文章编号:1671-4598(2016)02-0114-04 DOI:10.16526/j.cnki.11—4762/tp.2016.02.031 中图分类号:TP273 文献标识码:A

大数据环境下 MapReduce 准入控制的设计与实现

李亚如[1]，刘建华[2]
(1.西安邮电大学 通信与信息工程学院，西安 710061；2.西安邮电大学 信息中心，西安 710061)

摘要：为了保证 Hadoop 平台的安全性，确保执行 MapReduce 作业的 TaskTracker 节点符合 Hadoop 平台的安全管理要求，杜绝非法用户对 TaskTracker 节点访问并对其分配任务，提出了一种对 JobTracker 节点进行身份认证的方法，设计实现了对 JobTracker 节点进行身份认证的认证体系；该认证体系主要是基于 802.1x 进行准入控制，重点分析了 MapReduce 分布式计算的流程，提出了总体设计方案，完成了相关模块的配置实现，最终结果显示只有通过认证的 JobTracker 才能通过交换机的可控端口给相应的 TaskTracker 节点分配任务，有效地避免了非法用户的访问，增强了 Hadoop 平台的安全性、可靠性，为建设安全的大数据环境提供了很好的技术支持。

关键词：Hadoop；TaskTracker 节点；802.1x ；大数据

Design and Implementation of Mapreduce Access Control in Big Data Environment

Li Yaru[1], Liu Jianhua[2]
(1. School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710061, China; 2. Information Center, School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710061, China)

Abstract: In order to ensure the security of Hadoop platform, to ensure that the TaskTracker node of the MapReduce operation is in accordance with the safety management requirements of the Hadoop platform, Put an end to the illegal users access to TaskTracker node and the assigned task, A method of authentication for JobTracker node is proposed, Design and implementation of the authentication system for JobTracker nodes. The certification system is mainly based on 802.1x performs admission control, Focus on the analysis of the process of MapReduce distributed computing, Put forward the overall design scheme, Complete the configuration of the relevant module, The final result shows that the TaskTracker node assignment can be assigned to the corresponding JobTracker node through the control of the switch, Effectively avoid the illegal user access, Enhanced security and reliability of the Hadoop platform, For the construction safety of big data environment provides a good technical support.

Keywords: Hadoop; TaskTracker node; 802.1x; big data

0 引言

随着网络的普及和发展，数据正在以爆炸式的方式生产积累，世界已进入网络化的大数据时代，然而大规模数据的汇集无形中加大了信息泄漏的风险性，因此大数据的安全问题成为新的挑战。Hadoop 作为对大数据时代所特有的数据进行存储和处理的框架，以成本低、搭建灵活、便于管理等优势，在互联网的各个领域得到了广泛的应用和发展，但其安全机制薄弱缺乏一个安全认证机制，以致无法保证在集群上进行操作的用户为合法用户[1]。就 MapReduce 分布式计算框架而言，JobTracker 通过分配任务给 TaskTracker 节点运行，来协调管理全部运行在平台上的作业，如果任意用户都可访问 TaskTracker 节点并对其分配任务，会导致信息的泄露，产生严重后果，这将直接影响着整个平台的安全性能[2-3]。本文通过对 Hadoop 的 MapReduce 分布式计算框架的研究，改良其不足之处，设计实现对 MapReduce 进行基于 802.1x 的准入控制，从而避免非法用户访问 TaskTracker 节点并对其分配任务，对提高 Hadoop 平台的整体安全性能具有重要的现实意义。

收稿日期:2015-12-03； **修回日期**:2015-12-30。

作者简介:李亚如(1988-),女,山西晋城人,硕士研究生,主要从事网络与信息安全方向的研究。

刘建华(1963-),男,陕西西安人,教授,高级工程师,主要从事信息安全方向的研究。

1 系统总体设计

Hadoop 是一个能够在集群上对大规模数据进行分布式计算和处理的开源框架，实现了 Google 的 Map—Reduce 编程模型和框架，能够把应用程序分割成许多小的工作单元，并把这些单元放到集群的节点上执行处理[4]。Hadoop 框架最核心的两个设计模块就是 HDFS 分布式文件系统和 MapReduce 分布式计算框架。HDFS 以流式数据访问的模式实现了大规模数据的存储，MapReduce 实现了对海量数据进行并行计算处理的应用[5]。

当 client 节点向 Hadoop 集群提交一个 MapReduce 作业时，JobClient 调用 runJob（）方法创建 JobClient 实例并通过调用其 submitJob（）方法提交作业，与此同时 JobClient 会在 client 节点将运行作业所需的资源信息打包成 jar 文件存储到 HDFS 中。当 JobTracker 接收到 submitJob（）方法的调用后，会创建一个正在运行的作业对象，为创建任务运行列表，JobTracker 需从共享文件系统（Shared FileSystem）中检索 Hadoop 已划分好的独立的输入数据分片（input slipt)。为了确保

试图在 Hadoop 集群上对 TaskTracker 节点执行操作的用户为安全合法用户，JobTracker 将每一个分片创建的 task 任务发送给 TaskTracker 时需通过认证者进行身份验证，若为合法的安全用户，任务发送成功，若不合法，任务发送失败并重新进行身份验证。当 TaskTracker 接收到分配的任务后，首先将任务所包含的所有信息从共享文件中检索缓存，然后创建一个 TaskRunner 实例来执行该任务，TaskRunner 将启动一个新的 JVM 运行 Map 任务或 Reduce 任务，以确保 Map 和 Reduce 的独立性。具体系统的总体设计如图 1 所示。

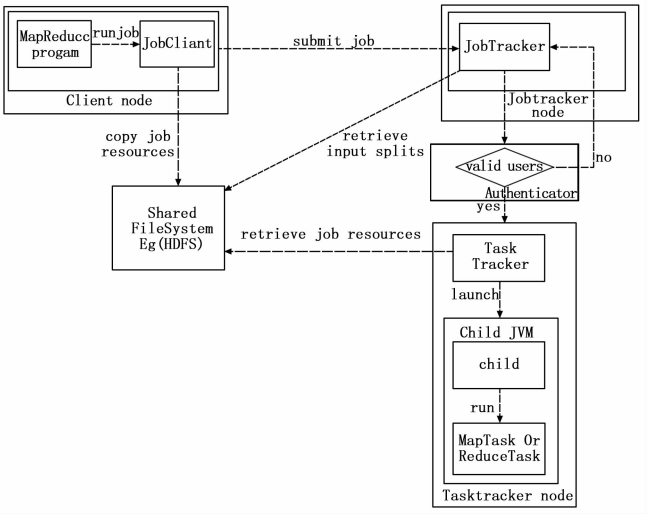


图 1　具体系统的总体设计

对于 hadoop 集群执行 MapReduce 作业的过程来讲，主要分为 5 个独立的部分：Client 节点、JobTracker 节点、TaskTracke 节点、HDFS 和认证者。具体各部分功能概述如表 1 所示。其中 JobTracker 节点、TaskTracke 节点和认证者构成了认证体系，在认证体系模块中进行详细的介绍。

表 1　各部分功能概述

client 节点	负责提交 MapReduce 作业
JobTracker 节点	负责各个作业任务的分配
TaskTracker 节点	负责具体任务的执行
HDFS	负责存储并共享任务
认证者	负责控制端口的状态

2　具体的设计方案

2.1　Hadoop 集群的配置方案

Hadoop 集群是一种典型的主从式架构，它可划分为 Master 和 Slave 两个角色，其中在 Master 节点上运行着 NameNode、JobTracker，而在每个 Slave 节点上，都部署一个 DataNode 和 TaskTracker。Hadoop 是基于 Java 编程语言的可运行在大型主机群上的应用程序，因此对 Hadoop 进行安装之前首先要配置 SSH 和支持 Java 运行环境的 JDK，其中 SSH 和 JDK 的配置安装没有先后顺序。配置完成 SSH 和 JDK 后，再进行 Hadoop 的安装配置，最后完成 Hadoop 集群的搭建。

2.1.1　SSH 和 JDK 的配置

Hadoop 通过 SSH（Secure Shell）来管理其守护进程，因此必须安装 SSH。在完全分布的模式下，Hadoop 控制脚本依赖 SSH 来执行针对整个集群的操作，为了支持无缝工作，需要在各个节点之间执行指令的时候进行无密码登陆的形式，所以在配置 SSH 需要无密码公钥认证的形式。

首先确保 Hadoop 集群的各个节点中 SSH 已经被安装。为了实现无密码登录，在集群的 master 节点下生成一组无密码登录的密钥，在默认的情况下将 authorized _ keys 文件复制到各个授权的 slave 节点的/. ssh 目录下。这样配置之后，master 节点可以无密码登录所有的 slave 节点。

配置支持 Java 运行环境的 JDK 之前，先对集群的各个节点安装 jdk1. 8. 0 _ 51 并保证其安装路径一致，在此基础上对/. bashrc 文件中的环境变量 JAVA _ HOME、CLASSPATH、PATH 和 JRE _ HOME 进行配置。

2.1.2　Hadoop 的配置

搭建 Hadoop 集群首先在 Master 节点上安装 Hadoop，本系统使用的 Hadoop 版本为 hadoop－1. 2. 1。安装完成之后需要在 hadoop/conf 目录下进行配置，其常用的 3 个配置文件为 core－site. xml 文件、hdfs－site. xml 文件、mapred－site. xml 文件，分别完成对 HDFS 地址和端口、HDFS 数据副本数量、MapReduce 的地址和端口的配置。启动 Hadoop 之前，需要对 Hadoop 的 HDFS 文件系统进行格式化，在 Master 节点 hadoop 的安装目录下输入 bin/hadoop namenode －format 命令格式化文件系统，自此 Hadoop 的文件配置全部完成。

2.1.3　Hadoop 集群的搭建

完成 Master 节点的 Hadoop 安装配置后，将该节点的 hadoop 文件夹拷贝至其他 Slave 节点处即可完成集群配置，在 Master 节点 hadoop 的安装目录下执行 bin/start－all. sh 命令启动 Hadoop 集群。

2.2　认证体系的配置方案

在 Hadoop 平台执行 MapReduce 作业的过程中，JobTracker 节点作为请求者向 TaskTracker 节点发送分配任务时，根据 802. 1x 认证程序，输入认证信息发起 802. 1x 认证，认证者通过非可控端口收到请求者提交的认证数据信息，发送至 TaskTracker 节点上的认证服务器进行验证，服务器将接收到的用户信息与数据库中的用户信息进行比对，如果用户信息一致则认证成功，可控端口打开任务发送成功，如果认证失败，交换机的端口保持关闭状态，任务分配失败。

802. 1x 是一种基于物理端口或逻辑端口（如 VLAN）的认证协议，是一种对用户身份进行认证的方法和策略。进行 802. 1x 认证的最终目的就是限制未授权的用户或设备通过端口接入网络[6-7]。它的体系结构包括二个模块：即请求模块、认证模块，其中认证模块包括认证者和认证服务器。具体的认证体系结构如图 2 所示。

2.2.1　请求模块的配置

请求模块一般是支持 802. 1x 认证的用户终端设备，用户通过启动客户端软件发起 802. 1x 认证，由认证系统对其进行身份认证来实现基于端口的接入控制[8-9]。在 MapReduce 作业执行的过程中 JobTracker 节点作为请求者访问被分配任务的 TaskTracker 节点，因此在 JobTracker 节点进行配置以实现对 802. 1x 认证标准的支持。

2.2.2　认证模块的配置

在 hadoop 集群中执行 MapReduce 作业的过程中，交换机

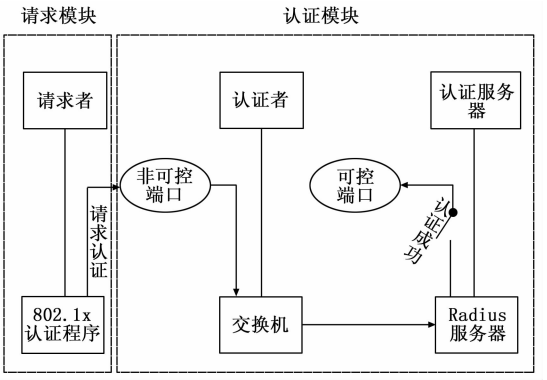


图 2 认证体系结构

和被分配任务的 TaskTracker 节点充当了认证体系中的认证模块。交换机作为认证者通常为两层可网管的交换机，其作用就是在端口上启用 802.1x 进行认证、指定认证服务器的信息以及根据认证结果控制端口状态等。

TaskTracker 节点作为认证服务器为认证系统提供认证服务的实体，本系统使用 freeradius 服务器来实现认证服务器的认证和授权功能。freeradius 是一款开源的、免费的服务器软件，不仅具有一般 radius 服务器所具有的认证、授权和计费功能，还可以灵活的配置和发送请求[10]。在 TaskTracker 节点上安装 freeradius 软件，并对其进行相关的配置，从而实现在 Hadoop 平台下的认证服务器。由于需要在服务器系统通过获取 mysql 数据库中存储的用户名和密码等认证信息，以此来校验 JobTracker 节点发送来的认证信息，所以需先安装 mysql 数据库。安装完数据库之后再安装 freeradius 并对其进行配置。配置完成后，启动服务器，在终端输入 radtest test test 192.168.0.147 100 testing123，可以看到服务器收到了请求，并返回成功接收信息。如图 3 所示。

```
ubuntu@hadoop:~$ radtest test test 192.168.0.147 100 testing123
Sending Access-Request of id 19 to 192.168.0.147 port 1812
        User-Name = "test"
        User-Password = "test"
        NAS-IP-Address = 192.168.0.147
        NAS-Port = 100
rad_recv: Access-Accept packet from host 192.168.0.147 port 1812, id=19, length=
20
ubuntu@hadoop:~$
```

图 3 配置成功后服务器状态

3 实验结果与分析

根据上述的设计方案，利用实验室的资源搭建了一个 Hadoop 集群的实验平台，实现对 TaskTracker 节点进行访问控制的验证。整个实验平台由 3 台普通的 pc 机和一个可网管的锐捷交换机组成，操作系统选择 ubuntu，支持 Java 运行环境的 JDK 版本为 jdk－8u51－linux，Hadoop 软件版本为 Hadoop－1.2.1，实现认证功能的服务器软件为 freeradius。

在 Hadoop 的实验平台，将 pc1 作为 JobTracker 节点，pc2 和 pc3 作为 TaskTracker 节点。平台结构如图 4 所示。

在 master 节点的 hadoop 安装目录下使用 bin/start－all.sh 命令开启 hadoop 集群，然后可以通过 web 模式看到 MapReduce 基本信息的管理界面如图 5 所示，通过该界面我们可以看到 MapReduce 系统的运行状态，启动时间等基本信息，以及集群正在运行的 maps 和 reduces 数目，活跃节点数，Maptask 的容量，Reducetask 的容量等的摘要信息。

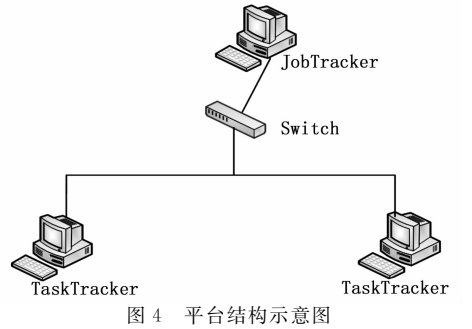
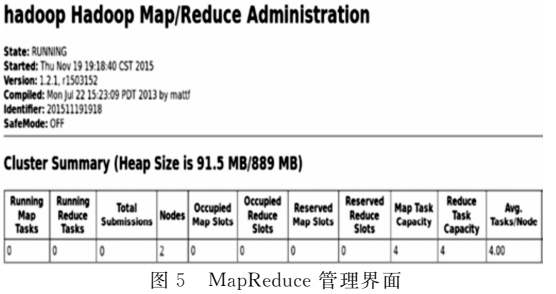


图 4 平台结构示意图



hadoop Hadoop Map/Reduce Administration

State: RUNNING
Started: Thu Nov 19 19:18:40 CST 2015
Version: 1.2.1, r1503152
Compiled: Mon Jul 22 15:23:09 PDT 2013 by mattf
Identifier: 201511191918
SafeMode: OFF

Cluster Summary (Heap Size is 91.5 MB/889 MB)

Running Map Tasks	Running Reduce Tasks	Total Submissions	Nodes	Occupied Map Slots	Occupied Reduce Slots	Reserved Map Slots	Reserved Reduce Slots	Map Task Capacity	Reduce Task Capacity	Avg. Tasks/Node
0	0	0	2	0	0	0	0	4	4	4.00

图 5 MapReduce 管理界面

为了实现对 TaskTracker 节点进行访问控制的验证，开启 TaskTracker 节点的 freeradius 服务器来进行实验测试。我们采用系统的 wordcount 程序对输入文件进行单词个数统计，并汇总所有统计结果输出。输入实验测试信息，弹出用户输入凭据，如图 6 所示。

图 6 用户输入凭据

输入正确的用户信息后，点击连接。实验运行结束，可以通过 web 模式看到实验结果信息的管理界面，如图 7 所示。

通过在 Hadoop 的实验平台运行 wordcount 程序作业对 TaskTracker 节点进行身份认证的实验，实现了 Hadoop 平台中对 TaskTracker 节点进行身份认证的安全控制。从实际效果来看，系统整体运行状况平稳，达到了系统设计目标。

4 结束语

Hadoop 作为一个能够对海量数据进行分布式存储和计算的平台，在各个领域得到了广泛的应用。MapReduce 分布式计算作为 Hadoop 的核心技术之一，主要对海量数据进行分布式计算处理。但是目前在 MapReduce 计算的环境中缺乏一个安全机制，无法保证对 TaskTracker 节点进行任务分配的是合法安全的用户，本设计通过使用 802.1x 协议的网络准入控制对访问者进行身份验证，有效地避免了非法用户访问 Task-

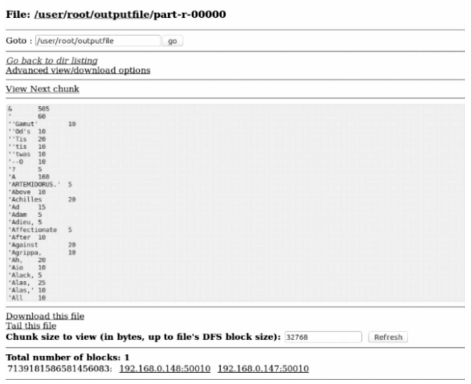


图 7　实验结果界面

Tracker 节点，增强了 Hadoop 平台的安全性、可靠性，为建设安全的大数据环境提供了很好的技术支持。

参考文献：

[1] 李晓蕾. 基于 Hadoop 社交网络分析平台的设计与实现 [J]. 计算机测量与控制，2014 (12)：4094－4097.

[2] 郑晓薇，项　明，张大为，等. 基于节点能力的 Hadoop 集群任务自适应调度方法 [J]. 计算机研究与发展，2014 (3)：618－626.

[3] 曹　旭. Hadoop 平台下海量日志数据处理模型的研究及改进 [D]. 杭州：浙江理工大学，2013.

[4] 何　翔，李仁发，唐　卓. 一种异构环境下的基于 MapReduce 任务调度改进机制 [J]. 计算机应用研究，2013 (11)：3370－3373，3379.

[5] 任萱萱. 基于 Hadoop 平台的作业调度研究 [D]. 天津：天津师范大学，2011.

[6] 王昌旭，周振柳，许榕生. 网络接入安全控制研究 [J]. 计算机应用与软件，2008 (11)：92－94.

[7] 朱　兵，周　爽，张　攀. 基于主机信息的 802. 1x 的改进方案设计 [J]. 信息安全与技术，2010 (7)：114－116.

[8] 鹿凯宁，韦乃文. 802. 1x 协议安全性能的改进 [J]. 电子测量技术，2007 (1)：107－109.

[9] 黄永锋，王　滨，许晓东. RADIUS 在 802. 1x 中的应用 [J]. 计算机工程与设计，2006 (5)：798－801.

[10] 杨凌凤. 使用 USBKey 提高 FreeRadius 证书认证的安全性 [J]. 计算机安全，2008 (2)：42－44.

(上接第 113 页)

表 1　机载收发信机接收灵敏度测试值

序号	1	2	3	4	5	6	7	8	9	10	11	12
可调衰减器衰减值	53	52	51	51	51	52	55	54	55	55	56	56
接收门限电平(≤－110dBM)	－116	－115	－114	－114	－114	－115	－118	－117	－118	－118	－119	－119

从 1 开始，每个周期发送数值累加 1，当数值累加到 100 并发送完成后，再将数值设回 1，循环累加往复发送；机载飞控计算机在遥控数据帧中取出相应的测试数据，并填到遥测数据帧中发送回地面飞行控制计算机。

图 9 为时间延迟测试数据，实线表示地面飞行控制计算机接收的测试数据，虚线表示地面飞行控制计算机发送的测试数据；横轴表示时间，纵轴表示设定的测试数据 1～100。

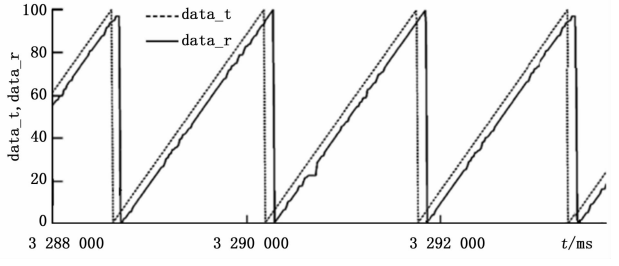


图 9　时间延时测试数据

选取图中每个锯齿波顶端为分析点，实线滞后于虚线，从图中可测出时间延迟平均为 90 ms，满足模型自由飞试验对于系统的实时性要求。

对于测控信号传输延迟主要由两个部分构成，一部分为采用编译码的传输延迟，一部分为接口处理时的缓存延迟。对于编译码来说，遥测和遥控采用了 (4，3，7) 卷积编码，信道延迟约为 370 个码元。若遥控速率按 50 kbps，遥测速率按 100 kbps计算，则遥控链路的编译码延迟为 7.4 ms，遥测链路的编译码延迟为 3.7 ms。接口处理时的缓存延迟一般为 3 个数据帧的延迟，同样按照上面假设的速率值进行计算，则遥控链路的编译码延迟为 30 ms，遥测链路的编译码延迟为15 ms。为此，遥控数据传输延迟总的时间可小于 40 ms，遥测数据传输延迟总的时间可小于 20 ms。所以传输延迟总时间约为 60 ms，但是系统总的延时需要考虑地面及机载计算机的处理时间，还有在其他介质中传播所花费的时间，所以测出时间延迟为 90 ms，大于理论计算值是合理的。

4　小结

从系统的试验结果来看，系统达到了某模型自由飞验证试验技术要求，并成功完成了某模型自由飞验证试验。系统设计中，多项测控与信息传输关键技术的突破与应用，使得系统的性能得到了很大的提升。

参考文献：

[1] 安玉娇，江辉军，郑　浩. 带动力模型自由飞试验测试系统设计与实现 [J]. 测控技术，2015，34 (6)：132－133.

[2] 刘尚民，赵　磊. 电传飞机模型自由飞试验飞行控制技术研究 [J]，飞行力学，2012，30 (1)：83－86.

[3] 周详生. 无人机测控与信息传输技术发展综述 [J]. 无线电工程，2008，38 (1)：30－33.

[4] 吴　潜，雷　厉. 多无人机测控与信息传输系统的技术与发展 [J]. 电讯技术，2008，48 (10)：107－111.

[5] 柴　霖. 临近空间飞行器测控与信息传输系统频段选择 [J]. 航空学报，2008，29 (4)：1007－1012.

[6] 方　威，王　锋，丁团结. 无人机数据链性能研究 [J]. 飞行力学，2010，28 (6)：68－71.