

cpFileToDir部署说明

cpFileToDir功能:

根据脚本的同级目录下配置文件cfg/cfg.cfg配置的条件将源文件拷贝到
目标目录

在脚本同级目录 tmp/back下会记录1天之内拷贝成功的文件，以防止重复拷贝

部署步骤

1.确定要拷贝的文件所属操作系统用户

比如要拷贝的文件是在root用户要生成的则,此文件操作系统用户属主为root

2. 用第1步确定的操作系统用户登录服务器

3. 将cpFileToDir压缩包导入服务器

4. 确定cpFileToDir放于系统哪个目录

比如常用的软件部署目录是 /fglyc/wpfs17/ 则将cpFileToDir压缩包解压后的文件夹cpFileToDir放于
/fglyc/wpfs17/ 目录下

5. 在部署的目录cpFileToDir下打开配置文件进行配置

例如:你将 cpFileToDir文件夹放于了 /fglyc/wpfs17/cpFileToDir 则打开配置文件

/fglyc/wpfs17/cpFileToDir/cfg/cfg.cfg 文件进行编辑,此文件主要配置文件夹配置部分,如下的样例配置的是 复制上传软件生成的短期和超短期文件到E文件软件接收处的例子:

```
#####
#           要处理文件目录及文件设置
#
# g_src_dir[0]="                #需要监视的第一个源文件目录
# g_dst_dir[0]="                #如果第一个源文件目录有符合要求文件，则拷贝到此目录(多个目录用|
分隔)
# g_file_name[0]="*.txt|*.xml"    #第一个源文件需要监视的文件（支持通匹符,如果有多个文件类型用|分
隔
# g_basicCondition_sec[0]=30      #设置第一个源文件对应文件多少秒没有出现修改时间变化就移动
#                                #例如30，则表示 g_src_dir[0]目录中的文件出现30秒内修改时间都
没变化则对其进行移动到g_dst_dir[0]目录
#
# 如果有第二个目录需要监视则把上面的变量都新增设置一遍，但[0]需要改成1
# 以此类推，有第n文件目录需要设置则新增上面设置，把[0]改成n-1
#
#
#####
#
#上传软件所在目录
g_dhp_dir="/home/code/Upload_general_V2_YN/build-Upload_JX_102-Desktop-Release"
#e文本接收目录
g_eds_dir="/zfmd/wpfs20/zg"

#上传目录的短期文件
g_src_dir[0]="${g_dhp_dir}/${g_curY}/${g_curm}/${g_curd}/DQ"
g_dst_dir[0]="${g_eds_dir}"
g_file_name[0]="yn_72wind_${g_tomYmd}.rb"
g_basicCondition_sec[0]=7

#上传目录的手动短期文件
g_src_dir[1]="${g_dhp_dir}/${g_curY}/${g_curm}/${g_curd}/hand_DQ"
g_dst_dir[1]="${g_eds_dir}"
g_file_name[1]="yn_72wind_${g_tomYmd}.rb"
g_basicCondition_sec[1]=7

#上传目录的超短期文件
g_src_dir[2]="${g_dhp_dir}/${g_curY}/${g_curm}/${g_curd}/CDQ"
g_dst_dir[2]="${g_eds_dir}"
g_file_name[2]="yn_4Cwind_${g_15mYmd}*.rb"
g_basicCondition_sec[2]=7
```

6. 将cpFileToDir脚本加入自启动中

将cpFileToDir加入自启动中有两种方法:

方法1: 将脚本cpFileToDir.sh加入系统的crontab中; 方法2: 将脚本cpFileToDir.sh加入自启动软件中,但前提是系统中已经部署了自启动软件.

推荐使用第2种方法

假定系统的自启动软件部署在 /fglyc/wpfs20/startup 下,则打开自启动的的配置文件(配置root用户下的软件自启动打开 rcfgRoot.cfg文件,其他用户下的软件自启动打开 rcfg.cfg文件) 添加如下配置项:

```
[cpFileToDir.sh]
logDir=/fglyc/wpfs17/cpFileToDir/ttylog
runPath=/fglyc/wpfs17/cpFileToDir
runPrePara=
runPara=
```

7. 检查cpFileToDir的日志是否有报错及其他情况

自启动配置好后,过几分钟(一般是1分钟后) cpFileToDir.sh脚本就会自动启动,然后查看日志是否正常,是否正常拷贝文件即可.

『其他说明』

1. 脚本不对源目录的文件进行删除操作
2. 脚本拷贝符合条件的源目录文件时执行的是如下的操作

```
# 为举例说明方便, 作如下假定:
#   假定源目录为${src}, 符合要求的文件名为xx.rb
#   假定要拷贝到的目标目录为${dst}
#   假定当前脚本所在目录为${cur}

# 操作序列如下:
cp -a ${src}/xx.rb ${cur}/tmp/xx.rb.tt
mv ${cur}/tmp/xx.rb.tt ${dst}/xx.rb

# 当此文件需要拷贝的目录都拷贝完后
# 进行touch操作, 以此来防止对同一个文件进行多次拷贝
touch ${cur}/tmp/back/xx.rb
```

3. 脚本自动对 脚本所在路径tmp/back 目录下的所有文件, 修改时间超过1天的文件进行删除
4. 脚本的日志文件只有一个即: log/cpFileToDir.log, 但脚本会对其大小进行限制处理(当文件大小超过6MB后, 脚本会删除文件内容(先删除文件开头的旧内容),直到文件大小小于6MB)