十年一日，深入成就深度
业精于专，专注成就专业

数字有机体工作平台及抗毁容灾系统

用户手册

成都天心悦高科技发展有限公司

2015 年 10 月

版权声明

数字有机体工作平台及其附属产品（含 Windows 客户端调度接口库）的版权属于成都天心悦高科技发展有限公司所有。任何组织和个人未经成都天心悦高科技发展有限公司许可与授权，不得擅自复制、更改该软件的内容及其产品包装。

本软件受版权法和国际条约的保护。如未经授权而擅自复制或传播本程序（或其中任何部分），将受到严厉的刑事及民事制裁，并将在法律许可的范围内受到最大可能的起诉！

成都天心悦高科技发展有限公司
地址：成都市武侯区棕南小区
电话：028－83318559
邮编：610054

目 录

1 引言

1.1 编写约定

非常感谢您使用成都天心悦高科技发展有限公司的产品，本公司将竭诚为您提供最好的服务。

本手册假定读者已参阅过《数字有机体系统安装指南》并已了解其中有关数字有机体工作平台的安装与配置的部分。手册中所有的描述与介绍均基于已安装完毕的数字有机体工作平台。

为统一对术语的理解，第二章《数字有机体工作平台介绍》将详细介绍数字有机体工作平台中的重要术语，附录 A 列出了平台中所有相关术语及释义。

本手册假定您对 Linux 的 shell 命令有基本的了解，附录 B 也提供常用 Linux 命令的速查。数字有机体工作平台用户命令的使用说明请参见本手册数字有机体工作平台维护部分。

文中出现的'#'号表示命令行提示符。

命令格式描述中的斜体字表示应由用户填充的部分，"[]"表示命令中可选的部分。

为了阅读方便，我们以灰底黑框的形式呈现某些重要的配置操作。

1.2 如何使用本手册

本手册内容大体分为四个部分。

- 第一部分：包括第 1 至第 2 章。此部分主要介绍数字有机体工作平台的功能及手册的使用
- 第二部分：包括第 3 至第 7 章。此部分主要介绍数字有机体工作平台的日常使用，所有在本部分介绍的命令均基于传统的字符界面。
- 第三部分：包括第 9 章。此部分主要介绍数字有机体工作平台的安全管理，对安全性要求较高的用户可着重参考。
- 第四部分：包括第 8 章，第 10 至第 12 章，此部分主要介绍通过 Web 界面对数字有机体工平台的日常管理以及维护，远程管理员可着重参考。

您既可以按照顺序阅读每一章，也可以根据目录索引直接获得所需的信息。

1.3 相关文档

数字有机体工作平台相关的手册共有三本。

- 《数字有机体系统安装指南》描述如何安装数字有机体系统。
- 《数字有机体工作平台及抗毁容灾系统用户手册》，即本手册，描述数字有机体工作平台的日常使用和维护，主要针对平台管理员、维护员和使用者。

- 《数字有机体工作平台及抗毁容灾系统开发手册》描述如何基于数字有机体工作平台研发应用系统，主要针对应用开发人员。

1.4 如何获得技术支持

在您遇到问题时，请首先联系您的产品提供商。大多数问题都可以在产品提供商的技术支持人员的帮助下得以解决。

您也可以通过产品提供商致电本公司的技术服务热线：028-83318559，获得电话技术支持。您还可以发送邮件，邮件地址是：tianxinyue@126.com。如果您确实需要本公司提供上门服务，本公司将竭诚为您服务。

2 数字有机体工作平台介绍

2.1 系统简介

数字有机体系统（英文名称为 Digital Organism System，缩写为 DOS）是在刘心松教授带领下，由成都天心悦高科技发展有限公司的研发人员前后千余人次，经过三十多年的技术积累，研发成功的基础系统。

研发这种系统的原始宗旨是向生物特别是人类个体和群体的结构、机理和特性逼近，是一种人能化的新的系统模式。这种系统集操作系统、数据库系统、大规模存储、抗毁容灾、高伸缩、高智能、高灵活、自搜索、自传播、自复制、自修复、自重构、自适应、系统间的兼容性、群体间的协作性、对资源的动态管理调度合理配置、大小新旧机器混合使用等特性为一体，是一个整体解决方案，是面向所有应用的统一的（应用）系统平台。

数字有机体系统主要由数字有机体工作平台、数字有机体抗毁容灾系统、数字有机体工作库、数字有机体大规模存储与管理系统、数字有机体安全系统组成。这是从底层作起的一个一体化平台，可以在此平台上开发任何应用，形成任何应用系统。例如现在已有的应用系统就有数字有机体流媒体系统、数字有机体监控系统、数字有机体会议系统、数字有机体网关、数字有机体管理系统、数字有机体控申系统、数字有机体侦查指挥系统等。

数字有机体工作平台是数字有机体系统的核心部分，提供大规模分布式并行文件系统服务、任务调度服务、用户管理服务、存储管理服务和安全支撑等，并具有 WEB 化的管理界面。

本文有时将数字有机体工作平台及抗毁容灾系统，数字有机体工作库及大规模存储与管理系统和数字有机体安全系统统称为数字有机体系统。数字有机体工作平台及抗毁容灾系统含盖常规操作系统但远高于常规操作系统，是一个在 Linux 之上的、面向很多应用的、统一的、人能化的应用系统平台。数字有机体工作库及大规模存储与管理系统含盖常规数据库系统但远高于常规数据库系统，是一个在 Mysql 之上的、面向很多应用的、统一的、人能化的应用数据平台。

有时将数字有机体工作平台及抗毁容灾系统简称为数字有机体工作平台甚至工作平台。

有时将数字有机体工作库及大规模存储与管理系统简称为数字有机体工作库甚至工作库。

数字有机体工作平台是数字有机体系统的基础系统。它将物理分布离散的主机有机地整合起来，将各主机提供的共享文件及服务均视为平台的资源，通过智能调度算法为每一个用户提供所请求的资源。它主要提供以下功能：

- 构建可以无限扩展的共享存储空间。它力图聚集所有计算机共享出的存储空间，以建立一个可以无限扩展的存储系统。每台计算机通过一定的方式共享出自己的一部分存储空间，作为共享存储空间的一部分。所有共享存储空间被统一使用。

每台计算机都可以在共享存储空间中创建和存储文件。

- 无处不在的文件访问能力。网络上的任何计算机都可以通过一定的方式加入系统，成为系统的使用者。加入系统的每台计算机都可以快速的访问系统中的文件，也可以在共享存储空间创建共享文件。
- 具有和传统文件系统类似的可管理能力。现有的各种研究主要强调对资源的共享能力。但是，只有共享能力却无法管理和控制的系统是无法产生有效价值的。因此，如何在保证共享能力的前提下，获得对资源的可管理能力即是关键的问题。数字有机体工作平台借鉴传统文件系统，采用单一的目录树管理所有文件；并通过有效的副本管理机制控制资源。
- 智能任务调度能力。系统能够根据每个节点的状况将任务分配到较优的节点上，并保证整个系统中负载的均衡性。
- 适当的安全控制机制。

数字有机体工作平台中最重要的资源即平台中所有主机所共享的文件，故平台对共享文件进行了高效的管理。文件管理的目的是将分布在各个数字有机体站内的文件资源组织起来，以两种方式提供给用户使用。一种方式是用户按照信息的名字访问资源，即传统文件访问方式；另一种方式是用户按照条件查找所有符合条件的信息，即资源发现。文件管理的主要目的是：

- 实现广域网络文件系统，并以传统文件系统方式提供给用户使用。
- 保证文件的可靠性，使其不因为一些计算机的死亡而不能被访问。
- 保证高速的访问文件。
- 对资源的更新保证一致性。
- 支持用户自定义规则以指导或者限制文件的副本放置。
- 能够将最热门的文件放置到从网络路由距离上讲最靠近大多数用户的地方。
- 保证系统的存储负载均衡，即各台计算机、各数字有机体站的负载应当与其性能相当，相互之间的存储负载是均衡的。
- 保证能够快速的找到文件。

数字有机体平台中资源的另一种形式是服务。平台的每个参与者都可以对外提供服务。数字有机体工作平台管理这些服务的描述信息，以便其他参与者可以尽快的找到最适合自己的服务。也就是说，数字有机体工作平台的任务是协助参与者寻找最适合自己的服务提供者。服务管理体现为两个方面，一是对系统中所有服务描述信息的记录、组织和对服务提供者的监控。其次是任务的调度，即帮助参与者找到最好的服务提供者。前一个任务我们将其划入资源管理的范畴，后一个任务作为独立的任务调度系统存在。服务资源管理提供的功能如下：

- 组织和保存系统中的所有服务描述信息，包括客户提供的服务信息。
- 提供服务信息注册和注销的机制。
- 监控服务的有效性，及时剔除失效的服务的描述信息。

- 提供快速的服务查找机制。

任务调度系统利用服务资源管理系统提供的服务查询能力，获得可以提供服务的节点，然后进行任务调度。任务调度子系统所提供的功能有：

- 对站和主机的性能差异进行适当处理。
- 有效的管理网络的带宽，从而可以找到网络情况最好的服务路线。
- 找到最靠近用户的可以提供资源的服务器。
- 在所有主机间进行有效的负载均衡。

服务管理所提供的其他的一些功能：

- 具有良好的缓存机制，包括对信息的缓存，对系统状况知识等的缓存等，从而有效的提高系统性能。
- 处理节点和站的异常，保证系统的可靠性。
- 保证节点和站的自由加入和退出，从而保证具有良好的可扩展性。

2.2术语释义

- **数字有机体系统**：将若干计算机通过宽带网络互联，根据需要采用现有的和最新的理论和技术，使互联而成的系统具有生物抗体之特性，则该系统即为数字有机体系统。数字有机体系统包括数字有机体工作平台和数字有机体工作库。
- **数字有机体工作平台**：数字有机体工作平台是一个基础平台，将分布在较大范围内的大量计算机聚合成一个有机的整体，以便向应用提供强大的处理、存储和通信等能力，从而满足大规模网络应用的需要，同时具有抗毁容灾功能。目前包括资源管理子系统、任务调度子系统、虚拟网络服务、抗毁容灾和安全子系统等。
- **数字有机体工作库**：它的全称是数字有机体工作库及大规模存储系统，简称为数字有机体工作库，有时也被称作数字有机体数据库系统。它是融合数据库技术、计算机网络技术和并行处理技术开发的数据库系统，简称为 DosSQL，是由分布在高速广域网内的大量高性能服务器构成的能够提供高度并行数据处理、大存储容量、高可靠性及可用性的数据库系统。它能满足当前高度复杂的数据处理需求和高可靠性要求，同时能够根据用户的需求变化进行动态配置，实现系统的动态伸缩和升级。数字有机体数据库系统不仅具有集中式数据库系统的所有功能，而且比集中式数据库系统具有更高的效率、可靠性及可用性，能满足多媒体等复杂数据的处理。该系统对外透明，即用户以使用单机数据库系统一样的方法使用这个系统，而不用了解这个系统的具体构成和规模。这样的系统具有极高的性价比，能够应用于大量复杂数据处理的环境，其高效性、高性价比和动态伸缩性已成为宽带网络运营商以及电子政务等应用的首选平台，能够为电子政务、视频点播等诸多应用提供数据管理，具有重大的应用前景。
- **数字有机体文件系统**：把广域网上加入系统的各节点的共享空间组成一个单一映

像的存储空间，并提供传统文件系统支持的分布式网络文件系统。它可以看作数字有机体工作平台的资源管理子系统。

- **站**：一个数字有机体系统由若干个站组成。一个站在一个 IP 广播域内，站内由多个紧密耦合的节点组成。这些节点能共享站内信息，且可直接操作站内副本资源。
- **节点**：组成数字有机体的任一计算机均是系统中的一个节点，总是属于系统中的一个站。

2.3 部署拓扑结构

数字有机体工作平台的对硬件的要求不高，且部署非常灵活。常见的应用环境如图 *2-1* 所示。系统由多个数字有机体站构成。每个站包括多个节点，即多台服务器。用户设备可以是各种计算机或者终端设备，如 PC 机、笔记本电脑、PDA、智能手机等。数字有机体工作平台运行在每台服务器上。

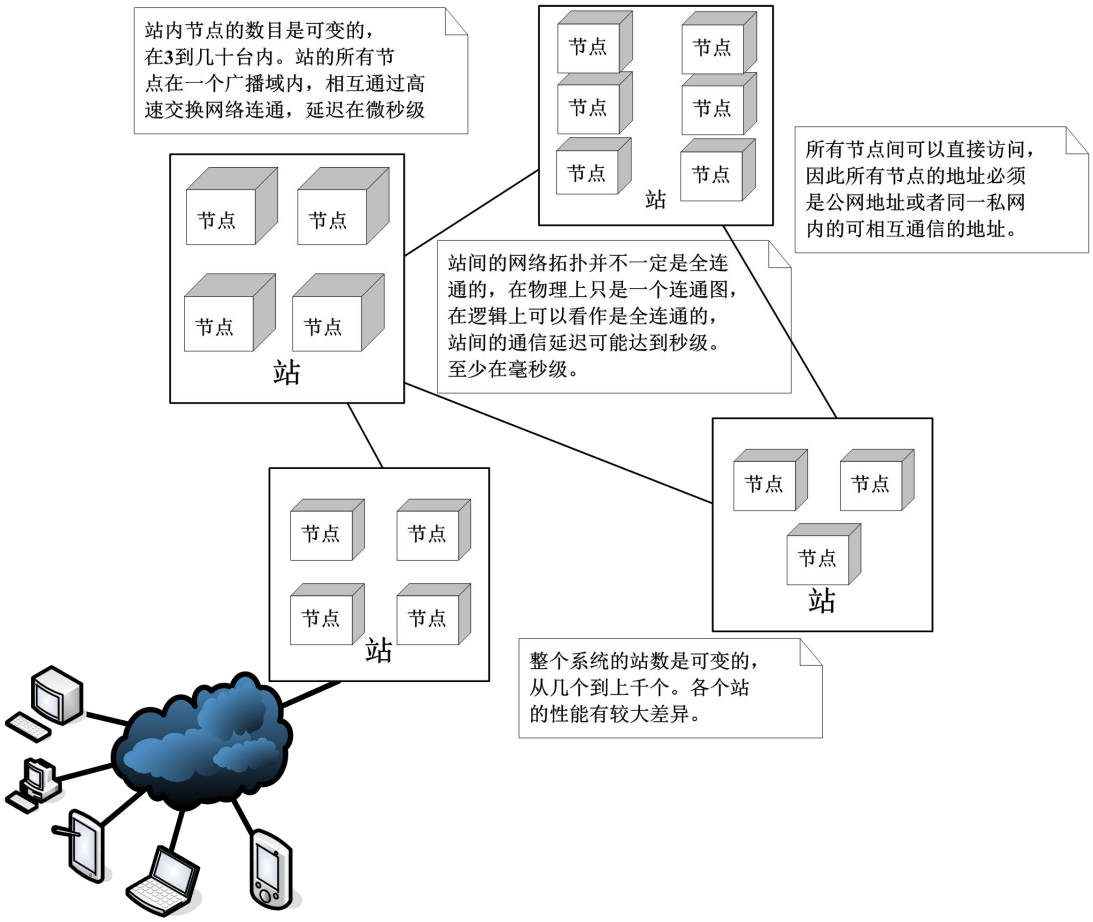


图 2-1 数字有机体系统部署结构

3 用户管理

数字有机体工作平台具有一套完整独立的用户管理机制。在本平台中，所有使用者均须持有一个平台账户作为其身份标志，而每个平台账户又隶属于一个账户组。平台对每个用户进行身份认证，非法的账户将不能登陆本平台继而使用其所提供的服务。注意，相同的账户可以在不同主机或不同终端上同时登陆并获得相同的权限。本章将具体介绍平台账户及账户组的管理方法。

3.1 用户管理相关命令

本节介绍的命令用于数字有机体工作平台的用户账户管理，包括用户账户的信息查询、增添、删除及冻结。

1) 命令名：do_user

功能：查看指定用户信息，如果不指定用户则默认查看 DOSroot 的信息。

DOSroot 用户可以查看任意用户的信息，其他用户则只能查看自己而不能查看别人的用户信息。

使用方法：**do_user user …**

{do_user 用户名 …}

2) 命令名：do_useradd

功能：为指定的用户组添加一个新用户。

本命令为特权命令，只有 DOSroot 用户有权限使用。

使用方法：**do_useradd [-u user] [-g group] [options]**

do_useradd [-u 用户名] [-g 组名] [选项]

参数释义：-q quota of user, default is 100MB
（新用户的配额，缺省则默认为 100MB）

-r real user, default is USER
（用户的真实姓名，缺省则默认与账户名相同）

-t type, it maybe: g - group manager, s - station manager, n - normal user
（新用户的类型，可以的取值是 g、s、n，分别表示组管理员，站管理员、普通用户）

-s real space of user, default is quota's 3 times
(新用户的真实可用存储空间，缺省则默认为配额的三倍)

-d description of user
（新用户的描述信息）

3) 命令名：do_userdel

功能：删除一个指定用户，用户账户名不能缺省，DOSroot 用户不能被删除。

本命令为特权命令，只有 DOSroot 用户有权限使用。

使用方法：**do_userdel user**

{do_userdel 用户名}

4) 命令名：do_usermod

功能：修改指定用户的属性

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_usermod –n user [options]**

{do_usermod –n 用户名 [选项]}

参数释义：-q quota of user, default is 100MB
（修改用户的配额，缺省则默认为 100MB）

-r real user, default is USER
（修改用户的真实姓名，缺省则默认与账户名相同）

-t type, it maybe: g - group manager, s - station manager, n - normal user
（修改用户的类型，可以的取值是 g、s、n，分别表示组管理员，站管理员、普通用户）

-s real space of user, default is quota's 3 times
(修改用户的真实可用存储空间，缺省则默认为配额的三倍)

-d description of user
（修改用户的描述信息）

5) 命令名：do_userfreeze

功能：冻结(暂停使用)指定用户账号

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_userfreeza user**

{do_userfreeze 用户名}

3.2 组管理相关命令

本节介绍的命令用于数字有机体工作平台用户组的管理，包括用户组的信息查询、增添以及删除。

1) 命令名：do_group

功能：查看系统用户组信息

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_group group …**

{do_group 组名 …}

2) 命令名：do_groupmod

功能：修改系统用户组描述

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_groupmod –n group –d description**

{do_groupmod –n 组名 –d 描述信息}

3) 命令名：do_groupadd

功能：为系统添加一个用户组

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_groupadd –n group –d description**

{do_groupadd –n 组名 –d 描述信息}

4) 命令名：do_groupdel

功能：删除指定的用户组

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_groupdel group**

{do_groupdel 组名}

5) 命令名：do_groups

功能：查看系统中所有的用户组

本命令为特权命令，只有 DOSroot 用户有权限使用

使用方法：**do_groups**

{do_groups}

3.3 口令管理相关命令

本节介绍的命令用于数字有机体工作平台用户账户口令的设置和修改。

1) 命令名：do_passwd

功能：为指定用户设置登陆口令，若用户名缺省则默认为 DOSroot。

DOSroot 用户可以为任何用户设置登陆口令，其他用户只能设置自己而不能设置他人的登陆口令。

使用方法 **do_passwd [user]**

{do_passwd [用户名]

4 进程用户标识与访问控制

在数字有机体工作平台自有用户系统的基础上，建立了独立于单个主机的访问控制机制。这样，数字有机体工作平台的用户可以从任何一台主机上登录数字有机体系统。系统依照全局一致的访问控制策略控制用户对文件等资源的访问。

对数字有机体文件系统来说，数字有机体工作平台采用 Linux 系统的访问接口作为文件访问接口，其访问控制策略和 Linux 文件访问控制策略相同。为此，用户登录系统后执行的所有操作都应以该用户的身份进行。但是，在数字有机体工作平台中，代表用户进行操作的实际上是进程。因此，需要给每个进程标识一个用户身份。

数字有机体工作平台实现了 Unix 文件访问控制（即自主访问控制机制）和文件访问控制列表。和单机 Linux 系统不同的是，访问控制机制中的用户是数字有机体系统用户，访问对象是数字有机体文件系统中的文件。

4.1 获取用户标识

系统提供三种方式给进程赋予身份。第一种方式是用户在终端中显式登录系统。第二种方式是通过配置文件指定进程用户身份。第三种方式是进程主动调用登录函数登录系统。第三种方式请参考《数字有机体工作平台及抗毁容灾系统开发手册》。以下将前两种方式的使用方法依次介绍。

4.1.1 通过登陆平台获取用户标识

用户可以在任意系统节点上使用 do_su 命令登陆数字有机体工作平台。登陆后，登录进程及其后继子进程均带有登录用户标识，并能进行与其身份权限相符的操作。通常，登录进程是一个 Shell 进程，因此在该 Shell 内执行的后继命令（转化为 shell 的子进程）自然也就有登录用户标识。 注意：登录前进程创建的子进程，或者用户在其他终端上的进程不会具有数字有机体系统的用户身份。

下面介绍 do_su 命令的使用方法。

命令名：**do_su**

功能：以指定用户身份登陆数字有机体工作平台

使用方法：**do_su 用户名**

(缺省用户名为 DOSroot)

成功登陆提示如下：

```
root@server190:/home/dos# do_su
Now it's DOSroot login:
Password:
Login DoOS successfull.
```

登陆失败提示如下：

```
server137:~# do_su
Now it's DOSroot login:
Password:
Login failed for password is error.
```

注意事项：do_su 可能会失败，常见的失败原因如下：

1） 用户身份认证未通过，即用户输入的账户或口令有误。

2） 数字有机体工作平台未启动或启动出错。

4.1.2 通过配置文件获取用户标识

对于某些既没有调用数字有机体工作平台的登录接口（例如已有的服务程序，Tomca 和 ftp 等），又不便使用 do_su 登录工作平台后再启动的进程，则需要采用本方法来赋予用户标识。我们称这样的进程为特殊进程。数字有机体工作平台的进程身份标识功能允许系统管理员通过配置文件安全有效的指定这些进程的身份。这样既可以兼容原有的应用程序，又可以保证访问的有效控制。

系统进程用户标识配置文件是位于/etc/下的 credent.conf 文件。文件中的每一行指定一个进程的身份标识，格式为：

可执行文件全路径名=本地用户名　授权用户名　授权组名

其中的空格可以有多个。以“#”号开始的行是注释行。该行的含义为：给由可执行文件全局路径名所决定并且由本地用户名所代表的用户的进程，指定由用户名和组名所决定的数字有机体用户身份。

注意：

1）进程的可执行文件全路径名应该是进程访问文件系统时的执行程序。某些服务，例如 Tomcat，通常通过脚本或者系统服务管理命令来启动；而最终驻留在系统中，访问数字有机体文件系统的程序其实不是启动脚本。Tomcat 服务的访问文件系统的程序其实是 Java。您可以在服务启动完成后，用“ps”命令查看服务的进程号，然后查看“/proc/进程号/exe”符号链接文件的指向的文件。例如：

```
root@server190:/proc# l /proc/19270/exe
lrwxrwxrwx. 1 tomcat6 tomcat6 0 5 15 11:58 /proc/19270/exe ->
/usr/lib/jvm/java-6-openjdk-amd64/jre/bin/java
```

其中“/usr/lib/jvm/java-6-openjdk-amd64/jre/bin/java”是 tomcat 服务进程的最终执行程序。

2）等号的前后不要有空格等任意字符。

3）本地用户名需要是进程的当前有效用户名。注意：有些进程的启动用户和有效用户是不同的，因为程序可以使用系统调用设置有效用户。如果不能确定进程的有效用户，则可以用星号（*）表示任意本地用户。

4）要指定的数字有机体工作平台用户通过授权用户名和授权组名指定，即指定进程所属的用户以及进程所属的用户组。这里必须明确指定，不能使用通配符等。如果指定的用

户或者组不存在，进程仍然是没有合法属性的。

5）配置行中的每项都是必须的，如果配置内容不全，则该行配置被丢弃。

建议管理员专门为每个应用注册一个数字有机体工作平台用户和组，并设置好他们的权限（如何设置可以参考本文档其它部分），然后再通过这个配置文件来授权。下面是以下常见的示例。

```
/usr/sbin/vsftpd=group2k     DOSroot     DOSroot
```

该行配置指定 vsftpd 的 group2k 用户映射为 DOSroot 用户和组。这时，group2k 用户通过 vsftpd 服务访问数字有机体文件系统时，将具有 DOSroot 身份，即系统超级用户身份。不过，建议尽量不要给服务指定超级用户身份。在下面的例子中，在数字有机体工作平台上已建立了一个名为 ftpuser 的用户和名为 ftpusers 的组。希望将所有通过 vsftpd 访问数字有机体文件系统的用户都映射到该用户和组，则可以使用以下配置行。

```
/usr/sbin/vsftpd=*    ftpuser     ftpusers
```

可以在数字有机体文件系统中创建一个空目录，将目录的所有者和组分别设置为 ftpuser 和 ftpusers ,并且设置其他用户不能访问该目录。然后将其作为 vsftpd 服务的根目录。这将限定通过 vsftpd 服务上传的文件只能写到该目录下，而且其他用户不能访问。

注意，如果同一个服务指定了两条以上的配置，则按照精确匹配优先的原则选择使用哪个配置行。例如下面这两行配置。

```
/usr/sbin/vsftpd=*              test          test
/usr/sbin/vsftpd=group2k     DOSroot     DOSroot
```

这样除了 group2k 被授予 DOSroot 的权限外，所有其他的用户都被授予 test 用户的权限。

注意：如果两行的可执行文件全路径名和本地用户名都相同，则以前一行为准，后面一行被丢弃。

4.2 文件访问控制

数字有机体工作平台实现了传统 Unix 文件访问控制和文件访问控制列表。

依据传统 Unix 文件访问控制标准，数字有机体工作平台中每个文件都有一个拥有者和所属组，即文件的 owner 属性和 group 属性。这里的用户和组必须是数字有机体工作平台的用户和组，否则系统将显示一串数字作为且用户和组，这时是无效的。文件的拥有者和超级用户（即 DOSroot）用户可以设置文件的拥有者和所属组。设置指令为 Linux 的 chown 命令。应用程序可以调用 chown 和 chgrp 函数。

传统的 Unix 文件访问控制无法精确设置每个用户对文件的访问权限，这就需要文件访问控制列表（ACL）机制了。文件访问控制列表也称为文件 ACL(Access Control List)，是用于限制不同用户对文件所能进行的操作的一种机制。数字有机体工作平台支持标准的 ACL 机制，并通过文件扩展属性方式实现。文件的扩展属性可以通过数字有机体工作平台的图形管理界面进行管理，也可以使用 Linux 原有的命令来管理。其中，查看文件 ACL 属性的命令为 stat 和 getfacl，设置和修改的命令为 setfacl, chacl。下面将依次介绍这些命令的

使用方法。应用程序可以通过对应的系统调用或者 C 库函数进行操作。

4.2.1 文件 ACL 属性查看

文件的 ACL 属性可以使用 getfacl 命令查看，下面介绍其使用方法

命令名：getfacl

功能：获取并展示文件的 ACL 属性。

使用方法：getfacl [options] file

getfacl [选项] 文件名

参数释义：-a Display the file access control list.
（展示指定文件的访问控制列表）

-d Display the default access control list.
（展示指定文件的默认访问控制列表）

-s Skip files that only have the base ACL entries (owner, group, others).
（跳过那些只含基本 ACL 项的文件）

-R List the ACLs of all files and directories recursively.
（递归列出所有文件及目录的 ACL 属性）

-L Logical walk, follow symbolic links to directories. The default behavior is to follow symbolic link arguments,and skip symbolic linksencountere in subdirectories. Only effective in combination with -R.
(逻辑遍历，也就是会跟随指向目录的软连接查下去。只有当指定了 -R 选项后才有效)

-P Physical walk, do not follow symbolic links to directories. This also skips symbolic link arguments. Onlyeffective in combination with -R.
（物理遍历，也就是不会跟随指向目录的软连接查下去。只有当指定了-R 选项后才有效

4.2.1 文件 ACL 属性设置和修改

与文件 ACL 属性设置相关的命令接口有 chmod、chown、setfacl、chacl。 chmod 可以改变文件的权限位。chown 可以改变文件的拥有者和所属组。setfacl 可以设置文件的 ACL 属性。chacl 可以改变文件的 ACL 属性。注意：只有超级用户和文件所有者才可以改变文件的属性。下面介绍最常用的 setfacl 和 chacl 命令的使用方法。

命令名：setfcal

功能：设置文件的 ACL 属性

使用方法：setfcal [options] [-m|-x acl_spec] [-M|-X acl_file] file...

{setfcal [选项] [-m|-x acl 属性]} [-M|-X acl 文件] 文件名...

参数释义： -m|-x 该选项后紧跟需要修改(-m)或删除(-x)的 ACL 属性

-M|-X 该选项后紧跟含有需要修改(-M)或需要删除(-X)的 ACL 属性的文件

-b Remove all extended ACL entries. The base ACL entries of the owner, group and others are retained.
（移除所有的扩展 ACL 项，基本的所有者、组、其他 ACL 项将被保存）

-k Remove the Default ACL. If no Default ACL exists, no warnings are issued.
（移除默认 ACL 属性，如果文件没有默认 ACL 也不会显示警告）

-R Apply operations to all files and directories recursively.
（操作将递归地作用于所有文件及其子目录）

-L Logical walk, follow symbolic links to directories. The default behavior is to follow symbolic link arguments,and skip symbolic links encountered in subdirectories. Only effective in combination with -R.
（逻辑遍历，即会跟随指向目录的软链接查下去，只有当指定了-R 选项后才有效）

-P Physical walk, do not follow symbolic links to directories. This also skips symbolic link arguments. Only effective in combination with –R.
（物理遍历，即不会跟随指向目录的软连接查下去，只有当指定了-R 选项后才有效）

命令名：chacl
功能：修改文件的 acl 属性
使用方法：chacl [options] pathname
{chacl [选项] 文件路径名}

参数释义：-b Indicates that there are two ACLs to change, the first is the file access ACL and the second the directory default ACL.

（表示将会修改两种 ACL 属性，第一种是文件访问控制 ACL 属性，第二种是目录的默认 ACL 属性）

-d Used to set only the default ACL of a directory.
（只设置目录的默认 ACL 属性）

-R Removes the file access ACL only.
（只移除文件的访问控制 ACL 属性）

-D Removes directory default ACL only.
（只移除目录的默认 ACL 属性）

-B Remove all ACLs.
（移除所有的 ACL 属性）

5 存储管理

5.1 主机共享存储管理

数字有机体工作平台实现了服务器间的存储共享。所谓“共享存储”是指平台中所有主机都能进行存取的存储空间，每个节点对该存储空间的操作都好像是在对本地的存储空间进行操作一样，而实际上该存储空间的物理位置分布可能相当分散。

在数字有机体工作平台上，每台主机都可以共享出一个存储空间，作为存储数字有机体文件系统的共享空间。数字有机体文件系统的文件和目录将由系统自动放置到这个空间中。即使某台主机没有共享出存储空间，它也可以在数字有机体文件系统中创建文件和目录。这时这些文件和目录将存储在其他主机的共享存储空间中。反之，一台主机共享出存储空间后，它创建的文件也不一定都放在它共享的空间中，仍然可能放置在其他主机上，而且其他主机创建的文件也可能放置在该共享空间中。

通常，每台主机都共享出一个存储空间。这样，随着主机数量增加，系统总的存储容量也就得到扩展。当然，各台服务器的能力不同，共享出的存储容量也可以是不同的。

要强调的是，每台主机只能共享出一个存储空间，即一个用输出目录指定的存储空间。例如，某台主机有 5 块硬盘，通过主机上的阵列卡整合为一个逻辑设备，例如为/dev/sdb。现在需要将这个逻辑设备作为共享存储设备输出，则可以在该逻辑设备上创建一个包含全部空间的分区，即/dev/sdb1。然后将该分区挂载到输出目录，例如常用的/raid/data。最后使用输出共享指令将/raid/data 目录输出。

实际上，数字有机体文件系统中的文件被存储在输出目录下，例如上例的/raid/data 目录下。这样，如果该目录下的子目录没有再挂接其他分区，则所有该目录下的文件实际上占用的是该目录对应的分区的空间。以此类推，可以用其他方式将多个物理存储设备整合为一个逻辑存储设备，然后作为存储本地输出目录的存储空间。

要注意的是另一种情况，即输出目录位于的分区同时用于存储其他数据。例如，输出目录所在的分区也是数字有机体工作库存储分区等。这时他们将共享存储空间。可能出大量非数字有机体文件系统的文件占用空间，从而用光存储空间的现象。

5.1.1 添加共享

所谓添加共享是指：

1、把本节点指定的输出目录添加到数字有机体工作平台中，用于存储平台中的共享文件。

2、输出本节点指定输出目录中的文件供数字有机体工作平台的所有用户访问。

在系统已经完成安装并成功启动后，可以通过下面的命令完成添加共享：

命令：

do_shareadd exportname [sharename]

功能：

输出本节点指定目录下的磁盘空间，并按规则将指定目录下的文件资源添加到数字有机体文件系统中。

运行方式：

同 shell 命令，在本节点任意目录下运行该命令即可。

参数配置：

exportname（本地输出目录名）是必须的。它指出了本地将要输出的空间及文件。该目录将作为本节点的输出空间，成为系统存储空间的一部分。目录下的文件以参数 sharename 为命名规则，添加到数字有机体文件系统中去。

sharename（共享目录名）是可选参数。如果用户指定此参数，则本地共享目录 exportname 下已有的文件将被放置到数字有机体文件系统的 sharename(共享目录名)所指定的目录下。否则,文件被放置到系统的根目录(/dpfs)目录下,即共享目录名缺省为/dpfs。

本地输出目录下的已有文件进入数字有机体文件系统后的命名规则为：共享目录名+本地文件的全路径名去掉 exportname 路径名剩余的部分。例如，输出空间的路径为/mnt/disk1，其下有文件/mnt/disk1/test/file；若指定的共享目录名为/dpfs/dianying，则文件 /mnt/disk1/test/file 将作为数字有机体文件系统中的文件/dpfs/dianying/test/file 存在；如果没有指定共享名（sharename），则它将被作为数字有机体文件系统中的文件/dpfs/test/file 存在。

执行结果：

Success to do_shareadd 表示输出共享空间成功；

add export dir to system error 表示输出共享空间失败；

add file in export dir to system error 表示输出文件失败；

输出共享空间中的文件与系统中已有文件出现文件命名冲突时，系统将以屏幕信息形式提示用户。用户可选择以下两种方式解决命名冲突：1、由系统自动生成随机数后缀方式；2、用户手动输入新的文件名方式。

备注：

本系统限定每个节点只能输出一块共享共间。因此，在使用本命令输出共享空间前，要确保本节点在此之前没有输出共享空间。如之前已输出了共享空间，可通过下面的删除共享空间命令先将原来的共享删除后再作添加。

并不建议将一个已经有很多文件的目录输出到数字有机体文件系统中。因为这些输出的文件将不会立即创建副本，而且可能因与系统中已有文件同名等而难以处理。

执行完命令后可以通过下列方法确认共享是否添加成功：

1）执行 cat /etc/export_info 命令查看登记输出目录信息的 export_info 文件，看是否有类似下面的提示信息：

/mnt/disk1/　　/.dpfs/server153　　server153

此时本节点是 server153。信息的第一项是输出空间的目录位置，第二项是本站的其他节点需要加载本节点输出目录的 mount 目录，第三项是本节点的主机名。

2）执行 ls /dpfs 命令查看/dpfs 全局目录下面是否有本节点输出目录的文件。如果输出目录下本来没有文件，则不会有文件被输出。

共享添加成功后，如果以后想把一些文件共享出来供大家访问，可以通过 cp 命令把想共享的文件拷贝到/dpfs 目录下。这样操作以后，共享文件就可以被大家访问了，但是这些共享出来的文件并不一定放在本节点的输出目录里。

5.1.1 删除共享

如果本节点不再想提供输出目录给系统所有节点存储文件，可以通过删除共享命令来完成操作。

命令：

do_sharedel exportname

功能：

撤销输出的本地共享空间，在系统中移除该共享空间下的文件副本。

运行方式：

同 shell 命令，在本节点任意目录下运行该命令即可。

参数配置：

其中命令行参数- exportname（本地输出目录名）是必须的，其指出了本地将要撤消的空间及文件路径。

执行结果：

Success to do_sharedel 表示撤消共享空间成功；

Failure to do_sharedel 表示撤销失败。

执行完命令后可以通过下列方法确认共享是否删除成功：

执行 cat /etc/export_info 命令查看登记输出目录信息的 export_info 文件，如果文件为空则已经成功移除共享。

注意：类似于.*dir1，.*dir2 等这样的以点星(.*)开头的文件是系统特殊文件，这些点星文件用于存储全局共享空间（/dpfs）里的目录文件，每个点星文件里保存的都是此目录下的所有目录项。

注意：移除本机共享目录时，如果某个数字有机体系统中的文件仅在要移除的共享目录下有副本，而且没有其他主机可以创建新文件副本，则移除共享后，该文件将被删除。如果大量文件都仅在要移除的共享目录下有副本，则可能需要大量的时间来在其他主机上创建新副本，从而需要很长时间才能完成操作。同时，移除共享后，如果许多文件的副本数无法满足要求，则可能由系统自动创建新副本，从而导致系统出现大量负载。

当前，移除共享后，共享目录下的文件并没有删除。某些特殊发行版本可能会删除所有文件，以免系统中的文件泄露出去。

5.2 用户存储配额管理

数字有机体工作平台具有配额机制。通过配额机制，可以为每个用户设置允许产生的数据量和使用的存储容量。当用户拥有的数据量超过限制时，数字有机体工作平台将拒绝用户继续写入数据。这有利于防止某些业务或者用户滥用存储空间。

和普通的单机文件系统不同，数字有机体文件系统中的文件可以有副本。这样，当某个用户写入 10MB 数据时，实际占用的存储空间可能是 30MB，而不是 10MB。如果仅仅限制用户可以写入的数据量，即用户拥有的文件的总数据量，则用户可能为每个文件创建大量副本，从而仍然过分的占用存储空间。因此，在数字有机体工作平台上，用户的存储配额由两个参数限制。一个是用户文件的总数据量，即逻辑配额。第二个是用户文件占用的总存储空间，即真实配额。当两个限制中的一个超出时，都将拒绝用户写入数据。

要提醒的是：在大规模的分布式环境下，由于用户可以在大量主机上并发操作文件，因此系统无法准确的统计用户的配额使用量。这时，可能因统计误差而拒绝用户写入数据。建议设置配额时，稍微提高限制。统计误差在几十兆 B 左右。

实现配额机制需要一定的处理开销。尤其在快速写入文件时，如果开启配额机制，写入的性能将降低 10%到 50%。因此，是否开启配额机制是由系统管理员在配置文件内配置的。在/etc/dos_exernel.cnf 中有 uses_quota 配置项。该配置项为 1 时表示开启配额机制，为 0 是表示关闭配额机制。但是，如果只有系统的部分节点启用了配额机制，而其他节点没有则是不行的。因此，要么全部节点都启用配额机制，要么就都不启用。

用户的配额信息保存在用户信息中，因此管理用户配额的命令就是用户管理命令。下面将介绍有关配额机制的命令的使用方法。

1) 命令名：do_useradd

功能：在数字有机体工作平台上增建一个新用户。增建用户时可以指定新用户的配额。

使用方法：**do_useradd -n USER -g GROUP [-r REAL_USER] [-t TYPE] [-d DESCRIPTION] [-q QUOTA] [-s SPACE] [-h HOST,...]**

其中，-q 选项指定新用户的逻辑配额，-s 选项指定用户的真实配额。由于用户刚刚创建，因此用户的剩余配额就是指定的配额。配额单位为字节。

例如下面的命令新建一个 qyj 用户，其逻辑配额为 1GB，真实配额为 3GB

```
do_useradd  -nqyj -gDOSroot -rqiuyuanjie -q 1000000000 -s 3000000000
```

可以用 do_user 查看用户信息，其中就包含配额信息。例如：

```
root@server190:/home# do_user qyj
USER                    qyj
REAL USER               qiuyuanjie
USER ID                 3188678431
GROUP                   DOSroot
GROUP ID                2244703089
VERSION                 1
VALID                   yes
NODES COUNT             0
ALL LOGIC QUOTA         1000000000
USABLE LOGIC QUOTA      1000000000
ALL REAL QUOTA          3000000000
USABLE REAL QUOTA       3000000000
DESCRIPTION             qyj
```

其中，ALL LOGIC QUOTA 为逻辑配额总量，USABLE LOGIC QUOTA 为剩余逻辑配额量，ALL REAL QUOTA 为真实配额总量，ALL REAL QUOTA 为剩余真实配额量。因为是新建用户，所以两者是相同的。

2) 命令名：do_usermod

功能：修改系统中已有用户的信息，这是也可以修改用户的配额限制。

使用方法：**do_usermod -n USER [-g GROUP] [-r REAL_USER] [-t TYPE] [-d DESCRIPTION] [-q QUOTA] [-s SPACE] [-h HOST,...]**

其中，-q 选项指定用户的新逻辑配额，-s 选项指定用户的新真实配额。如果是增加用户的配额，则系统会增加用户的剩余配额量。如果是减少用户的配额，则用户的剩余配额会减少。特殊情况是：减少用户配额后，用户的剩余配额变为负数。这是用户将无法再写入数据，直到他删除某些文件腾出配额后才能写入数据。

3) 命令名：do_user

功能：查看指定用户的当前配额设置

使用方法：do_user username

注意：只有 DOSroot 用户可以使用该命令。

6 文件副本管理

数字有机体工作平台具有智能副本管理机制。系统先将文件分成若干数据块，再按照一定的规则为每个文件的分块建立副本。副本的数量和位置都将根据系统的状况进行自动调整，不需人工指定，也不是固定不变的。不过，管理员可以通过配置规则来影响和限制系统放置副本的位置和数量。此处特别指出，本节后续小节中的副本均指文件的一个分块。

6.1 副本管理规则

数字有机体工作平台按照一定的规则管理平台中文件的副本。通过配置这些规则可满足不同用户的需求。数字有机体工作平台的配置文件是/etc/xfs_base.cnf。用户可通过修改配置文件调节副本数目、放置位置等相关规则。配置文件中配置项较多，下面只介绍与副本管理相关的配置项。

6.1.1 副本数量配置

1） 配置项：min_rep_num_in_station

效果：该配置项指定在有副本的站内的最小副本数。当因为某些原因(如：节点死亡)导致站内实际副本数小于最小副本数时，数字有机体工作平台将检测到这种情况并在站内某台主机上增加一个副本，保证实际副本数不小于最小副本数。

实际配置举例：min_rep_num_in_station=2

这样的配置即可将站内最小副本数设置为 2。

2） 配置项：max_rep_num_in_station

效果：该配置项指定在有副本的站内的最大副本数。当因为某些原因(如：节点死亡后又恢复并重新加入平台)导致站内实际副本数大于最大副本数时，数字有机体工作平台将检测到这种情况并选择站内某台主机，删除其上面的副本，保证实际副本数不大于最大副本数。

实际配置举例：max_rep_num_in_station=3

这样的配置即可将站内最大副本数设置为 3。

3） 配置项：min_rep_num_of_station

效果：该配置项指定平台中拥有副本的站的最小数量，本手册中称其为站间最小副本数。

实际配置举例：min_rep_num_of_station=2

这样的配置即可将站间最小副本数设置为 2.。

4） 配置项：max_rep_num_of_station

效果：该配置项指定平台中拥有副本的站的最大数量，本手册中称其为站间最大副本数。

实际配置举例：max_rep_num_of_station=3

这样的配置即可将站间最小副本数设置为 3。

总结：数字有机体工作平台将同一个站内的所有副本视为一个逻辑整体，称为站副本。对一个文件来说，创建文件的主机的配置指定了上述参数。在放置文件时，将遵守站内副本数量和站副本数量限制。注意，在计算副本数量时，没有区分主本和从本，所有拷贝都是副本。也没有主站和从站，所有站副本是等同的。

6.1.2 副本分布配置

1） 本机创建副本概率：host_replicas_pro

效果：该配置项设置新建文件时本机创建副本的概率。允许的取值在 0 至 100 之间并包括 0 和 100。100 表示新建文件时本机一定有文件的副本（空间不足除外），而设置为 0 则表示新建文件时副本的放置位置随机选择。注意，为 0 是不是表示本机不能放副本，而是表示随机放置。该配置项可由管理员根据网络情况具体配置。

实际配置举例：host_replicas_pro=90

这样的配置即可将本机创建副本优先概率设置为 90%。

2） 仅本地放置：only_local_set

效果：该配置项决定副本是否只存放于本地，0 表示否，1 表示是。该配置项的优先级非常高，配置时候须额外留意。系统记录创建文件的站的 ID，将该站作为文件的所属站。当该配置项的取值为 1 时，系统将只在文件的所属站上创建副本，而不会在其他站上创建副本。这样，站副本数量限制等都不在具有意义。

实际配置举例：only_local_set=1

这样的配置即指定副本只存放于所属站。

3） 文件默认放置站列表：file_default_local

效果：该配置项将副本放在以站 ID 指定的站内，配置值即为站 ID，多个站 ID 间以“，”隔开。

实际配置举例：file_default_local= 8566908956407194755+4380269852255007421

这样的配置即可指定副本放在站 ID 为 8566908956407194755+4380269852255007421 的站内。

总结：系统总是优先尝试在文件的创建站，即所属站上建立副本。如果 only_local_set 为 0，则系统按照最小站副本数限制选择其他站创建副本。所属站有副本时它也是一个站副本。因此，如果最小站间副本数为 1，则创建文件时在所属站上创建副本后，就不再其他站上创建了。只有最小站间副本数大于 1 时才会在创建文件时在所属站外创建副本。不过，如果最大站间副本数大于 1，则即使最小站间副本数为 1，在需要时，例如文件所属站故障了，或者其他站频繁访问文件，系统仍然可能在其他站上创建副本。

配置了文件默认放置站列表后，创建站外副本时，系统总是先按照设定的默认放置站顺序，依次创建副本。在站副本数大于等于站间最小副本数时停止创建。因此，不是说所有默认放置站都要创建文件副本。在默认放置站不足时，系统也会自动选择其他站创建副本。而其，创建文件的站，即文件所属站总是最优先的。

如果 only_local_set 为 1，则无论如何也不会在文件所属站外创建副本。

在站内，创建新文件时本机具有特殊性，即可以设定本机的创建副本概率。这使得管理员可以尽量将文件副本放置在创建主机上。不过，在其他时候则没有意义了。

注意：

1） 配置文件/etc/xfs_base.cnf 被修改后须重启数字有机体工作平台方可生效。

2） 同一个站内所有主机的配置文件/etc/xfs_base.cnf 中与副本管理相关的内容请务必保证相同，否则可能导致出错。

6.2 副本管理命令

除了系统自动管理文件副本外，管理员可能需要查看文件副本信息，也可能手动控制文件的副本位置。这就可以使用以下这些命令。如果应用程序需要控制副本位置和获取副本位置信息，可以使用数字有机体工作平台的接口函数。

1） 查看文件副本信息：do_replicas

功能：查看资源副本在系统中的分布。

使用方法：do_replicas　file-path

参数：file-path 为要查看的文件或者目录的路径名。可以是全路径也可以是相对路径。指定的文件或者目录必须是数字有机体文件系统中的。因此 file-path 实际上是数字有机体文件系统在本地挂载后的路径名。

例如：

```
root@server192:/dpfs# do_replicas ./A

file    block-number    station-id-high station-id-low  node-ip
/dpfs/A 0       5026522733777274790     2926722550748602854     192.168.2.192
/dpfs/A 0       5026522733777274790     2926722550748602854     192.168.2.193
/dpfs/A 0       5109026273992463603     7794464742246109241     192.168.2.79
/dpfs/A 0       5109026273992463603     7794464742246109241     192.168.2.72
/dpfs/A 1       5026522733777274790     2926722550748602854     192.168.2.192
/dpfs/A 1       5026522733777274790     2926722550748602854     192.168.2.193
/dpfs/A 1       5109026273992463603     7794464742246109241     192.168.2.78
/dpfs/A 1       5109026273992463603     7794464742246109241     192.168.2.72
/dpfs/A 2       5026522733777274790     2926722550748602854     192.168.2.193
/dpfs/A 2       5026522733777274790     2926722550748602854     192.168.2.192
/dpfs/A 2       5109026273992463603     7794464742246109241     192.168.2.72
/dpfs/A 2       5109026273992463603     7794464742246109241     192.168.2.78
```

2） 创建文件分块副本：do_repblockadd

功能：为资源在系统中指定节点上增加一个文件的分块副本。

使用方法： do_repblockadd　[options]　[ip/stationId]　file-path blockNum

参数：

-n 选项是和 ip 搭配的，用于为某个文件的分块在指定 IP 的节点上增加副本。

-s 选项是和 stationId 搭配的，用于为某个文件的分块在指定 ID 的站上创建副本。

blockNum 指定文件分块的块号，块号以 0 起始。

因此，本命令只有两种用法，即：

在指定节点上创建副本：do_repblockadd　-n ip　file-path blockNum

在指定站点上创建副本：do_repblockadd　-s stationId　file-path blockNum

注意事项：执行该命令时，在数字有机体工作平台收到指令后即返回用户成功消息。但是，真正创建副本的工作才开始。如果文件很大而网络速度又慢，则可能需要很长时间文件的副本才能真正创建成功。可以通过查看文件的副本信息来确认操作是否成功。

该命令常见失败原因如下：

1、所指定节点存储空间已经或者接近耗尽。

2、所指定资源为目录(系统中目录不能手动增加副本)。

3、所指定节点已经死亡。

3） 删除文件分块副本：do_repblockdel

功能：删除文件在系统中指定节点或者站点上的一个文件的分块副本。

使用方法：do_repblockdel [options] [ip/stationId]　file-path blockNum

参数：

-n 选项是和 ip 搭配的，用于删除某个文件在指定 IP 的节点上的副本。

-s 选项是和 stationId 搭配的，用于删除某个文件在指定 ID 的站上的副本。

blockNum 指明文件分块的块号。

因此，本命令只有两种用法，即：

删除指定节点上的副本：do_repblockdel　-n ip　file-path blockNum

删除指定站点上的副本：do_repblockdel　-s stationId　file-path blockNum

备注：和 do_replicaadd 命令相同，数字有机体工作平台收到指令后即返回成功。因此副本的删除工作异步完成的。不过删除副本操作通常都能成功。可以通过查看文件的副本信息来确认操作是否成功。

4） 设置文件分块大小：do_setblocksize

功能：设置文件的分块大小。

使用方法: do_setblocksize [dirname] [blocksize]

参数：

dirname 为需要设置块大小的路径名，必须是一个目录。

blocksize 是指定的块大小，以字节为单位。

备注：新文件的块大小继承自其父目录，所以设置文件分块大小也是针对目录而言，当且仅当指定路径是一个空目录时，本命令才能成功。

5） 获取文件分块大小：do_getblocksize

功能：获取文件的分块大小。

使用方法：do_getblocksize [filename]

参数：

filename 是需要查询的文件路径名。

6） 获取目录下文件的副本数量配置：do_getdirreplicas

功能：设置文件的分块大小。

使用方法: do_getdirreplicas DIR

参数：

DIR 为需要获取副本数量的路径名，必须是一个目录。

7） **修改目录下文件的副本数量配置：do_setdirreplicas**

功能：设置文件的分块大小。

使用方法：do_setdirreplicas -d DIR [-s MIN_REP_OF_STATION] [-S MAX_REP_OF_STATION] [-m MIN_REP_IN_STATION] [-M MAX_REP_IN_STATION]

参数：

DIR 为需要获取副本数量路径名，必须是一个目录。

MIN_REP_OF_STATION 表示站间最小冗余数，规定副本至少应该存在的站数量。

MAX_REP_OF_STATION 表示站间最大冗余数，规定副本最多存在的站数量。

MIN_REP_IN_STATION 表示站内最小冗余数，规定副本在站内至少应该存在的数量。

MAX_REP_IN_STATION 表示站内最大冗余数，规定副本在站内最多存在的数量。

备注：新文件的副本数量继承自其父目录，所以设置文件副本数量也是针对目录而言。

7 服务管理

服务信息指记录每台服务器可以提供的服务的相关信息。数字有机体工作平台收集、记录和管理系统中的服务信息，为应用程序提供服务查询和服务快速定位服务。

应用程序或者管理员通过服务注册机制向数字有机体工作平台注册每台服务器可以提供的服务的信息。数字有机体工作平台将监视每项服务的运行情况，在发现提供服务的服务器故障，或者服务程序死亡时，将自动注销对应服务信息。从而使得应用程序总能获得可用的服务。

7.1 服务管理器（rg 程序）

服务管理器主要用于实现服务程序的开机自启动功能、服务注册功能以及服务程序异常退出后的自动重启功能。程序开机自启动功能不是服务管理器特有的，而后两个功能则是服务管理器特有的功能，它可以简化服务管理的工作。

通常，服务程序启动后，都需要向数字有机体工作平台注册服务，否则客户无法找到他，也就无法从他这里获得服务。应用程序可以调用数字有机体工作平台的接口函数来注册自己提供的服务。但是，如果服务程序是旧程序，不能修改源码时，则需要通过外部命令来注册他提供的服务。这就是 7.3 节介绍的“服务的注册和注销”命令。服务管理器自动使用注册注销命令，从而减少管理员的工作。

有些服务程序可能因为各种原因而异常终止。如果需要继续提供服务则需要重启服务程序。由管理员来监视服务程序并手动重启是困难的。因此，服务管理器代替管理员完成这项工作。

服务管理器是由/usr/sbin/dpserver_start 调用启动的，而 dpserver_start 是由开机自动启动程序/etc/init.d/dpserver 程序启动，故服务管理器也是开机自动启动的。当然，服务管理器也可手动启动，使用方式为：

/usr/local/bin/rg　　start/stop/restart

或简写为：

rg　start/stop/restart

start 参数用于启动服务管理器，stop 参数则是关闭服务器管理器，而 restart 参数则是先关闭服务管理器，然后重新启动它，即重启服务管理器。

7.2 配置服务器管理的服务

服务管理器根据配置文件来管理各项服务。如果一项服务需要被服务管理器管理，则必须在服务管理器中增加相应的配置节。服务管理器的配置文件为/etc/regct2.cnf。该文件的头部为“[other]配置节，其格式如下：

[other]

interval=15;

path=/usr/local/bin/rg2/:/usr/local/sbin/:/usr/sbin/:/usr/local/bin/:/usr/local/mysql/bin;

RGLOG=/usr/local/bin/rg2/rg.log;

该节配置服务管理器的运行参数。参数 interval 定义服务管理器周期性检查各项服务的运行状态的时间间隔。参数 path 是服务管理器执行服务程序或者脚本的路径参数。参数 RGLOG 指定服务管理器运行日志的输出文件。

除了 other 节外，其他的配置节都是一项项要被服务管理器管理的服务。每项服务的配置内容是相同的，所以节以流媒体服务为例进行说明。其他的服务可以参照这个例子进行配置。以下是流媒体服务的配置。

```
[Darwinstreaming]
name=DarwinStreamingServer;
rgname=StreamingServer;
delay_start=5;
delay_stop=3;
start_command=/usr/local/sbin/DarwinStreamingServer -d >>/var/log/darwin.log &;
comm_after_start=;
stop_command=;
service_port=554;
pub_addr=192.168.1.1
reg_flag=1;
dependence=dos_schedule;
stop_sequ=2;
```

下面分别说明每项配置的含义和要求：

[Darwinstreaming]：服务标题，也是一个服务配置项的开始，在一个配置文件内需要时唯一的。

name：服务名称，必须与服务标题相同。

Rgname：服务注册时使用的名称。需要根据任务调度的需求来配置，可以和服务名称不同。应用客户查找服务时将以该名字作为参数。如果要注册服务，则该配置项不能为空。

delay_start：等待启动时间，单位为秒。

delay_stop：等待停止时间，单位为秒。

start_command：启动服务是执行的命令。服务管理器本身是一个 shell 脚本，因此他可以执行各种 shell 命令。这里就是其他服务需要执行得命令。

comm_after_start：附加启动命令项，指本服务启动完成后要执行的命令。

stop_command：表示停止本服务时执行命令。

service_port：表示服务使用的端口。该配置项只在服务需要注册时才有意义。它是注册服务时的参数。如果要注册服务则该配置项不能为空。

reg_flag：是否注册服务标记。如果为 1，则表示要注册服务，这时 rgname 和 service_port 不能为空。

Dependence：该服务依赖的其他服务，即本服务启动前必须保证此项所指服务已经启动。流媒体服务需要在 dos_schedule 启动后在启动，因此这里设置为 dos_schedule。这时，

配置文件中必须含 dos_schedule 的配置节。当服务依赖于多个服务时，只能将它依赖的服务再按照需要的启动顺序排列。这里只指定它相邻的前一项。

stop_sequ：停止服务的顺序号。值越小越先停止。在关闭或者重启服务管理器时，它将按照这个顺序关闭各项服务。

注意：

(1) 配置文件中以“#”号开始的行为注释行，服务管理器将忽略它。

(2) 每项服务在文件中的配置顺序并不等于启动顺序，服务启动的顺序是按照服务间的依赖关系决定的。只有两个服务没有依赖关系时，才按照配置顺序来启动。

(3) 服务的启动命令不能是阻塞式的，即服务管理器执行服务启动命令后应当能够快速返回，否则服务管理器将一直被阻塞，从而无法正常服务。可以采用后台运行的方式解决这个问题。例如流媒体服务器就是采用后台运行的。

(4) 服务管理器在监视服务是否正常运行时，将监视启动的进程是否仍然在运行，并且以启动进程消失作为服务需要重新启动的标志。因此，对于某些服务来说，这可能导致服务管理器误判服务死亡，从而反复重启服务。

(5) 每次修改配置后，必须重新启动服务管理器才能使新配置生效。

7.3 服务的注册和注销

如前所述，每台服务器能够提供哪些服务需要通过服务注册告诉数字有机体工作平台，否则数字有机体工作平台在提供服务查询和快速定位时将无从知晓。服务管理器提供了自动注册服务的功能。不过，某些时候，管理员还是希望自己启动服务，并自行决定何时注册或者注销服务。这时就需要相应的命令。下面将介绍这两个指令。

1） 注册服务

命令：do_serviceadd

用法：do_serviceadd -n SERVICE -h HOST -p PORT -i PID -t TYPE

参数：

这里的所有选项都是必须给出的，因此不能说是选项了。

-n SERVICE：给出服务注册的名称，也就是客户查询时需要的名称。

-h HOST：给出提供服务的 IP 地址。当服务器具有两个或者以上网络接口时，可能提供服务的网络地址并不和数字有机体工作平台内部通信的地址相同。在某些使用反向 NAT 时也会出现这种情况。这时，需要给出服务的实际服务地址，又成为公共服务地址。

-p PORT：提供服务的端口号。

-i PID：提供服务的进程 ID。注意，工作平台将监视这个 ID 指定的进程，当没有相应 ID 的进程存在时，工作平台认为该服务已经故障，将自动注销该服务。

-t TYPE：服务的类型，这里只有是服务器提供的服务（s）和客户机提供的服务（c）两种。服务管理器注册的都是服务器提供的服务。

除了 HOST 外，上述选项都是必须给出的，否则注册操作将失败。

2） 注销服务

命令：do_servicedel

用法：do_servicedel -n SERVICE -h HOST -p PORT -i PID -t TYPE

参数：该命令使用的参数和注册服务时相同，不再描述。

7.4 主机自动启动服务配置

初始安装的系统只自动启动了少量的服务。如果你希望某些服务或者程序在主机启动时就自动启动，可以使用 service_manager.sh 程序进行配置。在命令行中键入命令：service_manager.sh 即可启动配置程序。其配置界面如图 7-1 所示。

图 7-1：主机自动启动服务配置

系统自动检测当前哪些服务是设置为自动启动了的，并以星号表示已经是自动启动的。可以用上下键移动到要配置的服务，然后按空格键将其标记为自动启动或者取消自动启动。在完成配置后，在“确认”按钮上按回车键即可开始配置服务的自动启动。如果要放弃配置，在“取消”按钮上按回车键即可。

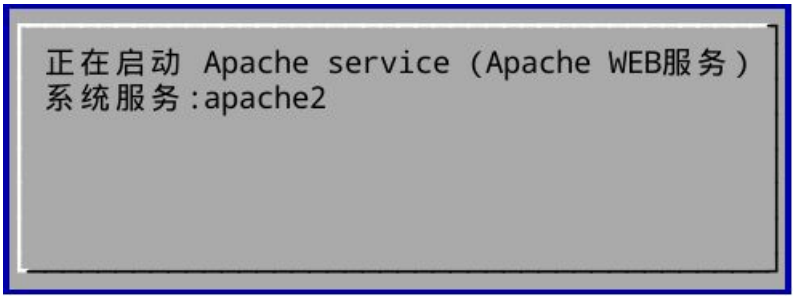


图 7-2：配置服务提示信息

如果设置了某个服务要自动启动，则程序将进行配置，并给出如图 7-2 所示的提示。

在完成配置后，该提示将自动消失。

如果设置了关闭某个服务的自动启动，则程序也将取消其启动配置，并给出如图 7-3 所示的提示。在完成配置后，该提示自动结束。

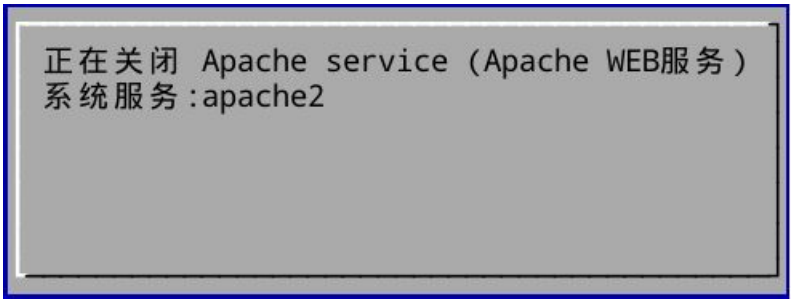


图 7-3：关闭服务自动启动配置的提示信息

如果改变了服务配置，则完成所有配置后，系统将提示是否立即重启主机。如果选择“是”则程序将重启主机。

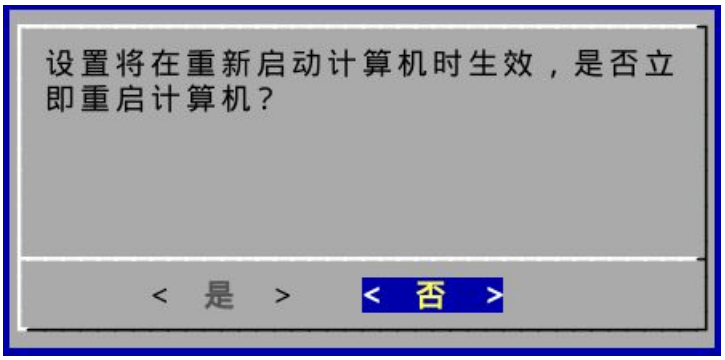


图 7-4：重启主机的提示

注意：某些配置只有在重启主机后才能生效。

8 网络虚拟服务管理

数字有机体工作平台为网络业务提供了单独的组件，称为数字有机体网络虚拟服务模块。它针对分布式网络服务的需要而设计，提供网络服务的单一 IP 入口、负载均衡、故障节点屏蔽、会话保持等功能。单一 IP 入口指多台服务器上的相同业务可以共享一个公共 IP 地址，网络用户通过这个公共 IP 地址即可获得服务，系统自动在多台服务器间均衡负载。故障节点屏蔽指系统在分配请求时，自动避开已经出现故障的节点。会话保持包括三个内容，一是分配请求时关联的请求分配到相同的服务器上处理。二是复制会话信息，在处理服务器故障时保证后继请求可以在其他服务器上继续执行。三是复制请求分配状态信息，保证后继请求在各台服务器上分配时都能到达同一台处理服务器。会话信息复制不能由数字有机体工作平台完成。其余的会话保持功能则由数字有机体工作平台实现。

当多台服务器同时提供一项服务时，服务的访问地址，即单一的 IP 入口，是一个虚拟的 IP。这个 IP 将同时绑定在所有提供该项服务的主机上，但是只有一台主机接受从网络上传来的请求。这台主机也被称为入口主机或者调度主机，因为它将按照一定的规则将请求转发给其他服务节点进行处理，并且由其他服务节点响应客户。在数字有机体工作平台上，每项服务的入口主机可能是不同的，因为这由系统自动选择决定。当某项服务的入口主机故障时，工作平台将自动选择其他主机担当服务的入口主机，从而保证服务能够继续下去。

工作平台需要知道哪些服务器可提供某项服务，即需要配置每项服务的真实主机。同时，工作平台将监视每台主机提供的服务是否可用。在主机提供的服务不可用（比如主机死亡或者服务程序故障等）时，工作平台自动向客户屏蔽该台主机，从而保证客户请求能够正常处理。

系统提供了多种负载均衡的策略。最简单的策略是按照管理员设定的比例在真实服务器间分配负载，即服务器默认权重策略。第二种策略为连接数权重策略，即工作平台认为每台服务器的处理能力相当，而且当前的活动连接数即表示了服务器的当前负载，因此调度的目的就是保证各台服务器的连接数均衡。第三种策略为负载权重策略，工作平台将周期性的统计并收集各台真实服务器的负载，然后调整分配系数以使各台真实服务器的负载均衡。服务器负载指服务器的 CPU、内存和磁盘的负载情况。第四种策略是综合权重策略，即综合考虑连接数和服务器负载来调整任务分配。

系统管理员需要配置网络虚拟服务的相关信息，以便系统能够正确组织。您可以通过数字有机体工作平台的管理网站进行配置。

8.1 系统参数配置

在使用网络虚拟服务功能之前，需要先配置网络虚拟服务模块运行的参数。在管理网站上配置的界面如图 8-1 所示。如果没有配置过，则看不到配置记录。这时需要通过添加按钮添加配置记录。如果已经有配置记录，则可以修改已有的配置记录。工作平台只使用

第一条配置记录。如果希望重新设置，可以先删除记录后再添加新记录。

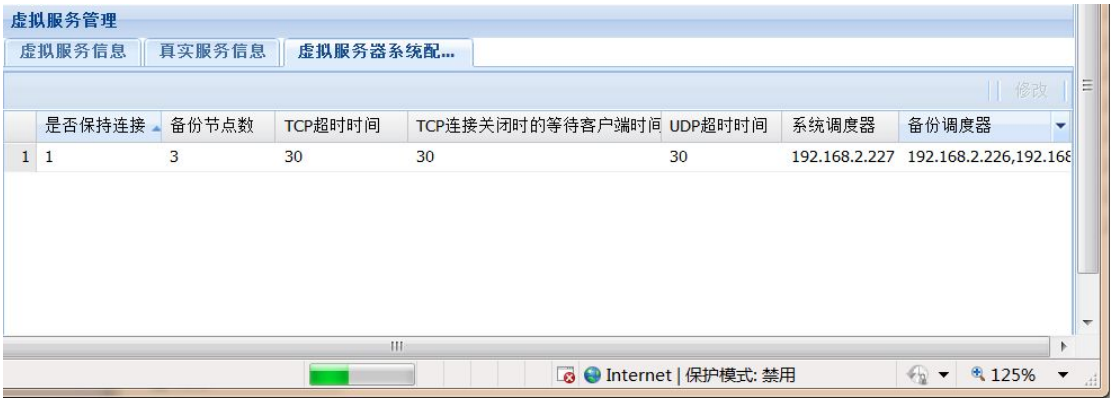


图 8-1 虚拟服务器系统配置

记录每个字段的含义如下：

1） 是否保持连接：当值为 1 时，表示需要数字有机体工作平台保持网络客户的连接状态信息。保持连接状态信息的目的是：当调度主机故障时，接替调度工作的主机可以继续按照已有的连接分配方式继续分配连接。这有助于保持业务回话的连续性。

2） 备份节点数：当启用连接保持后，数字有机体工作平台将把客户的连接状态信息同时复制到多台服务器上，以便在原有的调度主机死亡后，仍然能够保持客户的连接。本参数设置同时复制的节点数。

3） TCP 超时时间：在连接保持启用后，该参数指定调度入口服务器保持 TCP 连接的时间。当最后一个 TCP 连接请求超过该设定时间后，调度入口服务器可以将客户的请求调度到其他服务器。单位为秒。

4） TCP 连接关闭时的等待时间：在连接保持启用后，在服务主动关闭连接后，等待客户做出连接关闭的时间，超过该时间服务器强制关闭连接。

5） UDP 超时时间：该参数的含义和 TCP 超时时间类似，只是设定的是 UDP 的。

记录的另外两个字段无需配置，它将由工作平台自动填写，显示的目的仅是让管理员了解系统运行状况。当启用连接保持时，一个站只有一台主机作为调度节点，负责所以业务的调度。反之，则各个虚拟服务可以由不同的节点调度。

配置参数后，需要等待一定时间，以便工作平台能够完成配置工作。

8.2 虚拟服务信息

对于每一项虚拟服务，都需要为它配置运行参数。每项虚拟服务的参数作为一条记录看待。管理员可以在数字有机体工作平台的管理网站中配置它，配置界面如图 8-2 所示。每项服务需要配置的内容较多，下面分别介绍各个配置项的含义和配置注意事项。

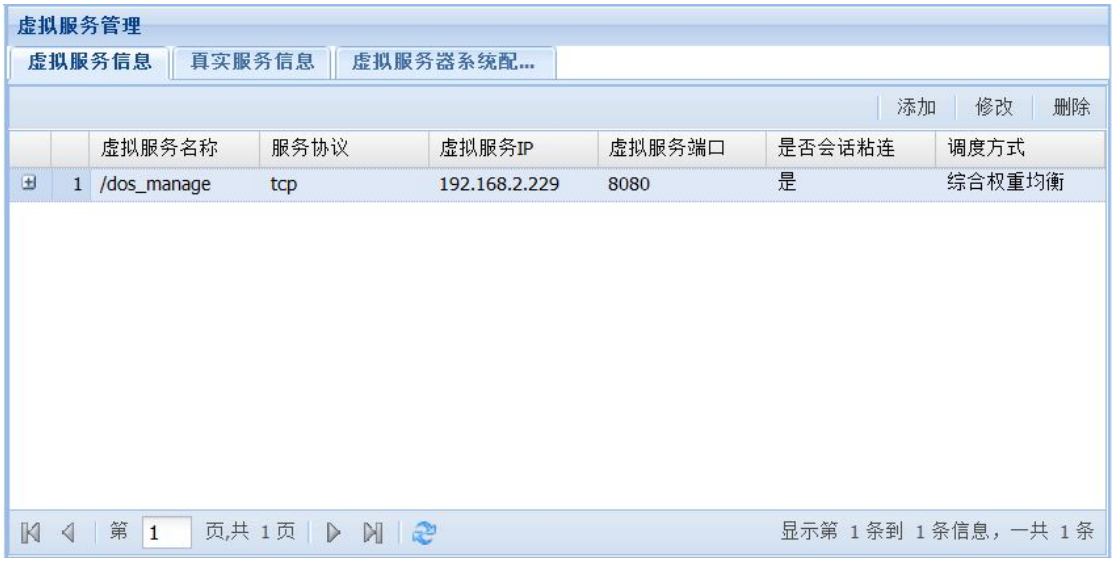


图 8-2 虚拟服务信息

服务名称：用来区别服务的名称。不过，数字有机体工作平台还使用该名称来检查 HTTP 服务是否可用。检查时，将使用“真实服务器的 IP：真实服务端口/服务名称”来请求服务，因此它必须为可访问路径。例如 http://192.168.2.190:8080/demo/demo.jsp 的服务名称为 demo/demo.jsp。

服务协议：当前主要支持 TCP、UDP 协议，无需区分大小写。

虚拟服务 IP：此即虚拟服务的单一 IP 入口地址。

虚拟服务端口：一般情况下与服务的端口一致。如果虚拟服务端口和真实服务的端口不同，则需要外部网络设备来完成映射，例如反向 NAT 服务器。

是否会话粘连和会话粘连超时：该参数设置：分配客户时是否将同一个客户的连接分配到同一台服务器上出来。该参数与系统参数中的保持连接没有必然联系。不过，如果要严格保证回话的连续性，在设置会话粘连的同时，也要设置保持连接。

当值为 1 时，表示需要数字有机体工作平台保持网络客户的连接。这时同一个客户的所有连接都将被同一台服务器处理。数字有机体工作平台用请求中的源 IP 地址区分客户。因此，如果几个客户通过同一个代理或者 NAT 请求服务，则系统将把它们作为一个客户看待。源 IP 地址和网络屏蔽掩码（见下一个参数）与操作后得到地址被看作“区分地址”。来至相同“区分地址”的请求分配到同一台服务器。因此通常将网络屏蔽掩码设置为“255.255.255.255”。

网络屏蔽掩码：该屏蔽码是会话粘连时使用的。具体含义见前一项说明。。

调度方式：当前有四种调度方式：

负载权重：根据每台真实服务器的 CPU、网络和磁盘负载情况均衡调度服务；

连接数权重：根据每台真实服务器的连接数情况均衡调度服务；

综合权重：根据连接数、负载情况以及原始分配比例均衡调度服务；

默认权重：根据真实服务器的原始分配比例调度服务。

附加路由：该配置项用于设置服务入口主机绑定虚拟 IP 地址后，需要设置的附加路由信息。在某些网络中，使用服务的虚拟 IP 需要指定单独的路由信息，以便服务主机能够将

响应发回客户。每条附加路由信息的格式为：

目的 IP,目的网络 mask,网关地址/网络设备名。

网关地址/网络设备名任选一项，根据配置需要选择。

多条附加路由用分号分割。

8.3 真实服务器信息

对每一项虚拟服务，需要配置其提供服务的真实服务器信息。同样，可以在数字有机体工作平台管理网站中配置它，其界面如图 8-3 所示。

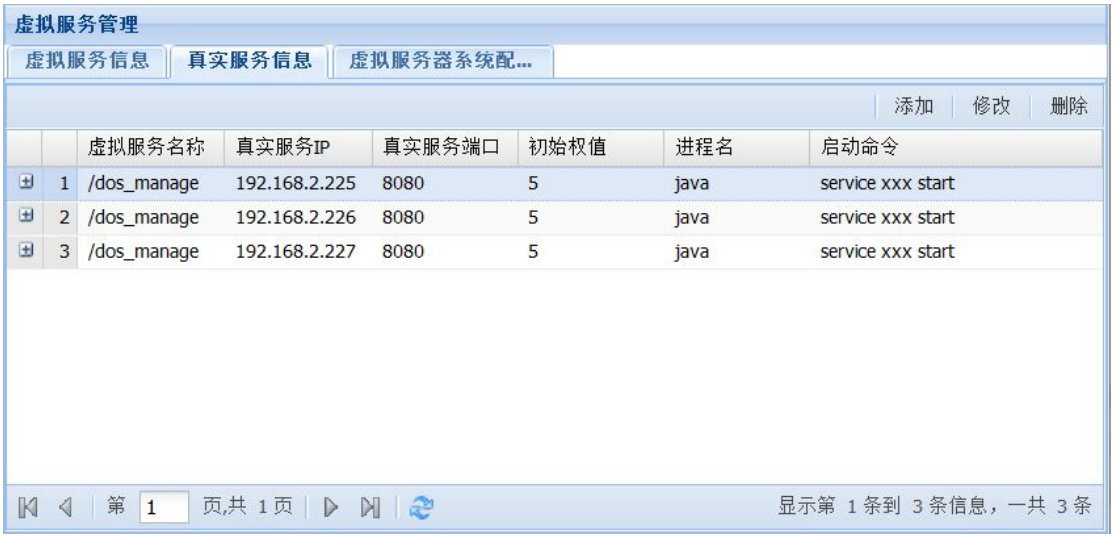


图 8-3 真实服务器信息

一项虚拟服务可以由多台服务器提供，因此应该对应多条真实服务器信息。每条真实服务器信息的包含以下内容：

虚拟服务：选择已经配置好的虚拟服务，即真实服务器提供的服务。

真实服务 IP：指定服务器 IP。该 IP 是内部 IP，即数字有机体工作平台内部通信 IP。

真实服务端口：指定真实服务器提供服务所用的端口，通常和虚拟服务端口相同。

初始权重值：表示服务器能力的一个权重值。不要求所有真实服务器的初始权重值的和为 100，通常以能力最差的服务器为基准，例如设能力最差的服务器的初始权重值为 10。如果另一台服务器的能力是它的 1.5 倍，则它的初始权重值可以设为 15。以此类推，即可设定每台服务器的初始权重值。

进程名：提供服务的进程的名称。注意，数字有机体工作平台将根据该名称监控服务是否存在。某些服务，例如 tomcat，其提供服务的进程的名称并不是服务的名称。Tomcat 的服务进程名为 java，而不是 tomcat，因此这里需要填写 java，而不是 tomcat。

启动命令：启动服务使用的命令，当前该项配置还未使用。未来将考虑由系统自动启动服务，从而动态调节整体服务能力。

增加的路由：该项配置的方式和虚拟服务的附加路由配置方式相同。不过，这些路由是设置给非服务入口主机的，即普通服务主机的附加路由信息。

完成系统配置、虚拟服务配置和虚拟服务的真实服务配置后，即可启动数字有机体工作平台使用网络虚拟服务功能。不过，以下几点需要注意：

1) 系统参数是给所有虚拟服务配置的，因此作用于所有虚拟服务。
2) 网络虚拟服务是以站为单位进行配置的，即这里所作的配置仅作用于配置服务器所在的站，各个数字有机体站的配置可以互不相同。
3) 数字有机体工作平台将周期性获取虚拟服务和真实服务器配置，因此配置他们后系统将自动使新配置起效。但是，系统参数不会周期性获取，因此需要重新启动数字有机体工作平台后才有效。而且，要重启同一个站的所有服务器。幸好，系统参数在设置后通常不会更改。
4) 网络虚拟服务配置需要的虚拟 IP、路由信息都和提供服务的网络结构相关，因此需要管理员了解相应的网络知识，否则无法正确配置。

8.4 查看虚拟服务运行状况

网络虚拟服务利用了 Linux 系统的 lvs 机制。系统提供 ipvsadm 命令来管理和查看虚拟服务的配置状况。

对同一项虚拟服务来说，它只有一个服务入口主机。在服务入口主机上可以用“ipvsadm –L”或者“ipvsadm –l”可查看虚拟服务当前状况。执行他们将列出当前正在运行的虚拟服务，以及各台真实服务器的状况。

如果需要看每个虚拟服务的连接状况，可以使用“ipvsadm –L –c”命令，它将显示出虚拟服务当前的网络连接情况。

如果在任何一台真实服务器上都无法用上述命令查出虚拟服务的信息，则该虚拟服务没能成功启动。

也可以使用 ifconfig 命令来查看虚拟 IP 绑定的情况。通常，在服务入口主机上，虚拟 IP 地址绑定在真实网络接口上，而在普通真实服务主机上，虚拟 IP 地址绑定在 lo 上。

8.5 使用 Tomcat 实现会话保持

当真实服务器故障时，如何业务会话的信息没有备份（即复制到其他节点），则客户的会话将丢失。数字有机体工作平台并没有提供业务会话复制的功能，因此需要业务系统自己完成会话复制工作。

实现会话信息复制的方式有多种。第一种方式是将会话信息实时保存在数据库中，利用数据库备份会话信息。第二种方式是使用 Cookie，但这种方式需要客户的浏览器支持。第三种方式是利用业务服务系统的会话复制功能。例如 Tomcat 提供的会话复制功能。

在 Tomcat 6 以上版本中，都提供了会话变量复制功能。启用会话复制需要配置三个文件。第一个是 Tomcat 的 server.xml 文件，它在 tomcat 安装目录下的 conf 目录下。该文件中配置复制的各项参数。第二个文件时 Tomcat 的 web.xml 文件，它配置所有 tomcat 项目

都要进行会话复制。第三个配置文件时项目的配置文件，即项目目录下的WEB-INF/web.xml 文件。它配置本项目需要进行会话复制。

8.5.1 Tomcat 的 server.xml 参数配置

这个文件中“Cluster”元组用于配置会话复制（即 tomcat 集群）。下面说明其子元组和选项的含义。

1、channelSendOptions 这个参数说明了集群内部服务器消息的发送方式，实际上用了二进制数的表示方式，第一位是 1 说明使用 ack(收到确认)，第二位是 1 说明使用同步 ack，第三位是 1 说明使用异步消息的模式，在这个配置的例子当中使用 6=2+4，说明使用了同步 ack 的方式发送。

```
<Cluster
className="org.apache.catalina.ha.tcp.SimpleTcpCluster"  channelSendOptions="6">
```

2、session 拷贝方式 BackupManager 只拷贝部署当前应用的服务器，DeltaManager 拷贝方式 all to all。就目前来说 tomcat 提供 2 种 manager，一种是 BackupManager，一种是 DeltaManager，后者需要 cluster 的每个节点都严格部署了完全一样的同一个应用，会话被复制到了每一个服务器上。

```
  <Manager
className="org.apache.catalina.ha.session.BackupManager"  expireSessionsOnShutdown="fals
e"   notifyListenersOnReplication="true"    mapSendOptions="6"/>
<!--  <Manager
className="org.apache.catalina.ha.session.DeltaManager"  expireSessionsOnShutdown="false"
   notifyListenersOnReplication="true"/>    -->
```

3、Channel 子元组负责对 tomcat 集群的 IO 层进行配置。tomcat 服务使用该通道完成会话信息复制。常用的方式是组播。即通过组播完成一到多的通信。

```
<Channel className="org.apache.catalina.tribes.group.GroupChannel">
```

4、Membership 用于发现集群中的其他节点，这里的 address 用的是组播地址（Multicast address），使用同一个组播地址和端口的多个节点同属一个子集群，因此通过自定义组播地址和端口就可将一个大的 tomcat 集群分成多个子集群。 这里的地址 228.0.0.4 是一个默认的广播地址，只要是使用默认的广播地址的 tomcat 实例，均认为是工作在同一个 cluster 上，也就是说这些 tomcat 之间的会话会被复制。

```
      <Membership className="org.apache.catalina.tribes.membership.McastService"
          address="228.0.0.4"
          port="45564"
          frequency="500"
          dropTime="3000"/>
```

5、Receiver 用于各个节点接收其他节点发送的数据，在默认配置下 tomcat 会从 4000-4100 间依次选取一个可用的端口进行接收。自定义配置时，如果多个 tomcat 节点在一台物理服务器上注意要使用不同的端口。

```
      <Receiver className="org.apache.catalina.tribes.transport.nio.NioReceiver"
```

```
        address="auto"
        port="5001"
        selectorTimeout="100"
        maxThreads="6"/>
```

6、Sender 用于向其他节点发送数据，具体实现通过 Transport 配置，PooledParallelSender 是从 tcp 连接池中获取连接，可以实现并行发送，即集群中的多个节点可以同时向其他所有节点发送数据而互不影响。

```
    <Sender className="org.apache.catalina.tribes.transport.ReplicationTransmitter">
        <Transport
className="org.apache.catalina.tribes.transport.nio.PooledParallelSender"/>
    </Sender>
```

7、Interceptor 有点类似下面将要解释的 Valve，起到一个阀门的作用，在数据到达目的节点前进行检测或其他操作，如 TcpFailureDetector 用于检测在数据的传输过程中是否发生了 tcp 错误。

```
 <Interceptor
className="org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"/>
 <Interceptor
className="org.apache.catalina.tribes.group.interceptors.MessageDispatch15Interceptor"/>
 <Interceptor
className="org.apache.catalina.tribes.group.interceptors.ThroughputInterceptor"/>
    </Channel>
```

8、Valve 用于在节点向客户端响应前进行检测或进行某些操作，ReplicationValve 就是用于用于检测当前的响应是否涉及 Session 数据的更新，如果是则启动 Session 拷贝操作，filter 用于过滤请求，如客户端对图片，css，js 的请求就不会涉及 Session，因此不需检测，默认状态下不进行过滤，监测所有的响应.JvmRouteBinderValve 会在前端的 Apache mod_jk 发生错误时保证同一客户端的请求发送到集群的同一个节点。

```
    <Valve className="org.apache.catalina.ha.tcp.ReplicationValve"
    filter=".*\.gif;.*\.js;.*\.jpg;.*\.png;.*\.htm;.*\.html;.*\.css;.*\.txt;"/>
```

9、Deployer 用于集群的 farm 功能，监控应用中文件的更新，以保证集群中所有节点应用的一致性，如某个用户上传文件到集群中某个节点的应用程序目录下，Deployer 会监测到这一操作并把这一文件拷贝到集群中其他节点相同应用的对应目录下以保持所有应用的一致。这是一个相当强大的功能，不过很遗憾，tomcat 集群目前并不能做到这一点，开发人员正在努力实现它，这里的配置只是预留了一个接口。

```
    <Deployer className="org.apache.catalina.ha.deploy.FarmWarDeployer"
      tempDir="/tmp/war-temp/"
      deployDir="/tmp/war-deploy/"
      watchDir="/tmp/war-listen/"
      watchEnabled="false"/>
```

Listener 用于跟踪集群中节点发出和收到的数据，也有点类似 Valve 的功能。

```
    <ClusterListener className="org.apache.catalina.ha.session.ClusterSessionListener"/>
  </Cluster>
```

以下是 server.xml 的配置示例：

```
<?xml version='1.0' encoding='utf-8'?>
      <!-- 这里省略了无关的配置 -->

        <!--For clustering, please take a look at documentation at:
              /docs/cluster-howto.html  (simple how to)
              /docs/config/cluster.html (reference documentation) -->
  <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"
                        channelSendOptions="6">

              <Manager className="org.apache.catalina.ha.session.DeltaManager"
                          expireSessionsOnShutdown="false"
                          notifyListenersOnReplication="true"/>
              <Channel className="org.apache.catalina.tribes.group.GroupChannel">
                <Membership className="org.apache.catalina.tribes.membership.McastService"
                                  address="228.0.0.4"
                                  port="45564"
                                  frequency="500"
                                  dropTime="3000"/>
                <Receiver className="org.apache.catalina.tribes.transport.nio.NioReceiver"
                                  address="192.168.2.189" <-- 默认配置是 auto，这里要和虚拟服务一同使用，
一定要改为主机的 IP 地址 -->
                                  port="5000"
                                  selectorTimeout="100"
                                  maxThreads="6"/>

                <Sender className="org.apache.catalina.tribes.transport.ReplicationTransmitter">
                  <Transport className="org.apache.catalina.tribes.transport.nio.PooledParallelSender"/>
                </Sender>
                <Interceptor
className="org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"/>
                <Interceptor
className="org.apache.catalina.tribes.group.interceptors.MessageDispatch15Interceptor"/>
                <Interceptor
className="org.apache.catalina.tribes.group.interceptors.ThroughputInterceptor"/>
              </Channel>

              <Valve                                  className="org.apache.catalina.ha.tcp.ReplicationValve"
filter=".*\.gif;.*\.js;.*\.jpg;.*\.png;.*\.htm;.*\.html;.*\.css;.*\.txt;"/>

              <!--Deployer className="org.apache.catalina.ha.deploy.FarmWarDeployer"
                          tempDir="/tmp/war-temp/"
```

```
                    deployDir="/tmp/war-deploy/"
                    watchDir="/tmp/war-listen/"
                    watchEnabled="false"/-->

          <ClusterListener className="org.apache.catalina.ha.session.ClusterSessionListener"/>
        </Cluster>
<-- 其余部分省略 -->
</Server>
```

8.5.2 Tomcat 的 web.xml 的配置

最后在 Web.xml 里面加上<distributable/>。官方文档没有这个，但测试发现加上更好，因为按照标准的 tomcat 启动，当 Host 对象被创建时，一个 Cluster 对象（默认配置下是 SimpleTcpCluster）也同时被关联到这个 Host 对象。当某个应用在 web.xml 中设置了 distributable 时，Tomcat 将为此应用的上下文环境创建一个 DeltaManager。SimpleTcpCluster 启动 membership 服务和 Replication 服务。

配置的实例如下：

```
<-- 前面内容省略 -->
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
        <welcome-file>index.htm</welcome-file>
        <welcome-file>index.jsp</welcome-file>
    </welcome-file-list>
    <distributable/>    <--这是新加的行 -->
</web-app>
```

8.5.3 项目的 web.xml 配置文件

每个项目也有一个 web.xml 配置文件。在<web-app>元组下也应当加上<distributable/>元组，以表明该项目需要集群服务。

在完成配置后，重新启动各个 tomcat 服务，在日志文件 catalina.out 中可以看到 cluster 成员的加入和退出情况，从而表明集群的构建。

在实际应用中，最常见的配置错误时没有配置 Receiver 元组下的 address 的值。该值最好是主机的 IP 地址。采用 auto 时可能绑定到不合适的地址，从而导致集群无法构建。

8.5.4 WEB 开发注意事项

1、启动 Tomcat 有可能报错，请查看网关是否配置。

2、Tomcat Session 时间应配置足够长，以防止调度到新服务器时 session 却已经超时。

3、应用 web 上的等待超时时间应设置足够长，以防止当调度成功前网页退出。

4、Session 中的结构体必须序列化，否则虚拟服务子系统之间无法进行 session 复制。这也是 tomcat 集群的要求。

9 安全

9.1 SELINUX 介绍

9.1.1 简介

SELinux(Security-Enhanced Linux) 即安全增强的 Linux。它是 Linux 最重要的一个安全模块。传统的 Unix 和 Linux 系统只有“自主访问”控制（即由文件、进程等的属主控制访问权限）机制，无法满足高安全应用的需求。SELinux 实现基于“类型”的强制访问控制机制，还包含一个多级别安全（MLS，有些文献翻译为多层安全）机制，以及基于角色的访问控制（RBAC）机制，是当前国内各个安全操作系统的核心和基础。数字有机体工作平台支持 SELinux，并用其实现特权用户分割，即“最小特权集”。

和自主访问控制不同，强制访问控制引入独立的访问控制机构，并采用统一的规则来判定访问是否许可。在访问控制中，访问者被称为“主体”，被访问者就是“客体”。早期的强制访问控制机制是多级别安全机制。MLS 机制给每个主体和客体设定一个固定的安全等级，主体可以读写相同等级的客体，但只能写高等级的客体，读低等级的客体。通过限制机密数据流向非安全人员，从而保证系统的安全性。MLS 机制过于僵化，因此 SELinux 主要提供类型强制机制（TE）。在类型强制下，所有主体和客体都有一个类型标识符与它们关联，要访问某个客体，访问规则中必须有主体的类型到客体的类型的访问授权，而不管主体的用户标识符。这样，管理员可以配置访问规则，从而建立起适合自身安全需求的访问控制。配置的访问规则又叫 SELinux 策略（policy）。

所有操作系统访问控制都是以关联的客体和主体的某种类型的访问控制属性为基础的。在 SELinux 中，访问控制属性叫做安全上下文。所有客体（文件、进程间通讯通道、套接字、网络主机等）和主体（进程）都有与其关联的安全上下文。一个安全上下文由三部分组成：用户、角色和类型标识符。为了便于将 Linux 用户映射到进程类型，这里引入角色的概念，从而实现基于角色的访问控制。

SELinux 的详细介绍和使用方法请参考 Linux 操作系统的相关文档，这里不再进一步描述。本章仅仅介绍如何使用数字有机体工作平台上的 SELinux。

9.1.2 配置启用 SELinux

9.1.2.1 开机启动 SELinux 配置

SELinux 是 Linux 的一个内核安全机制。可以配置 Linux 的内核启动参数来启用或者禁用 SELinux。数字有机体工作平台采用 GRUB 作为系统启动引导工具，因此需要修改 GRUB 配置文件，以设置内核启动参数。

可以用 vi 打开/boot/grub/grub.cfg 配置文件，在相应启动项的 linux 配置中增加“security=selinux selinux=1”参数。这里设置 Linux 安全机制为 selinux，selinux 的启用参数为 1。如果 selinux 的启用参数为 0 则 selinux 仍然不被启用。数字有机体工作平台的发

行版默认将 selinux 启用参数设置为 0，即没有启用。要启用时，linux 配置将如下所示：

linux /boot/vmlinuz-3.2.34 root=/dev/sda1 ro quiet security=selinux selinux=1 splash

重新启动主机，并选择设置了 selinux 起效的启动项后，启动的系统中 selinux 将被启用。

9.1.2.2 SELinux 配置文件

SELinux 的启动配置文件为/etc/selinux/config。其内容如下所示。

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
# enforcing - SELinux security policy is enforced.
# permissive - SELinux prints warnings instead of enforcing.
# disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of these two values:
# default - equivalent to the old strict and targeted policies
# mls         - Multi-Level Security (for military and educational use)
# src         - Custom policy built from source
SELINUXTYPE=default
# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0
```

该配置文件只是三个配置项，他们分别如下：

(1) 配置项：**SELINUX = enforcing**

效果：SELINUX 有 3 个选项：permissive（许可模式）、enfocing（强制模式）、disabled（禁止 SELinux）。许可模式是一种测试模式，访问控制机制将检查每一次访问，并对将禁止的访问给出拒绝消息，但是并不真正阻止访问。强制模式下则会阻止访问。而 disabled 模式下 selinux 不做访问检查。

(2) 配置项：SELINUXTYPE=default

效果：配置使用哪一组策略。在/etc/selinux 目录下有多个策略目录。通常包括 default 和 MLS。其中 default 是发行版默认的自带策略。MLS 是实现多级安全机制的策略。您也可以编写自己的策略，然后命名为 src。

(3) 配置项：SETLOCALDEFS=0

效果，是否检查本地定义变化，默认是 0，即不检查。

9.1.2.3 更换或者启用策略文件

SELinux 的策略是经过编译后的二进制文件。如果你需要更换原有的策略，你可以编写新策略，并将 拷贝策略到/etc/selinux 下，并将策略目录名称改为 SELINUXTYPE 选项所指明的名称。当系统重启 selinux 生效时，会读取本目录下的策略作为启动时使用文件的策略，如果违反策略，则系统无法启动。数字有机体工作平台已经包含启动时所需要的基本策略以及其他基本服务的策略，非策略允许的非法操作将被禁止。

9.1.2.4 重新标记文件系统

通常，在启用一个新策略后，或者系统的某些文件需要标记上下文时，需要进行一次文件重新标记工作。如果文件没有标记安全上下文，或者标记的安全上下文是错误的，则 SELinux 的访问控制机构就无法进行正确的访问合法性判断。

给所有文件标记安全上下文的命令如下：

```
Fixfiles  –F  relabel
```

为所有文件标记安全上下文需要较长的时间。具体花费的时间与机器性能和文件的数量有关。完成文件安全上下文标记后，SELinux 就能有效运行了。

9.1.3 SELinux 用户常用命令

SELnux 常用命令主要针对 SELinux 的状态查看，策略管理，用户管理的相关命令，其他 SELinux 命令此处不做介绍。其中部分命令必须在 selinux 为许可模式下才能使用，故只有管理员可以执行，该命令与策略加载，模块加载，添加用户相关，非管理员用户无法修改 SELinux 状态，故无法执行相关命令。

即便使用已有的策略，在不同的软硬件环境也可能出现 AVC 的 denied 事件。为了快速查找并弥补 AVC 规则，用户应该首先进入到 permissive 模式，待出现 AVC 的 denied 事件后，执行“dmesg | grep avc | audit2allow -m local”命令，即可自动生成所需的 AVC 规则，并提示正常运行所需解决的问题（一般为约束）。

9.1.3.1 ls 命令

说明：该命令可查看文件安全上下文，需要使用–Z 或者 –context 选项。

用法：ls -Z 或 ls --context

例如：

```
root@server214:/etc/selinux# ls -Z /etc/selinux/config
system_u:object_r:selinux_config_t:s0 /etc/selinux/config
root@server214:/etc/selinux# ls --context /etc/selinux/config
system_u:object_r:selinux_config_t:s0 /etc/selinux/config
```

9.1.3.2 chcon 命令

说明：该命令用于修改文件或者目录的安全上下文。

用法： chcon [选项]... 安全上下文 文件...

或: chcon [选项]... [-u 用户] [-r 角色] [-l 范围] [-t 类型] 文件...

或: chcon [选项]... --reference=参考文件 文件...

说明：该命令将文件的安全上下文变更至指定安全上下文。

使用 --reference 选项时，把指定文件的安全上下文设置为与参考文件相同。

-h, --no-dereference 影响符号连接而非引用的文件。

--reference=参考文件 使用指定参考文件的安全环境，而非指定值。

-R, --recursive 递归处理所有的文件及子目录

-v, --verbose 为处理的所有文件显示诊断信息

-u, --user=用户　　设置指定用户的目标安全环境

-r, --role=角色　　设置指定角色的目标安全环境

-t, --type=类型　　设置指定类型的目标安全环境

-l, --range=范围　　设置指定范围的目标安全环境

以下选项是在指定了 -R 选项时被用于设置如何穿越目录结构体系。

如果您指定了多于一个选项，那么只有最后一个会生效。

-H　　如果命令行参数是一个通到目录的符号链接，则遍历符号链接

-L　　遍历每一个遇到的通到目录的符号链接

-P　　不遍历任何符号链接(默认)

9.1.3.3 mount 命令

说明：在安装文件系统时可以指定新安装文件系统中文件的安全上下文。

用法： mount -t type dev dir -o context=”用户：角色：类型：安全等级”

例如：安装数字有机体工作平台到/dpfs 下，并指定安全上下文。

mount –t dpfs none /dpfs –o context=“system_u:object_r:mnt_t:s0“

说明：该功能主要用于处理那些不支持通过文件扩展属性保存文件安全上下文的文件系统。

9.1.3.4 ps 命令

说明：可在查看进程时显示进程的安全上下文。这时需要使用-Z 或者—context 选项。

用法：ps –Z 或 ps –context

例如：

```
root@server214:/etc/selinux# ps -Z
LABEL                             PID TTY      TIME CMD
sysadm_u:sysadm_r:sysadm_t:s0    4136 pts/0    00:00:00 bash
sysadm_u:sysadm_r:sysadm_t:s0   15782 pts/0    00:00:00 ps
```

9.1.3.5 restorecon 命令

用来恢复文件在策略文件（/etc/selinux/default）里定义的安全上下文，

用法：restorecon [-o outfilename] [-R] [-n] [-p] [-v] [-e directory] pathname...

restorecon -f infilename [-o outfilename] [-e directory] [-R] [-n] [-p] [-v] [-F]

说明：该命令选项和参数众多，请参考 SELinux 的文档。

9.1.3.6 sestatus 命令

说明：显示当前 SELinux 的信息，只有-b 和-v 两个选项。-b 选项使其显示当前 selinux 策略中所有布尔值的当前值。-v 使其显示当前的进程和文件的上下文。如果不带选项则只显示 selinux 的基本信息。

用法：sestatus [OPTION]

例如：

```
root@server214:/# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
```

```
SELinux root directory:          /etc/selinux
Loaded policy name:              default
Current mode:                    permissive
Mode from config file:           permissive
Policy MLS status:               enabled
Policy deny_unknown status:      denied
Max kernel policy version:       26
```

显示的内容中，SELinux status 是 SELinux 机制的状态，enabled 表示起效。Loaded policy name 表示使用的策略。Current mode 表示当前的 SELinux 模式，介绍 SELinux 配置文件时已经说明。

9.1.3.7 getenforce 命令

说明：得到当前 SELinux 模式的值。返回值为 permissive(许可)、enforcing（强制）或 disabled（无效）之一。
用法：getenforce

9.1.3.8 setenforce 命令

说明：设置当前 SELinux 的模式值
用法：setenforce [Enforcing | Permissive | 1 | 0]
备注：可以使用 1 和 0 表示强制模式和许可模式。改变模式后将立即生效。

9.1.3.9 load_policy 命令（许可模式）

说明：用于加载设定的策略，在修改配置文件指定要使用的策略，或者替换策略文件后，可以使用该命令载入新策略。
用法：load_policy

9.1.3.10 setfiles 命令

说明：功能给跟 chcon 相同，用于设定文件或者目录的安全上下文。
用法：setfiles [-dnpqvW] [-o filename] [-r alt_root_path] spec_file pathname...
setfiles -c policyfile spec_file
setfiles -s [-dnpqvW] [-o filename] spec_file
备注：该命令参数和选项众多，请具体参考 SELinux 的文档。

9.1.3.11 semanage 命令（许可模式）

说明：由于 semanage 用法复杂，此处介绍其常见用法，详细用法请参照 man 手册，常见用法是用于查看，添加，删除其 SELinux 用户以及 SELinux 用户到本地用户的映射。
用法：
SELinux 用户相关用法：semanage user -{a|d|m|l|D|E} [-LnrRP] selinux_name
SELinux 用户到本地用户映射相关用法：semanage login -{a|d|m|l|D|E} [-nsr] local_name
例如：

1) 查看所有 SElinux 用户及其所扮演的角色
   semanage user –l
2) 查看本地用户到 SELinux 用户映射
   semanage login –l
3) 添加一个 SELinux 用户 test_u 的用户，并扮演 SELinux 中的 sysadm_r 角色。

semanage user –a –R “sysadm_r” test_u

4) 删除一个 SELinux 用户 test_u

semanage user –d test_u

5) 添加一个 SELinux 用户 test_u 到本地用户 dos 的映射

semanage login –a –s “test_u” dos

6) 或删一个 SELinux 用户到本地用户 dos 的映射

semanage login -d dos

9.1.3.12 semodule 命令（许可模式）

说明：semodule 用于模块的增加，删除，更新等，此处列举其几种常用选项，其他的选项可参考 man 手册，就不一一列举。

用法：semodule [options] [模块名]

例如：

```
#semodule –r xxx.pp #移除模块
#semodule -i xxx.pp #安装一个模块
#semodule -b base.pp #安装或替换 base.pp
#semodule -l #查看模板列表
#semodule -DB #使”dontaudit”失效，即使与该规则相关的 AVC 消息写入 日志
#semodule -I *.pp #安装或替换当前目录的所有非基础模块的 pp
#ls *.pp | grep –Ev ”base.pp|enableaudit.pp” | xargs semodule –b base.pp –i #安装当前目录所有 pp
```

9.1.3.13 sesearch 命令

说明：用于策略查询，常用 grep 命令配合查询，此处列举其常用选项的用法。

用法：sesearch [OPTIONS]

例如：

```
#sesearch --allow | grep “bin_t” | grep “user_home_t” #搜索 allow 策略 一般和 grep 配合搜索
#sesearch --all | -- #搜索所有 --表示同上
#sesearch --dontaudit | -- #搜索不允许打印消息的策略
#sesearch --type | -- #搜索类型转换策略
#sesearch --role_allow | -- #搜索角色许可
#sesearch –role_trans | -- #搜索角色转换
```

9.1.3.14 fixfiles 命令

说明：用于重新标记文件系统中所有文件的安全上下文。

用法：fixfiles [-F] [-l logfile] { check | restore| [-f] relabel | verify } [[dir/file] ...]

例如：

fixfiles –F relabel

9.1.4 数字有机体工作平台上的 SELinux 用户角色

数字有机体工作平台的发行包中自带了一个默认安全策略，它配置了大多数应用需要的安全策略，并实现了 Linux 超级用户的分割，满足“最小特权集”的要求。如果有需要，您也可以编写自己的安全策略，并且使用它。如何编写安全策略请参考 SELinux 的相关文档。下面仅介绍数字有机体工作平台自带安全策略。

数字有机体工作平台将用户角色分为管理员、普通用户、安全管理员、日志管理员和重启管理员。此五个用户角色皆为 SElinux 用户角色，必须与本地用户相关联才能使用。如何关联本地用户与 SELinux 用户角色请参考 semanage 命令（9.1.4.11）的使用。下面分别介绍各个角色：

1) 管理员用户，位于 adm 组下的用户，关联的 selinux 用户为 sysadm_u，可以以普通用户的身份远程登录，并且执行 sudo su 命令切换为管理员用户。该用户可以进行所有的管理操作，类似于超级用户，但无法直接访问 shadow 文件。
2) 普通用户，即不在 adm 或 root 组的用户,其关联的 selinux 用户为 user_u，可以远程登录系统，可以对自己主目录(/home/用户名/)下的文件进行任意管理操作,可以使用一些常用操作命令,如：ls、grep、find 等，但无法执行管理员相关命令,例如：用户管理命令（useradd、groupadd 等）、服务启动命令（/etc/init.d/tomcat start、/etc/init.d/vsftpd start 等）、进程管理命令等，总而言之，该用户只是为了用于管理个人文件信息而创建。
3) 安全管理员，位于 adm 组下的用户，关联的 selinux 用户为 secadm_u,可以以普通用户身份远程登录，并且通过执行 sudo su 命令切换为安全管理员用户，该用户主要负责策略管理工作,强制模式（enfocing）下只能进行策略检查工作，无法修改或安装策略，需要修改、安装重新加载策略必须由管理员将其模式改为许可模式（permissive）后才能进行。该用户只关心系统策略安全问题，故用户没有个人文件资料管理目录，对大部分文件如：日志文件（/var/log/目录下）,配置文件(/etc 目录下)仅有查看功能，无法修改。
4) 日志管理员，位于 adm 组下的用户，关联的 selinux 用户为 logadm_u，可以远程登录，主要负责日志查看工作，发现异常或错误，并向相关人员报告。该用户仅有读取功能且主要工作目录为/var/log/。
5) 重启管理员，位于 adm 组下的用户，关联的 selinux 用户为 sdadm_u,可以以普通用户身份远程登录，并且执行 sudo su 命令切换为重启管理员，该用户能执行的 shell 命令仅为 shutdown –r，即该用户无法操作任何文件。

数字有机体系统默认只使用了“管理员用户”角色，与该角色对应的 linux 用户名为“root”和“dos”，不可删除。其余的角色没有启用，启用的方式是：

1) 新建 linux 的用户，例如为 loguser；
2) 使用 semanage 指令添加 linux 用户对应的角色，例如 semanage login -a -s "logadm_u" loguser；
3) 重新启动计算机，使之起效。

如果以上的角色无法满足用户的需求，用户还可以修改或者编写新的规则来满足自身需求，selinux 源码位于“/usr/src/selinux”目录。

9.2数字有机体防火墙

9.2.1 系统简介

数字有机体工作平台整合分布在网络中的大量服务器来给网络用户提供服务。为了保护服务系统不受外界的网络攻击，系统使用 iptables 构建统一的主机防火墙。但是，如此众多的服务器使得管理员难以进行有效地配置和管理。为此，数字有机体工作平台实现了一套防火墙配置系统，即数字有机体防火墙配置系统。它包括两部分：前台配置页面和后台守护进程。用户在页面上设置防火墙的配置，守护进程定时读取配置信息，并控制 iptables，使其满足系统的网络安全要求。

主机防火墙 iptables 的策略是由一组有序的规则建立。它告诉内核应如何处理某些类别的数据包。每一个 iptables 规则应用于一个表中的一个链。一个 iptables 链是一个规则集，这些规则按序与包含共同特征的数据包进行比较。下面是一些重要概念的说明。

表：表是 iptables 构建块。iptables 中共有 4 个表：filter、nat、mangle 和 raw。防火墙过滤规则应用于 filter 表。防火墙配置系统只配置 filter 表。

链：每个表都有自己的一组内置链。用户还可以对链进行自定义，这样用户就可以建立一组规则，它们都关联到一个共同的标签。对于配置系统来说，主要使用的内置链是 filter 表中的 INPUT 和 OUTPUT 链。当一个数据包由内核中的路由计算确定为指向本机系统之后，它将经过 INPUT 链的检查。OUTPUT 链用于过滤从主机发送出去的数据包。通俗的说， INPUT 代表了主机的输入检查规则链，OUTPUT 代表了系统的输出检查规则链。

匹配与目标：每个 iptables 规则都包含一组匹配以及一个目标。Iptables 匹配指的是数据包必须匹配的条件。只有当数据包满足所有的匹配条件时，iptables 才能根据由该规则的目标所指定的动作来处理数据包。

数字有机体防火墙配置系统抽象了上述概念，使得管理员更加方便地配置系统中各台服务器的防火墙策略。

9.2.2 安装软件

数字有机体防火墙配置系统由守护进程和用户 UI 页面组成。守护进程是一个独立的安装包，需要单独安装。如果您使用的是数字有机体系统发行版，则该软件已经安装在系统中。用户 UI 页面即数字有机体工作平台的管理网站。在数字有机体系统发行版中也已包含。

数字有机体防火墙配置系统守护进程安装包全称名为 dosfirewall-1.1-1.x86_64.rpm，其中 dosfirewall 为包名字，-1.1.1 为版本号，x86_64 表示用于 64 位的 linux 操作系统。其中安装步骤如下：

第一步：拷贝安装包到 Linux 操作系统 home 目录下，注意安装包的版本是否符合你的操作系统。

第二步：执行安装命令：rpm -ivh --nodeps dosfirewall-1.1-1.x86_64.rpm

执行结果如图 9-1 所示。

```
root@server214:/home# rpm -ivh --nodeps dosfirewall-1.1-1.x86_64.rpm
rpm: RPM should not be used directly install RPM packages, use Alien instead!
rpm: However assuming you know what you are doing...
Preparing...                          ################################# [100%]
Updating / installing...
   1:dosfirewall-1.1-1                ################################# [100%]
```

图 9-1 安装守护进程

第三步：打开配置文件：vi　所示。

db_name=ridi_1

db_user=dba

db_passwd=sql

db_host=localhost

db_connects_num=4

with_tcp_track=0

该配置文件设置守护进程的配置信息数据源，即获取配置信息的工作库。通常，配置信息保存在数字有机体工作平台的文件属性库中。为了统一管理，数字有机体系统中的某个站被设置为管理站。全局的配置信息则保存在管理站的文件属性库中。因此，该配置文件就是设定管理站的文件属性库的信息。

配置项“with_tcp_track”用于设置是否启用 TCP 连接跟踪功能。启用该功能可以提升安全性，但是会导致虚拟网络服务异常。如果没有需要，建议不要启用。设置为 1 时启用 TCP 连接跟踪功能。

第四步：启动防火墙配置进程 dosfw。可以直接在终端输入 dosfw 命令来启动进程。然后采用命令 pidof 所示。

```
root@server214:/home# dosfw
root@server214:/home# pidof dosfw
6718
```

图 9-2 查看进程

也可以将防火墙配置进程作为系统服务在主机启动时自动启动。将其设为自动启动的方法如下：

```
update-rc.d    dosfw enable
```

如果要去除自动启动，则可以使用以下命令：

```
update-rc.d    dosfw disable
```

如果你使用的是数字有机体工作平台的 4.0 及以上版本，则在安装好系统后已经内置了数字有机体防火墙配置系统。你只需从第三步开始配置即可。在数字有机体工作平台内，防火墙使用的数据库必须是系统管理站的文件属性库（ridi）。要注意的是，所有服务器必须访问相同的配置库，这样才能形成一致的，完整的结构。如果各自访问不同的配置库，将使得系统的配置不一致。

如何安装数字有机体工作平台的管理网站请参见第十章。

9.2.3 防火墙规则配置说明

数字有机体系统中的服务器都提供相同的服务，或者许多服务器提供相同的服务；同时，这些主机通常面临相同的网络环境，对访问的限制也大多数相同。因此，没有必要分别为每台服务器配置防火墙规则，可以采取分类配置的方式。

为了简化配置，数字有机体防火墙配置系统采用默认设置和个性设置相结合的方法。基本的思想是：最大化限制输入，最小化限制输出。因此，首先用规则禁止所有输入和允许所有输出，并对所有非法输入和输出记录日志；最后设置允许输入规则和禁止输出规则。另外，考虑到各台服务器间的通信要求，允许系统内节点间相互访问，本机可以访问自己，允许 ICMP echo。这些被称为默认规则，是管理员无需进行配置的部分。

另一个问题是：是否启用连接跟踪策略。对 TCP 协议来说，启动连接跟踪策略可尽早丢弃不合法的包，从而加强控制并防范攻击。但是，启用连接跟踪后，会影响虚拟网络服务的访问，并且可能延迟系统启动的时间。因此，在使用虚拟网络服务时不要启用连接跟踪。如果不是特别需要，也不要启用连接跟踪。

根据基本思想，管理员要设置的是允许的输入规则和禁止输出规则。对服务器来说，允许某个输入即允许客户访问某项服务。因此，可以将允许输入规则看作允许访问某项服务的规则。而某项服务往往是由系统全部节点或者某些节点提供的。为了方便，将系统全体节点都提供的服务看作一类，称为系统对外服务。如果仅是部分节点提供的服务，则需要分别配置每个节点，即配置节点对外服务。最后管理员根据特殊需要，配置自定义规则。因此，需要配置的规则分为三类：系统对外提供的服务、某个节点单独对外提供的服务（指定服务提供者），用户自定义规则。

数字有机体防火墙只使用 iptables 的过滤表（filter），且只配置输入链和输出链，其他规则无法通过防火墙配置系统进行配置。

1) 过滤表的默认规则

过滤表的默认规则有四条，如下所示。其功能分别是默认禁止所有输入，允许所有输出，开启连接跟踪。

```
iptables –P INPUT DROP
iptables –P OUTPUT ACCEPT
modprobe ip_conntrack
modprobe ip_conntrack_ftp
```

2) 输入链的配置

根据设计思想，输入链的规则顺序为：默认输入连接跟踪处理（开启连接跟踪时）、默认允许规则、系统对外提供服务的规则、节点对外提供服务的规则、用户自定义输入允许规则、默认输入日志记录规则。他们的定义分别如下。

输入连接跟踪处理部分(开启连接跟踪时有效)：

```
iptables -A INPUT -m state --state INVALID -j LOG --log-prefix="DROP INVALID " --log-ip-options
--log-tcp-options
iptables -A INPUT -m state --state INVALID -j DROP
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

默认允许规则：

该部分核心的是允许系统内的节点相互访问。如果没有开启连接跟踪则没有“--syn -m state --state NEW”部分。下面类似的规则也是一样的。

```
##allow ping this host.
iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
##allow all tcp and udp for dos system node
iptables -A INPUT -p udp -s $DOS_NODE_LIST -j ACCEPT
iptables -A INPUT -p tcp -s $DOS_NODE_LIST --syn -m state --state NEW -j ACCEPT
```

##allow local host

```
iptables -A INPUT -p tcp -s 127.0.0.1/32 --syn -m state --state NEW -j ACCEPT
iptables -A INPUT -p udp -s 127.0.0.1/32 -j ACCEPT
```

系统对外提供服务的规则：

该部分根据管理员配置系统对外服务信息，生成允许访问服务的输入规则。对配置的每个服务，在每台服务器上转化为一条运行规则。如果协议为 TCP，则格式为：

```
iptables -A INPUT -p tcp  －s 允许范围 --dport 服务端口 --syn -m state --state NEW -j ACCEPT
```

如果协议是 UDP，则格式为：

```
iptables -A INPUT －p udp  －s 允许范围 --dport 服务端口 －j ACCEPT
```

如果服务范围要取反，则在-s 前加叹号和空格。

节点对外提供服务的规则

该部分根据管理员配置的节点对外服务信息，由守护进程按照自己的 IP 地址查询自己提供的服务记录。每条记录转化为一条运行规则。如果是 TCP 协议，格式为：

```
$IPTABLES -A INPUT -p tcp –s 允许范围 --dport 服务端口 --syn -m state --state NEW -j ACCEPT
```

如果是 UDP 协议，则格式为：

iptables -A INPUT -p udp －s 允许范围 --dport 服务端口 -j ACCEPT

如果服务范围要取反，则在-s 前加叹号和空格。

限制在系统公共服务表里已有的服务端口在各个节点上不能再出现。

用户自定义输入规则：

该部分根据管理员配置的自定义输入规则生成防火墙规则。每条规则转换为一条允许规则，如果是 tcp 且需要记录则为：

```
iptables -A INPUT -p tcp -s 源地址范围 --sport 源端口 –d 目的地址范围 --dport 目的端口 --syn -m state --state NEW –j LOG --log-prefix "ACCEPT" --log-ip-options --log-tcp-options
iptables -A INPUT -p tcp -s 源地址范围 --sport 源端口 –d 目的地址范围 --dport 目的端口 --syn -m state --state NEW -j ACCEPT
```

如果不需要进行记录则不增加第一条规则。

如果是 UDP 协议且需要记录日志则改为：

```
iptables -A INPUT -p udp -s 源地址范围 --sport 源端口 –d 目的地址范围 --dport 目的端口 -j LOG --log-prefix " ACCEPT " --log-ip-options
```

```
iptables -A INPUT -p udp -s 源地址范围 --sport 源端口 –d 目的地址范围 --dport 目的端口 -j ACCEPT
```

如果不需要进行记录则不增加第一条规则。

源地址范围和目的地址范围都可以取反。

默认输入日志记录规则：

最后是默认的日志记录规则，管理员无需配置。

```
### defualt INPUT LOG rule
iptables -A INPUT  ! -i lo -j LOG --log-prefix "DROP" --log-ip-options --log-tcp-options
```

3) 输出链的配置

输出链的配置要简单的多。由于默认允许所有输出，而限制输出往往是特殊的情况，因此输出链配置的只有禁止输出的自定义规则。

禁止输出的自定义规则和自定义输入规则一起配置，只是选择的是输出链。生成规则的方式也类似，只是这里是“DROP”处理，记录日志的标记是“DROP”。

9.2.4 使用配置系统

如图 9-3 所示，管理员用浏览器打开数字有机体管理系统网页，然后用 DOSroot 账号登入系统。只有该用户能够配置系统的防火墙。

图 9-3 数字有机体管理系统

配置工作可以分为两部分。一部分是配置系统的数字有机体站和每个站的服务器。第二部分是配置防火墙的过滤规则。下面分别进行描述。

9.2.4.1站点节点配置

如图 9-4 所示，进入数字有机体管理系统后，点击服务管理的站点节点配置菜单进入站点配置页。每个站点内包含有若干个节点。

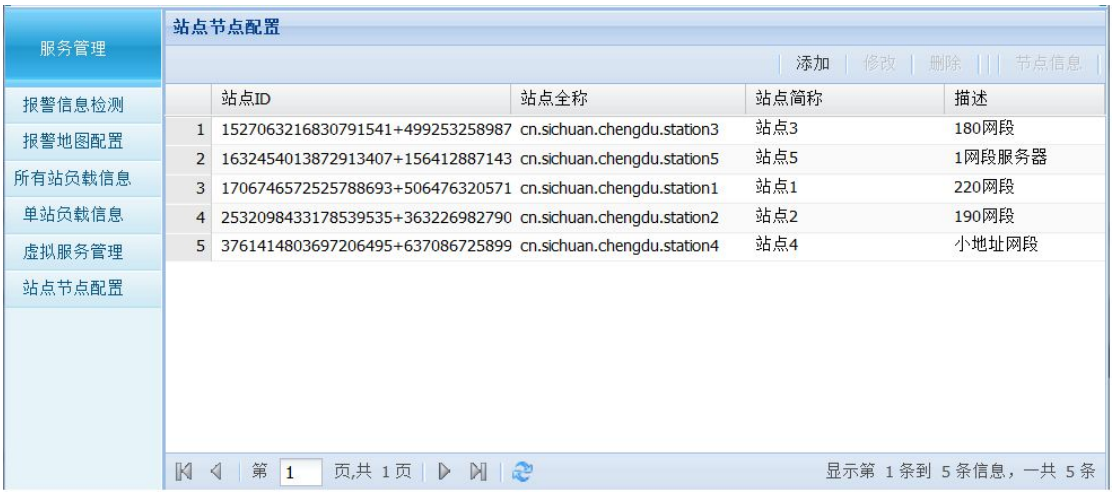


图 9-4 站点节点配置

(1) 添加站点

点击页面的添加按钮即可添加站点。如图 9-5 所示。其中，站点 ID 要求使用数字有机体工作平台中每个站的 ID。可以从站内服务器的/etc/dos_exernel.cnf 文件中查到。站点全称则建议使用/etc/dos_exernel.cnf 文件中站点描述信息。

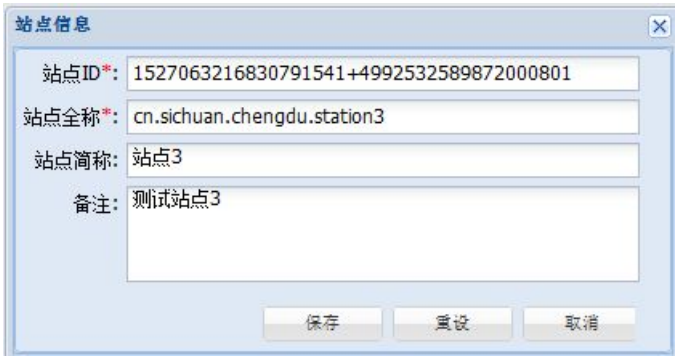


图 9-5 添加站点信息

(2) 修改站点

修改站点即在已有的站点信息的基础上进行修改。在站点列表中选择要修改的站点后点击“修改”按钮即可进入修改对话框。该对话框和添加站点的对话框相同，只是不能修改站点的 ID 和全称，其余信息都可修改。

(3) 站点删除

先选中相应的站点信息，点击删除按钮，弹出如图 9-6 所示的提示框。点击是即可删除相应的站点信息

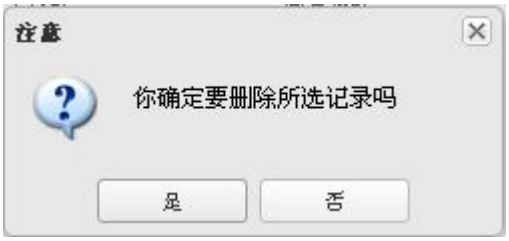


图 9-6 删除站点信息提示

(4) 节点信息

每个站点内包含了若干个节点。点击节点信息查看，如图 9-7 所示。

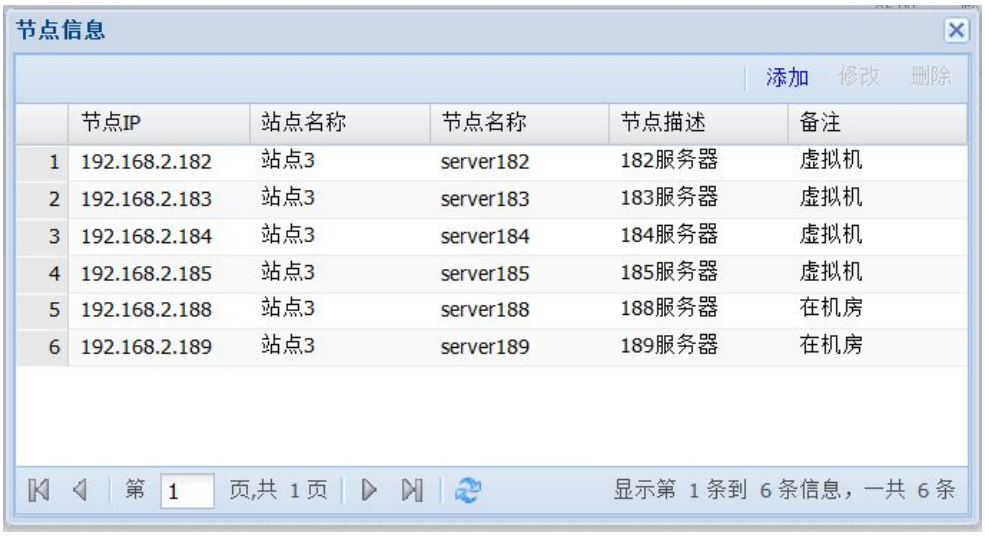


图 9-7 站内节点信息

(5) 节点添加

点击添加按钮可在相应的站点内添加节点，如**错误！未找到引用源。**所示。

图 9-8 节点添加

(6) 节点修改

选中相应的节点点击站内节点修改按钮弹出对话框如图 9-9 所示。

图 9-9 节点修改

(7) 节点删除

先选中要删除的节点，点击删除按钮弹出提示框如图 9-10 所示。点击是即可删除相应节点。

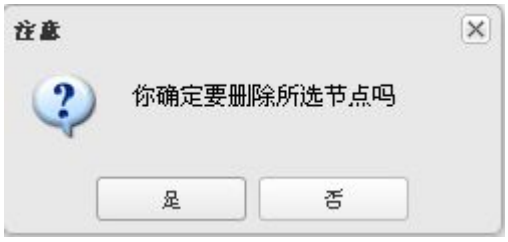


图 9-10 节点删除提示框

9.2.4.2规则配置

前面已经描述防火墙规则的配置原理，下面介绍如何在 UI 界面上进行配置。

1) 系统对外服务规则配置

系统对外的服务规则即用于限定系统所提供的服务。系统服务在数字有机体系统中为每个节点主机都能提供的服务。

系统对外的系统规则配置如图 9-11 所示。在这页面用户可以添加、删除和查询系统规则。

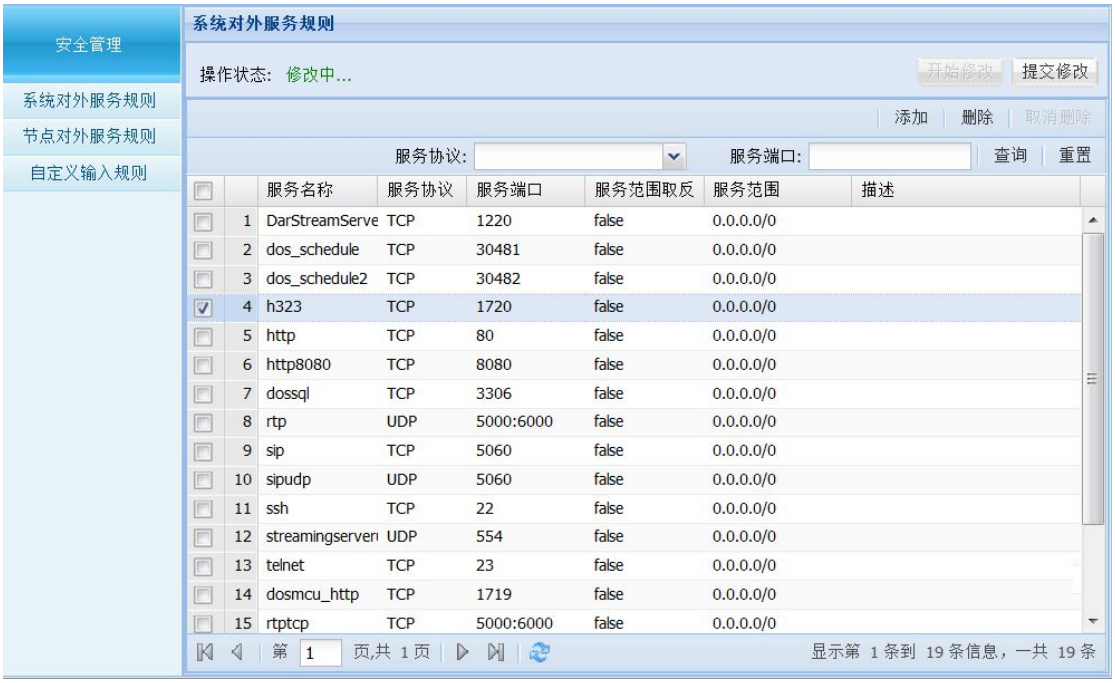


图 9-11 系统对外的系统规则配置

注意，由于规则间有依赖关系，因此对规则的修改以事务的模式进行。需要先点击“开始修改”按钮启动一次修改事务，在修改完成后，再点击“提交修改”按钮保存整个修改内容。如果未点击“提交修改”就关闭页面，则修改的信息仅仅保存在临时数据表中，并不生效。

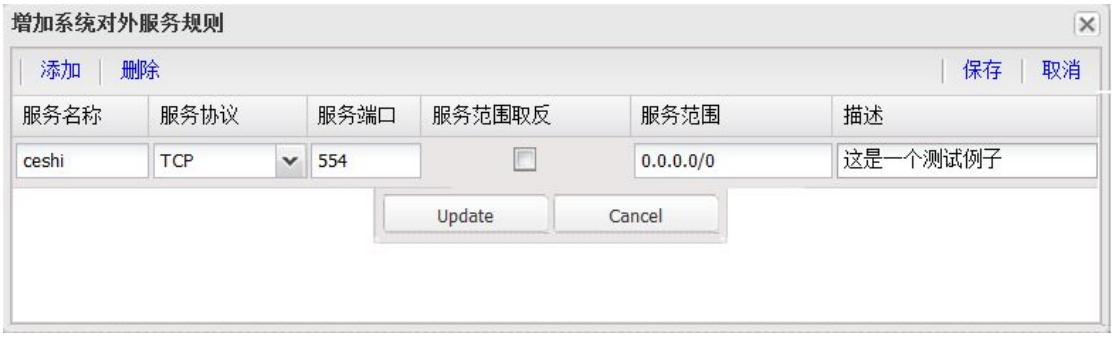


图 9-12 添加系统对外服务规则

点击“开始修改”按钮后，“添加”等按钮就变为可用的。点击“添加”按钮弹出系统规则添加窗口如图 9-12 所示。在图 9-12 窗口中点击添加按钮在表单中插入一行对话框。填写相关数据后，点击 update 按钮把相关数据插入到表单，点击 cancel 则取消操作。这个时候插入的数据为临时数据，点击右上角的保存按钮后表单数据才插入系统规则配置主页表单中，点击取消则不插入表单。关闭页面时临时数据将清空。在配置主页点击“提交修改”按钮将表单数据最终写入数据库中保存。如果这时还要进行新的配置添加，再次点击“开始修改”按钮即可。

设置对外服务时，服务名称可用根据需要填写，长度不超过 25 个字节。服务协议可以是 TCP 和 UDP，这是由服务本身所使用的协议决定的。如果服务同时使用了两个协议，则需要分为两条配置记录。服务端口为服务使用的端口号。如果是单个端口则写入端口号即可，如果是一个范围，可用“起始端口:结束端口”表示。服务范围为网络地址，可以对

服务范围取反。

2) 节点对外服务规则配置

节点对外服务的规则用于设定数字有机体工作平台中某个节点所提供的服务。在这里，制定的规则针对具体某个节点。

节点对外服务规则的制定页如图 9-13 所示。用户可以添加、删除和查询相关的规则信息。

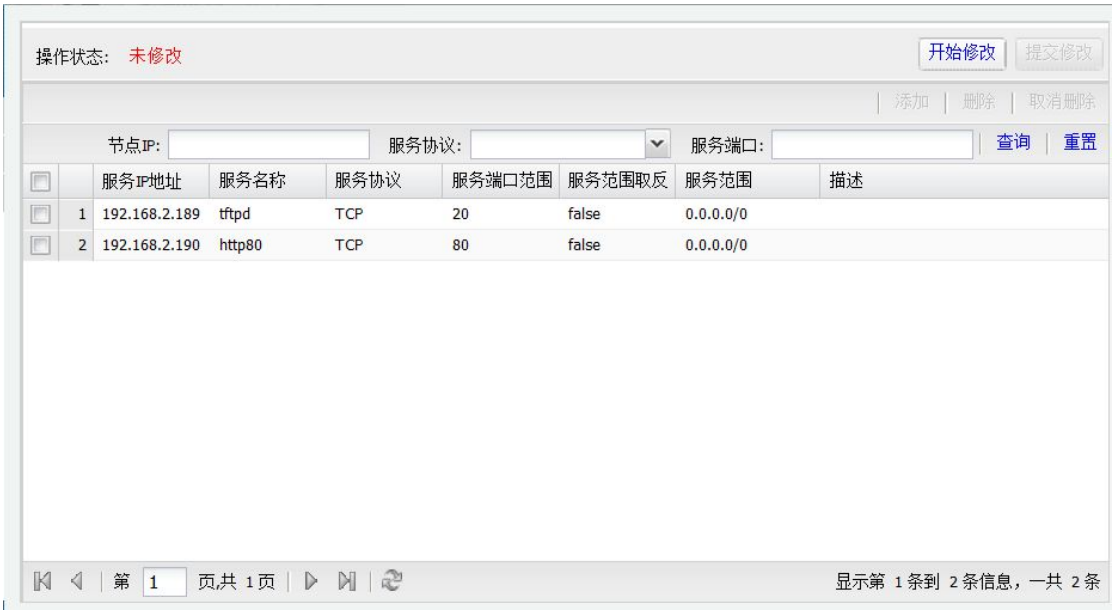


图 9-13 节点对外服务规则制定

与添加系统对外服务规则类似，仍然以事务的模式进行操作。在主配置页面点击“添加”按钮弹出节点添加服务规则窗口如图 9-14 所示。在弹出的窗口中点击添加，在表单中插入一行添加规则对话框。与系统添加规则类似，不同在于此处多一个 IP 地址参数。该参数指定提供服务的节点。填好相关数据项后点击 update 插入到表单，点击 cancel 取消。当规则添加完毕，相应的数据并没有写入数据库，只有点击右上角的保存，相应的数据才插入节点规则配置主页表单中，点击取消即不保存。最后在配置主页点击提交修改按钮将表单数据最终写入数据库中保存。如果这时要进行新的配置添加，点击开始修改即可。

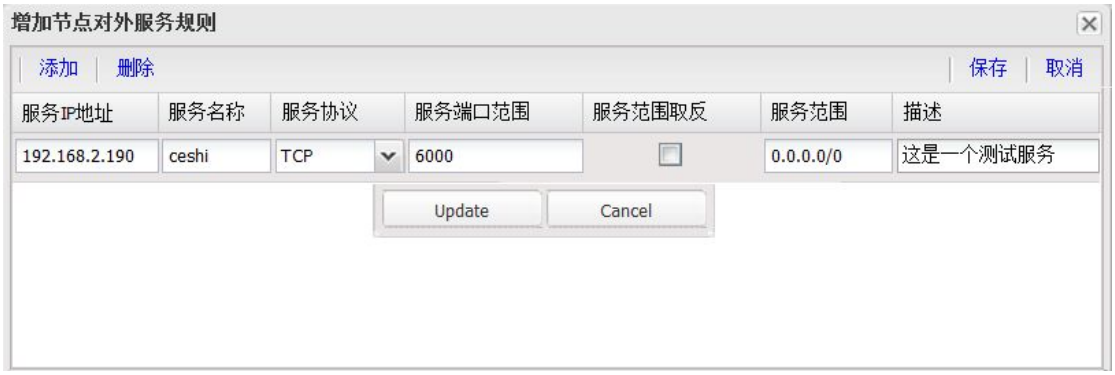


图 9-14 添加节点对外的服务规则

3) 自定义输入规则配置

用户自定义规则配置主页面如图 9-15 所示。用户可以在此页面添加、删除和查询相关的规则操作。

用户自定义规则主要用于设置节点的输入和输出特殊规则。要注意的是：对输入规则来说，这里设定的是允许输入的情况；对输出规则来说，这里定义的却是禁止输出的情况。两者是相反的。

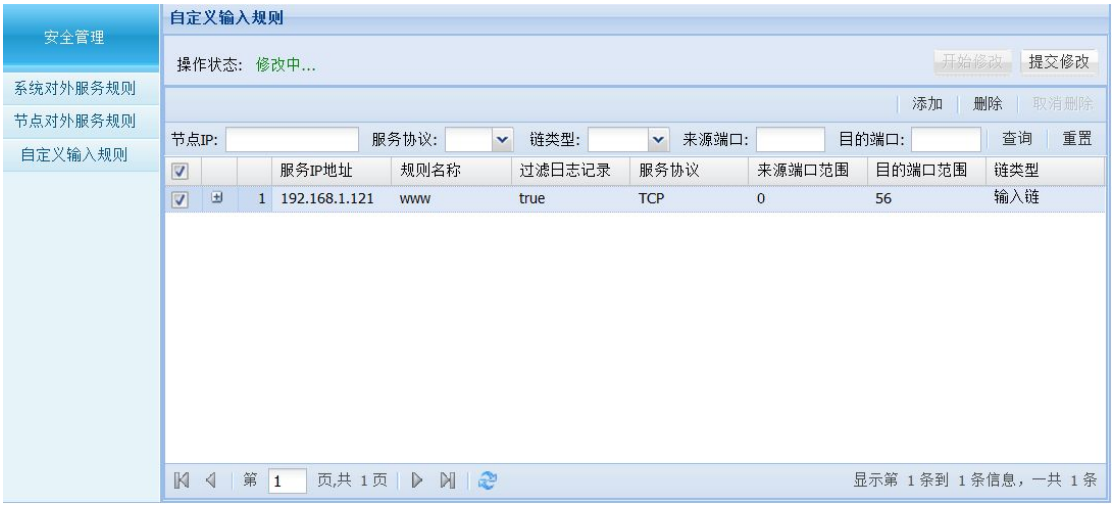


图 9-15 添加用户自定义规则

点击主页面添加按钮，弹出用户自定制添加规则窗口。如图 9-16 所示。其添加规则与系统对外服务规则和节点对外服务规则类似。

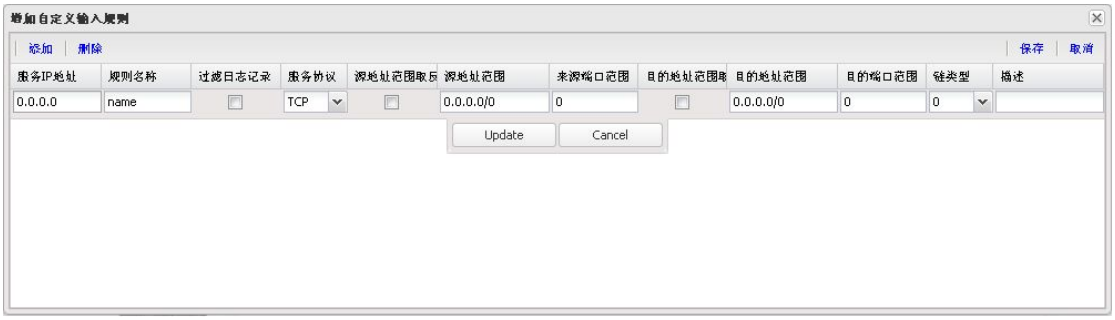


图 9-16 添加用户自定义规则窗口

如果不知道服务 IP 地址，则认为是所有节点都要设置的自定义规则。这里可以选择链类型，即是配置输入链还是输出链。

9.3 主机鉴别和安全通信

为了防止攻击者假冒系统的主机，数字有机体系统采用主机间相互认证的方式实现主机真伪鉴别，并且采用随机加密方式和 ssl 协议通信实现安全通信。主机间的鉴别是通过为每台主机发放经数字签名的证书，然后在通信时鉴别证书真伪来实现的。经过数字签名的证书可以防止篡改和假冒。同时，为了防止通信过程中，攻击者通过截获网络数据包来窃取数据，通信时采用 ssl 通信机制，并使用随机加密算法和随机密钥方式保证通信过程的安全，从而保证数字有机体系统的安全。

9.3.1 证书制作

数字有机体工作平台使用的数字证书是使用 OpenSSL 制作的证书。该证书内不仅封装

了颁发证书机构的信息，还包含了证书持有者的 IP。在鉴别证书时，不仅鉴别对方的证书真伪，还鉴别对方的 IP 和证书持有者 IP 是否相同，从而防止攻击者窃取证书后假冒系统主机。不过，现在攻击者仍然可以通过同时假冒 IP 地址来达到攻击的目的。因此，仍然需要妥善保管每台主机的证书。

1） 制作机构证书时，将提示输入加密证书的密码，请按照如下的提示输入并再次输入以进行确认。这个密钥是后面制作主机证书时要求输入的，因此不要忘记了。

```
开始制作cacert.pem
Generating a 1024 bit RSA private key
..++++++
.............................................++++++
writing new private key to '/usr/local/bin/create_ca/CA/private/cakey.pem'
Enter PEM pass phrase:
```

```
Country Name (2 letter code) [CN]:
State or Province Name (full name) [sichuan]:
Locality Name (eg, city) [chengdu]:
Organization Name (eg, company) [txy]:
Organizational Unit Name (eg, section) [txy]:
Common Name (eg, YOUR name) [192.168.0.100]:qyjyuanjie
Email Address [tianxinyue@126.com]:
```

2） 制作主机证书时，也将首先生成主机的私钥，并提示输入封装私钥的密码，如下所示。在输入并确认密码后，即开始输入主机的信息，这些信息条目和机构证书的相同，只是有一个可选的挑战密码和公司名称，建议直接回车略过。

```
开始制作key.pem
Generating a 1024 bit RSA private key
.........++++++
..................++++++
writing new private key to '192.168.0.100/server-key.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

3） 完成主机证书信息输入后，将询问是否采用密码封装证书，建议选择是，这时将提示输入密码。该密码必须是第 6 步时输入的私钥封装密码，否则将出错。而且在配置参数时将使用该密码。

```
Enter pass phrase for 192.168.0.100/server-key.pem:
```

4） 输入主机的私钥封装密码后，将提示输入证书的私钥封装密码。该密码是在第 4 步时输入的，这里一定要输入正确。

```
开始制作cert.pem
Using configuration from ./openssl.cnf
Enter pass phrase for ./CA/private/cakey.pem:
```

5） 在正确输入后，将显示主机证书的信息，确认无误后，在下面提示中输入“y”即可完成主机证书制作。

```
Certificate is to be certified until May 20 09:53:52 2015 GMT (365 days)
Sign the certificate? [y/n]:y
```

6） 生成的证书在/usr/local/bin/create_ca/主机 IP/目录下。这里共有三个文件，他们分

别是：cacert.pem 是机构证书文件，server-cert.pem 是主机的公开证书文件，server-key.pem 是主机的私钥证书文件。现在需要将它们拷贝到运行主机上。

7） 您可以将上述三个文件放在不同的目录下。通常 server-key.pem 放在/etc/security/目录下，并且除了 root 用户外其他人都不能读取，所有人都不能修改。而另外两个文件也可以放在/etc/security/，或者放在/etc/do 目录下。单仍然要保证其只有 root 用户可读，所有用户都不可写。

上述证书制作过程中，要注意两点：

1） 如果重新为一台主机制作证书，则需要先删除其原来的证书目录，并且从文件 CA/index.txt 中删除原来的证书记录，否则将无法重新生成证书。

2） 机构私钥证书的密码不能忘记，否则将无法制作证书。同样，主机私钥证书的密码也不能忘记，否则无法配置下面的参数。

9.3.2 配置主机鉴别和加密通信参数

如果需要使用主机身份鉴别和加密通信，则需要在配置文件中进行配置。注意，数字有机体工作平台和数字有机体工作库可以使用同一套证书。当然，您也可以分别为他们制作一套证书。而且，不是两个系统必须同时启用，可以只在一个系统中启用主机鉴别和加密通信功能，而另一个不启用。

要强调的是：在启用主机身份鉴别和加密通信后，系统的性能将下降，因此这将增大主机通信处理开销和消息传输延迟。其次，如果要启用主机身份鉴别和加密通信，则系统的所有主机都必须启用，不能只有部分主机启用该项功能。因为这将使得他们之间无法正常通信。

在数字有机体工作平台的配置文件/etc/dos_exernel.cnf 中有关于 ssl 的配置，修改这些配置来启用主机身份鉴别和加密通信。默认这些配置是注释了的，即不启用。

```
ssl_ca=/etc/de/cacert.pem
ssl_cert=/etc/do/server-cert.pem
ssl_key=/etc/do/server-key.pem
ssl_key_pwd=123456
```

其中 ssl_ca 为根证书文件名，即机构证书文件名，ssl_cert 为主机公钥证书文件名，ssl_key 为主机私钥证书文件名，ssl_key_pwd 为私钥证书的密码，在制作主机证书时设置的。如果前面三项中的任意两项没有配置则不会启用主机身份鉴别和加密通信。不过，只有在所有项都正确配置的情况下才能正常运行。否则，主机将因为无法和其他节点通信而不能融入系统。

在数字有机体工作库配置文件/etc/dossql.cnf 中配置是否启用主机身份鉴别和加密通信。其配置如下：

```
use_ssl=0
ssl_ca=/etc/demo/cacert.pem
ssl_cert=/etc/demo/server-cert.pem
ssl_key=/etc/demo/server-key.pem
ssl_password=123456
```

use_ssl 为是否启用的控制变量，为 0 表示不启用，为 1 则表示启用。在启用主机身份鉴别和加密通信后，必须正确设置后面四项，否则主机将无法和其他主机通信。后面四项的设置和数字有机体工作平台相同。

9.4 系统安全加固

数字有机体系统的发行版已经对系统安全进行了加固。下面将说明这些加固设置。某些加固可能影响系统的某些功能的正常使用，请仔细阅读。

9.4.1 配置加固

这些加固措施大都已经配置在发行版内。您可以逐一检查每项配置，以确保系统的安全使用。

(1) 禁止共享内存文件系统可写，禁止执行其中的程序和进行 su 操作。

攻击一个运行中的服务(如 httpd)时经常要使用/run/shm，修改/etc/fstab 使其更安全。

sudo vi /etc/fstab 添加下面这一行内容：

```
none        /run/shm            tmpfsro,defaults,noexec,nosuid    0      0
```

(2) 禁止 root 用户通过 SSH 登录

sudo vi /etc/ssh/sshd_config 将 PermitRootLogin 设为 no。当然，如果需要通过 SSH 访问服务器，在禁用 root 使用 SSH 之前，确保其它用户可以正常使用 sudo 工作。设置后重启 sshd 服务将生效。

(3) 非 root 用户 SSH 登录配置

系统默认配置允许 dos 用户通过 SSH 登录，但新增用户是无法通过 SSH 登录的，故需要在配置文件/etc/sshusers-allowed 配置，允许该用户通过 SSH 登录。配置方法为在该文件中换行新增该用户的用户名称即可。

(4) 只允许特定用户使用 su 和 sudo 命令

这里只允许 adm 组的用户使用 su 和 sudo 命令。

```
sudo dpkg-statoverride --update --add root root 4750 /bin/su
sudo dpkg-statoverride --update --add root adm 4750 /usr/bin/sudo
```

并修改了/etc/sudoers 文件，只允许 root、管理组的成员用 sudo 命令。不再包含配置目录，并且文件设置为强制只读。如果需要可以配置每个用户具体可以执行的 sudo 后的命令。

(5) 不允许跟踪入站数据包的源路由

```
sudo sysctl -w net.ipv4.conf.all.accept_source_route=0
sudo sysctl -w net.ipv4.conf.default.accept_source_route=0
```

(6) 为 grub 启动时配置增加了密码，密码用户为 root，缺省密码是“123456”，建议在安装完成后修改它。

(7) 设置/etc/inetd.conf 只读，并且只有 root 用户可以访问，禁止了所有 inetd 服务。如果确实需要使用某项 inetd 服务，可以修改其配置。

(8) 配置 vsftpd 不能匿名登录，限制每个 IP 终端只能有 3 个连接，最大连接数为 10，

端口改为 1144。配置文件变为只读的。可以根据实际需要进行调整。如果不使用 ftp 服务，建议将其关闭，并取消自动启动。

(9) 设置 su 的行为。只有 adm 组成员可以用 su。

(10) 设置登录限制，限制口令有效期为 100 天，长度为 6 到 30，只有 adm 组用户可以 su，记录登录日志。这样，在口令有效期过去后，再次登录时系统将提示其修改口令，否则无法登录系统。

(11) 修改/etc/passwd 文件，对不需要登录的用户设置其 shell 为 false。注意：root 用户的 shell 不能修改为 false。这样防止攻击者使用非常规的账号登录系统。

(12) 修改用户和口令等文件为只读。注意：这项加固使得用户不能更改自己的口令。在服务器上，通常只有管理员登录使用，不存在其他用户。而且这样做可以防止攻击者更改用户的口令。在需要更改口令时，先将只读属性去除，即执行以下这些指令：

```
#chattr -i /etc/passwd
#chattr -i /etc/shadow
#chattr -i /etc/group
#chattr -i /etc/gshadow
```

在修改密码后，再将上述文件设为只读的，即使用”+i”选项。

(13) 默认的 init 启动级别为 5（在/etc/inittab 中设置）。移除用 rc.init 启动的不需要的服务程序，主要是 snort、bind9、winbind、tomcat6 和 apache2，在需要时用户可以用 update-rc.d 命令来恢复。kerneloops 服务不能起，它是向某个组织报告系统信息的。

(14) 移除在/etc/init 下配置的其他的服务。由于系统采用数字有机体防火墙，因此也去除了 ufw。

(15) 将/etc/hosts.equiv 文件清空，然后将其设为只读文件，防止攻击者或者其他应用利用该机制进入系统。

(16) 将/etc/services 文件设为只读的，修改宿主为 root.root。

(17) 修改/etc/security/limits.conf 文件，使得默认所有人都不产生 core，ftp 不能执行程序。

(18) 周期性执行任务（cron）的配置应该只能由 root 用户完成。即所有文件的属主和所属组都应是 root。

当然，还可以根据自身需要对其他系统配置进行加固，从而提升系统的安全性。要注意的是，您应当清楚每项加固的影响，以免限制了正常服务的提供。

9.4.2 安装前和安装过程中的安全考虑

(1) 分区设置

建议建立多个分区来存储不同用途的数据。

(1) 建立系统根分区，用于保存系统基本信息，安装在/目录下。

(2) 建立单独的 swap 分区，用于保存系统交换数据，大小等于内存大小或者两倍。

(3) 建立独立的用户主目录分区，安装在/home 目录下，用于存储各个用户的数据。

(4) 建立动态数据存储分区，安装在/var/目录下，用于存储各个应用产生的动态数据。

(5) 建立临时数据存储分区，安装在/tmp 目录下，用于存储各个应用产生的临时数据。

(6) 建立数字有机体工作库数据存储分区，安装在/usr/local/mysql/var 目录下。

(7) 建立独立的系统日志存储分区，安装在/var/log/目录下。

(8) 建立存储数字有机体文件的分区，安装在/raid 目录下。

对不同的分区，其安装时使用的参数有所不同，以便限制起始文件的行为。下面是一个典型的例子（前面的设备名可以忽略，因为各台主机的设备和分区可能不同，关注的是不同目录下文件应该有的限制）：

```
/dev/sda6   /usr   ext4  defaults,ro,nodev         0 2
/dev/sda7   /var   ext4  defaults,nodev,usrquota,grpquota   0 2
/dev/sda8   /tmp  ext4  defaults,nodev,nosuid,noexec,usrquota,grpquota      0 2
/dev/sda10 /var/log      ext4  defaults,nodev,nosuid,noexec 0 2
/dev/sda13 /home         ext4  rw,nosuid,nodev,exec,auto,nouser,async,usrquota,grpquota        0 2
/dev/sda2   /usr/local/mysql/var     ext4 defaults,rw, nodev, nosuid,noexec     0 2
/dev/sda3   /raid/          ext4  defaults,rw, nodev, nosuid,noexec          0 2
/dev/sr0     /mnt/cdrom         iso9660     ro,users,nodev,nosuid,noexec        0
```

注意：/usr 分区下存储的是安装的程序和库，在不再安装新程序和库的情况下，建议将其设置为只读模式。

当然，如果安全不是那么重要，也无需建立如此多的分区。这里只是尽量详细的列出各个目录的安全需求。

建议除交换分区外，系统的其他分区的文件系统类型选择 ext4。

(2) 准备好前不要连入互联网

系统在安装过程中不应该被连入互联网。这听起来很蠢, 因为网络安装是最常用的方法。但是，系统安装后服务马上被激活，如果系统连入互联网但服务没有被正确配置，那么您将面临着被攻击的风险。

同时应当注意到您所安装软件包中的一些服务可能存在着未被修复的安全漏洞。如果您的系统是由老版本安装的, 通常会是如此。在这种情况下，安装完成前您的系统是非常脆弱的!

(3) 安装后立即修改 root 和 dos 用户的口令

安装完成后，请立即修改 root 用户和 dos 用户的口令。建议口令长度不小于 8 个字符，最好包含字母、数字和特殊字符。不要采用易被他人猜出的口令。

(4) 运行最少服务需求

服务就是程序，如 ftp 服务和 web 服务。因为它们必需侦听连接请求，并响应请求，这样外部计算机就可以和您的计算机建立连接。服务器有时候是非常脆弱的(即可能在遭受一次攻击后瘫痪)，因此存在安全风险。

您不应该在您的服务器上安装不需要的服务。每个安装的服务都可能在您的计算机上产生新的，或许不明显(或不知道)的安全漏洞。

系统默认安装了一些网络应用服务。在完成安装后，请尽量关闭那些不用的服务。如果确定是不需要的，可以把它从系统中卸载。

(5) 禁用不需要的守护进程

禁用一个守护进程非常简单. 有以下不同的方法:

- 删除 /etc/rc${runlevel}.d/目录下的链接或将其更名(即不以 'S' 开头)。
- 更名脚本文件 (/etc/init.d/_service_name_) 为其它文件 (如 /etc/init.d/OFF._service_name_)。
- 取消 /etc/init.d/_service_name_ 文件的可执行属性。

您可以手动或者使用 update-rc.d 删除 /etc/rc${runlevel}.d/目录下的对应链接. 例如, 您可以在多用户运行模式下执行下边的命令禁用一个服务：

```
update-rc.d stop XX 2 3 4 5 .
```

(6) 禁用 inetd 服务

注意：默认没有启动 inetd 服务，并且关闭了所有通过 inetd 实现的服务。

现在，您应当检查一下是否真的需要 inetd 守护进程。inetd 一直是对内核不足的一个补偿。但是那些问题已经在最新的内核中得到了解决，可能会因为 inetd 而存在拒绝服务(它将会极大的增加机器的负载),，并且很多人喜欢直接使用守护进程而不是通过 inetd 加载。如果您仍然想使用 inetd 类的服务，请使用更加结构化的 inet 守护进程如 xinetd 或 rlinetd 或 rlinetd。发行版中使用的是 xinetd。

您应当停掉您系统中所有不必要的服务，如 echo、chargen、discard、daytime、time、talk、ntalk 和被认为极不安全的 r-services (rsh, rlogin 和 rcp，可用 ssh 替代)。

您可以通过直接编辑/etc/inetd.conf 来禁用服务, 但 Debian 提供一个更好的选择：update-inetd(当您要启用服务的时候会更方便)。您可以通过执行下边的命令来改变文件设置并重起守护进程以删除 telnet 服务(这样 telnet 就被禁用了)：

```
/usr/sbin/update-inetd --disable telnet
```

如果您想保留一项服务,但又不想让其监听您的主机的所有 IP 地址，那么您可以使用 inetd 的非归档特性 (服务名称用 service@ip 代替)或者使用其他的 inetd 守护进程如 xinetd.

(7) 限制可以通过 sshd 远程登录的用户

默认安装不提供 telnet 服务，远程登录服务只有 ssh。默认配置设定只有出现在文件 /etc/sshusers-allowed 中的用户才能通过 sshd 远程登录系统。你可以修改该文件中的用户，满足您的安全需求。

9.4.3 其他安全措施

在发行版中还安装了一些安全工具，可以利用这些工具检测系统漏洞，并抵御某些攻击。

(1) denyhosts

运行 denyhosts 抵御 sshd 等攻击。该程序的原理是：扫描登录日志文件，在发现多次登录失败时，将尝试登录者的 IP 加到/etc/hosts.deny 文件中。该文件将被各种远程访问服务使用，包括 ssh，从而拒绝远程访问。

该服务现在是自动启动的。其配置文件为/etc/denyhosts.conf。主要配置项有：

```
SECURE_LOG = /var/log/auth.log   #配置登录日志文件的路径。
HOSTS_DENY = /etc/hosts.deny   #配置拒绝服务文件的路径。
```

```
PURGE_DENY = 5d #配置拒绝解禁时间。d 表示天，为空表示永不解禁。
BLOCK_SERVICE = ALL #配置拒绝所有服务，如果只是 ssh 可以用 sshd，
DENY_THRESHOLD_INVALID = 2     #允许无效用户失败的次数
DENY_THRESHOLD_VALID = 5    #允许有效用户登录失败的次数
DENY_THRESHOLD_ROOT = 3     #允许 root 登录失败的次数
AGE_RESET_VALID=5d     #（h 表示小时，d 表示天，m 表示月，w 表示周，y 表示年）
AGE_RESET_ROOT=25d
AGE_RESET_RESTRICTED=25d
AGE_RESET_INVALID=10d
#用户的登陆失败计数会在多长时间后重置为 0
RESET_ON_SUCCESS = yes
#如果一个 ip 登陆成功后，失败的登陆计数是否重置为 0
DAEMON_SLEEP = 30s
#当以后台方式运行时，每读一次日志文件的时间间隔。
DAEMON_PURGE = 1h
#当以后台方式运行时，清除机制在 HOSTS_DENY 中终止旧条目的时间间隔,这个会影响
PURGE_DENY 的间隔。
```

还有一些难以一一介绍。可以在配置文件中看到相应的解释。

(2) 系统安全性检查：Tiger 和 chkrootkit

tiger 是一个用于检查系统安全设置是否完备的工具。它包含了 chkrootkit，能够检查各种安全设置。该软件已经安装在系统中，可以用 tiger 命令启动一次检测工作。可以每隔一段时间运行一次，以便查找系统是否存在安全漏洞。

tiger 将生成系统的审计日志到/var/log/tiger 目录下，文件名以执行日期结尾。可以直接查看该文件获得审计结果。如果对某些错误看不明白，可以用下面命令来获得具体解释。

```
tigexp –f 审计日志文件名
```

注意，系统内核被替换为数字有机体工作平台的内核，检查时可能报内核的所有文件不属于任何软件包。您可以忽略这些警告。

Chkrootkit 扫描系统中是否存在 rootkit。可以周期性运行一下。可以检查其打印出来的报告来发现入侵。如果发现入侵，可以寻找相应的工具进行处理。

9.5数字有机体文件系统加密服务

数字有机体工作平台在文件级别上为用户提供数据加密服务，满足对数据安全性要求较高的用户的需求。加密只针对普通文件而并不会对目录加密。由于平台中用户众多，并不是每个用户的每个文件都需要进行加密，故而平台使用以目录为单位的加密设置，即用户可以指定平台共享存储空间中某一目录下所有文件的加密属性。要么该目录下所有文件都加密，要么该目录下所有文件都不加密。不同需求的用户可以通过创建不同加密属性的目录来满足各自的要求。

对于加密后的文件，其数据在物理存储介质上处于加密状态，即使设备被窃取也能保证敏感数据的安全。用户只需要指定文件是否需要加密，实际的加解密过程对用户完全透

明，用户读写加密文件与读写普通文件没有任何区别。当向需要加密的文件写入数据时，平台随机选择一种对称加密算法并生成密钥进行加密，密钥则被分为多段分散地存放于数字有机体工作库中以提高安全性。目前平台使用的加密算法有 des、3des、aes 这三种，后续版本将支持更多的加密算法。

9.5.1 设置目录加密属性

建立加密目录时，通常先建立一个新目录，其加密属性继承至上级目录。然后使用 do_setdirctrypt 命令设置目录的加密属性，即设置目录下的文件是否需要加密。命令格式为：

do_setdirctrypt 目录名 [1/0]

目录名必须是已经存在的数字有机体文件系统的目录，而且必须是空目录。已经有文件或者子目录的目录不能再改变加密属性。这样限制的目的仅仅是方便管理。

第二个参数为 1 时表示要加密，为 0 则表示不加密。

9.5.2 数字有机体文件系统加密服务的规则

由于数字有机体工作平台中不同用户对安全性要求不完全相同，数字有机体文件系统中的不同目录及其下的文件可能被设置成不同的加密属性，故而需要一定的规则来协调用户的操作。下面介绍使用数字有机体文件系统加密服务时的加密规则：

1）数字有机体文件系统的根目录，即/dpfs 目录初始时是非加密的，如果需要，在根目录空时可以修改其加密属性。

2）当一个目录为空时，可以改变目录的加密属性。但是目录非空时不能修改。

3）目录下的文件继承父目录的加密属性，即文件所在目录的加密属性。新创建的文件的加密属性就是父目录的加密属性。因此，加密目录下的文件都是加密的，反之则是未加密的。

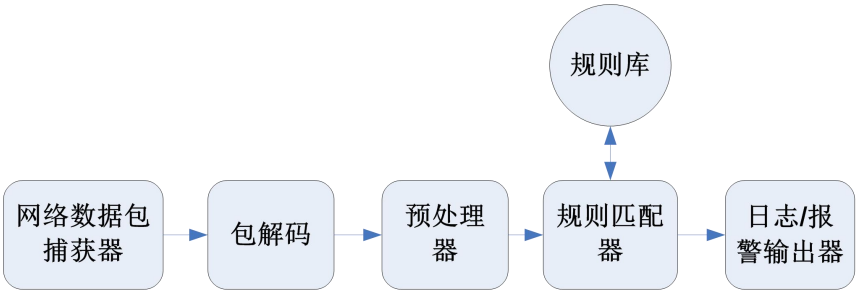
4）目录下的子目录继承父目录的加密属性，即目录所在目录的加密属性，但是目录为空时可以修改加密属性。因此，可能出现加密目录下又有非加密子目录的情况。当然，非加密目录的子目录也可以是加密目录。

5）文件的加密属性继承则父目录，而且不能更改。即无法将加密文件改为不加密的，也不能就非加密文件改为加密文件。

9.6 入侵检测系统

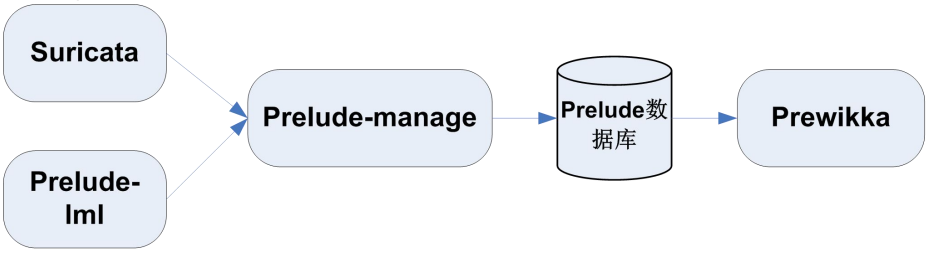
9.6.1 简介

入侵检测包括网络入侵检测（NIDS）和主机入侵检测（HIDS）。在数字有机体系统中，网络入侵检测使用 Suricata 实现。其工作原理如图所示。



Suricata 支持多种方式捕获网络数据包。数据包被捕获后，将进行解码，然后交由预处理器进行处理。Suricata 具有许多预处理器，用来完成各种不同数据包的预处理。主要的预处理是包重组、应用协议规范化、异常检测等。预处理将简化规则匹配的复杂度，从而提升规则匹配效率。

Suricata 入侵检测的原理是按照已知的入侵特征构建检测数据包的规则，通过规则匹配发现入侵现象。因此，Suricata 需要及时更新规则库，以便应对新出现的入侵手段。规则匹配器发现入侵或者需要记录日志时，通过日志（报警）输出器将日志和报警输出。Suricata 支持将日志和报警发送给 prelude-Manage。本方案采用将日志和报警发送给 prelude-manage 进行统一报警的方式。

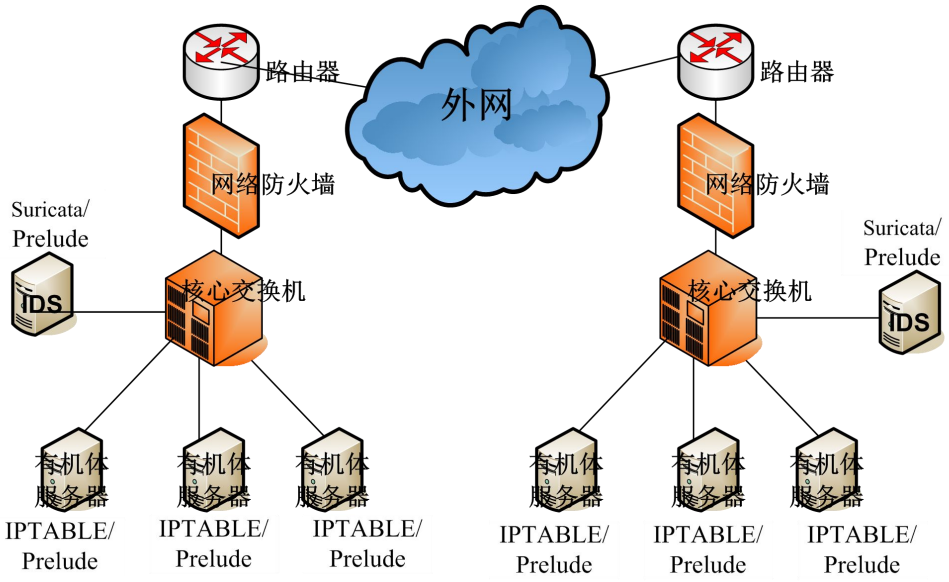


Suricata 和 Prelude-lml 等称为 sensor，即传感器，用于实际完成入侵检测工作。检测产生的报警和日志等发送给 Prelude-manage，由他统一存储到 Prelude 数据库中。然后通过 Prewikka 网站以 WEB 服务的形式显示出来。

Prelude-lml 是一个主机入侵检测器，它分析主机上的各种日志文件，发现可能的入侵现象。该入侵检测器采用规则匹配方式，能够检测内核和许多种应用产生的日志信息，从而可以较好的检测入侵状况。

9.6.2 部署方式

由于 Suricata 采用的是网卡的混杂模式来抓包，因此将抓获所有到达网卡的数据包。这样，如果同一个局域网内的多台计算机都运行 Suricata，将出现大量重复的数据包分析，从而产生重复报警。通常，网络入侵来自外部网络，因此只需在外部网络到内部网络的入口处部署检测即可。根据这个思路，利用核心交换机通常具有的端口镜像功能，可以将入侵检测服务器连接在镜像外部出口的核心交换机出口上。这也是当前网络入侵检测通常的部署方式。整个系统的部署结构如图所示。在这种方案中，由 Snort 实现的入侵检测系统（IDS）取代了通常由硬件安全设备实现的网络入侵检测系统。



在每台数字有机体服务器上部署防火墙，由数字有机体防火墙系统进行管理。再部署 Prelude-lml 实现主机入侵检测。它不仅检测系统日志，也检测防火墙的日志，并将报警统一发送给 Prelude-manage 服务。在每台服务器上部署 Prelude-manage 服务，从而各主机上的传感器只需将数据发给本机服务即可。所有主机上的 Prelude-manage 服务访问同一个数字有机体工作库，即 Prelude 库。这样就实现了系统中所有服务器的日志和报警统一存储、管理、显示和统计分析。

可以选择一些服务器部署 Prewikka 服务，以便安全管理员可以通过浏览器实时监控系统的安全状况。

9.6.3 Prelude 的安装配置

9.6.3.1 prelude 的安装

在完成数字有机体系统发行版安装后，系统中已经包含了 prelude-lml、prelude-manager、prewikka 等软件，并且安装到了相应目录，无需再进行安装。

根据 prelude 的结构，prelude-manager 是需要首先运行起来的服务，然后再启动各个传感器，即实际完成入侵检测的守护进程。

prelude 各个软件共用的配置放置在/etc/prelude/default 目录下。这里包含有客户程序共用配置文件（client.conf），全局共用文件（global.conf），标准协议客户配置文件（idmef-client.conf）和 tls 通信配置文件（tls.conf）。对当前的配置方案来说，通常无需修改这些配置文件。每个配置文件都是自解释的，因此不再介绍他们。

9.6.3.2 prelude-manager 的配置

注意：在一个系统中只需要建立一个 prelude 库。所有的服务器上的 prelude-manager 都访问该库即可。

prelude-manager 使用数据库来保存收集到的入侵日志和报警，因此需要在数字有机体工作库系统中建立一个数据库。建议数据库的名称为 prelude，当然您也可以采用其他的名

称。同时，你需要为该数据库建立一个访问用户，并向该用户授予操作 prelude 库数据需要的权限。当然，您也可以建立多个用户并使用不同的口令。每个用户分配给不同的服务器上的 prelude-manager 守护进程使用。

完成数据库建立后，您需要将数据库的表和初始数据导入相应的数据库中。数据库表结构和初始数据保存在文件/usr/share/dbconfig-common/data/prelude-manager/install/mysql 中。可能的命令如下：

cat /usr/share/dbconfig-common/data/prelude-manager/install/mysql |mysql –u 用户名 –p 口令 –h 数据库服务器地址 数据库名

如果本机是数字有机体工作库服务器，则可以不用-h 参数。

Prelude 的传感器和管理者间通过 TLS 协议通信。该协议需要一套安全证书。系统安装是自带了安全证书，但建议重新初始化 TLS 通信需要的证书信息。为此，需要先删除/etc/prelude/profile 目录下的所有文件和子目录。然后使用如下命令重新构建管理者的证书。

prelude-admin add "prelude-manager" --uid 119 --gid 127

其中 uid 后为 prelude 用户的 id，gid 为 prelude 用户的组 id，可以在/etc/passwd 文件中查到。

完成证书初始化后，需要重新启动 prelude-manager 服务才有效。

9.6.3.3 prelude-manager 配置参数

首先，要用 init 程序启动 prelude-manager 服务需要修改/etc/default/prelude-manager 文件，以允许它允许。配置文件的第二行需要进行修改，即将“RUN=”后的 no 修改为 yes，如下所示。

RUN=yes

完成上述修改后，prelude-manager 即可使用 init 程序启动了。

prelude-manager 的主要配置文件是/etc/prelude-manager/prelude-manager.conf。这个文件可根据部署情况进行配置。

prelude-manager 将收集到的入侵日志和报警等保存在数字有机体工作库中。因此，在该文件中需要进行配置的主要是工作库连接参数，即[db]组下的子项。这些子项如下所示：

type = mysql，该项配置数据库类型。数字有机体工作库的类型是 mysql。

host = localhost，该项配置数据库服务器地址。通常每台服务器上都运行了数字有机体工作库，而且他们是组织成一个系统的，因此可以用“localhost”。如果您是单独部署的数据库服务器，则填写该服务器的 IP 地址。

port = 3306，这是数据库服务端口，通常无需修改。

name = prelude，这是保存日志和报警等的数据库的名称。您需要在数字有机体工作库中先建好这个数据库，并将初始数据导入。

user = prelude，这是访问数据库使用的用户名，下面是名称。你可以根据需要在数字有机体工作库中建立用户并且授权。建议采用新的用户和口令。

pass = preludea2

该文件中的其他项目可以不进行修改。

9.6.3.4 prelude-lml 配置

prelude-lml 用于对主机日志进行分析，以便发现可能的入侵。全局的配置已经在/erc/prelude 目录下配置。针对 lml 的配置在/etc/perlude-lml/目录下。这个目录下有两个文件和一个子目录。其中 plugins.rules 文件和 ruleset 子目录用于配置检测入侵的规则。配置 prelude-lml 运行参数的文件是 prelude-lml.conf。

对于检测规则和插件处理的配置已经在安装时完成，而且规则的书写也比较复杂。如果没有特殊的需要，建议不要修改它们。

9.6.3.5 prelude-lml 运行配置

运行配置文件为/etc/perlude-lml/prelude-lml.conf 文件。该文件主要配置的内容是要检测的日志文件以及文件的编码，格式等。这些已经进行过配置，您可以直接使用默认配置。

向 prelude-manager 注册

prelude-lml 用于对主机日志进行分析，以便发现可能的入侵。它需要和 prelude-manager 使用 ssl 通信，并且需要向其注册，使其成为可信的客户。向 prelude-manager 注册主机入侵检测传感器的过程如下：

prelude-lml 需要使用命令：

prelude-admin register "prelude-lml" "idmef:w admin:r" 127.0.0.1 --uid 0 --gid 0

prelude-admin 是 prelude 系统用于管理的命令。执行上述命令的终端暂时称为注册终端。上述命令执行时将建立安全通信需要的证书等。等待其完成处理后，注册终端将出现以下提示：

You now need to start "prelude-admin" registration-server on 127.0.0.1:

example: "prelude-admin registration-server prelude-manager"

这时，需要在主机的另一个终端（暂时称为服务终端）上输入以下命令：

prelude-admin registration-server prelude-manager

注意其打印的第一行，通常内容如下：

The "v4te3e50" password will be requested by "prelude-admin register"

其中"v4te3e50"是临时生成的一个密码，需要记下这个密码，在注册终端中输入。在注册终端，将看到如下的提示：

Enter the one-shot password provided on 127.0.0.1:

这时就需要输入上述显示的临时密码。输入密码后，如果输入正确，则在服务终端将出现以下提示：

Approve registration? [y/n]:

输入 y 则同意客户注册。这时注册终端才能收到注册成功的回应，显示：

Successful registration to 127.0.0.1:5553.

这样则表示 prelude-lml 已经作为一个客户成功注册。注册完成后，在/etc/prelude/profile 目录下将出现一个新子目录，即 prelude-lml。在该子目录下存放着 prelude-lml 连接 prelude-manager 需要的证书等文件。

注意：注册成功后就不要删除这些文件，也不要尝试再次注册，否则会在数据库中看到重复的代理，使管理和显示出现麻烦。

9.6.4 prewikka 配置

在系统中已经默认安装了 prewikka 程序。这个程序提供一个 WEB 网站，以便管理员可以通过 WEB 页面监控由 prelude-manager 们收集到的日志和报警。prewikka 提供的页面是英文版的。

prewikka 也需要一个自己的数据库，建议名称为 prewikka。这个数据库用于保存网站登录的用户信息等。网站默认的登录用户是 admin，口令也是 admin。

在启动网站之前，需要先建立好 prewikka 数据库，并为其分配用户，以及给用户授权。创建数据表和初始数据的文件为/usr/share/prewikka/database/mysql.sql。在创建好数据库和授权后，您可以用如下的命令将表和数据导入数据库。

cat /usr/share/prewikka/database/mysql.sql |mysql –u 用户名 –p 口令 –h 数据库服务器 iP prewikka

注意：系统中只需要一个 prewikka 数据库。如果已经建立了，您无需再进行上述操作。

在导入数据后，您需要配置 prewikka，以便它能够访问该数据库。prewikka 的配置文件为：/etc/prewikka/prewikka.conf。您可以使用这个文件中大部分的默认配置。对数据库的配置主要是两组。一组是报警和日志存储数据库，即 prelude-manager 使用的数据库，默认名称是 prelude。这个数据库的配置组是[idmef_database]。在该组下，您需要配置如下各项：

host: localhost　数据库服务器的地址，如果本机由数字有机体数据库服务，可以不变

user: prelude　访问报警数据库的用户，可以使用 prelude-manager 相同的账号。

pass: preludea2　访问报警数据库的用户的口令

name: prelude　报警数据库的名称，需要和 prelude-manager 中的相同。

prewikka 除了访问报警数据库外，就是自己使用的数据库。他的配置组为[database]。其配置项是相似的，只是需要使用 prewikka 数据库的配置。通常如下：

```
host: localhost
user: prewikka
pass: preludea2
name: prewikka
```

完成配置后，即可使用 prewikka-httpd 命令启动 WEB 服务。当然，您也可以通过 apache 来启动网站服务。但是为了维护方便，建议不采取这种方式。

启动服务后，您可以在浏览器中输入如下的地址：http://localhost:8000 访问本机的服务。如果是在其他计算机上访问，则需要使用 http://服务器地址:8000 来访问。服务默认的端口为 8000。您可以在配置文件中修改它。初始登录的账号为 admin，口令也是 admin。建议登录后即修改用户口令，以增强系统的安全新。

prewikka 的启动需要等待数据库系统启动完成，可以不依赖其他服务。因此，除了手动启动外，要自动启动需要进行复杂的处理。

9.6.5 Suricata 的安装配置

在发布的数字有机体系统中，安装完成后已经自带了 Suricata 系统，因此无需单独安装。

9.6.5.1 Suricata 的配置

Suricata 的配置文件在/etc/suricata/目录下。安装包中已经进行了默认配置，如果没有特殊要求，可以不做修改。系统默认将日志发送给本机的 prelude-manage。

Suricata **向 prelude-manager 注册**

Suricata 向 prelude-manager 注册的过程和 prelude-lml 是类似的，不过执行的注册命令有所不同。请先检查/etc/ suricata suricata-debian.yaml 文件是否配置为向 Prelude-manager 发送报警。需要的配置如下所示：

```
- alert-prelude:
        enabled: yes
        profile: suricata
```

其中“enabled”配置是否启用向 prelude-manager 发送报警的功能，“profile”配置 prelude 客户的名称。这个名称需要向 prelude-manager 注册后才能使用。为此，您需要向 prelude-manager 注册，以生成必要的 profile 文件。命令如下：

```
prelude-admin register "suricata" "idmef:w admin:r" 127.0.0.1 --uid 0 --gid 0
```

-uid 和-gid 是运行 Suricata 的用户的 ID 及其组 ID。这里使用 root 用户来运行 Suricata。也可以创建普通用户来运行该服务。其后的注册处理过程和 prelude-lml 的相同，这里不再描述。

9.6.5.2 运行 Suricata

可以用以下命令手动启动 Suricata：

```
service suricata start
```

如果是重新启动，则可以用 restart 参数。

通常情况下，希望系统开启时自动启动 Suricata 服务。但是，Suricata 需要等待 prelude-manager 启动后才能启动。而 prelude-manager 启动需要等待数字有机体工作库系统启动好后才能启动。

9.6.5.3 Suricata 策略管理

Suricata 的策略放在/etc/suricata/rules 目录下。如果您需要修改这些策略，可以参考“snort 用户手册”中策略设计相关的内容。Suricata 使用和 snort 相同的策略文件。

最常见的操作是从网上下载最新的策略文件，更换已有的策略。可以从如下地址下载到最新的策略文件：

```
http://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz
```

下载的策略文件可以用于替换已有的策略文件。这时，需要进行以下的操作。

1）停止 Suricata 的运行：

```
/etc/init.d/suricata stop
```

2）删除原有的策略文件：

rm /etc/suricata/rules/*

3) 复制新的策略文件到策略存放目录：

tar –zxvf emerging.rules.tar.gz

cp 新策略文件存储目录/* /etc/suricata /rules/

4) 修改配置文件设置要使用那些策略

在/etc/suricata/ suricata-debian.yaml 中有使用策略文件的配置项。配置项名称为“rule-files:”。每个策略文件使用一行，例如：

- emerging-activex.rules

必须要用“-”开始，后面是策略文件的名称。

4) 重新启动 suricata 程序：

/etc/init.d/ suricata start

9.6.6 手动启动入侵检测系统

(1) 确保数字有机体工作库系统已经启动完成

您可以使用分配给 prelude-manager 的数据库账号和口令访问 prelude 数据库。在确定数据库能够成功访问后，即可确认数字有机体工作库系统已经启动完成。

(2) 启动 prelude-manager 服务

您可以通过命令：/etc/init.d/prelude-manager restart 启动服务。为了避免服务已经启 动，可采用重启方式。服务启动后，您可以在/var/log/syslog 中查看服务是否正常启动。如果没有报错，则服务正常启动了。您也可以用 pidof prelude-manager 命令查看服务的进程号。如果服务的进程号未打印出来，则表明服务没有启动。

(3) 启动 prelude-lml 服务

在确认 prelude-manager 启动成功后，您可以通过命令/etc/init.d/prelude-lml restart 命令启动 prelude-lml 服务。启动日志可以在/var/log/syslog 中看到。启动成功后，系统中将存在 prelude-lml 进程。您可以用 pidof prelude-lml 命令查看其进程号。

(4) 启动 Suricata 服务

在确认 prelude-manager 启动成功后，您可以通过 service suricata restart 命令启动 Suricata 服务。该服务的启动日志保存在/var/log/suticata/suricata-start.log 中。如果有错误，可以在这个文件中看到原因。服务启动后将驻留一个名称为 suricata 的进程。您可以使用 pidof suricata 来查看其进程号。如果进程号不存在，则服务必然已经故障了。

(5) 启动 prewikka 服务

在确认数据库系统启动成功后，即可启动 prewikka 服务。该服务的启动命令是 prewikka-httpd。启动后将出现一个“/usr/bin/python /usr/bin/prewikka-httpd”的进程。您可以用 ps 命令查看到它。同时，您可以通过浏览器访问该网站来查看启动是否正常。

9.6.7 常见问题及其解决方法

(1) 用/etc/init.d/prelude-lml restart 命令后服务没有启动

注意，该命令似乎存在一个 BUG，在某些时候使用时没有任何结果。您可以再执行一遍，以便 prelude-lml 服务能够正常启动起来。

(2) 因日志保存目录不存在而无法启动 Suricata

现象：在启动 Suricata 时，出现 ERR: logging directory /var/log/suricata does not exist 提示，Suricata 启动失败。

解决办法：可能原因是/var/log/suricata 目录没有创建或者不能访问。需要创建该目录，并且修改目录的权限为可以读写和查询。

10 数字有机体管理系统

10.1 概述

数字有机体管理系统是对数字有机体资源文件、用户信息、服务器状态进行管理的综合应用软件，共分为四大部分：

用户管理：用户管理使管理员对数字有机体工作平台的所有用户进行一体化的管理。

副本管理：副本管理能够对存储在数字有机体工作平台的所有文件的副本进行统一的管理，并且非系统管理用户也能对自己所属文件的副本进行管理。

服务管理：服务管理是对数字有机体工作平台的所有站和节点进行监控和管理。

安全管理：该部分是数字有机体防火墙配置界面，它和数字有机体防火墙的后台程序配合，实现系统内所有主机的内置防火墙的统一配置。

这里所进行的管理是在 Internet 浏览器中通过 WEB 方式进行的。下面将对各部分的操作方式和内容进行详细的说明。

数字有机体工作平台管理网站是 Tomcat 的一个应用，需要部署在 Tomcat 下，并由 Tomcat 提供服务。

10.2 配置数字有机体管理网站

数字有机体管理网站已经包含在发行版中。在安装数字有机体系统发行版后，可以在/usr/local/tomcat/webapps/目录下找到 dos_manage.war 文件。该文件是网站的压缩包，可以通过启动 Tomcat 服务的方式来部署这个网站。启动 tomcat 服务的命令如下：

```
service    tomcat6 start
```

在需要关闭服务时，可以用 stop 参数停止服务。

Tomcat 服务启动后，将自动解压网站的压缩包。您可以找到/usr/local/tomcat/webapps/dos_manage 目录。

网站的配置文件为/usr/local/tomcat/webapps/dos_manage/WEB-INF/applicationContext.xml。通常，只需配置网站的数据源即可，例如：

```
<property name="url"
value="jdbc:mysql://192.168.2.190:3306/ridi190?useUnicode=true&amp;characterEncoding=utf-8">
</property>
```

这里配置的是数字有机体工作平台的管理站的文件属性库（ridi 库）。在系统中，选择一个最稳定、最可靠、性能最高的、管理员经常使用的站作为系统的管理站。在管理站的节点上执行 do_setmainm 命令即可设置该站为管理站。通常，系统的管理网站也部署在管理站的节点上。这里设置的就是管理站的文件属性库的访问方式。

配置中的“192.168.2.190”是管理站文件属性库服务地址，如果本机是数字有机体工作库系统的成员，可以直接配置为“localhost”，否则指定为有管理站文件属性库的服务

器的地址。“ridi190”是示例中的管理站文件属性库的名称，需要根据实际的配置进行修改。文件中的其他内容请不要修改。

完成配置后，需要重新启动 tomcat 服务器才能使配置生效。重新启动命令如下：

```
service    tomcat6 start
```

在重新启动服务后，您即可在浏览器中使用“http://服务器地址:8080/dos_manage”目录访问管理网站。建议使用 IE 7 浏览器。“8080”是 tomcat 服务器通常使用的服务端口，如果您配置过 tomcat 服务的端口，请改变为配置后的端口号，否则将无法访问网站。

10.3 用户登陆

数字有机体管理系统的首页是登录界面。在登录系统后才能使用系统的各项功能。这里的用户账号是数字有机体工作平台的用户账号。只有合法的用户才具有数字有机体管理系统的使用权限，并且根据不同类型的用户，具有不同的操作权限。

数字有机体管理系统共有两种类型的用户：

系统管理员用户：该用户具有数字有机体管理系统的最高权限，它能够进行系统所有功能的操作。数字有机体工作平台的管理员账号是“DOSroot”，默认的口令是“123456”。在首次登录系统后，请修改口令。

普通用户：这类用户只能管理自己的信息和操作属于自己的资源文件。

10.3.1 系统管理员登陆

图 10-1 用户登陆界面

因为数字有机体管理系统中只有 DOSroot 一个系统管理员，因此要具有超级系统权限，只能用 DOSroot 用户登陆，如图 10- 1 所示。

输入 DOSroot 用户名和密码以后，单击“登陆”按钮，登录成功后就以系统管理员的身份登陆了数字有机体管理系统，并具备了管理系统的所有管理权限，其操作界面如图 10- 2

所示。

图 10-2 系统管理员的操作界面

10.3.2 普通用户登陆

数字有机体管理系统中存在大量的普通用户。普通用户只对自己的信息和属于自己的文件具有管理权限。登陆方式和系统管理员相同，只是使用普通用户账号和口令。

在输入用户名和正确的密码后，单击【登陆】按钮后，就以普通用户的身份登入了数字有机体管理系统。普通用户的操作界面和系统管理员的操作界面不同，如图 10- 3 所示。

图 10-3 普通用户的操作界面

10.4 用户管理

在导航栏中，单击【用户管理】页表头，这时操作界面会进入用户管理子菜单中，并且在右侧的正文中显示出组用户和用户列表，如图 10- 4 所示。

图 10-4 用户管理的界面

10.4.1 获取所有的组用户及用户

进入用户管理的菜单以后，单击【组和用户】按钮,右侧正文栏中就出显示出系统中所有组用户和用户的名称，并以层次的方式显示，如图 10- 6 所。当单击组用户图标时，会收折起该组的用户列表，如图 10- 5 所示，再次单击会展开该组的所有用户。

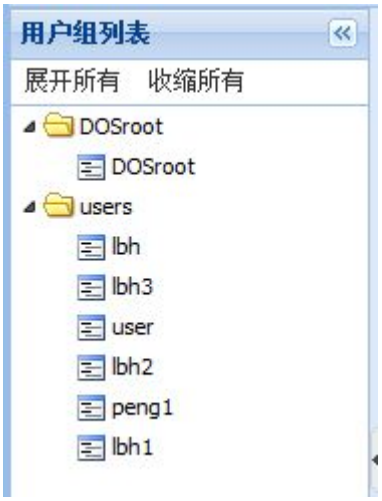


图 10-5 用户列表展开时界面

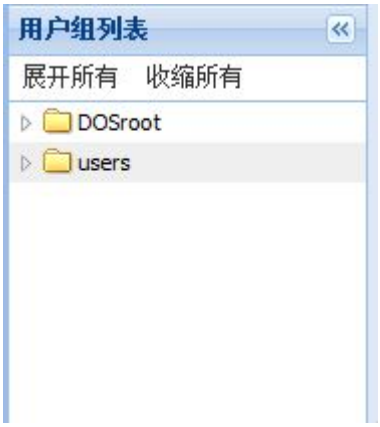


图 10-6 用户列表收折时界面

10.4.2 注册用户组

进入用户管理的菜单以后，单击【注册用户组】按钮，在右侧正文栏中弹出注册用户组的窗口，如图 10- 7 所示。

图 10-7 注册组用户的界面

在组名的输入框中输入要注册用户组的名称，在组描述输入框中输入该组的描述信息，这时系统会自动检测输入格式是否有误。若要重新填写，单击【重置】按钮，则输入框中的内容会自动置空。单击【提交】按钮，系统会注册该用户组，若系统已有该用户组，则会提示“注用户组失败”；否则，系统会提示成功注册，这里会跳至组和用户显示列表处，刚注册的用户组会显示在列表中，如上图 10- 5 所示。

10.4.3 查看用户组

打开用户组和用户列表，单击用户组名（如“DOSroot”）这时在右侧栏中显示出该组的详细信息，如图 10- 8 所示。

图 10-8 组用户详细信息

10.4.4 注销用户组

在打开的【用户组详细信息】栏中，单击【注销组】按钮，将弹出如图 10- 9 所示的提示框，单击【确定】按钮后，该用户组将从系统中删除，注销成功。

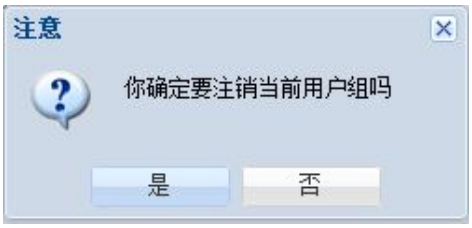


图 10-9 注销组用户的提示框

10.4.5 修改用户组

在打开的【用户组详细信息】栏中，只有用户描述信息才能修改，其它信息均不能修改，修改好组描述信息后，单击【修改组】按钮，即可将修改后的用户组信息保存至系统中，这时也会弹出修改成功的提示，如图 10- 10 所示。

图 10-10 修改组用户后的成功提示框

若是没有进行任何修改，则会弹出不能修改的提示，如图 10- 11 所示。

图 10-11 修改组用户失败的提示框

10.4.6 注册用户

单击【用户管理】菜单中的【注册用户】按钮，即可在右侧正文栏弹出注册用户的窗口，如图 10- 12 所示。

在用户信息输入框中输入用户详细信息。若是出现输入格式出错，则右侧会出现红色字体的的相应提示。输入完后，如果要重新输入，则单击【重置】按钮，所有的输入框中会重置为默认值。若要注册该用户，则单击【提交】按钮，若系统中没有该用户，则会成功注册该用户，这时直接跳至用户组和用户列表栏中，就会发现刚注册的用户已存在系统中了；若该系统中已存在该用户了，则出弹出“[用户名已存在，更换用户名！]”的提示。

图 10-12 用户注册的界面

10.4.7 获取用户信息

打开用户组和用户列表，单击要查看的用户，则在右侧正文栏中弹出用户详细信息的窗口，如下图 10- 13 所示。

图 10-13 用户详细信息

10.4.8 注销用户

在【用户详细信息】栏中，单击【注销用户】按钮，如下图 10- 14 所示。

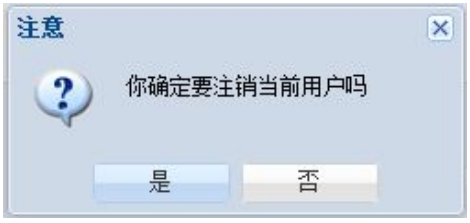


图 10-14 删除用户文件的提示框

选择【是】这时系统就会删除该用户，并弹出注销成功的提示，并且用户列表也不存在该用户了，如图 10- 15 所示。

图 10-15 注销用户成功的提示

10.4.9 修改用户信息

选择要修改的用户，在弹出的【用户详细信息】栏中，如上图 10- 13 所示，只有“真实姓名”和“用户描述”才能修改，填入修改后的值后，单击【提交】按钮，这时该用户信息将会被更新至系统中，并弹出修改成功的提示，如图 10- 16 所示。

图 10-16 用户修改成功的提示

10.4.10 查看用户配额信息

选择要查看配额的用户，在弹出的【用户详细信息】栏中，会弹出用户配额详细信息栏，如下图 10- 17 所示，这时显示出该用户的配额总数与剩余配额。

图 10-17 用户配额信息

10.4.11修改用户配额信息

在上节的用户配额信息栏中，重新输入新的配额总量，这里系统会自动计算剩余配额，完成输入后，单击【保存】按钮，这时新的用户配额被更新至系统中并弹出修改成功的提示，如下图 10- 18 所示。

图 10-18 配额修改成功的提示

10.4.12 修改用户密码

在用户详细信息栏中单击【修改密码】按钮，弹出修改用户密码的对话框，如下图 10- 19 所示。

图 10-19 用户密码修改的界面

在密码输入框中输入要“新密码”和“确认新密码”后，单击【保存】按钮，这时，该用户的登陆密码将被更新，并弹出修改成功的提示框，如图 10- 20 所示。

图 10-20 用户密码修改成功的提示

10.4.13 冻结用户

在用户详细信息栏中，若该用户处于激活状态，则在对话框的顶部则会出现【冻结用户】的按钮，这时该单击该按钮，弹出如图 10- 21 所示的提示对话框，然后，单击【是】按钮，这时该用户就被冻结了，并且在用户详细信息对话框底部的【冻结用户】按钮会转变为【恢复用户】按钮。

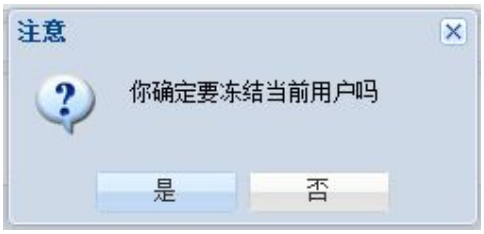


图 10-21 冻结用户的提示

10.4.14 恢复用户

在用户详细信息栏中，若该用户处于冻结状态，则在对话框的顶部则会出现【恢复用户】按钮，单击该按钮，弹出如图 10- 22 所示的提示对话框，然后，单击【是】按钮，这时该用户就会被激活，并且在用户详细信息对话框顶部的【恢复用户】按钮会转变为【冻结用户】按钮。

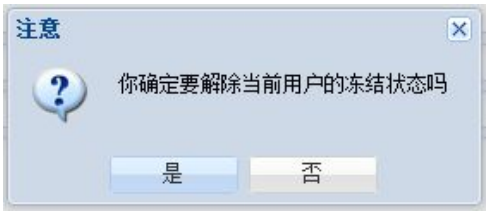


图 10-22 恢复用户的提示

10.5 副本管理

在导航栏中，单击【副本管理】按钮,这时操作界面就会进入副本管理子菜单中，如图 10- 23 所示，并且在右侧的正文中显示系统中所有的文件信息。

图 10-23 副本管理菜单

10.5.1 获取数字有机体操作系统根目录下的文件

单击【副本管理】子菜单中的【系统文件】按钮，则会自动列出开数字有机体操作系统根目录（/dpfs）下的所有文件和目录，如图 10- 24 所示。

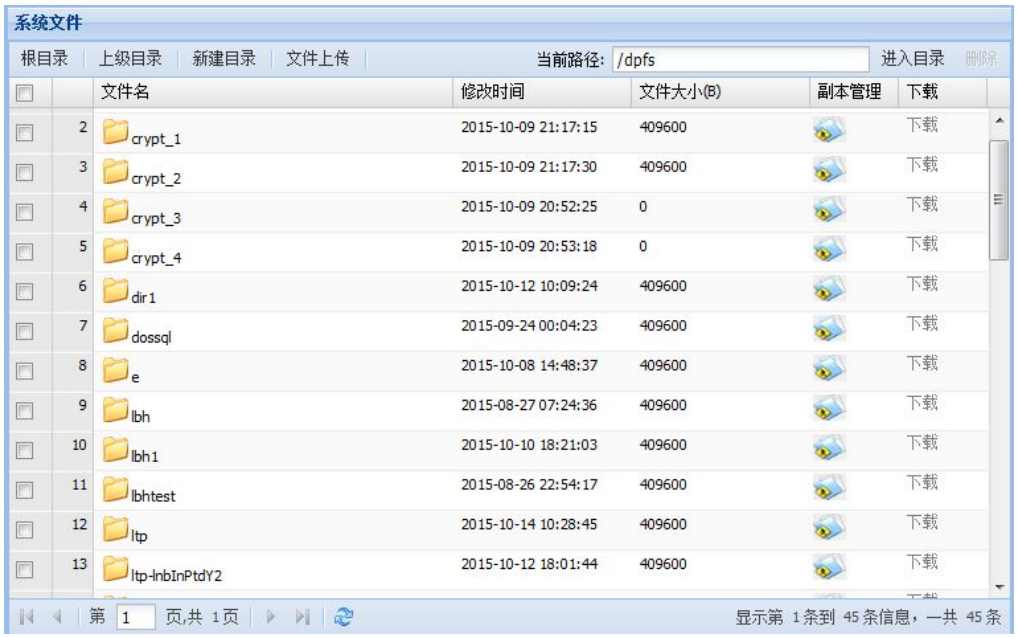


图 10-24 /dpfs 目录下的所有文件与目录

如果文件列表中的文件是目录文件，则会以  图标的方式显示，这时该目录名也是按钮，当单击时，就会显示该目录文件下的文件。如果是普通文件，则会以  图标的方式显示。

单击【根目录】按钮回到“/dpfs”目录；【上级目录】则是回到当前目录的上级目录。

10.5.2 新建目录

在【系统文件】页面中如上图 10- 24 所示，单击【新建目录】按钮，弹出如图 10- 25 所示的对话框，然后输入所需文件名，单击【确定】按钮，这时应该在当前目录下建立一个新文件夹。

图 10-25 新建文件夹

10.5.3 上传文件

在【系统文件】页面中如上图 10- 24 所示，单击【文件上传】按钮，弹出如图 10- 26 所示的对话框，单击【选择文件】按钮选择要上传的文件，支持批量上传，单击【上传】按钮，所选择的文件会自动上传。

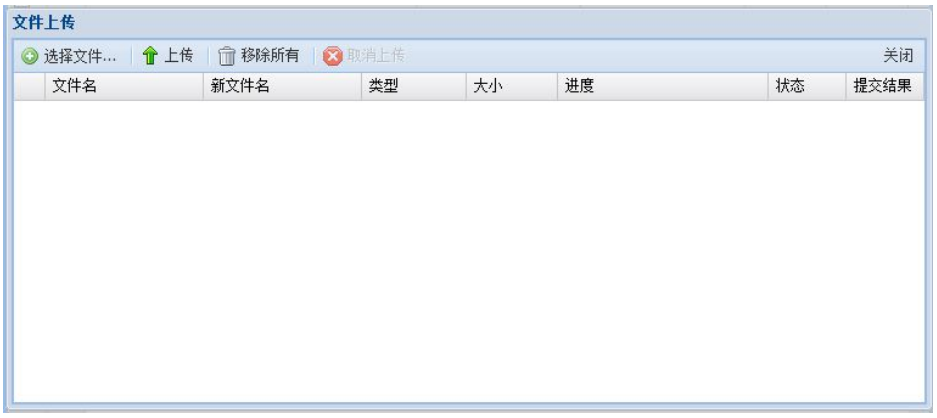


图 10-26 文件上传

需要注意的是浏览器的 Flash Player 版本不能低于版本 10，否则会出现如下图 10- 27 错误。

图 10-27 错误提示

10.5.4 下载文件

在【系统文件】页面中如上图 10-24 所示，单击文件的【下载】按钮，将文件下载到本地磁盘，文件夹不能下载。

10.5.5 删除文件

在【系统文件】页面中如上图 10- 24 所示，单击【删除】按钮删除系统的文件或目录。在删除文件时，如果要选中所有的文件，则可单击顶部【 】按钮；当然也可在多选框中选中您要删除的文件；这时单击【删除】按钮，您选中的文件将从系统中删除，如下图 10- 28 所示。

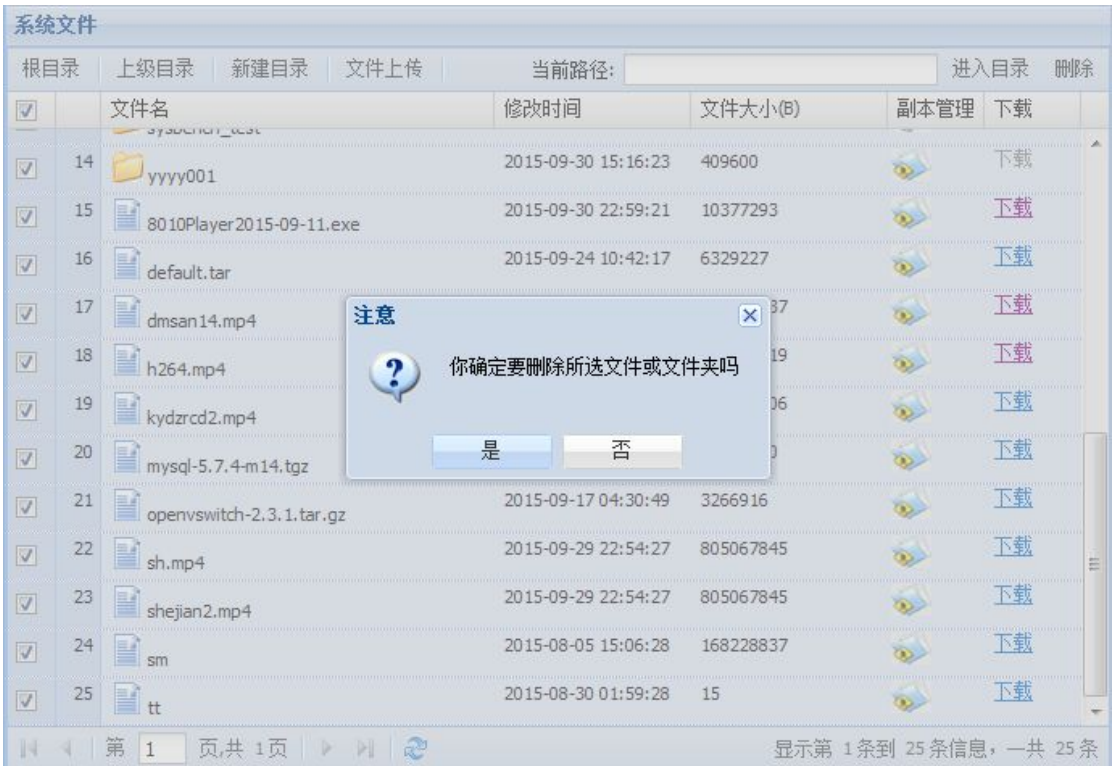


图 10-28 删除所选文件

10.5.6 获取文件副本的分布

获取文件的副本分布信息的方法有三种：

1）单击【副本管理】子菜单中的【副本管理】按钮，则显示出副本管理的对话框，如下图 10- 29 所示。

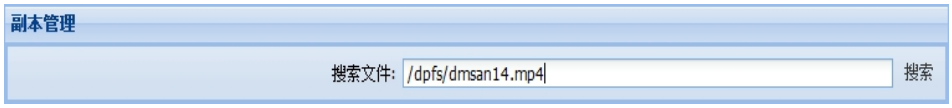


图 10-29 查询文件副本的界面

输入文件的路径名后，单击【提交】按钮，则会显示该文件的副本分布信息。

2）在【系统文件】列表图中单击文件名后的【】副本管理按钮，则会显示出该文件的副本分布信息。

3）在【系统文件查询】列表图中单击文件名后的【】副本管理按钮，则会显示出该文件的副本分布信息。

若文件为目录文件，目录文件的副本分布信息如下图 10- 30 所示，因为目录文件的副本分布情况不能随意更改，所以不显示目录副本的修改栏；

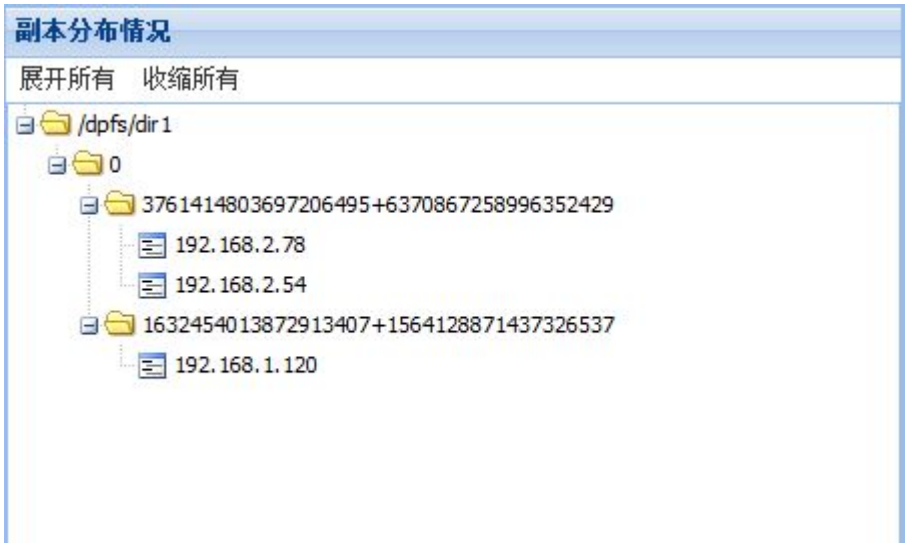


图 10-30 目录文件的副本分布图

若文件为普通文件，则文件的副本分布情况如下图 10- 31 所示，因为普通文件的副本分布是可以手动修改的，所以有副本修改栏。

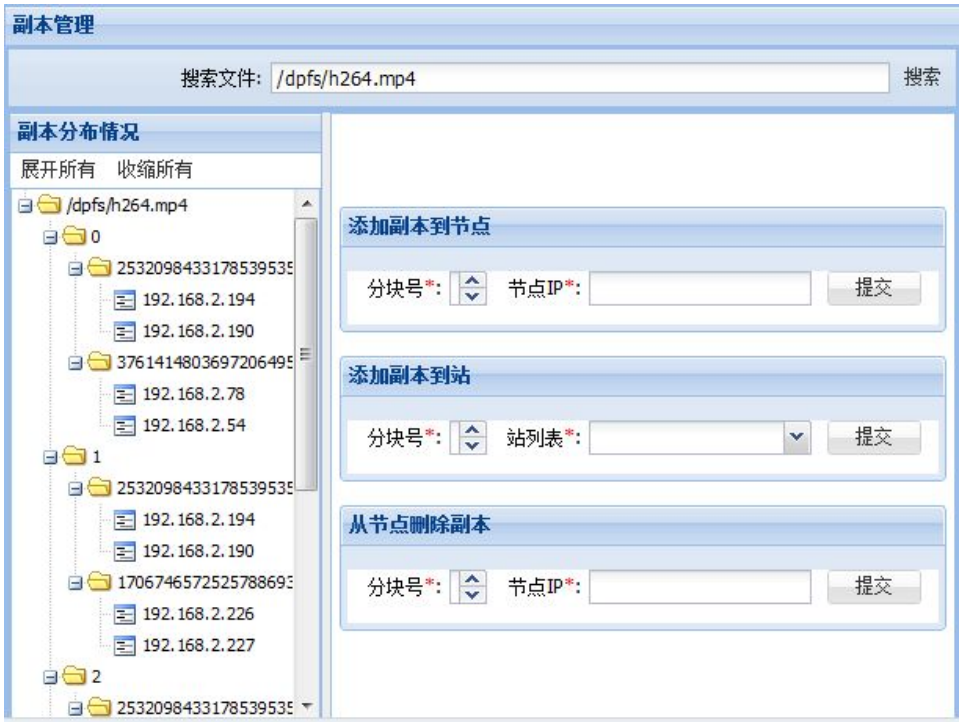


图 10-31 普通文件的副本分布图

10.5.7 添加副本到节点

在文件副本分布详细栏中，定位到【添加副本到节点】输入框中，如下图 10- 32 所示。

图 10-32 添加副本到节点的界面

在分块号输入框中，输入要添加的块号；在节点 IP 输入框中，输入您要添加该文件副本的节点 IP，前提是该节点并不存在该文件块的副本，然后单击【确定】按钮，该文件副本就添加到目标 IP 上了，并在副本分布情况栏中显示出，这时可能会出现，添加成功，但没有显示出来，这是因为文件过大时，会出现延时的现象，过段时间后，单击【  】刷新按钮，这时在新节点上的副本才会显示出来。

10.5.8 从节点删除副本

在文件副本分布详细框中，定位到【从节点删除副本】输入框中，如下图 10- 33 所示。

图 10-33 从节点删除副本的界面

输入要删除该文件副本的节点 IP 和块号，前提是该节点上必需存在该文件块副本，然后单击【提交】按钮，这时该节点上的副本文件被删除了。

10.5.9 添加副本到站

在文件副本分布详细框中，定位到【添加副本到站】输入框中，如图 10- 34 下所示。

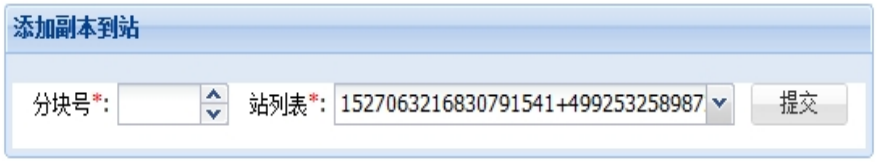


图 10-34 添加副本到站的界面

选择要添加到的块号和站 ID，然后单击【提交】按钮，该文件块副本会被添加到目标站上。同样可能会出现添加成功，但没有显出来，这是因为文件块过大时，会出现延时的现象，过段时间后，单击【  】刷新按钮，这新站上的文件副本就会显示出来。

10.6 服务管理

在导航栏中，单击【服务管理】按钮，就进入了服务管理子菜单中了，如下图 10- 35 所示。这里包括了报警信息检测、所有站负载信息、单站负载信息、报警地图配置、站点节点配置和虚拟服务管理，并在右侧栏中显示出系统的报警信息。

图 10-35 服务管理菜单

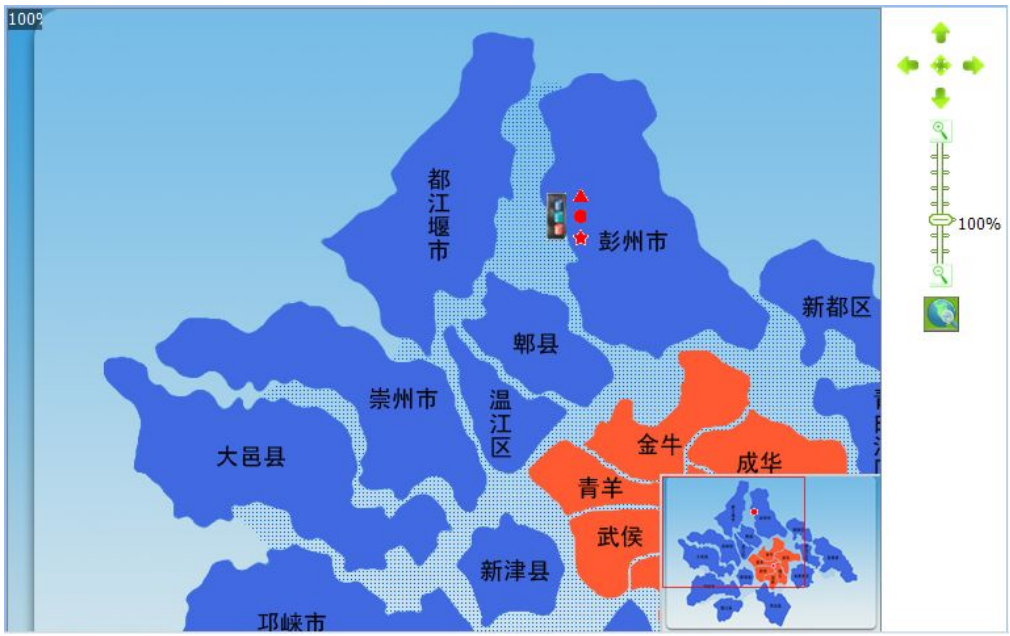


图 10-36 报警信息显示图

10.6.1 报警信息检测

在【服务管理】子菜单中，单击【报警信息检测】按钮，即可显示出数字有机体操作系统的报警示意图，如图 10- 36 下所示。

图 10- 36 中的【▲】表示是该站的活动状态，若是绿色，则是表示该站所有节点正常；若为红色，则表示有节点处于异常。这时单击该按钮，会显示出该站的所有节点状态，如下图 10- 37 所示，在这里，可以看到节点的详细状态。

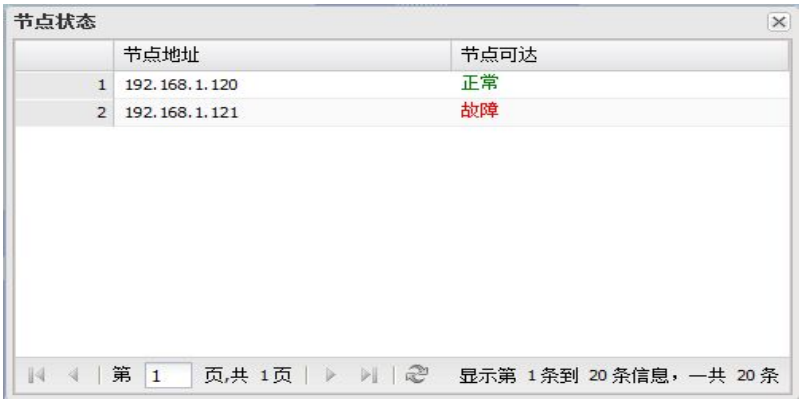


图 10-37 站内节点状态图

图 10- 36 中的【●】表示该站的负载报警状态，若为绿色，则表示没有负载过重；若为红色，则表示该站负载过重。单击该按钮，则显出该站的负载报警信息，如下图 10- 38 所示。其中红色表示该项负载过重，已超出了最高限制。其中，【删除】按钮表示删除该条负载报警信息；【处理】按钮表示处理该条负载报警信息；【已处理记录】按钮可显示出本站内所有已处理的报警信息；单击【未处理记录】按钮可显示出本站内所有未处理的报警信息。

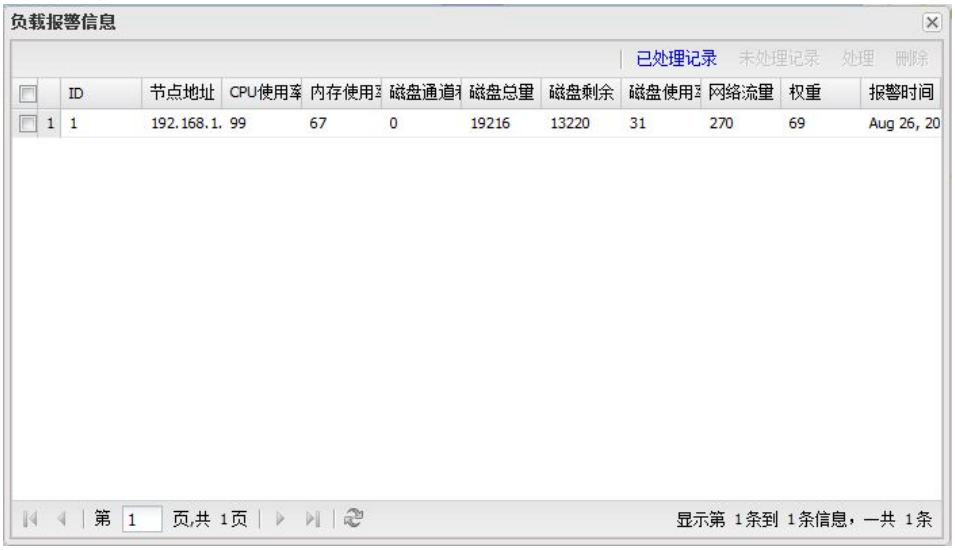


图 10-38 负载报警详细信息图

图 10- 36 中的【★】按钮表示该站内的服务报警，用于检测该站内被监控的服务是否正常。若为绿色则表示该站内的服务一切正常，若为红色则表示，该站有服务出现异常。单击该按钮，则显示出该站的所有服务报警信息，如下图 10- 39 所示。其中【删除】按钮表示删除该条服务报警信息；【处理】按钮表示处理该条服务报警信息；【已处理记录】按钮表示显示出本站内所有已处理的服务报警记录；【未处理记录】按钮表示处理本站内所有未处理的服务报警记录。

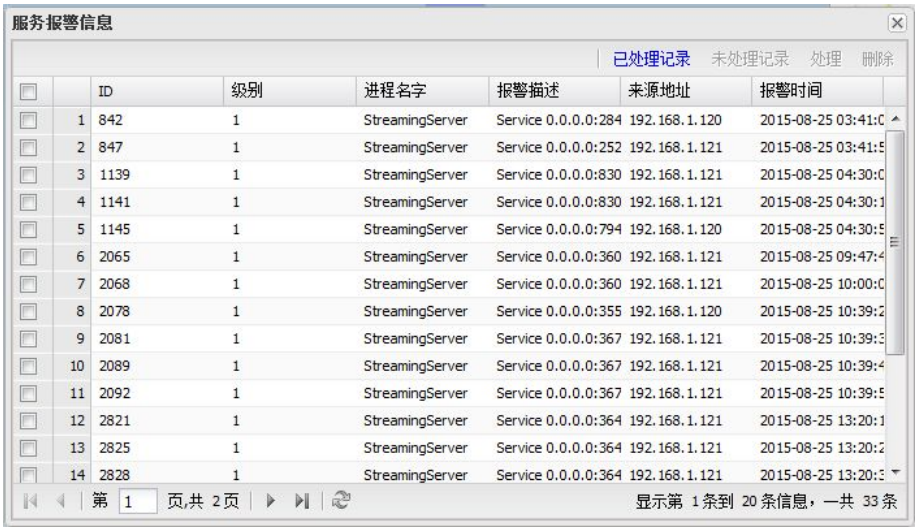


图 10-39 服务报警详细信息图

10.6.2 所有站负载信息

在服务管理子菜单中，单击【所有站负载信息】按钮，即可查看到系统中所有站的详细负载信息，如下图 10- 40 所示。

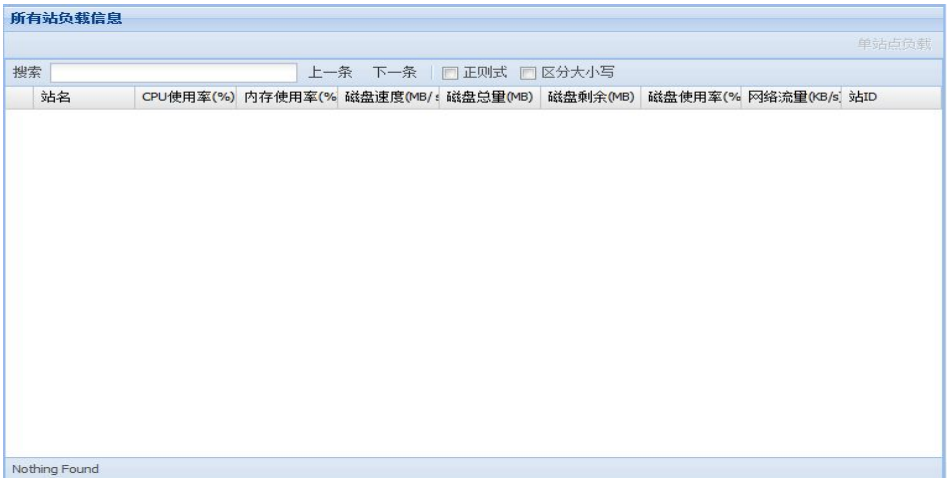
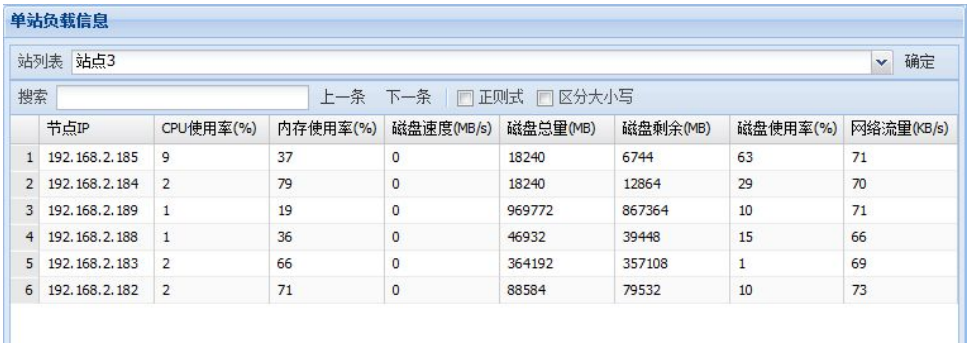


图 10-40 系统所有站的负载信息

其中包括站名称、站 ID、每站的节点数、CPU 使用率、内存使用率、磁盘速度、磁盘总量、磁盘剩余、磁盘使用率和网络流量相关数据，其中红色显示的数据表示该项指标已超标了，即负载过重。

10.6.3 单站负载信息

在【服务管理】子菜单中，单击【单站负载信息】按钮，即可查询到目标站的负载信息，如下图 10- 41 所示。同样，单站的负载包括 CPU 使用率、内存使用率、磁盘速度、磁盘总量、磁盘剩余、磁盘使用率和网络流量相关数据，其中红色显示的数据表示该项指标已超标了，即负载过重。



	节点IP	CPU使用率(%)	内存使用率(%)	磁盘速度(MB/s)	磁盘总量(MB)	磁盘剩余(MB)	磁盘使用率(%)	网络流量(KB/s)
1	192.168.2.185	9	37	0	18240	6744	63	71
2	192.168.2.184	2	79	0	18240	12864	29	70
3	192.168.2.189	1	19	0	969772	867364	10	71
4	192.168.2.188	1	36	0	46932	39448	15	66
5	192.168.2.183	2	66	0	364192	357108	1	69
6	192.168.2.182	2	71	0	88584	79532	10	73

图 10-41 节点负载的详细信息

10.6.4 报警地图配置

在【服务管理】子菜单中，单击【报警地图配置】按钮，即可弹出报警地图配置的窗口，如下图 10- 42 所示。

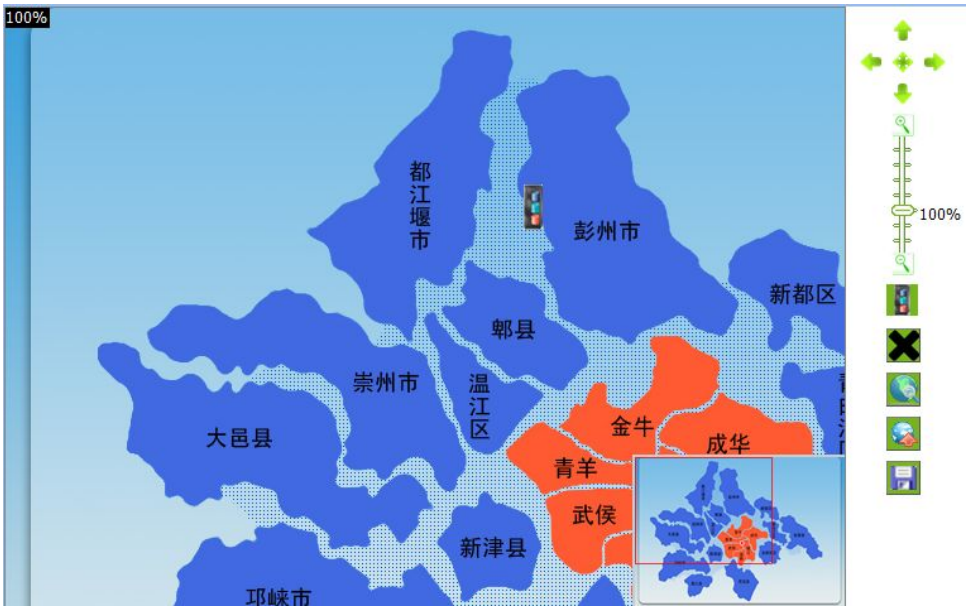


图 10-42 报警地图配置窗口

在图 10- 42 中的【⬆】【⬇】【⬅】【➡】表示地图的移动方向，【✥】表示使地图居中显示，【🔍】【🔍】表示地图的放大和缩小，【🚦】表示添加数字有机体站，【✖】表示清除数字有机体站，【🌐】表示显示或隐藏鸟瞰图，【🖼】表示上传地图，【💾】表示保存地图信息。

【🚦】表示添加数字有机体站，先点击此按钮，然后在地图上单击左键确定有机体站的位置并弹出有机体站的选择窗口，如图 10- 43。地图上的有机体站可以通过拖动来移动它的位置，并且可以双击弹出有机体站的选择窗口来改变对应的有机体站。

图 10-43 有机体站选择窗口

【】表示上传地图，点击此按钮会弹出地图上传窗口，如图 10- 44。

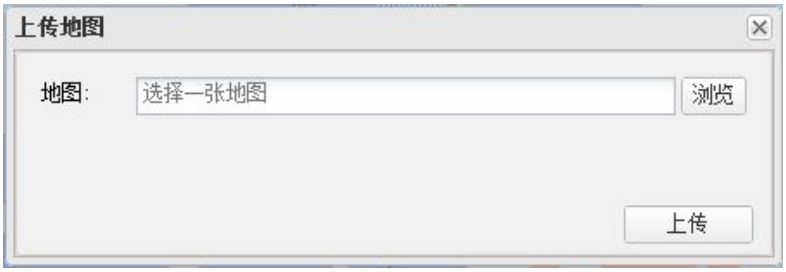


图 10-44 地图上传窗口

10.6.5 站点节点配置

在【服务管理】子菜单中，单击【站点节点配置】按钮，即可显示出数字有机体操作系统的站点配置窗口，如下图 10- 45 所示。

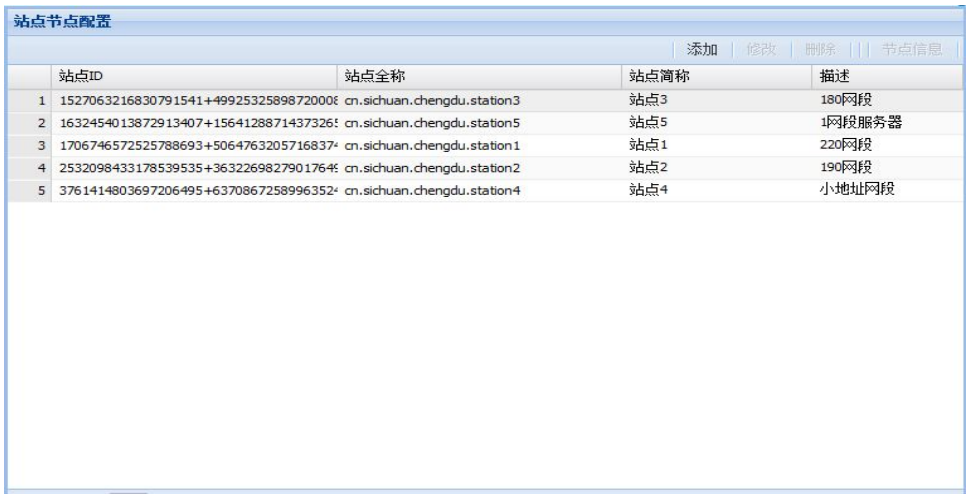


图 10-45 站点节点配置窗口

点击图 10- 45 中的【添加】按钮会弹出站点新增窗口，如图 10- 46 所示。有红色*标记的为必填项，其中的站点 ID 为不可修改项。

图 10-46 站点新增窗口

点击图 10- 45 中的【修改】按钮会弹出站点修改窗口，与图 10- 46 一样，但是其中的站点 ID 不可编辑。

图 10- 45 中的【节点信息】会弹出该站点下的节点信息窗口，如图 10- 47。在此窗口中可以做节点的增、删、改操作，同站点配置一样，节点名称也不可修改，所以在添加输入时要留心一点。

图 10-47 节点信息窗口

10.6.6 虚拟服务管理

本部分是网络虚拟服务的 WEB 配置界面，在第 8 章章已经详细介绍，这里不再描述。

10.7 安全管理

本部分是数字有机体防火墙系统的统一配置 UI 界面，在 9.2 节“数字有机体防火墙”中已经详细介绍过，这里不再描述。

10.8 出错处理

如果数字有机体管理系统运行中出现了“系统出错，请检查数字有机体操作系统”的提示，这时请与数字有机体操作系统管理与维护人员联系，等数字有机体操作系统完成启动以后，再次运行数子有机体管理系统。

如果数字有机管理系统在登陆时总是登陆不成功，这时说明数字有机体操作系统启动未完成，这时要等待数字有机体操作系统启动完成后，才能登陆数字有机体操作系统。

11 日常维护

11.1 系统运行状况监控（是否有主机死亡）

对系统的监控可通过管理网站的服务管理（10.6 章节）进行查看。通过管理网站可以查看系统中各节点的网络情况，各节点的负载情况以及各站的服务情况。

11.1.1 各节点网络状态

该状态由报警信息检测（10.6.1 章节）地图上的三角形表示。若三角形为绿色，表示正常状态；若为红色，表示存在异常。点击红色三角形可查看各个节点的可达情况，从而找出网络连接错误的节点。

11.1.2 各节点的负载情况

该状态有报警信息检测（10.6.1 章节）地图上的圆形表示。若圆形为绿色，表示正常状态；若为红色，表示存在异常。点击红色圆形可查看各个节点的 CPU、内存、磁盘的使用情况，同时也能查看到该报警出现的时间。

11.1.3 各站的服务情况

该状态有报警信息检测（10.6.1 章节）地图上的五角星表示。若五角星为绿色，表示正常状态；若为红色，表示存在异常，点击红色五角星可查看该站各服务的进程名，报警级别，报警时间，报警来源等信息，从而找出异常服务出错的原因。

11.2 数据备份与恢复

在本系统中，为了保证数据的安全性，每个文件都有相应的备份文件（存储于其他节点上的副本文件），所以不必对 dpfs 下的文件进行备份处理。如果有特别需要，可以通过 cp 命令把 dpfs 下的文件拷贝到本机的非全局目录下保存或刻录到光盘保存。

如果因大量节点故障而使的某些文件丢失，可以通过 cp 命令将备份的文件拷贝到系统中。操作可以在任意节点上完成。

11.3 在线扩充服务器

11.3.1 概述

数字有机体具有良好的在线扩充能力，以便能以最经济的方式部署应用系统。最小的数字有机体工作平台可以只有一两台服务器，然后随着应用需求的不断增长，数字有机体工作平台可以增长到具有成千上万台服务器的系统。因此，系统的初期投入很低，也无需

为未来发展预留设备，即不存在闲置设备的问题。某些系统虽然也具有较好的扩展能力，但往往要求预先估计系统的最大规模，从而可以为某些集中设备预留足够的能力。

数字有机体工作平台不仅是能扩展系统，更支持在线的扩充系统。当发现现有部署系统的能力无法满足应用的需求时，可以通过三个层面提升系统的能力：

1）提升系统中每台服务器的能力，例如给服务器增加内存、磁盘空间、处理器或者其他硬件设备等。由于系统由大量服务器构成，且系统具有良好的容灾能力，因此升级某台服务器完全无需暂停业务运行，即可实现完全的在线升级。

2）提升某个站的服务能力。如果发现某地的数字有机体站无法满足当地应用的需求，可以提升该站的服务能力以更好的满足应用需求。提升单个站的服务能力除了提升站中已有服务器的能力外，也可以增加新的服务器。新的服务器只需按照前述的安装配置介绍进行配置后，即可启动服务器加入已有的站，从而提升单个站的服务能力。同样，增加服务器也无需暂停正在运行的业务。

3）提升系统规模。随着业务服务范围的增长，现有少数几个地点提供的服务将无法更好的就近为异地网络的客户提供服务，这时就需要增加新的数字有机体站。新增加的数字有机体站可以分流计算、存储和网络负载等，尤其是就近分流网络负载可以提供更好的服务体验。同样，增加数字有机体站也无需暂停现有的业务。

11.3.2 提升某台服务器的能力

如果需要提升某台服务器的能力，则需要关闭该台服务器以便扩展服务器的设备。常见的系统都是由单台服务器提供服务的，因此关闭服务器就要暂停服务。数字有机体工作平台由多台服务器构成，因此可以在不暂停服务的情况下关闭某台服务器来升级。关闭服务器的方式很多，最好的是用 root 用户登录要关闭的服务器，然后运行以下命令关闭服务器。

halt 或者 shutdown –t time

halt 命令将立即关闭系统，而 shutdown 命令的-g 参数可以指定延迟关闭的秒数，并提醒其他已经登录系统的用户系统将要关闭。

关闭服务器后，即可对服务器进行升级，包括扩充服务器的内存、增加 CPU 等。升级完成后，启动服务器即可使服务器重新回到系统中，继续向用户提供服务。不过，如果涉及到服务器硬盘存储容量的扩充，需要按照后面“扩充服务器存储容量”的描述进行一些配置操作。

11.3.3 为已有站增加服务器

如果要给一个已有的站增加新的服务器，则可以按照以下步骤进行。

1）按照《数字有机体系统安装指南》中的步骤对新服务器进行安装配置。注意服务器的名称需要和系统中现有所有服务器的名称都不同。

配置时最简单的办法是用 config_dos 程序进行配置，选择为已有站增加新服务器即可。也可以手动配置，例如从站内已有服务器上拷贝 dos_exernel.cnf 和 dossql.cnf 两个配置文件。从其他服务器上拷贝文件可以使用 scp 命令（在 ssh 服务开启的情况下）

scp 用户名@已有服务器的 IP 地址:/etc/dos_exernel.cnf /etc/dos_exernel.cnf

系统将提示输入用户的口令，正确输入口令后即可将文件拷贝到本地。获得 dossql.cnf 也可以采用同样的方式。

如果数字有机体内部通信都是使用的默认配置（即第一块网卡）因此没有在配置文件中指明主机地址，甚至无需修改这两个配置文件即可使用。如果在配置文件中配置了主机地址，请将其修改为新服务器的内部通信地址。在 dos_exernel.cnf 中主机内部通信地址的配置项为 host_addr，在 dossql.cnf 中为 local_ip。

11.3.4 建立新的数字有机体站

如果需要给系统增加新的数字有机体站，请先规划好新数字有机体站中各台服务器的 IP 地址，并指定新站的描述名（需要和现有站不同）和数字有机体工作库的站 ID（需要和已有的站都不相同）。

这里首先安装配置并启动后新站的第一台服务器，其他的服务器则只需按照“为已有站增加服务器”的方式进行操作即可。

新服务器上数字有机体发行版的安装如前“数字有机体发行版安装”章节所述。完成发行版安装后，即可配置服务器以作为一个新的站加入系统。简单的方法是使用 config_dos 程序进行配置，下面说明配置中的一些注意项。

1）配置新服务器的数字有机体工作库及大规模存储管理系统

新服务器作为一个新的站加入系统，因此需要设置一个新的站 ID，即配置 dossql.cnf 中的 station_id 项。如果新的站和其他站都不在一个网段内，可以使用和其他站相同的站内广播端口。如果要在同一个网段内部署两个及以上的站，则每个站都要具有不同的站内广播端口，即需要配置 dossql.cnf 中的 station_cast_port 项。新站点加入已有的系统需要配置一个系统内正常运行的服务器的地址，即配置 dossql.cnf 文件中的 neighbor_ip 项目。注意系统内所有服务器的 manager_port 都必须是同一个。

其他的项目根据实际的情况进行配置。例如要采用主机鉴别和安全通信则要为新服务器制作证书并进行配置。

如果站点间部署有防火墙，请确保在防火墙上进行了适当的配置，以保证各台服务器间可以顺畅的通信。通常的办法是在防火墙上设置信任主机组，这些主机间的通信不受任何限制。如果觉得不够安全，可以配置这些主机间开放的协议和端口。数字有机体工作平台的站间通信都采用 tcp 协议。使用的端口包括：

manager_port 及其后的 4 个端口，min_tcp_port 和 max_tcp_port 间的端口（含上下边界），数据库服务端口 3306。

2）配置服务器的数字有机体工作平台及抗毁容灾系统

工作平台的配置文件为 dos_exernel.cnf。新的站点需要一个新的站描述名，即 station_locality_desc 配置项，同时需要清除 station_id 配置项的设置，以便由系统重新自动生成新站点的 ID。如果在一个网段来部署多个站点，则需要保证 group_port 各不相同。net_port 则需要和系统中也有的站点相同，并且需要配置一个已经在系统中的节点作为邻

居站点，以便新服务器可以通过该节点加入系统。

新的站点需要一个新的元数据存储库，因此需要在启动好数字有机体工作库后创建一个新的数据库。建议将该数据库设置为只在本站内放置。

如果新的数字有机体站的文件需要按照某种规则方式，请参见“副本放置规则”章节的描述进行配置。

11.4 给服务器扩充存储设备

给服务器增加存储设备除了需要关闭服务器更换和增加存储设备外，有以下几点需要注意。

更换存储设备或者扩充存储设备都可能使原有存储设备上的数据丢失，因此需要确认原有存储设备上的数据在系统中是否都有副本。如果服务器是某个站的唯一节点，或者最少副本允许为 1，或者某些文件刚加入系统，都可能存在某些文件没有副本在其他节点上的情况。因此，可能造成数据丢失。可靠的办法是先备份存储设备上的数据到其他节点，然后再进行存储扩充或者替换。

每台服务器只能输出一个共享目录。因此，如果要输出的存储空间由多个物理设备构成，例如过个磁盘，或者磁盘和外部存储设备等，就需要将这些存储设备整合为一个逻辑存储空间。常用的方式是 LVM（即逻辑卷）和软 raid 技术。相应配置请参考 Debian 7 系统配置手册。虚拟设备再通过 gparted 等分区后即可挂载在输出目录下（默认为/raid/data）。

在扩充存储容量后，可以将旧设备上原有的数据还原到新设备上，启动数字有机体工作平台后将自动融入现有系统。如果旧设备上的文件都有副本，则可以不恢复旧设备上的数据。系统将自动安排文件副本的放置。

扩充存储容量后，系统可能重新调整副本的放置位置，以更好的均衡各台服务器上的存储空间和负载。这个动作将产生明显的额外负载，因此可能降低系统的服务能力。不过这个过程是缓慢的，以尽量减少影响。

11.5 节点关闭和重启

可以使用 shutdown –y –g0 关机，也可以在图形界面下依次选择 N->Logout->Menu->Shutdown，然后输入 root 用户密码，即可关闭。

节点重新启动自动加载数字有机体工作平台，并按照配置文件的配置加入数字有机体工作平台。如果需要修改已有配置，可以在启动后执行/usr/local/bin/rg stop 停止数字有机体软件的加载。

如果因为硬件或者是软/raid 或者是文件系统的原因，造成系统无法正常启动而进入单用户模式（机器名为 none），可以采用以下办法来解决：

如果是硬件问题：一般是磁盘阵列的故障造成文件系统的损坏，需要首先解决磁盘阵列的问题，正常启动后重新部署数字有机体的输出目录即可。

如果是软/raid 的原因，可以先把/etc/fstab 中软/raid 的挂载信息注释掉（用#注释）：

#/dev/md0 /raid reiserfs acl,user_xattr 1 2

然后重启，打开注释，重新挂载软/raid 的挂载目录：mount /dev/md0 /raid。

如果是文件系统的原因，根据系统给出的出错日志，用 fsck.reiserfs 对出错分区进行检查，如果需要重建文件系统，可以加上--rebuild-tree 选项。

如果不是上面的原因造成系统无法正常启动，需要根据具体的问题具体操作。

12 常见问题解决

12.1 常见数字有机体工作平台问题

12.1.1 添加共享空间失败

问题描述：

执行 do_shareadd /mnt/disk1/ /dpfs/movie/命令返回 addshare error！(说明添加失败)。

可能原因：

a) 系统外核 dos_exernel 程序没有启动或正在启动

b) 外核虽然已启动但没有加载内核模块

c) 本节点共享空间在此之前已经添加成功

处理方法：

执行 ps –ax 命令，查看是否有./dos_exernel –d 进程；如果有该进程，则继续执行 lsmod 查看显示结果中是否有 dpfs 模块，如果有则表明内核模块已被加载。没有 dpfs 模块则需要执行命令:

cd /lib/modules/2.6.5-7.244-dpfs/kernel/fs/dpfs

insmod dpfs.o

#mount –t dpfs none /dpfs 加载内核模块。

此时可以执行命令：# tail –f /tmp/dos_exernel_stdout 查看外核启动进度，如果外

核已正常启动成功，则可重新执行添加共享命令。如果还是添加失败，则可打开文件 /etc/export_info 查看是否已有共享空间被添加，如看到类似记录：/raid/data/ /.dpfs/server162 server162（该条记录的第一项是输出空间的目录位置，第二项是本站的其他节点需要加载本节点输出目录的 mount 目录，第三项是本节点的节点名）则表明已有共享空间被添加。可以不用添加共享空间了，如想换调该共享空间，则可先执行 do_sharedel /raid/data/ 命令，然后再执行添加共享命令。

12.1.2 删除共享空间失败

问题描述：

执行 do_sharedel /mnt/disk1/命令返回 delshare error!(说明删除失败)。

可能原因：

a) 系统外核 dos_exernel 程序没有启动或正在启动

b) 外核虽然已启动但没有加载内核模块

c) 本节点没有添加共享空间

处理方法：

类似于添加共享失败的检查方案。

12.1.3 数字有机体工作平台管理系统的服务器信息收集总是失败

问题描述：

数字有机体工作平台启动一切正常，但管理系统的服务器信息收集功能模块所有功能执行都是失败，即不能收集到节点或站以及系统的负载信息。

可能原因：

a) 系统负载信息收集定时器没有启动，即 collect_timer_value=0

b) 系统负载信息收集定时器虽然启动，但定时器时间值设置过大，如：collect_timer_value=10000

处理方法：

使用 ssh 或 telnet 远程登陆命令，登陆到对应节点上打开文件/etc/dos_exernel.cnf，找到配置项 collect_timer_value，将其设置为 collect_timer_value=5 或 collect_timer_value=10；然后退出保存，并将该节点所在站其他节点外核全部杀死，然后首先在本站内系统该节点的外核，等启动好再逐个启动系统内其他节点外核程序。

12.1.4 在字符界面下登陆 dpfs 文件系统失败

问题描述：

在字符界面下登陆 dpfs 文件系统失败。

可能原因：

a) 系统外核 dos_exernel 程序没有启动或正在启动

b) 外核虽然已启动但没有加载内核模块

处理方法：

执行 ps –ax 命令，查看是否有./dos_exernel –d 进程；如果有该进程，则继续执行 lsmod 查看显示结果中是否有 dpfs 模块，如果有则表明内核模块已被加载。没有 dpfs 模块则需要执行命令：

cd /lib/modules/2.6.5-7.244-dpfs/kernel/fs/dpfs

insmod dpfs.o

mount –t dpfs none /dpfs 加载内核模块。

此时可以执行命令：#tail –f /tmp/dos_exernel_stdout 查看外核启动进度，如果外核已正常启动成功则可以尝试再次登陆 dpfs 文件系统。

13 常用命令速查

表格 13-1 命令常用命令速查

编号	系统指令	帮助说明
1	do_su	功能： 切换数字有机体用户，默认为数字有机体超级用户（DOSroot） 用法： do_su [USER] 选项： --help display this help and exit --version output version information and exit
2	do_shareadd	**功能：** 输出本节点指定目录下的磁盘空间，并按规则将指定目录下的文件资源添加到数字有机体文件系统中去。 用法： do_shareadd EXPORT_DIRECTORY [SHARED_DIRECTORY] **选项：** --help display this help and exit --version output version information and exit
3	do_sharedel	功能： 撤销输出的本地共享空间，在系统中屏蔽掉该共享空间下的文件副本。 用法： do_sharedel EXPORT_DIRECTORY [SHARED_DIRECTORY].
4	do_replicas	功能： 查看资源副本在系统中的分布。 用法： do_replicas FILE. 选项： --help display this help and exit --version output version information and exit

5	do_repblockadd	功能： 为资源的文件块在系统中指定节点上增加一个副本。 用法： do_repblockadd [OPTION] [IP/STATION_ID] FILE BLOCKNUM 选项： -s add to station, should give STATION_ID -n add to node, should give node's IP --help display this help and exit --version output version information and exit
6	do_repblockdel	功能： 删除资源的块在系统中指定节点上的一个副本。 用法： do_repblockdel [OPTION] [IP/STATION_ID] FILE BLOCKNUM 选项： -s add to station, should give STATION_ID -n add to node, should give node's IP --help display this help and exit --version output version information and exit
7	do_checkridi	功能： 对数字有机体文件系统进行错误检查，排除可能存在的不一致。 用法： do_checkridi [FILE/DIRECTORY]… 选项： --help display this help and exit --version output version information and exit
8	do_serviceadd	功能： 添加一个服务，让调度程序管理该服务的请求。 用法： do_serviceadd -n SERVICE -h HOST -p PORT -i PID -t TYPE 选项： -n NAME of SERVICE -h HOST of SERVICE -p PORT of SERVICE -i PID of SERVICE -t TYPE of SERVICE c is client node,s is server node. --help Give this help list --usage Give a short usage message --version Print program version

9	do_servicedel	功能： 从调度程序中撤销一个服务。 用法： do_servicedel -n SERVICE -h HOST -p PORT -i PID -t TYPE 选项： -n NAME of SERVICE -h HOST of SERVICE -p PORT of SERVICE -i PID of SERVICE -t TYPE of SERVICE c is client node s is server node --help Give this help list --usage Give a short usage message --version Print program version
10	do_groupadd	功能： 添加一个数字有机体用户组。 用法： do_groupadd -n GROUP -d DESCRIPTION 选项： -n Name of group -d Description of group --help Give this help list --usage Give a short usage message --version Print program version
11	do_groupdel	功能： 删除一个数字有机体用户组 用法： do_groupdel GROUP… 选项： --help Give this help list --usage Give a short usage message --version Print program version

12	do_groupmod	功能： 修改一个数字有机体用户组。 用法： do_groupmod -n GROUP -d DESCRIPTION 选项： -n Name of group -d Description of group --help Give this help list --usage Give a short usage message --version Print program version
13	do_groups	功能： 查询所有数字有机体用户组 用法： do_groups 选项： --help Give this help list --usage Give a short usage message --version Print program version
14	do_group	功能： 查看一个数字有机体用户组的详细信息。 用法： do_group GROUP... 选项： --help Give this help list --usage Give a short usage message --version Print program version
15	do_userdel	功能： 删除一个数字有机体用户。 用法： do_userdel USER 选项： --help Give this help list --usage Give a short usage message --version Print program version

16	do_useradd	功能： 添加一个数字有机体用户。 用法： do_useradd -n USER -g GROUP [-r REAL_USER] [-t TYPE] [-d DESCRIPTION] [-q QUOTA] [-s SPACE] [-h HOST,...] 选项： -n USER of DoOS -g GROUP of DoOS -r REAL USER, default is USER -t TYPE, it maybe: g -- group manager s --station manager n --normal user By default, it's normal user. -d DESCRIPTION of USER -q QUOTA of USER, default is 100MB -s real SPACE of USER, default is QUOTA's 3 times -h login HOST(s), default is all --help Give this help list --usage Give a short usage message --version Print program version
17	do_users	功能： 查询所有数字有机体用户 用法： do_users 选项： --help Give this help list --usage Give a short usage message --version Print program version
18	do_user	功能： 查看一个数字有机体用户信息 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_user USER...

19	do_usermod	功能： 修改一个数字有机体用户 选项： -n USER of DoOS -g GROUP of DoOS -r REAL USER, default is USER -t TYPE, it maybe: g -- group manager s --station manager n --normal user By default, it's normal user. -d DESCRIPTION of USER -q QUOTA of USER, default is 100MB -s real SPACE of USER, default is 3 QUOTA's time -h login HOST(s), default is all --help Give this help list --usage Give a short usage message --version Print program version 用法： do_usermod -n USER -g GROUP [-r REAL_USER] [-t TYPE] [-d DESCRIPTION] [-q QUOTA] [-s SPACE] [-h HOST,...]
20	do_userfreeze	功能： 冻结一个数字有机体用户 选项 --help Give this help list --usage Give a short usage message --version Print program version 用法： do_userfreeze USER...
21	do_userrecover	功能： 恢复一个已冻结的数字有机体用户 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_userrecover USER...

22	do_passwd	功能： 修改一个数字有机体用户的密码 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_passwd [USER]
23	do_services	功能： 查看注册的服务所在节点 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_services SERVICE
24	do_stations	功能： 查看所有站的信息，包括 IP 地址 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_stations
25	do_getmainm	功能： 获取管理节点的 IP 地址 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_getmainm

26	do_setmainm	功能： 修改管理节点的 IP 地址 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_setmainm “password”
27	do_setdircrypt	功能： 设置一个空目录是否为加密目录，加密目录下的文件全部加密，加密为 1，否则为 0 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_setdircrypt dpfs_dir_name encrypt_flag
28	do_umount	功能： 卸载一个目标为“/dpfs”下的目录 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_umount TO_DIRECTORY
29	do_mount	功能： 挂载一个目标为“/dpfs”下的目录 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_mount FROM_DIRECTORY TO_DIRECTORY 'options'

30	do_getblocksize	功能： 获取目录“/dpfs”下的目录的文件块大小 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_getblocksize [dirname|path]
31	do_setblocksize	功能： 修改目录“/dpfs”下的目录的文件块大小 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_setblocksize [dirname|path] [blocksize]
32	do_getdirreplicas	功能： 获取目录“/dpfs”下的目录的文件（块）副本数量配置 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_getdirreplicas [DIR]
33	do_setdirreplicas	功能： 修改目录“/dpfs”下的目录的文件（块）副本数量配置 选项： --help Give this help list --usage Give a short usage message --version Print program version 用法： do_setdirreplicas -d DIR [-s MIN_REP_OF_STATION] [-S MAX_REP_OF_STATION] [-m MIN_REP_IN_STATION] [-M MAX_REP_IN_STATION]