

TARS 部署文档

目录

1. 部署准备	2
2. 环境准备	2
a. 部署服务器	2
b. 创建相关用户与用户组	2
c. 相关文件放置	2
d. 创建后台目录	2
3. 安装基本组件	3
a. 安装基本组件	3
b. 配置 rsync	3
c. 配置 svnserve	4
d. Mysql 配置	5
e. Nginx 域名配置	6
f. Php 配置	8
4. 修改 API 配置文件	10
a. API	10
b. PkgWorker	11
c. 命令通道	12
5. 修改 portal 配置文件	12
6. TARS 工具代码部署	12
7. Php 扩展	13
a. mhash	13
b. libmcrypt	13
c. mcrypt	13
d. mcrypt-php	13
8. TARS 的第一次使用	14

1. 部署准备

基本组件:

Ngnix	服务配置
Php	版本大于 5.3
Svn server	版本管理
Rsyncd	文件传输
Mysql	数据库

开源文件:

文件名	说明
tars-api-v0.1.1.tar.gz	API 源代码
tars-portal-v0.1.1.tar.gz	前台源代码
Tars.sql	数据库文件

依赖文件:

文件名	说明
mcrypt.tar.gz	Php 扩展/加解密

2. 环境准备

a. 部署服务器

文档中部署的服务器 IP 为 192.168.1.1

部署时可参照修改

b. 创建相关用户与用户组

这里创建了 tars.tars

```
groupadd tars
```

```
useradd tars -g tars
```

```
passwd tars
```

之后配置操作均在 tars 用户下操作

c. 相关文件放置

```
mkdir -p /data/tars_deploy;
```

```
chown -R tars.tars /data/tars_deploy
```

上传文件

```
[root@GK-todo-1412171818 /data/tars_deploy]# ll
total 3304
-rw-r--r-- 1 tars tars    8837 Apr 14 17:19 Tars.sql
drwxr-xr-x 7 tars tars    4096 Apr 15 20:33 mcrypt
-rw-r--r-- 1 tars tars 533467 Apr 14 17:46 tars_api.tar.gz
-rw-r--r-- 1 tars tars 2784052 Apr 14 17:54 tars_portal.tar.gz
```

d. 创建后台目录

```
mkdir -p /data/pkg/pkg_home
```

```
chown -R tars.tars /data/pkg/
```

3. 安装基本组件

a. 安装基本组件

```
yum -y install nginx
yum -y install mysql-server
yum -y install php-fpm php php-mbstring php-mysql
service nginx restart
service mysqld restart
service php-fpm restart
yum -y install subversion
yum -y install rsync
```

b. 配置 rsync

```
vim /etc/rsyncd.conf
```

添加以下内容:

```
use chroot = true
transfer logging = true
log format = %h %o %f %l %b
log file = /var/log/rsyncd.log
pid file = /var/run/rsyncd.pid
#hosts allow = trusted.hosts
slp refresh = 300
address = ****
```

```
[pkg_home]
```

```
uid = tars
```

```
gid = tars
```

```
path = /data/pkg/pkg_home
```

```
ignore errors
```

```
read only = false
```

```
use chroot = true
transfer logging = true
log format = %h %o %f %l %b
log file = /var/log/rsyncd.log
pid file = /var/run/rsyncd.pid
#hosts allow = trusted.hosts
slp refresh = 300

#[Example]
#   path = /home/Example
#   comment = An Example
#   auth users = user
#   secrets file = /etc/rsyncd.secrets

[pkg_home]
uid = tars
gid = tars
path = /data/pkg/pkg_home
ignore errors
read only = false
~
~
~
```

启动:

```
rsync --daemon
```

c. 配置 svnserve

这里配置到/data/svntars 目录

mkdir /data/svntars

svnadmin create /data/svntars/

vim /data/svntars/conf/passwd

添加:tars=tars

```
[users]
# harry = harrysecret
# sally = sallysecret
tars = tars
```

vim /data/svntars/conf/authz

在 groups 下添加:

admin=tars

[/]

@admin = rw

```
### grant read ( r ) access, read-write ( rw ) access and
### ( 'w' ).

[aliases]
# joe = /C=XZ/ST=Dessert/L=Snake City/O=Snake Oil

[groups]
# harry_and_sally = harry,sally
# harry_sally_and_joe = harry,sally,&joe

# [/foo/bar]
# harry = rw
# &joe = r
# * =

# [repository:/baz/fuz]
# @harry_and_sally = rw
# * = r
#
#
admin = tars

[/]
@admin = rw
```

vim /data/svntars/conf/svnserve.conf

添加以下内容:

anon-access = none

auth-access = write

password-db = passwd

authz-db = authz

```
[general]
### These options control access to the repository for unauthenticated
### and authenticated users. Valid values are "write", "read",
### and "none". The sample settings below are the defaults.
# anon-access = read
# auth-access = write
### The password-db option controls the location of the password
### database file. Unless you specify a path starting with a /,
### the file's location is relative to the directory containing
### this configuration file.
### If SASL is enabled (see below), this file will NOT be used.
### Uncomment the line below to use the default password file.
# password-db = passwd
### The authz-db option controls the location of the authorization
### rules for path-based access control. Unless you specify a path
### starting with a /, the file's location is relative to the the
### directory containing this file. If you don't specify an
### authz-db, no path-based access control is done.
### Uncomment the line below to use the default authorization file.
# authz-db = authz
### This option specifies the authentication realm of the repository.
### If two repositories have the same authentication realm, they should
### have the same password database, and vice versa. The default realm
### is repository's uuid.
# realm = My First Repository
anon-access = none
auth-access = write
password-db = passwd
authz-db = authz
```

修改客户端忽略配置(需要切换为 tars 用户)

执行一下 svn up

vim /home/tars/.subversion/config

global-ignores =

(此处置空, 避免 svn 过滤特殊文件)

```
### '*' matches leading dots, e.g. '*.rej' matches '.foo.rej'.
# global-ignores = *.o *.lo *.la *.al *.libs *.so *.so.[0-9]* *.a *.pyc *.pyo
global-ignores =
# *.rej *~ ##* .* *.swp .DS_Store
### Set log-encoding to the default encoding for log messages
```

启动

svnserve -d -r /data/svntars/

svnserve --listen-host 192.168.1.1 -d -r /data/svntars/

d. Mysql 配置

mysql 导入

修改 mysql 安装路径对应的 my.cnf

sql_mode=ANSI

```
#sql_mode=NO_ENGINE_SUBSTITUTION,STRICT_TRANS_TABLES
sql_mode=ANSI
```

mysql> source /data/tars_deploy/Tars.sql

添加用户密码

mysql> grant all privileges on *.* to tars@192.168.1.1 identified by 'tars';

mysql> flush privileges;

验证 `mysql -utars -ptars -h 192.168.1.1`

e. Nginx 域名配置

修改默认配置

添加以下内容:

```
#for tars
fastcgi_connect_timeout 300;
fastcgi_send_timeout 3600;
fastcgi_read_timeout 3600;
fastcgi_buffer_size 1024m;
fastcgi_buffers 4 1024m;
fastcgi_busy_buffers_size 1024m;
fastcgi_temp_file_write_size 1024m;
client_header_buffer_size 32m;
large_client_header_buffers 4 32m;
client_max_body_size 1024m;
#for tars end
```

```
user nginx;
worker_processes 1;

error_log /var/log/nginx/error.log warn;
pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;

    sendfile on;
    #tcp_nopush on;

    keepalive_timeout 65;

    #for tars
    fastcgi_connect_timeout 300;
    fastcgi_send_timeout 3600;
    fastcgi_read_timeout 3600;
    fastcgi_buffer_size 1024m;
    fastcgi_buffers 4 1024m;
    fastcgi_busy_buffers_size 1024m;
    fastcgi_temp_file_write_size 1024m;
    client_header_buffer_size 32m;
    large_client_header_buffers 4 32m;
    client_max_body_size 1024m;
    #for tars end

    #gzip on;

    include /etc/nginx/conf.d/*.conf;
}
```

将代码解压到 web 目录

`mkdir -p /data/webroot_tars/tars.qq.com;`

```
chown tars.tars /data/webroot_tars -R;
cd /data/webroot_tars/tars.qq.com;
cp /data/tars_deploy/tars-api-v0.1.1.tar.gz ./
cp /data/tars_deploy/tars-portal-v0.1.1.tar.gz ./
tar -xzf tars-api-v0.1.1.tar.gz
tar -xzf tars-portal-v0.1.1.tar.gz
```

nginx 目录下创建目录
[/etc/nginx]# mkdir logs

配置 API 域名

vim /etc/nginx/conf.d/api.tars.isd.com.conf

(注意配置中的空格符，特别是"/"后的)

```
server {
    listen 80;
    server_name api.tars.isd.com;
    index index.php index.html;

    access_log logs/api.tars.isd.com.conf-access.log;
    error_log logs/api.tars.isd.com.conf-error.log error;

    root /data/webroot_tars/tars.qq.com/pkg_opensrc;

    set $appname "";
    if ($uri ~ "^(/[^/]+)/") {
        set $appname $1;
    }

    location / {
        try_files $uri $uri/ $appname/index.php;
    }

    location /pkg/ {
        root /data/pkg/pkg_home;
    }

    location ~ \.php$ {
        fastcgi_pass 127.0.0.1:9000;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME $document_root/$fastcgi_script_name;
        include fastcgi_params;
    }
}
```

配置 portal 域名

vim /etc/nginx/conf.d/tars.isd.com.conf

```
server {
    listen 80;
    server_name tars.isd.com;
```

```

index index.php index.html;

access_log logs/tars.isd.com-access.log;
error_log logs/tars.isd.com-error.log error;

root /data/webroot_tars/tars.qq.com/open-pkg/www;

location / {
    try_files $uri $uri/ /index.php;
}

location ~ /api/.*\.php {
    try_files $uri $uri/ /index.php;
}

location ~ ^/download/ {
    rewrite /download/(.*) $1 break;
    proxy_pass http://127.0.0.1/pkg/$1;
    proxy_set_header Host api.tars.isd.com;
}

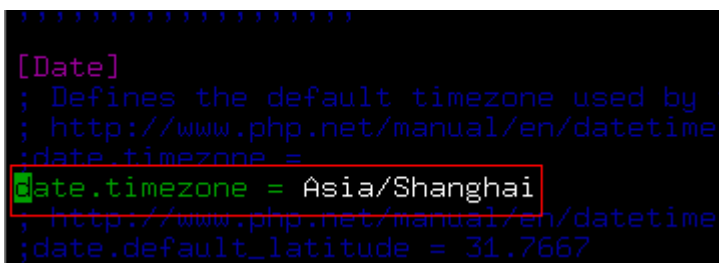
location ~ \.php$ {
    fastcgi_pass 127.0.0.1:9000;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME $document_root/$fastcgi_script_name;
    include fastcgi_params;
}
}

```

f. Php 配置

vim /etc/php.ini

date.timezone = Asia/Shanghai

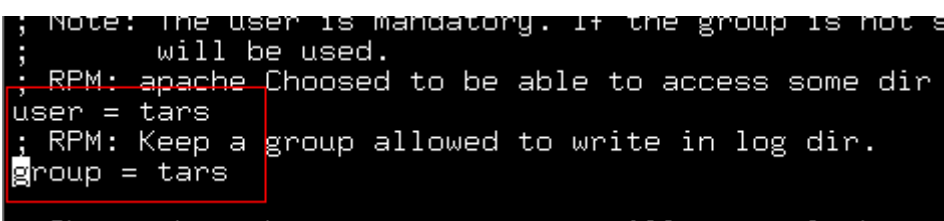


```

;date.timezone =
date.timezone = Asia/Shanghai
;date.default_latitude = 31.7667

```

vim /etc/php-fpm.d/www.conf



```

; Note: The user is mandatory. If the group is not set, the user
; will be used.
; RPM: apache Choosed to be able to access some dir
user = tars
; RPM: Keep a group allowed to write in log dir.
group = tars

```


修改 php session 权限

chmod 777 /var/lib/php/session/ -R

```
[root@GK-todo-1412171818 /var/lib/php]# chmod 777 /var/lib/php/session/ -R
```

修改 php.ini 文件

上传文件大小限制

```
; Maximum size of POST data that PHP will accept.  
post_max_size = 1024M
```

```
; Maximum allowed size for uploaded files.  
upload_max_filesize = 1024M
```

; Maximum allowed size for uploaded files.

upload_max_filesize = 1024M

; Maximum size of POST data that PHP will accept.

post_max_size = 1024M

重启服务

service php-fpm restart

4. 修改 API 配置文件

a. API

vim /data/webroot_tars/tars.qq.com/pkg_opensrc/publicsrc/conf/Conf.ini

如图修改:

```
[Open]
open='true'

[Setting]
private_key='myPassword_!'
max_file_size=41943040
php_path=/usr/bin/php      改为系统php路径
; to store all package-related files
package_path=/data/pkg/

[ApiServer]
server=192.168.1.1      修改为部署的机器IP与域名
port=80
hostname=api.tars.isd.com

[SynServer]
svn_user=tars      用户密码均为tars
svn_password=tars      server改为部署机器IP
svn_repo=svn://192.168.1.1

[Database]
pkg_db_host=192.168.1.1      数据库IP改为部署机器IP
pkg_db_port=3306
pkg_db_name=PackageCenterOpensrc
pkg_db_user=tars      用户密码均为tars
pkg_db_password=tars
pkg_db_info_table=sPackage
pkg_db_instance_table=sInstanceNet
pkg_db_usefav_table=sUserFav
pkg_db_task_table=ZY_taskInfo
pkg_db_task_result_table=ZY_taskDetail      server改为部署的机器IP/跟随api server,域名改为配置的
pkg_db_device_table=devicePassword

[FileServer]
;file_manage_host=
file_manage_host=192.168.1.1
file_manage_hostname=http://api.tars.isd.com
file_manage_mainurl=/filemanage/
file_manage_suburl_deletecache=delete_cache
file_manage_suburl_getfilelist=get_update_file_list
```

b. PkgWorker

vim /data/webroot_tars/tars.qq.com/pkg_opensrc/pkgworker/conf/Conf.ini

```
Open
open='true'

[DefaultSettings]
defaultTimeout = 600;
defaultRequestTimeout = 1200;
defaultBatchNum = 500;
defaultBatchInterval = 5;

[Database]
;this setting is the same with the public src Conf.ini
mysqlServer = '192.168.1.1';
mysqlPort = '3306';
mysqlDbName = 'PackageCenterOpensrc';
mysqlUserName = 'tars';
mysqlPassword = 'tars';
mysqlTaskTableName = 'ZY_taskInfo';
mysqlTaskResultTableName = 'ZY_taskDetail';

[RsyncServer]
;the server stores the package source
rsyncServer = '192.168.1.1';

[CommandTunnel]
;the tunnel runs the command to operate packages
commandHost = 'c.isd.com';
commandIp = '192.168.1.1';
commandUrl = 'http://api.tars.isd.com/command/BeginShellTask.php';
commandQueryUrl = 'http://api.tars.isd.com/command/QueryShellTask.php';

[APISetting]
;file manage API config
FileManageUrl = 'http://api.tars.isd.com/filemanage/';
FileManageHostName = 'api.tars.isd.com';
FileManageHost = '192.168.1.1';
;filemanagendost

;below do not need to change, recommend setting
mainurl = '/filemanage/';
suburlCheckCache = 'export';
suburlUploadPkg = 'upload';
suburlExport = 'export_update';
suburlDeleteIgnore = 'delete_ignore';
suburlIgnoreFile = 'ignore_file';

;below settings do not change, not for open version
[Rabbitmq]
;use rabbitmq to deal task or not
useRabbitmq = false;
```

数据库server改为配置的服务器IP, 端口默认3306, 用户密码tars

传输server改为配置的服务器IP

命令执行server改为配置的服务器IP, 同时url域名改为配置的api.tars.isd.com

API配置跟随部署的服务
器IP, 域名也是配置好的,
这里是api.tars.isd.com

c. 命令通道

vim /data/webroot_tars/tars.qq.com/pkg_opensrc/command/config.php

```
<?php
class AREA_CONFIG
{
    public static $server_type = "formal";

    public static $dbHost = "192.168.1.1"; //数据库host改为部署服务器IP,
    public static $dbPort = 3306; //用户名和密码为tars
    public static $dbName = "TARS";
    public static $dbUser = "tars"; //密码访问API配置部署的服务器,URL域名改为配置好的,这里
    public static $dbPassword = "tars"; //为api.tars.isd.com
    public static $passwordUrl = "http://api.tars.isd.com/query/device_password";
    public static $passwordHost = "192.168.1.1";
    ① //public static $authentication = "password"; //publickey .password
    public static $authentication = "publickey"; //publickey .password
    ② public static $privatekey = "/data/webroot/tars.qq.com/pkg_opensrc/command/key/id_rsa"; //

    public static function getLocalIP()
    {
        $cmd = "/sbin/ip route|grep ^src 172\.\.1src 10\.\.1awk '{print $9}'|head -n 1";
        $ret = shell_exec($cmd);
        $ip = trim($ret);
        return $ip;
    }
}
```

绿色框框部分为机器登录方式:

1. 密码登录, 需要将设备密码导入数据库 <http://tars.isd.com/devices>
2. 公钥登录, 设置配置项\$authentication 值为"publickey"
同时指定私钥文件目录, 默认上配置为 command 下的 key/id_rsa(绝对路径)
以上方式 2 选 1

5. 修改 portal 配置文件

vim /data/webroot_tars/tars.qq.com/open-pkg/tars.ini

pkg.api_url = "http:// 192.168.1.1/"

pkg.api_host = "api.tars.isd.com"

```
; TARS config file

[pkg]

pkg.api_url = "http:// 192.168.1.1 /"
pkg.api_host = "api.tars.isd.com"
```

6. TARS 工具代码部署

cd /data/pkg/

cp /data/webroot_tars/tars.qq.com/pkg_opensrc/tools/framework /data/pkg/ -a

cp /data/webroot_tars/tars.qq.com/pkg_opensrc/tools/pkg_tools /data/pkg/pkg_home -a

vim /data/pkg/pkg_home/pkg_tools/report.conf

```
rpt_ip=192.168.1.1
rpt_port=80
rpt_host=api.tars.isd.com
~
IP改为API部署的IP, 域名
同样修改为配置的域名
```

vim /data/pkg/pkg_home/pkg_tools/rsyncd.conf

将 rsync ip 配置为当前机器 ip:192.168.1.1

7. Php 扩展

mcrypt 扩展

```
[root@GK-todo-1412171818 /data/tars_deploy/mcrypt]# ll
total 2960
-rw-r--r-- 1 root root 875425 Apr 13 13:07 libmcrypt-2.5.8.tar.bz2
-rw-r--r-- 1 root root 471915 Apr 13 11:26 mcrypt-2.6.8.tar.gz
-rw-r--r-- 1 root root 725842 Apr 15 18:07 mcrypt.tar.gz
-rw-r--r-- 1 root root 931437 Apr 13 11:41 mhash-0.9.9.9.tar.gz
[root@GK-todo-1412171818 /data/tars_deploy/mcrypt]# tar -xzf *
```

全部解压之后

```
tar xzf mhash-0.9.9.9.tar.gz
```

```
tar xzf mcrypt-2.6.8.tar.gz
```

```
tar fzx libmcrypt-2.5.8.tar.bz2
```

a. mhash

```
cd mhash-0.9.9.9; ./configure ; make ; make install;
```

b. libmcrypt

```
cd libmcrypt-2.5.8; ./configure ;make;make install;
```

c. mcrypt

```
vim /etc/ld.so.conf
```

加入

```
/usr/local/lib/
```

```
include ld.so.conf.d/*.conf
/usr/local/lib/
```

ldconfig

```
cd mcrypt-2.6.8; ./configure && make && make install;
```

d. mcrypt-php

```
yum install php-devel;
```

```
phpize
```

```
cd /data/tars_deploy/mcrypt/php-5.3.27
```

```
sed -i 's|PHP_FE_END|{NULL,NULL,NULL}|' ./ext/**/*.c
```

```
sed -i 's|ZEND_MOD_END|{NULL,NULL,NULL}|' ./ext/**/*.c
```

```
cd ext/mcrypt/
```

```
phpize
```

```
./configure -with-php-config=/usr/bin/php-config
```

```
make
```

```
make install
```

```
[root@GK-todo-1412171818 /data/tars_deploy/mcrypt/php-5.3.27/ext/mcrypt]# make install
Installing shared extensions: /usr/lib64/php/modules/
```

```
vim /etc/php.ini
```

```
extension=/usr/lib64/php/modules/mcrypt.so
```

```
; If you only provide the name of the extension,
; the default extension directory will be used.
```

```
extension=/usr/lib64/php/modules/mcrypt.so
```

```
;;;
```

重启服务

8. TARS 的第一次使用

手动执行一次

在 fpm 的执行用户下执行:

```
svn co svn:// 192.168.1.1/ ./;
```

按照提示, 输入用户名 **tars**, 密码 **tars**

配置好本地访问 host 之后, 打开网页即可:

<http://tars.isd.com/>

初始登录管理员账号:

用户:admin

密码:admin