### Autowork帮助手册

[**1、 autowork介绍** 2](#_Toc493081975)

[**1.1、autowork来源** 2](#_Toc493081976)

[**1.2、功能简介** 2](#_Toc493081977)

[**2、 autowork操作介绍** 2](#_Toc493081978)

[**2.1、目录结构** 2](#_Toc493081979)

[**2.2、操作界面** 3](#_Toc493081980)

[**2.3、目标区域** 4](#_Toc493081981)

[**2.4、功能区域** 4](#_Toc493081982)

[**2.4.1、windows运维工具** 4](#_Toc493081983)

[**2.4.2、win网络策略检查** 7](#_Toc493081984)

[**2.4.3、win站点部署** 8](#_Toc493081985)

[**2.4.4、win程序更新** 9](#_Toc493081986)

[**2.4.5、win更新回退** 11](#_Toc493081987)

[**2.4.6、linux运维工具** 12](#_Toc493081988)

[**3、 结果区域** 17](#_Toc493081989)

[**4、 存在bug** 17](#_Toc493081990)

[**4.1、服务更新问题** 17](#_Toc493081991)

[**4.2、登陆锁定** 17](#_Toc493081992)

[**4.3、退出告警** 17](#_Toc493081993)

[**4.4、退出日志丢失** 18](#_Toc493081994)

[**4.5、执行时界面卡住** 18](#_Toc493081995)

[**4.6、ip组执行偶有失败** 18](#_Toc493081996)

[**4.7、工具界面卡死** 18](#_Toc493081997)

[**5、 工具使用样例** 18](#_Toc493081998)

[**5.1、远程推送执行脚本1** 18](#_Toc493081999)

[**5.3、远程推送执行脚本2** 19](#_Toc493082000)

[**5.3、批量配置主机python环境** 19](#_Toc493082001)

[**6、 日志说明** 20](#_Toc493082002)

[**7、 附录脚本** 20](#_Toc493082003)



欢迎各位使用者提供bug信息和改进建议

1. **autowork介绍**

**1.1、autowork来源**

在日常的运维工作中，我们会写很多运维脚本，但日积月累的脚本渐渐多起来，有时候要完成一些任务，需花一点时间把对应脚本找出来，而且脚本执行也相对比较麻烦。一些不太熟悉命令的运维人员经常需要通过界面来进行windows系统的运维操作，不仅繁琐，效率也比较低。为了解决日常运维工作中的这些问题，我想到了将各类常用的运维脚本集中化管理，并通过简单的操作界面来实现自动化运维，于是autowork诞生了。

在这里要向各位运维人员推荐一个学习powershell的好地方：http://www.pstips.net。在我学习powershell过程中，它给了我很大帮助，饮水思源，在这里我也帮忙推广一下。

**1.2、功能简介**

autowork自动化运维工具能实现服务器集中化管理操作，方便运维人员对大批量的服务器进行远程操作，自动化部署web站点，自动化上线等功能，用户所有操作记录均会写入日志文件，提供日志审计排障功能。

目前工具只能单用户操作，当有人在使用时，其他用户启动程序会提示“已有用户登录”并退出。因为工具绝大部分功能都是用powershell实现，运行本程序的主机需要先开启本机powershell脚本执行策略，命令：set-executionpolicy remotesigned -force。被操作的远程windows主机只需要能连通445端口即可，被操作的远程linux主机需要能连通22端口。

autowork的权限是登陆用户的权限，推荐在域环境下使用，如果在工作组环境下，需要将远程被操作主机的window凭据添加到运行autowork的主机，具体操作为：控制面板—凭据管理器—添加 Windows 凭据，测试windows环境为windows 2008 R2，linux环境为centos 6.6。



1. **autowork操作介绍**

**2.1、目录结构**

autowork目录下有scripts、log、temp、config和resource这几个目录，各目录说明如下：

Scripts目录：存放各类需要调用的脚本文件。

Temp目录：各种操作生成的临时文件，其中有一个比较重要的登陆锁文件，该文件的作用具体见bug部分。

Config目录：存放各种配置文件。

Resource目录：各种资源文件如图标，文档等。

Log目录：记录登陆和退出日志，记录所有操作日志和输出结果日志，日志格式以年月日命名，一天产生一个日志。日志格式样例如下：

18:08:16 qinws logout.

18:09:25 qinws login.

18:09:40 qinws start 10.0.0.192 service sql\*

service sql\* not exist

18:09:45 qinws execute get service

18:09:55 qinws get 10.0.0.192 service

18:10:03 qinws stop 10.0.0.192 service SQLSERVERAGENT

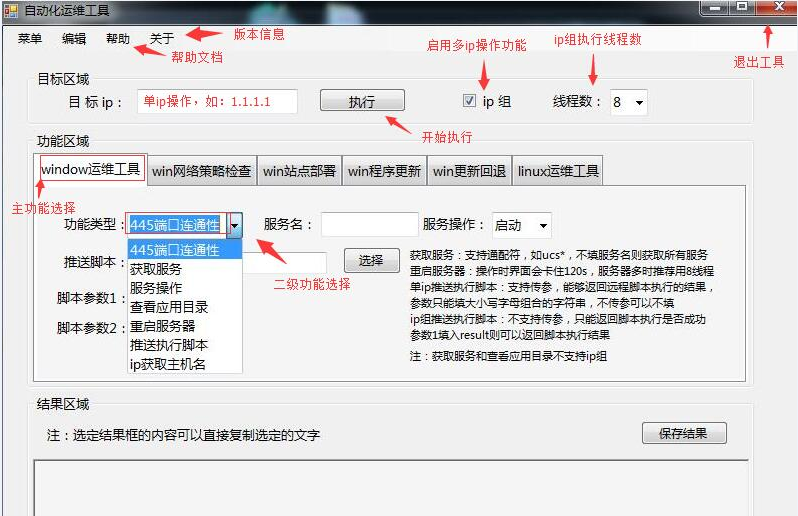
SQL Server 代理 (MSSQLSERVER) 服务正在停止.

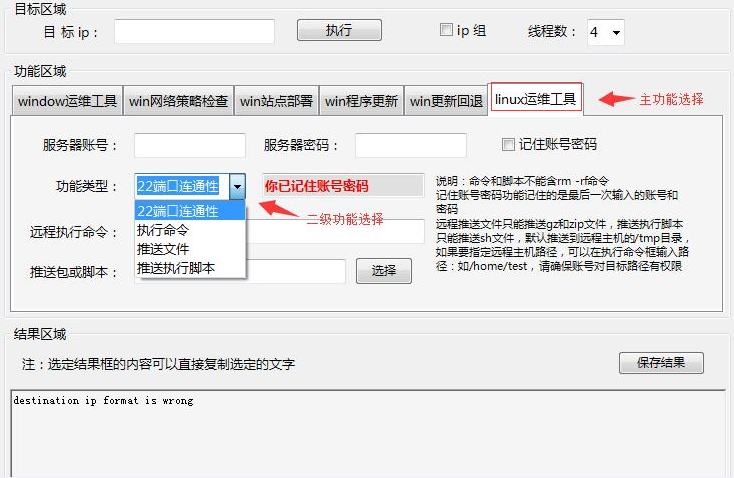
SQL Server 代理 (MSSQLSERVER) 服务已成功停止。

**2.2、操作界面**

autowork操作界面分为三大区域：目标区域、功能区域和结果区域。目标区域有远程要操作的服务器ip，是否启用ip组功能，执行ip组功能时的线程数量。

具体操作主界面如下：





**2.3、目标区域**

目标区域操作如下：

目标ip：指需要远程操作的机器ip，在网络策略检查功能中，它指的是网络策略源ip。

执行：开始对远程机器或者ip组执行命令或功能脚本。

ip组：对多台主机进行操作，先将ip.txt（一行一个ip）放到程序相同目录下后，勾选后生效，默认采用4线程执行。

线程数：采用ip组执行功能时的线程数。

**2.4、功能区域**

功能区域分为windows运维工具、win网络策略检查、win站点部署、win程序更新、win更新回退和linux运维工具几个部分。这个区域主要是选择各种需要执行的功能、资源包和运维脚本，只有linux运维工具是针对linux系统操作，其他都是针对windows系统的功能。

**2.4.1、windows运维工具**

Windows运维工具是针对windows服务器做各种操作，具体说明如下：

445端口连通：检测操作主机到远程主机的端口是否能通。

获取服务：获取远程主机服务运行情况，支持\*通配符，不输入服务名则获取远程主机的全部服务运行情况。

服务操作：对远程主机的服务进行启动、停止、重启操作，也能对远程的服务进行配置，支持配置成启动、手动和禁用状态。

查看应用目录：显示远程主机d盘和e盘的目录和文件，如果盘符根目录有websites和winservice目录，则追加显示这两个目录下的文件和目录。

重启服务器：重启命令发送后，界面卡住120s后，再尝试去获取远程主机的开机时间（获取时间仅适用虚机，物理机一般还没进入系统）。

推送执行脚本：能够将用户编写的powershell脚本推送到远程主机并执行，单ip推送最大支持传2个参数，脚本能返回脚本执行所有结果，ip组推送不支持传参数，脚本默认只能返回脚本执行是否成功。如果要支持返回全部执行结果，可以在参数1框里填入result参数。脚本具体格式请见push\_example文件。

ip获取主机名：通过输入ip获取对应的服务器名。

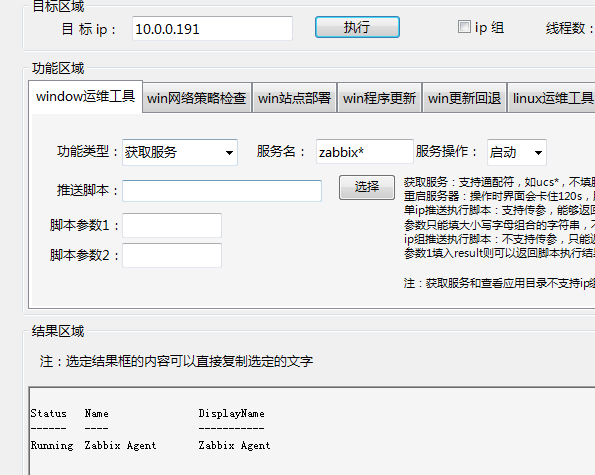
Windows运维工具各项功能需填写内容和操作演示如下：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 功能类型 | 服务名 | 服务操作 | 推送脚本 | 脚本参数1 | 脚本参数2 | ip组 | 多线程 |
| 445端口连通 |  |  |  |  |  | 支持 | 支持 |
| 获取服务 | 选填 |  |  |  |  | 不支持 | 不支持 |
| 服务操作 | 必填 | 启动/停止/重启/自动/手动/禁用 |  |  |  | 支持 | 支持 |
| 查看应用目录 |  |  |  |  |  | 不支持 | 不支持 |
| 重启服务器 |  |  |  |  |  | 支持 | 支持 |
| 推送执行脚本 |  |  | 必填 | 选填 | 选填 | 支持 | 支持 |
| ip获取主机名 |  |  |  |  |  | 支持 | 支持 |

445端口连通检测：



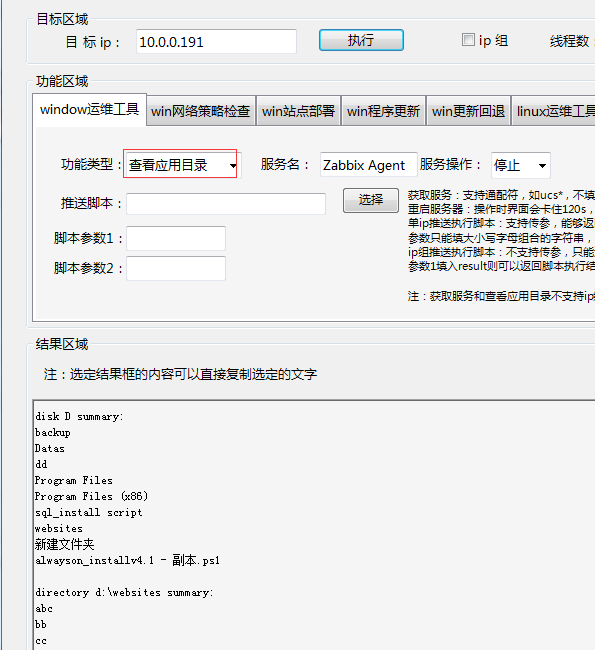
获取服务：



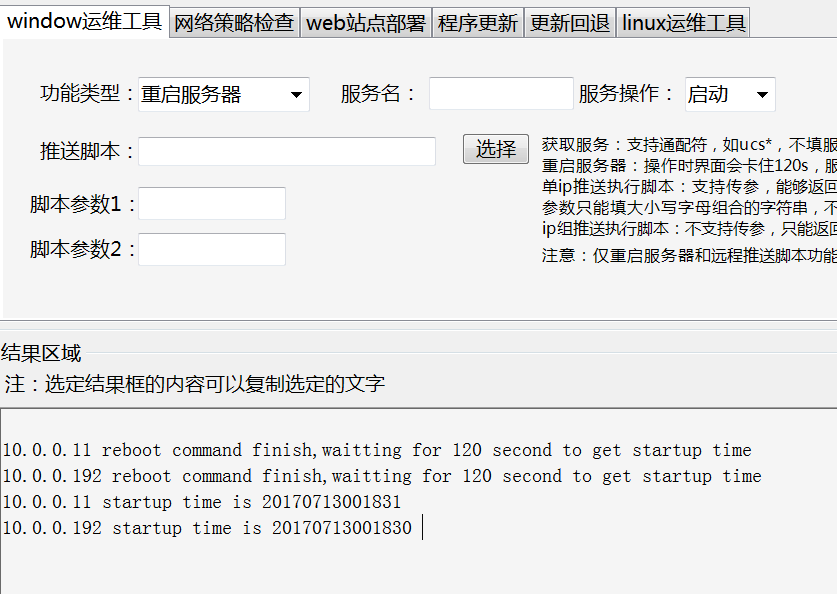
服务操作：



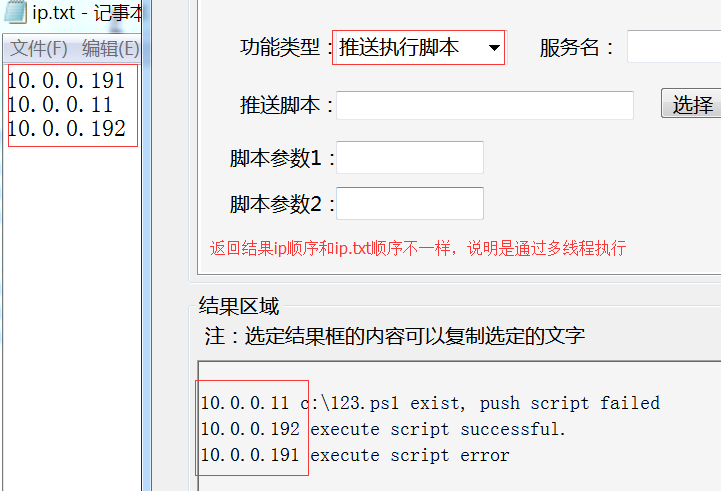
查看应用目录：



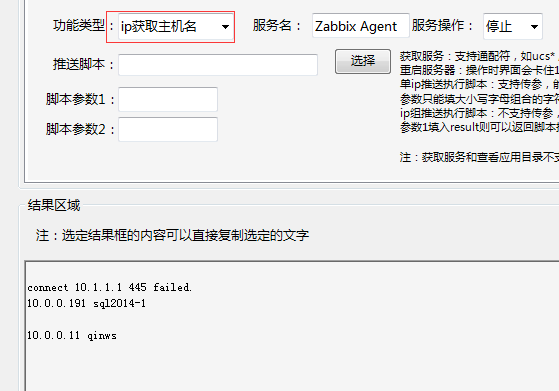
重启服务器：



推送执行脚本:

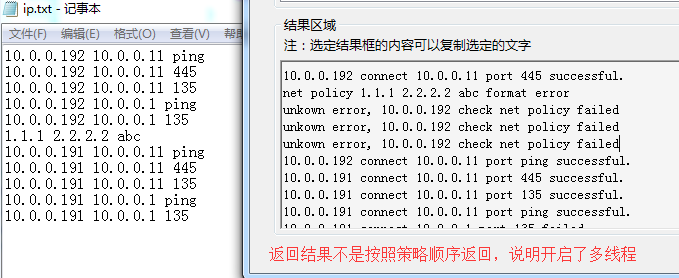


ip获取主机名:



**2.4.2、win网络策略检查**

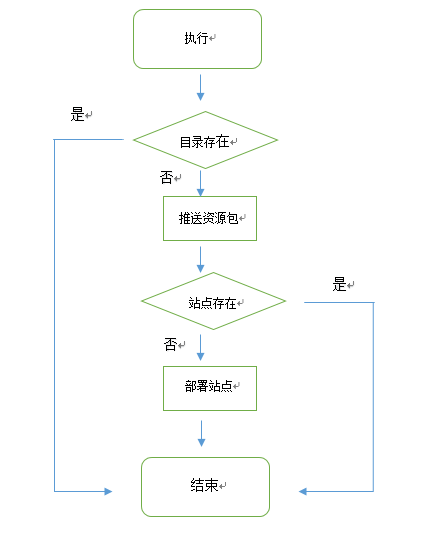
网络策略检查目前只支持windows环境，支持ip组功能，支持多线程。源ip是目标功能区的目标ip，目的ip是功能区的目标ip，服务端口只能填写ping或0-65535间的任意数字。ip.txt内容比较特殊，格式样例：10.0.0.1 172.16.1.1 80（一条策略一行）。操作结果演示：



**2.4.3、win站点部署**

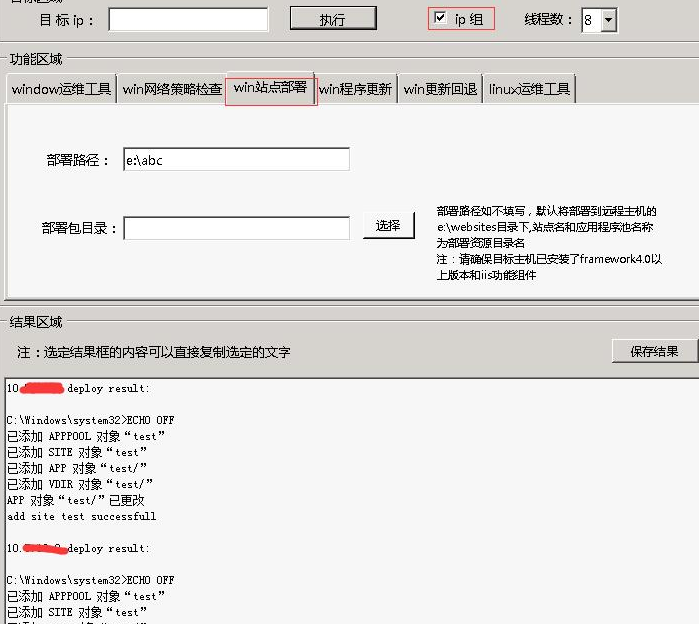
web站点部署实现远程部署站点功能，部署路径是资源包目录（如：abc）需要放在哪个路径下（如e:\websites），则站点程序路径就是e:\websites\abc。部署包目录直接选择资源包目录即可。

部署前请确保目标主机已安装了framework4.0以上版本和iis功能组件。站点名和应用程序池名称为部署资源目录名，应用程序池framework版本为4.0，队列长度为10000，如果需要调整相关参数，请修改scripts目录下的addsite.bat脚本即可。站点部署流程如下：



建站操作结果演示：

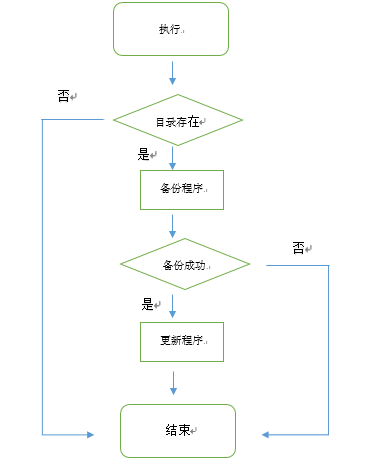




**2.4.4、win程序更新**

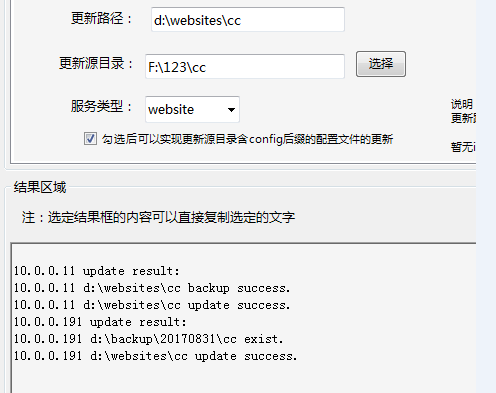
这个功能能够实现自动化上线，如果更新源目录包含config配置文件，请勾选相应勾选框。更新路径是远程主机的站点路径，更新源目录直接选择更新包的目录即可，服务类型选择website是web站点更新，选service是服务更新。website更新会直接替换文件更新；service更新分两种情况，如果服务是运行状态，先将服务停止，再更新程序文件，最后再启动服务；如果服务是停止状态，则直接更新程序文件。在更新程序前会先备份原来的程序目录，程序目录如果小于150M，则直接备份，如果超过150M，则备份时会排除程序目录下的log、logs、tmp、temp和FileTemp\_SFTP这5个目录。150M和5个目录名字可以通过scripts下面的update脚本进行修改，备份路径是和程序相同盘符下的backup目录，一天只保留一个相同备份程序，备份路径样例：e:\backup\20170707\abc。

website更新流程如下：

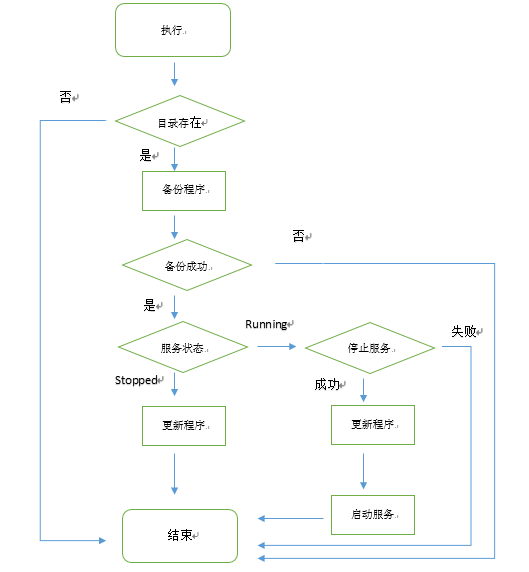


we

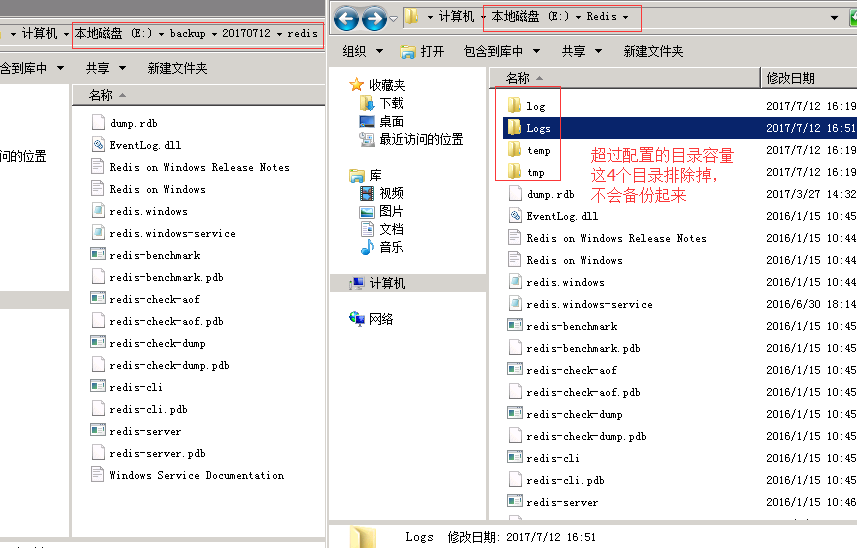
website更新结果演示：

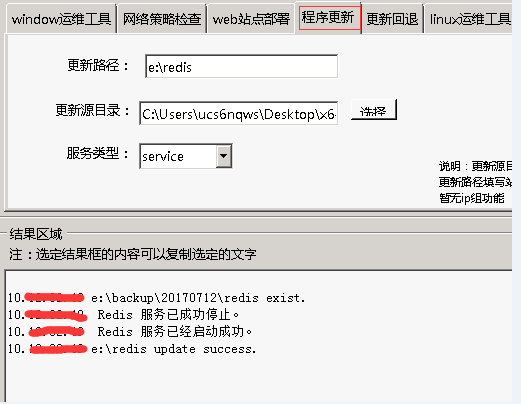


Service更新流程如下（使用service更新前请先了解一下bug，请见4.1章节）：



Service更新结果演示：

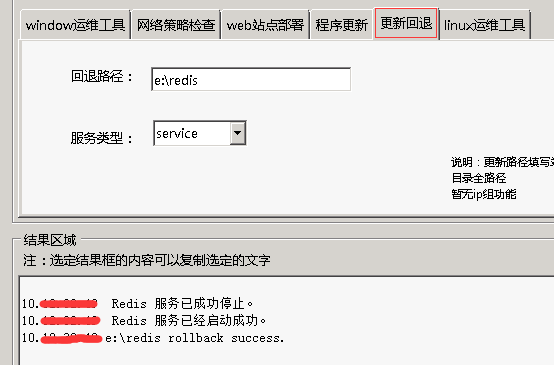


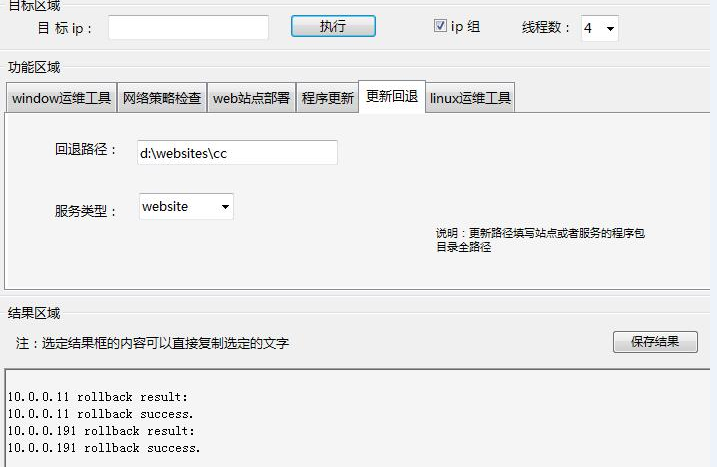




**2.4.5、win更新回退**

这个一般是更新失败后的回退，操作部分和程序更新差不多。服务类型选择website是web站点回退，选service是服务回退。Website回退会直接将备份的文件进行替换；service回退分两种情况，如果服务是运行状态，先将服务停止，再将备份的文件进行替换，最后再启动服务；如果服务是停止状态，则直接将备份的文件进行替换。Service回退结果演示：





**2.4.6、linux运维工具**

linux运维工具是针对linux服务器做各种操作，具体说明如下：

服务器账号：远程服务器的账号。

服务器密码：远程服务器的密码。

记住账号和密码：记录填入的账号密码，相同账号密码的主机再次操作无需输入账号密码。记录账号密码只记录最后输入的账号密码，所以如果服务器密码改变了，只需要再次输入账号密码，勾选“记住账号密码”，执行一次命令即可更新，如果不想要记录的账号密码，删除temp目录下对应用户名文件即可。

22端口连通：检测操作主机到远程主机的端口是否能通。

执行命令：对远程主机执行相应命令，命令不能包含rm –rf，命令支持管道，多条命令分隔等功能。

推送文件：推送本地文件到远程主机，只能推送gz和zip后缀的文件，默认推送到远程主机的/tmp目录，如果要推送到指定目录，请在远程命令框内输入相应路径，如：/home/test。

推送执行脚本：推送本地shell脚本到远程主机并执行，执行后返回脚本所有输出结果，脚本不能含有rm -rf命令。只能推送sh后缀的文件，默认推送到远程主机的/tmp目录，如果要推送到指定目录，请在远程命令框内输入相应路径，如：/home/test。

服务器账号和密码是必填项（如果已经记录了账号密码，则无需填写），单ip操作支持检测密码是否正确，ip组操作则不支持。推送文件或脚本时必须确保输入账号能执行相关命令权限和对相应目录的读写权限。推送脚本样例请见附录脚本部分。

Linux运维工具的ip组多线程功能是通过python的paramiko模块和multiprocessing模块来实现的，如果需要使用该功能，必须安装好python环境和相应的模块。安装相关模块方法如下：

Python安装：下载python2.7版本的安装包进行安装即可，我使用的版本是2.7.13，安装建议采用默认路径：c:\python27

添加python环境变量：执行建议执行附录的addpyenv.ps1脚本（记得要用管理员权限执行），其中$pyenv改成你实际安装python的路径，如果安装路径是默认的c:\python27，就不需要改这个参数。

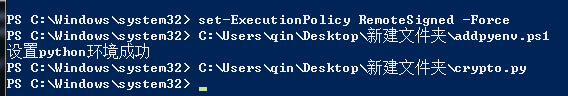
安装pycrypto模块： paramiko依赖这个模块，推荐用编译好的exe文件安装，下载地址：http://www.voidspace.org.uk/python/modules.shtml#pycrypto，把PyCrypto 2.6 的32位和64位的包都下载来安装，安装后用import Crypto命令测试一下是否能安装成功。

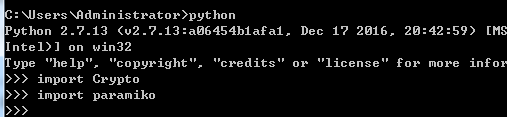
安装paramiko模块：下载地址：https://pypi.python.org/pypi/paramiko/1.7.7.1，进入解压后的文件目录，执行安装命令：python setup.py build 和python setup.py install，安装后用import paramiko命令测试一下是否能安装成功。

如果import出现问题，推荐联网对两个模块升级，具体命令如下：

C:\Python27\Scripts\pip.exe install paramiko –upgrade

C:\Python27\Scripts\pip.exe install cryptography --upgrade





安装可能出现的问题及解决方法：

问题1： 安装pycrypto找不到注册表里的python。

解决：执行附录的crypto.py脚本修复。

问题2：报错import winrandom ImportError: DLL load failed: %1 不是有效的 Win32 应用程序。

解决：无安装pycrypto 32位版本，安装pycrypto-2.6.win32-py2.7.exe即可。

问题3：低版本paramiko在连接高版本openssh会报错：paramiko.SSHException: Incompatible ssh peer (no acceptable kex algorithm)

解决：在用paramiko1.7.7的版本连openssh7.3的服务器时报错，升级paramiko版本即可解决，推荐联网用C:\Python27\Scripts\pip.exe install paramiko –upgrade命令升级。

问题4：报错ImportError: No module named pyasn1.type.univ

解决：同问题3一样的解决方法。

问题5：ImportError: No module named cryptography.hazmat.backends

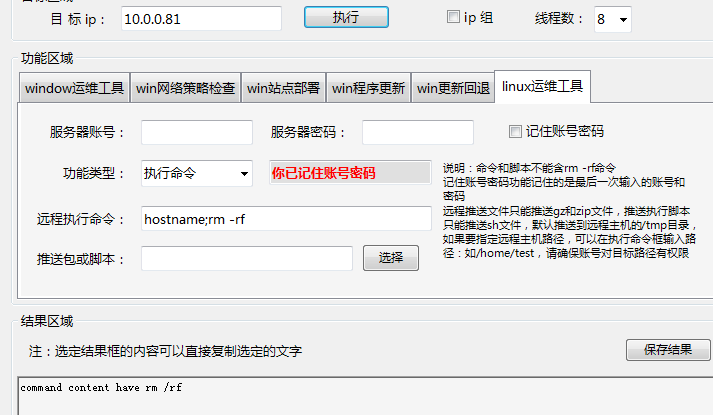
解决：cryptography 模块没装好，重安装一下，推荐联网用C:\Python27\Scripts\pip.exe install cryptography --upgrade命令升级。

其他：由于各种不同的环境都安装这两个模块出现的问题也不一定相同，如果出现一些奇怪的原因，只能自己去寻求谷歌大神了。

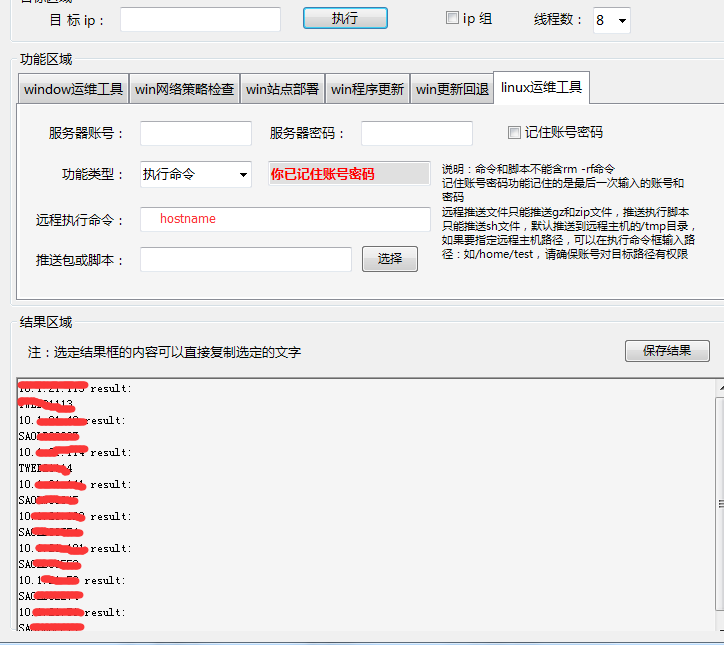
Linux22端口检测结果演示：



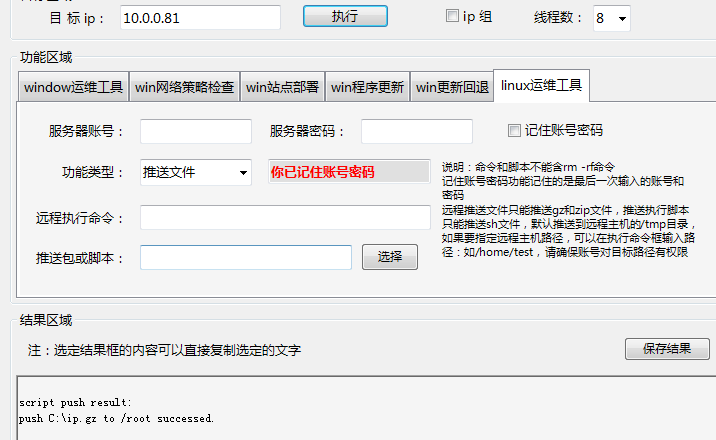
Linux命令操作结果演示：

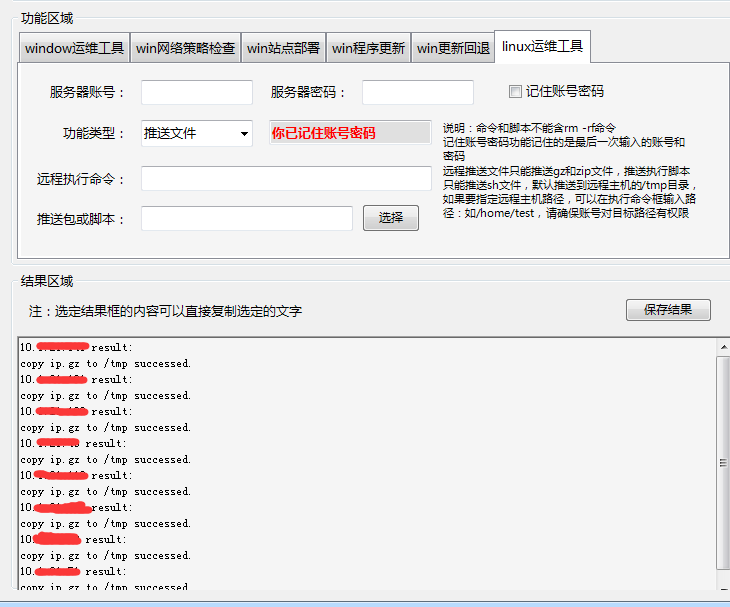




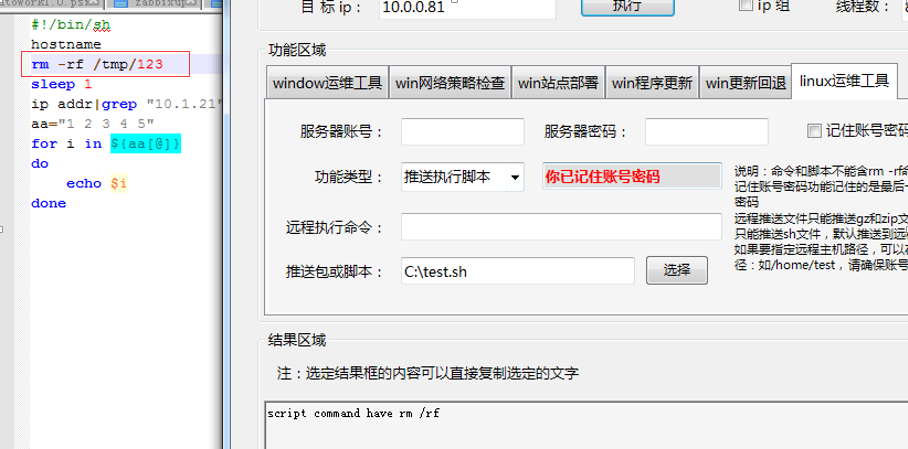


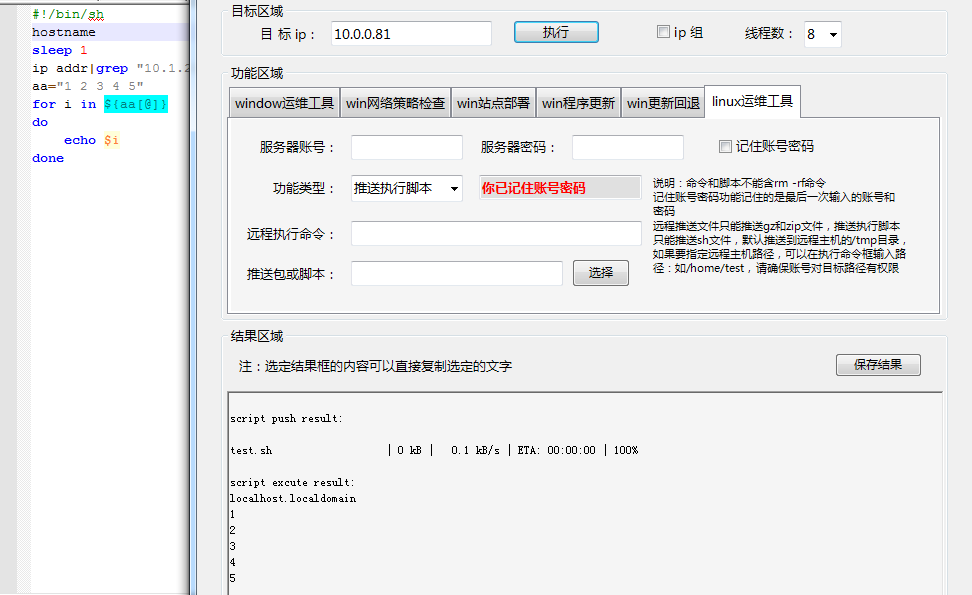
Linux推送文件包结果演示：





Linux脚本推送结果演示：







1. **结果区域**

结果区域显示每次执行完操作的结果，内容框支持选定复制（直接选定内容，在别的地方粘贴即可），每次点击执行按钮都会清空结果区域内容，对于比较重要的操作结果，可以将其保存起来，点击保存结果按钮，选择保存路径即可，结果的存储格式为text文件。对大量服务器操作时（比如数百台服务器），用默认功能来做可能输出信息太多，推荐用远程推送执行脚本的方法来做。

1. **存在bug**

**4.1、服务更新问题**

这个非常重要，进行服务更新之前需要先了解这个bug会造成的影响。因为服务更新是通过更新路径去服务列表里匹配从而找到服务名，如果在同级目录存在类似的目录名，而它也是注册了服务，如：目录一是redis，目录二是redis6380，这样如果更新成功的话会出现一个问题，它会导致两个目录的服务都被更新，而其中一个被更新的服务是没有备份文件的。

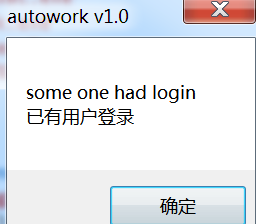
解决方法：由于考虑到service服务程序目录的同级目录出现类似目录名的情况极少，所以暂不考虑修复。如有这种情况的环境，可以通过在填写更新路径时，将服务名填上，在scripts的update脚本将$d\_path拆分为更新程序全路径和服务名两个变量，然后修改update脚本执行流程来解决。



**4.2、登陆锁定**

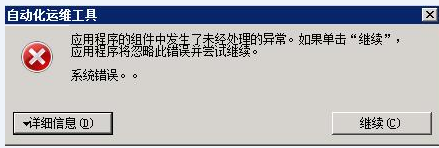
由于工具只能单用户操作，当有用户登陆时，在temp目录会生成一个login\_lock锁文件，防止其他用户登陆操作，当用户正常退出时，会释放这个文件。当用户处于登陆状态时，工具软件崩溃或者非法关闭，这个锁文件是无法释放的，导致用户登录时提示“已有用户登录”。

解决方法：如果确认没有其他用户在使用，人工去将锁文件删除即可正常登陆。



**4.3、退出告警**

在点击退出工具后，在弹出的确认框点击“否”，这时会弹出错误告警框，这是为了采用系统的控件，方便控制工具缩放和退出，没有采用窗体的Form.Close方法来退出，而是直接调用系统的退出控件，点击“否”后，会强制中断退出过程导致窗体告警，这个并不会造成任何影响，直接点继续或叉掉告警窗口，继续操作会有任何影响。



**4.4、退出日志丢失**

这个也是工具软件崩溃或者非法关闭导致，必须正常退出时才会用户的logout日志。

解决方法：暂无解决方法。

**4.5、执行时界面卡住**

这个是正常现象，尤其对大量服务器操作时，在测试中试过用60MB的更新包8线程对80台服务器进行更新，耗时约30分钟，也就是界面要卡住30分钟。

解决方法：暂无解决方法。

**4.6、ip组执行偶有失败**

这种情况主要是对大量服务器做操作时，偶尔有主机会存在这种情况，在测试中尝试用推送执行脚本功能对600+的服务器进行更新组策略和重启zabbix agent的操作，大概会有10台左右机器出现失败的情况。

解决方法：这种情况可以将结果保存起来，再将失败的主机过滤出来尝试重新执行一遍。过滤结果的命令如下（c:\result.txt是保存的执行结果文件）：

Get-Content c:\result.txt|Select-String "failed|error"|% {(([string]$\_).split())[0]} > c:\ip.txt



**4.7、工具界面卡死**

这个一般是在远程推送执行脚本，推送的脚本本身有问题，导致执行一直没有结束，无法返回结果，导致工具界面卡死。一旦出现这种情况，只能通过结束autowork进程来结束运维工具，并将登陆锁定文件删除，删除登录锁定文件请看4.2。

解决方法：这种情况针对linux服务器的推送执行脚本可以通过修改paramiko模块的代码来设置超时时间（具体方法请自行找度娘），windows暂无法解决该问题。

1. **工具使用样例**

在运维工具的功能中，远程推送执行脚本应该是会经常用到的，因为它是推powershell脚本去远程机器上本地执行的，理论上啥操作都能做。有时想要实现的功能运维工具并没有提供，这时可以考虑用它通过几种功能的组合来实现。

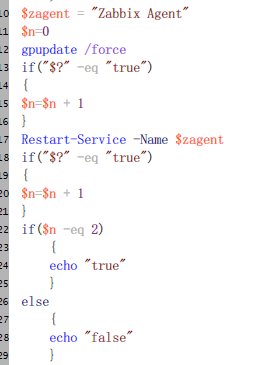
**5.1、远程推送执行脚本1**

推送执行脚本并且不返回输出结果的方式可以让结果看起来非常简洁，推荐在对大量服务器操作时使用这个功能。



**5.3、远程推送执行脚本2**

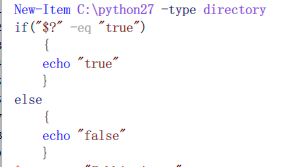
通过推送执行脚本来更新远程主机的组策略和重启zabbix agent，实际测试中用8线程对600+的服务器操作大概要耗时40分钟。推送脚本内容如下：



**5.3、批量配置主机python环境**

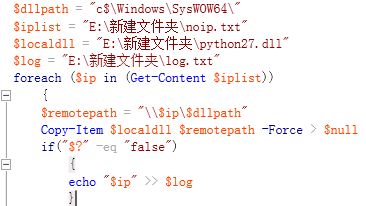
工具本身并没有这个功能，我们可以先通过远程推送执行脚本建立一个目录，然后用websites更新的方法来实现，这样操作会导致相关磁盘存在一个backup的无用目录，它备份的是刚刚建立起来的空目录。以配置python27的环境为例：

（1）将建立目录的脚本推送到ip组执行，脚本内容如下：

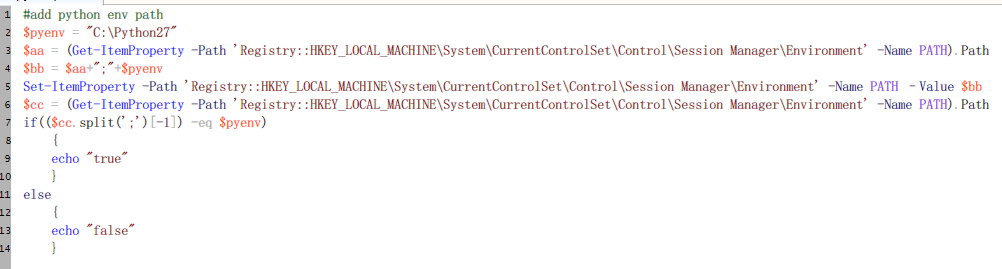


（2）用win程序更新功能的website对ip组进行更新，但拷贝速度比较慢，60MB的更新包8线程对60台服务器进行更新，耗时20多分钟。

（3）执行拷贝python27的dll的脚本，因为只有一个小文件，拷贝速度很快。脚本如下：



（4）将增加python环境变量的脚本推送到ip组执行，脚本内容如下：



1. **日志说明**

以下是对一些重要功能操作记录的日志关键字说明，其中带group的是服务器组操作日志，带single是单机操作。

服务操作：[group] start/stop/restart/set x.x.x.x service

重启服务器：[group] reboot

推送执行脚本: [group] push script

网络策略检查：[singgle|group] net policy

部署站点：[group] deploy website

程序更新：[single|group] update

更新回退：[single|group] rollback

Linux执行命令：[single|group] execute command

Linux推送文件：[single|group] push file

Linux推送执行脚本：[single|group] push script

1. **附录脚本**

推送脚本内容会记录到日志，请勿在脚本内填写账号密码等敏感信息。各种推送脚本样例如下：



组ip功能的ip.txt内容样例如下：



组ip功能网络策略的ip.txt内容样例如下：



添加python环境变量脚本：



安装pycrypto模块可能要用到，将python加入注册表：

