

RackTables 安装及应用指南

一、Racktables 简介	2
1.1 存在的问题.....	2
1.2 Racktables 简介	3
1.3 参考资源.....	3
二、Racktables 安装及初始设置	4
2.1 前期准备.....	4
2.2 安装相关组件.....	4
2.3 配置数据库.....	4
2.4 配置 apache.....	5
2.5 安装 racktables	6
三、Racktables 配置	7
3.1 关于中文.....	7
3.2 界面说明.....	7
3.3 权限设置.....	9
四、Racktables 升级	11
五、Racktables 其它设置	11
5.1 可选属性说明.....	11
5.2 字典（字段）属性说明.....	12
5.3 软件参数设定说明.....	14
5.4 供应商筛选.....	15
5.5 增加端口映射.....	16
5.6 增加 NATv4 关联.....	17

一、Racktables 简介

1.1 存在的问题

随着公司服务器规模的增加，以往手工记录服务器资产的方式存以下的问题：

无法准确统计 IT 资产

- 1、有多少需要处理的 IT 资产？
- 2、资产中有哪些特殊的设备供应商
- 3、过去的三年中每年新增的服务器是多少？有多少无法使用的或是资源利用率很低的？
- 4、资产设备什么时候过保？
- 5、如何操作资产更新（例如服务器 IP 变更或硬件资源变更）
- 6、现有环境有多少 IP 资源可用？核心的设备是哪些？哪个组哪个部门对其负责

硬件资源维护

- 1、迁移服务器时需要多少 IT 设备才能保证工作正常，设备的更新列表有哪些？
- 2、有多少已经过保的设备？有多少建议淘汰的服务器？
- 3、有哪些服务器会因为 UPS 管理维护导致不能使用？
- 4、重要设备到交换机的网络接入点以及交换机之间的网络连线是如何分配的？

软件 License 管理

- 1、需要多少必备软件的 License？
- 2、每个服务器的 License 情况如何？
- 3、有哪些是已失效的 License？

文档管理

- 1、相关的配置文件能否及时恢复，保证软件的正常应用？
- 2、服务器及周边设备的配置、处理文档是否完整？

以上这些都成了实际工作中所面临的问题，再加上人员的变动手工记录的方式已越来越不适应。在这中间尝试过用 iTop（太庞大，主要面向 ITSM）、otrs（主要面向 ITSM 的服务台）、cmdbuild（配置复杂，驾驭不了）后发现 racktables 比较适合我们的现有需求（当然也有可能是对其它的工具了解较少）。

1.2 Racktables 简介

Racktables 是一个用来管理机房资产的开源工具，可以用来管理成百上千台的服务器及更多的 IP 和 MAC 地址。适用于机房和数据中心的服务器管理。


summary

机柜行列: 第一排
名称: 二号机柜
高度: 42
资产标签: rack-02
% 使用情况:
对象: 6
Explicit tags: 标准机柜
Implicit tags: 机柜

files (1)

文件 注释

2号机柜20140120.jpg
94.2k



Rack diagram

第一排: ← 二号机柜 →

	Front	Interior	Back
42			
41			
40			
39			
38		192.168.0.33	
37			
36			
35			
34			
33			
32		192.168.0.145	
31			
30			
29			
28			
27		192.168.0.24	
26			
25			
24			
23			
22		192.168.0.55	
21			
20			
19			
18		s-192.168.0.24	
17			
16			
15			
14			
13			
12		s-192.168.0.24	
11			
10			
9			
8			
7		s-192.168.0.24	

1.3 参考资源

http://wiki.racktables.org/index.php/Main_Page

二、Racktables 安装及初始设置

2.1 前期准备

在使用 yum 指令前建议先设定好 linux 软件源（如下图所示，可以参考 <http://waringid.blog.51cto.com/65148/904201>）。需要注意的是 epel 源中的 epel-release-6-7.noarch.rpm 已经换成了 epel-release-6-8.noarch.rpm 版本（同时注意 32 位和 64 位版本的区别）。

2. 增加软件源

1. `wget http://mirrors.163.com/.help/CentOS6-Base-163.repo`
2. `mv CentOS6-Base-163.repo /etc/yum.repos.d/`
3. `wget http://mirrors.ustc.edu.cn/fedora/epel//6/i386/epel-release-6-7.noarch.rpm`
4. `rpm -ivh epel-release-6-7.noarch.rpm`

```
yum -y install gcc gcc-c++ autoconf libjpeglibjpeg-devel libpng libpng-devel freetype \
freetype-devel libxml2 libxml2-develzlib zlib-devel glibc glibc-devel glib2 glib2-devel \
bzip2 bzip2-devel ncursesncurses-devel curl curl-devel e2fsprogs e2fsprogs-devel krb5 \
krb5-devel libidnlibidn-devel openssl openssl-devel make patch pcre-devel
cd /root
wget
http://downloads.sourceforge.net/project/racktables/RackTables-0.20.6.tar.gz?r=http%3A%2F%2F
racktables.org%2F&ts=1388393599&use_mirror=jaist
tar zxvf RackTables-0.20.6.tar.gz
```

2.2 安装相关组件

安装运行需要的相关组件，如数据库、PHP 支持等，如下所示。

```
yum install mysql-server mysql
yum install php php-mysql php-pdo php-gd php-snmp php-mbstring php-bcmath httpd
httpd-tools
mv -r RackTables-0.20.6 /usr/share/
mkdir /var/www/html/it
cd /var/www/html/it
ln -s /usr/share/RackTables-0.20.6/wwwroot/index.php ./index.php
```

2.3 配置数据库

通过以下命令初始化数据库，如下所示。

```
service mysqld start
mysql -uroot -p
create database racktables;
```

```
grant all on racktables.* to root;
grant all on racktables.* to root@localhost;
grant all on racktables.* to rackuser;
grant all on racktables.* to rackuser@localhost;
set password for rackuser@localhost=password('password');
exit
```

2.4 配置 apache

通过以下命令配置 apache，如下所示。

```
<VirtualHost *:81>
    ServerAdmin zabbix
    DocumentRoot /var/www/html/it
    ErrorLog /var/log/httpd/racktables.log
    HostnameLookups Off
    UseCanonicalName Off
    ServerSignature Off

    <Directory />
        Options FollowSymLinks
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    <Directory "/var/www/html">
        Options Indexes FollowSymLinks
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
~
```

```
vim /etc/httpd/conf.d/it.conf
```

```
<VirtualHost *:81>
    ServerAdmin zabbix
    DocumentRoot /var/www/html/it
    ErrorLog /var/log/httpd/racktables.log
    HostnameLookups Off
    UseCanonicalName Off
    ServerSignature Off

    <Directory />
        Options FollowSymLinks
        AllowOverride None
        Order allow,deny
        Allow from all
```

```
</Directory>
```

```
<Directory "/var/www/html">
```

```
Options Indexes FollowSymLinks
```

```
AllowOverride None
```

```
Order allow,deny
```

```
Allow from all
```

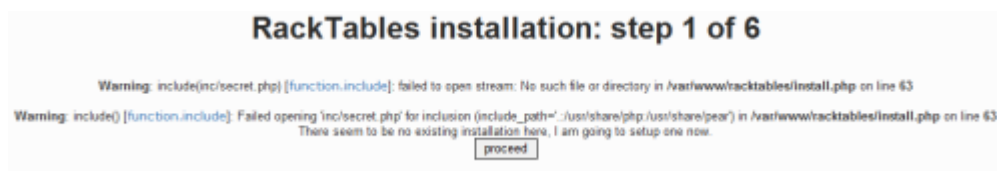
```
</Directory>
```

```
</VirtualHost>
```

```
service httpd start
```

2.5 安装 racktables

打开浏览器按设置输入地址，例如：<http://192.168.88.123:81> 进入安装界面，同时选择继续。如下所示。



下一步进入到环境检测界面，如果不需要用到 snmp 协议（关于 snmp 方面的内容请参考“[zabbix 中文配置指南续](#)”）可以忽略该提示，如果没有问题选择继续。如下图所示。

RackTables installation: step 2 of 6

check item	result
PDO extension	Ok
PDO-MySQL	Ok
hash functions	Ok
SNMP extension	Ok
GD functions	Ok
HTTP scheme	HTTP (all your passwords will be transmitted in cleartext)

接下来会提示按要求建立配置文件，主要是设定 racktables 的数据库连接。如下图所示。

```
touch /var/www/html/it/inc/secret.php
```

```
chmod 666 /var/www/html/it/inc/secret.php
```

RackTables installation: step 3 of 6

The inc/secret.php file is not writable by web-server. Make sure it is. The following commands should suffice:

```
touch inc/secret.php  
chmod 666 inc/secret.php
```

RackTables installation: step 3 of 6

MySQL host:
database:
username:
password:

The above parameters did not work. Check and try again.

RackTables installation: step 3 of 6

The configuration file has been written successfully.

完成配置文件的建立后，按页面的提示完成后续的安装。如下图所示。

RackTables installation: step 6 of 6

Congratulations! RackTables installation is complete. After pressing Proceed you will enter the system. Authenticate with **admin** username.
RackTables web-site runs some [wiki](#) pages and a [bug tracker](#).
We have also got a [mailing list](#) for users. Have fun.

三、Racktables 配置

3.1 关于中文

该软件不支持多语言，在英文版本的状态下可以正常显示中文字符（需数据支持中文编码），如果需要中文化需自行修改代码中的英文描述（可以通过修改 php 代码实现初步的中文化）。

3.2 界面说明

Rackspace

机柜配置。该功能用来配置机房的位置以及机柜的信息。包含机柜所在的位

置、行列以及机柜的名称、高度、排列次序、服务器在机柜中的位置等，如下图所示。



Objects

设备对象配置。该功能用来配置设备信息。包含该设备的配置信息（例如服务器的硬件配置、IP 信息、KVM 接口信息、归属的管理部门、线路的接口、保修期限、在机柜上的位置等），如下图所示。



IPv4 space

IP 地址管理。该功能用来配置 IP 地址信息。根据实际情况自定义使用到的 IP 地址，包含 VLAN 信息、网段状况、各网段 IP 应用情况以及和服务器设备的产联信息等。如下图所示。

192.168.0.0/24
服务器网段0

summary

192.168.0.0/24
服务器网段0
VLAN 10

%% used: 218/256
Netmask: 255.255.255.0
Netmask: 0xFFFFF00
Wildcard bits: 0.0.0.255

VLAN: VLAN10 @ 192.168.0.21
Routed by: 192.168.0.21@eth0

Explicit tags: 中型网络
Implicit tags: 网络

details

Address	Name	Comment	Allocation
192.168.0.0	网络地址 (保留)	RESERVED	
192.168.0.1	网关地址 (保留)	RESERVED, L1: vlan10@192.168.0.2	
192.168.0.2	服务器网段0	Cluster@192.168.0.2 Virtual MSSQL:1433/TCP→	
192.168.0.3	服务器网段0	eth0@192.168.0.3	
192.168.0.4	RouteOS路由服务器	vmnic@192.168.0.4	
192.168.0.5	服务器网段0	vmnic@192.168.0.5	
192.168.0.6	服务器网段0	eth0@192.168.0.6	
192.168.0.7	服务器网段0	(保留)	
192.168.0.8	服务器网段0	(保留)	
192.168.0.9	服务器网段0	vmnic@192.168.0.9	

Configuration

配置管理。该功能最灵活。根据实际情况配置各项目的内容以及详细的字段内容、各组件的属性、对象的包含关系、用户的界面、用户及权限等。如下图所示。

View Edit attributes Edit map

Optional attributes

Attribute name	Attribute type	Applies to
base MAC address	[S] string	Server
CPU, MHz	[U] unsigned integer	Network chassis Network security Network switch Router VM Resource Pool
CPU型号	[D] dictionary record	Server (values from 'CPU型号')
CPU数量	[D] dictionary record	Server (values from 'CPU数量') VM (values from 'CPU数量')
DRAM, MB	[U] unsigned integer	Network chassis Network security Network switch Router VM Resource Pool
flash memory, MB	[U] unsigned integer	Network chassis Network security Network switch Router

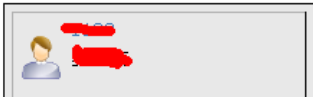
3.3 权限设置

软件是初始化安装时提示输入管理员的密码，初始化完成后登陆系统在“Main page:Configuration:Local users”对用户进行相关设定。

Main page : Configuration : Local users

View Edit

User accounts (7)



The screenshot shows the RackTables Admin interface. At the top, there are 'View' and 'Edit' tabs. Below them is the 'Add new' section, which includes input fields for 'Username', 'Real name', and 'Password', along with a 'Tags' field containing the text 'new tags here...'. Below this is the 'Manage existing (7)' section, which displays a table of existing users with columns for 'Username', 'Real name', and 'Password'. Each row has a small edit icon to its right.

Username	Real name	Password
138	
144	
3238	
3238	
1138	
1138	
3083	
3083	
3755	
admin	RackTables Administrator
user	普通用户

首先根据实际情况建立使用用户，设定好系统的应用帐号。其中“edit”用来修改原有用户或新建用户，如上图所示。

用户权限的配置是一件比较复杂的事情，和 windows 应用程序不同的是无法通过图形的方式实现权限的修改，只能通过修改配置文件（“Main page:Configuration:Permissions”）的方式来实现。具体可参考：<http://wiki.racktables.org/index.php/RackTablesAdminGuide>。

```

Main page : Configuration : Permissions

View Edit

1 allow {$userid_1}
2 allow {$username_user} and {$tab_default}
3 deny {$username_144} and {$page_config}
4 allow {$username_144}
5 deny {$username_3238} and {$page_config}
6 allow {$username_3238}
7 deny {$username_1138} and {$page_config}
8 allow {$username_1138}
9 deny {$username_3083} and {$page_config}
10 allow {$username_3083}
11 deny {$username_3755} and {$page_config}
12 allow {$username_3755}

```

`allow {$userid_1}`

管理员用户可应用所有功能

`allow {$username_user} and {$tab_default}`

user 用户可只读查看所有功能

```
deny {$username_abc} and {$page_config}
allow {$username_abc}
```

abc 用户无法查看（也无法更改）“Configuration” 功能，其它项目的功能都是可读写

四、Racktables 升级

Racktables 的升级很简单，只要版本的跨度不大（尝试过从 0.20.5 升级至 0.20.7）都可以自动完成升级。

- 1、首先做好数据备份（主要是数据库），可以通过以下的指令进行：

```
mysqldump --user=racktables_user --password=www.myj123.com racktables_db >
/tmp/backup_racktables_db_20140318.sql
```

- 2、将新版本的文件存放至指定文件夹，同时保留原有版本的配置文件，如下所示：

```
cp -R /root/RackTables-0.20.7/ /usr/share/
cp /usr/share/RackTables-0.20.5/wwwroot/inc/secret.php
/usr/share/RackTables-0.20.7/wwwroot/inc/
```

- 3、删除原有的链接文件同时建立新的文件，如下所示：

```
rm /var/www/html/it/index.php
ln -s /usr/share/RackTables-0.20.7/wwwroot/index.php /var/www/html/it/index.php
```

- 4、完成后重新登录 RackTables 系统按页面提示完成后续的升级操作。

五、Racktables 其它设置

Racktables 的功能十分强大，系统自带了大部份的设备及相关配置信息，但毕竟实际环境会有些差距，可以通过自定义的方式达到实用效果。

5.1 可选属性说明

可以通过“Main page” - “Configuration” - “Attributes” 选项来增加自定义的字段。

“Attribute type” 属性类型用于定义字段的类型，用于确定在录入该字段信息时输入内容的属性，有以下几种：

- [S] string 字符串类型，一串字符。例如：主机名称
- [U] unsigned integer 整数类型，输入整数。例如：内存大小
- [F] floating point 浮点类型，输入小数。例如：电压
- [T] date 日期类型，输入日期。例如：合同过期时间
- [D] dictionary record 字典类型，录入时可选。例如：CPU 型号

“Applies to” 指将定义的的这个字段应用到哪一类的设备上，例如为“server” 类型增加 CPU 型号，如下图所示。

CPU, MHz	[U] unsigned integer	Network chassis Network security Network switch Router VM Resource Pool
CPU型号	[D] dictionary record	Server (values from 'CPU型号')
CPU数量	[D] dictionary record	Server (values from 'CPU数量') VM (values from 'CPU数量')

完成字段的添加后再进入设备的资产录入界面就可以发现在指定类型的设备属性中增加了自定义的 CPU 型号和 CPU 数量两个字段，如下图所示。

Attributes

Type: Server

Common name:

Visible label:

Asset tag:

Select container:

CPU型号:

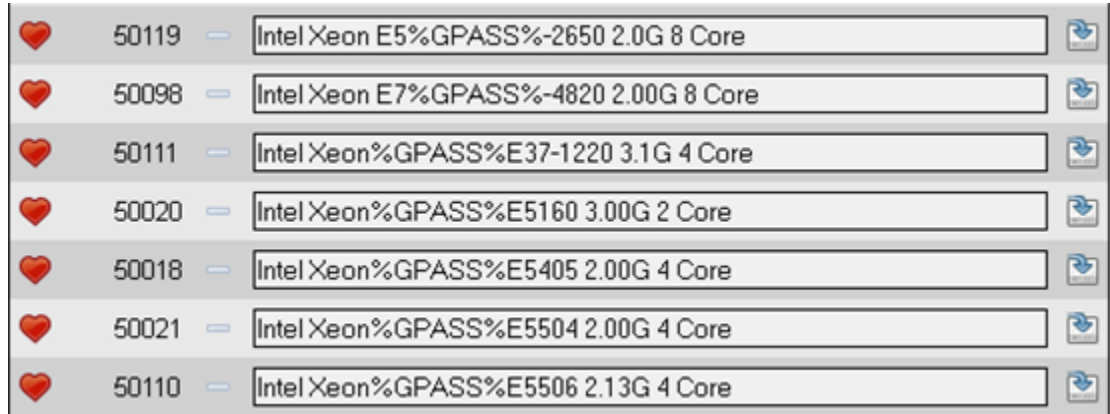
CPU数量:

5.2 字典（字段）属性说明

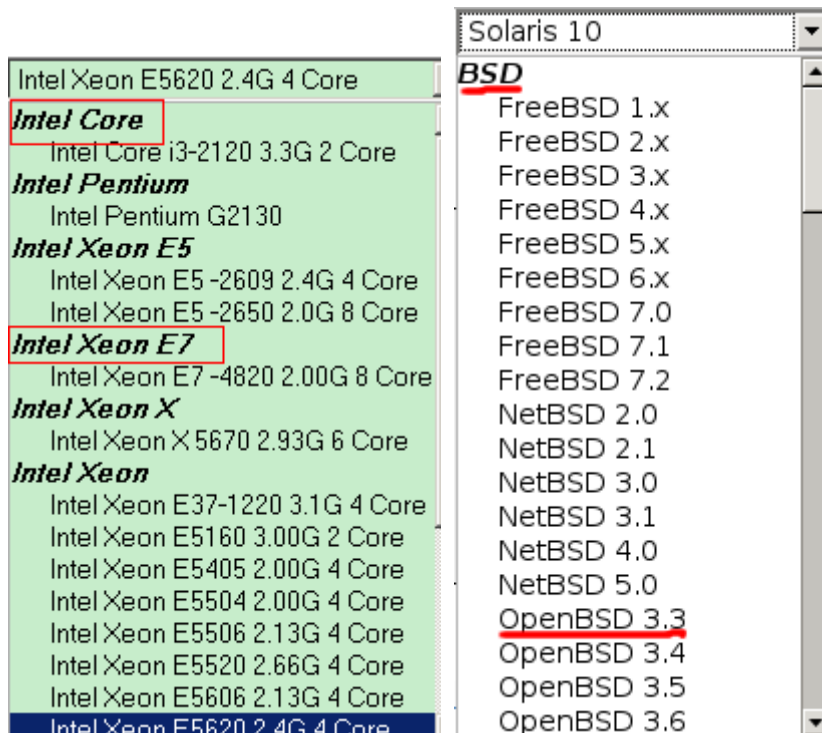
字段的内容可以自定义（参见“5.1 可选属性说明”），当建立的字段内容不多时可以将其定义为“[D] dictionary record”属性，这样用户在录入时只需

选取指定的内容就可以了，减少了输入的工作和出错的比率。

通过“Main page” - “Configuration” - “Dictionary” 进入字段字典设定界面，该界面中的字段信息应该和“5.1 可选属性说明”中的内容保持一致。例如在“CPU 型号”字段上可以通过以下方式增加对应的型号，如下图所示。



在内容录入时可以通过上图的方式进行，其中“%GPASS%”指用户在选择时该字符前的内容为一个大类。而如果是“%GSKIP%”则表示忽略该字段沿用原有的内容。例如输入“BSD%GSKIP%OpenBSD 3.6”。如下图所示。



也可以在录入时加入该内容的链接信息，例如在录入 CPU 型号时加入该 CPU 的详细信息，例如通过这种方式的录入：“[[Intel Xeon E5620 2.4G 4 Core |

<http://ark.intel.com/products/47925/Intel-Xeon-Processor-E5620-12M-Cache-2-40-GHz-5-86-GTs-Intel-QPI>]]”，其中[]表示该信息有对外的链接，设置好后更新原有的内容就会增加网站链接，如下图所示。



5.3 软件参数设定说明

打开“Main”-“Configuration”-“User interface”界面进入用户的界面及显示内容的设定，如下图所示。

Current configuration	
Option	Value
8021Q_DEPLOY_MAXAGE <i>802.1Q deploy maximum age (system-wide)</i>	3600
8021Q_DEPLOY_MINAGE <i>802.1Q deploy minimum age (system-wide)</i>	300
8021Q_DEPLOY_RETRY <i>802.1Q deploy retry timer (system-wide)</i>	10800
8021Q_EXTSYNC_LISTSRC <i>List source: objects with extended 802.1Q sync (system-wide)</i>	false
8021Q_INSTANT_DEPLOY <i>802.1Q: instant deploy</i>	no
8021Q_MULTILINK_LISTSRC <i>List source: IPv4/IPv6 networks allowing multiple VLANs from same domain (system-wide)</i>	false
8021Q_WRI_AFTER_CONF1_LISTSRC <i>802.1Q: save device configuration after deploy (RackCode) (system-wide)</i>	false
ADDNEW_AT_TOP <i>Render "add new" line at top of the list</i>	yes
ASSETWARN_LISTSRC <i>List source: object, for which asset tag should be set (system-wide)</i>	{typeid_4} or {typeid_7} or {typeid_8}
AUTOPORTS_CONFIG <i>AutoPorts configuration (system-wide)</i>	4 = 1*33*kvm + 2*24*eth%u;15 = 1*446*kvm
CACTI_LISTSRC <i>List of object with Cacti graphs (system-wide)</i>	false

enterprise 标记使用该软件的公司或企业名称

DEFAULT_RACK_HEIGHT 设定默认的机柜高度

DATETIME_FORMAT 设定日期格式

QUICK_LINK_PAGES 需要快捷显示的功能页

5.4 供应商筛选

RackTables 在安装的时候默认加载了市场上常见的设备及操作系统型号，而在实际的应用中并不会所有的内容都需要，可以通过设定供应商筛选的方式来屏蔽显示（仅不显示而已，数据仍存在数据库中）。

例如当用户在输入服务器操作系统时发现有很多的操作系统版本，假设实际的环境中用不到“FreeBSD”这个版本（如下图所示），可以通过以下的方法来屏蔽。

Visible label:

Asset tag:

Container: 192.168.0.159

Select container:

CPU数量: 2个

主机名称:

内存大小,GB: 2

操作系统: Windows 2003

硬盘大小,GB:

磁盘柜大小,GB:

联系人,电话:

Has problems:

Actions: **FreeBSD**

- FreeBSD 1.x
- FreeBSD 2.x
- FreeBSD 3.x
- FreeBSD 4.x
- FreeBSD 5.x
- FreeBSD 6.x
- FreeBSD 7.0
- FreeBSD 7.1
- FreeBSD 7.2
- FreeBSD 8.x
- FreeBSD 9.x

生活易官网服务器

首先打开“Main” - “Configuration” - “User interface”界面进入用户的界面及显示内容的设定，然后在“VENDOR_SIEVE”中输入需要屏蔽的操作系统名称，如下图所示。

VENDOR_SIEVE
Vendor sieve configuration

操作完成后保存该设定然后回到之前的设备录入界面并选择“操作系统”选项，可以看到有关“FreeBSD”的操作系统已屏蔽显示。

CPU数量:	2个
主机名称:	
内存大小,GB:	2
操作系统:	Windows 2003
硬盘大小,GB:	
磁盘柜大小,GB:	
联系人,电话:	
Has problems:	
Actions:	<ul style="list-style-type: none"> CentOS <ul style="list-style-type: none"> CentOS V2 CentOS V3 CentOS V4 CentOS V5 CentOS V6 Debian <ul style="list-style-type: none"> Debian 2.0 (hamm) Debian 2.1 (slink) Debian 2.2 (potato) Debian 3.0 (woody) Debian 3.1 (sarge) Debian 4.0 (etch) Debian 5.0 (lenny) Debian 6.0 (squeeze) Debian 7.0 (wheezy) Gentoo <ul style="list-style-type: none"> Gentoo 10.0

官网服务器

5.5 增加端口映射

通过输入属性内容“Properties”来记录设备的硬件配置信息（包括保修期限等），同时通过端口信息来记录该设备的端口连接情况，如下图所示。

Ports and interfaces

Local name	Visible label	Interface	Start Number	Count
<input type="text"/>	<input type="text"/>	1000Base-T	<input type="text"/>	<input type="text"/>

[Clear port list](#)

Local name	Visible label	Interface	L2 address	Remote object and port	Cable ID	(Un)link or (un)reserve
<input type="text"/>	<input type="text"/>	1000Base-T	<input type="text"/>			<input type="text"/>
KVM	12号	KVM (host)		kvm-192.168.0.81 kvm12	<input type="text"/>	<input type="text"/>
eth0	1	1000Base-T		h3c-01 G11	<input type="text"/>	<input type="text"/>
eth1		1000Base-T			<input type="text"/>	<input type="text"/>

和端口“Ports”相关（网络接口）的会有 IP 地址，在录入端口映射地址前必须完善 IP 地址，如下图所示。

Allocations

OS interface	IP address	network	routed by	type	misc
<input type="text"/>	<input type="text"/>			Connected	<input type="text"/>
eth0	192.168.0.83 (便民WEB服务器)	192.168.0.0/24 .256 服务器网段0 中型网络 VLAN 19	192.168.0.21	Connected	<input type="text"/>

完成 IP 地址输入后就有地址映射“NATv4”的选项。在该项中，“Match endpoint”指本地的 IP 地址，后面输入本地的端口号；“Translate to”是用于映射的外网地址和端口中；“Target object”指该外网 IP 所经过的设备（例如路由器等），如下图所示。

locally performed NAT					
	Match endpoint	Translate to	Target object	Comment	
+	TCP eth0: 192.168.0.83:8282				+
=	TCP/eth0: 192.168.0.83:8282	58.30.150.100:8282	192.168.0.21(eth5)	192.168.0.21:8282	
=	TCP/eth0: 192.168.0.83:8282	121.19.138.100:8282	192.168.0.21(eth3)	192.168.0.21:8282	

5.6 增加 NATv4 关联

系统默认的内容并不是所有带 IP 地址的就有“NATv4”关联项，正常情况下“server”类的设备会有，而“VM”类的就没有，如下图所示。

Main page : Objects : 192.168.0.31

View Properties Log Ports IP Tags Files 802.1Q order

View Properties Log Rackspace Ports IP **NATv4** Tags Files 802.1Q order

192.168.0.83

可以通过以下的步骤增加“VM”类的“NATv4”选项。首先需要记录“VM”的类型 ID（其实是数据库中的对应 ID），在查看页面选中“Object type”的内容然后查看它的链接，例如本例中就是 1504，如下图所示。

summary

Common name: 192.168.0.31

Object type: **VM**

Container: 192.168.0.159

CPU数量: 1个

192.168.0.240/racktables/index.php?page=depot&tab=default&cfe={ \$typeid_1504 }

接着进入“Main page” - “Configuration” - “User interface”中编辑“IPV4NAT_LISTSRC”按该项目中的原有格式增加记录，如图所示。

```
IPV4NAT_LISTSRC {typeid_4} or {typeid_1504} or {typeid_7} or {typeid_8} or {typeid_798}
List source: IPv4 NAT performers (system-wide)
```

设置完成后好下图所示。

