	entries														
					1											
	Cluster M Apps Submittee 0 Schedulet Show 20 ID User Showing 0	Cluster Metrics Apps Apps Summite Pendin 0 Scheduler Metrics Scheduler Metrics Scheduler Scheduler Brow 20 - Jenties 10 User Name 5 Showing 0 to 0 of 0	Cluster Metrics Apps App Apps Sumittee Pending Running 0 0 Scheduler Metrics Scheduler Type Capacity Scheduler 10 User Name Application 10 User Name Application 10 Showing 0 to 0 of 0 entries	Cluster Metrics Apps Apps Apps Apps Submitted Pending Running Complete 0 of Charles Source Complete 0 of Charles Source Complete 0 of Charles Source Complete 10 User Name Application Queue Ap 10 User Name Application Queue Ap 10 User Name Application Queue Ap 10 User Name Application Queue Ap	Cluster Metrics Apps Apps Apps Apps Apps Containe Submitted Pending Running Completed Running 0 0 0 0 0 0 0 0 Scheduler Metrics Scheduler Type Scheduler (MEMORY) Show 20 + Intries 10 User Name Application Queue Application S Showing 0 to 0 of 0 entries	Cluster Metrics Apps Apps Apps Apps Containers Mem Sumitie Fending Kunning Completed Running Use 0 0 0 0 0 0 0 0 0 0 0 8 Scheduler Metrics Scheduler Metrics Scheduler (MEMORY) Shevi 20 - Sinhelise 10 User Name Application Queue Application StartTime 2 Showing 0 to 0 of 0 entries	Cluster Metrics Apps App Apps Containers Memory Memory Summited Pending Kunning Compated Ranning Used Total 0 0 0 0 0 0 8 8 868 Scheduler Metrics Scheduler Type Scheduling Resource Type Capacity Scheduler (MEMORY) Shew 20 reinties 10 User Name Application Queue Application StartTime FrishTime 7 9 1 ype 2 9 Priority 0 StartTime FrishTime 2 Schewing 0 to 0 of 0 entries	Cluster Metrics Apps Apps Apps Apps Containers Memory Mem	Cluster Metrics Apps App Apps Containers Memory Memory Memory Summited Pending Kunning Compated Ranning Used Total Reserved 0 0 0 0 0 0 0 0 0 0 8 8 68 0 8 Scheduler Metrics Scheduler Metrics Scheduler Metric (MEMORY) cmemory Memory Memory Memory Memory Capacity Scheduler Compatibility Scheduling Resource Type Capacity Scheduler (MEMORY) Show 20 remines 10 User Name Application Queue Application StartTime FinishTime State FinalSt 10 User Name Application Queue Application StartTime FinishTime State No data avail Showing 0 to 0 of 0 entries	Cluster Metrics Apps App Apps Apps Containers Memory Memory Memory VCores Submitted Pending Running Completed Running Used Total Reserved Used 0 0 0 0 0 0 0 0 0 0 8 8 68 0 8 0 Scheduler Metrics Scheduling Resource Type (Memory 1024, Scheduler Type (Memory) (Scheduling Resource Type (Memory 1024, Scheduling Resource Type (Memory 1024, Scheduling Resource Type (Memory 1024, Scheduling Version Scheduling Resource Type (Memory 1024, Schedul	Cluster Metrics Summited Apps Apps Containers Memory Memory VCores VCores 0	Cluster Metrics Sumittee Apps Apps Apps Containers Memory Memory Vicores Vic	Cluster Metrics Apps Apps Containers Memory Memory VCres VCores Active Submitted Pending Munning Completed Running Used Total Reserved Used Total Reserved Nodes Scheduler 0	Cluster Metrics Cluster Metrics Submitted Pending Running Completed Running Used Total Reserved Used Total Reserved Nodes 0 <td>Cluster Metrics Cluster Metrics Apps Apps Containers Memory Memory Memory Vcores Vcores Vcores Active Decommissioned Scheduler Metrics 0</td> <td>Cluster Metrics Schwinter Apps Apps Containers Memory Memory</td>	Cluster Metrics Cluster Metrics Apps Apps Containers Memory Memory Memory Vcores Vcores Vcores Active Decommissioned Scheduler Metrics 0	Cluster Metrics Schwinter Apps Apps Containers Memory Memory

十五、测试实验

1. 输入命令 vim input.txt 来编辑一个文本文件

[root@hadoop 2.6.3.0-235]# vim input.txt_

2. 文本文件内容如下图所示(也可自行编辑)

hello	
world	
hadoop	
yarn Î	
mapreduce	
resourcemanager	
kevin	
allen	
he <u>l</u> en	

 输入命令 hdfs dfs -put input.txt /input 上传文本文件到 HDFS 集群中 并输入命令 hdfs dfs -cat /input 查看文件是否上传成功



5. 输入如下命令进行测试

bash-4.2\$ hadoop jar /usr/hdp/2.6.3.0-235/hadoop-mapreduce/hadoop-mapreduce-exam ples-2.7.3.2.6.3.0-235.jar wordcount /input /output_

- 6. 测试结果反馈
- 1). 任务运行反馈如下图所示

18/01/10 16:55:07 INFO input.FileInputFormat: Total input paths to process :	1
18/01/10 16:55:07 INFO mapreduce.JobSubmitter: number of splits:1	
18/01/10 16:55:07 INFO mapreduce.JobSubmitter: Submitting tokens for job: jo	b_15
15573119828_0001	
18/01/10 16:55:08 INFO impl.YarnClientImpl: Submitted application applicatio	n_15
15573119828_0001	
18/01/10 16:55:08 INFO mapreduce.Job: The url to track the job: http://hadoo	p:80
88/proxy/application_1515573119828_0001/	
18/01/10 16:55:08 INFO mapreduce.Job: Running job: job_1515573119828_0001	
18/01/10 16:55:45 INFO mapreduce.Job: Job job_1515573119828 0001 running in	uber
mode : false	
18/01/10 16:55:45 INFO mapreduce.Job: map 0% reduce 0% 样条正在运行	
18/01/10 16:56:35 INFO mapreduce.Job: map 100% reduce 0%	
18/01/10 16:56:46 INFO mapreduce.Job: map 100% reduce 100%	
18/01/10 16:56:49 INFO mapreduce.Job: Job_job_1515573119828_0001 completed s	ucce
ssfully	
18/01/10 16:56:50 INFO mapreduce.Job: Counters: 49	
File System Counters	
FILE: Number of bytes read=128	
FILE: Number of bytes written=234931	
FILE: Number of read operations=0	
FILE: Number of large read operations=0	
FILE: Number of write operations=0	
HDFS: Number of bytes read=157	
HDFS: Number of bytes written=86	

2). 输入命令 hdfs dfs -cat /output/part-r-00000 查看 mapreduce 任务输出

[root0]	hadoop ~1	# hadoop	fs -cat /outp	ut/part-r-0000	U		
allen	1						
hadoop	1						
helen	1						
hello	1				直復	mapreduce	
kevin	1						
mapredu	lce	1					
resourd	cemanager	• 1 🙀					
world	1						
yarn	1						
[root0]	hadoop ~1	# _	输出内容				

3). Web 界面运行结果反馈

		① 192.168.216.100:8088/cluster										88	☆		l		9	1, 9 Official	1
hee	loop						AII A	ppli	cati	ons						Lo	gged i	n as: dr.	.wl
er	Cluster Me	trics																	
s Labels	Apps Submitted	Apps Pending	Apps Running	Apps Complet	Containers ed Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decom	missioned odes	Lost Nodes	Unhea Nod	lthy es	Reboo Node	te es
ations EW	1 Scheduler	0 Metrics	0	1	0	0 B	8 GB	0 B	0	8	0	1	0		Q	0		0	
JEMITTED CCEPTED	So Capacity So	Scheduler Type Scheduling Resource Type Capacity Scheduler [MEMORY]						<men< td=""><td>N Nory:1024</td><td colspan="6">Minimum Allocation Maximum Allocation 24, vCores:1> <memory:8192, vcores:8=""></memory:8192,></td><td></td><td></td></men<>	N Nory:1024	Minimum Allocation Maximum Allocation 24, vCores:1> <memory:8192, vcores:8=""></memory:8192,>									
NISHED	Show 20 💌 entries												Search:						
ILLED duler		ID		User 0	Name Ap	plication ype °	Queue 0	StartTim	e Finis	hTime 0	State 0	FinalSta	atus p	rogress 0	Track	ing UI 0	Bla	odes	3
	application	15155731	19828 000	loot	word MAPF count	REDUCE	default	Wed Jan 10 16:55:08 +0800 2018	Wed 16:56 +080	Jan 10 5:48 10 2018	FINISHED	SUCCEE	DED		Histor	Υ	N/A		
	Showing 1 t	to 1 of 1 er	ntries m	preducet	壬务提交成功							napredu	ice任务执	行成功			1 N		