服务

Windows服务-SC

简介

SC 是用来与服务控制管理器和服务进行通信的命令行程序。

```
用法：sc <server> [command] [service name] <option1> <option2>...

语法示例
sc query                - 枚举活动服务和驱动程序的状态
sc query eventlog       - 显示 eventlog 服务的状态
sc queryex eventlog     - 显示 eventlog 服务的扩展状态
```

创建服务

```
sc create 服务名 binpath==执行路径      DisplayName==显示名称

服务名称是Arknights
C:\Windows\system32>sc create Arknights binPath=D:\game\Arknights.exe
DisplayName= Arknights
```

```
C:\Windows\System32>sc create Arknights binPath=D:\game\Arknights.exe DisplayName= Arknights
[SC] CreateService 成功

C:\Windows\System32>
```

查询服务

```
使用query参数
C:\Windows\system32>sc query Arknights

SERVICE_NAME: redis
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 1077  (0x435)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0

使用queryex参数（比query查询多出两个结果）
C:\Windows\system32>sc queryex redis

SERVICE_NAME: redis
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 1077  (0x435)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0
```

```
        PID                : 0
        FLAGS              :
```

```
C:\Windows\System32>sc query Arknights

SERVICE_NAME: Arknights
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 1077  (0x435)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0

C:\Windows\System32>sc queryex Arknights

SERVICE_NAME: Arknights
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 1  STOPPED
        WIN32_EXIT_CODE    : 1077  (0x435)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0
        PID                : 0
        FLAGS              :
```

启动服务

```
sc start 服务名
```

停止服务

```
sc stop 服务名
```

删除服务

```
sc delete 服务名
```

实例

```
sc create bbs binPath= "cmd /K start" type= own type= interact start= demand

sc create bbs: 创建一个名为“bbs”的服务。
binPath= "cmd /K start": 指定服务的可执行文件路径，这里是 cmd /K start，意思是启动一个新的命令行窗口并保持窗口打开。
type= own: 服务运行在它自己的进程中。
type= interact: 允许服务与用户桌面交互。
start= demand: 服务设置为手动启动。

sc create fpx binPath= "cmd /K start" type= own type= interact start= auto

sc create fpx: 创建一个名为“fpx”的服务。
binPath= "cmd /K start": 指定服务的可执行文件路径，这里是 cmd /K start，意思是启动一个新的命令行窗口并保持窗口打开。
```

```
type= own: 服务运行在它自己的进程中。
type= interact: 允许服务与用户桌面交互。
start= auto: 服务设置为自动启动。
```

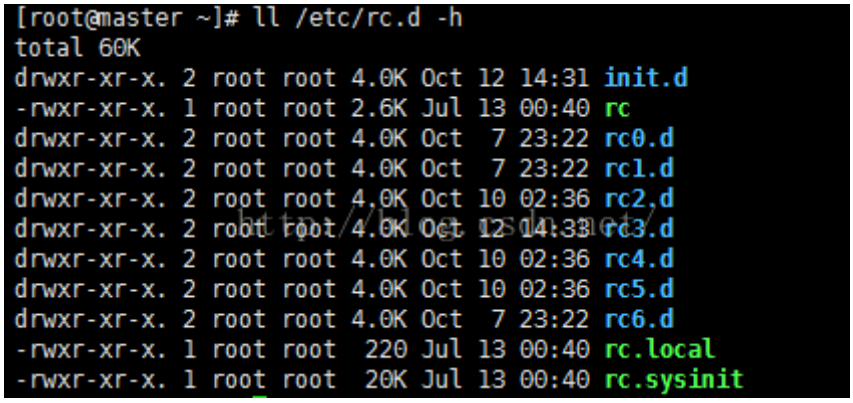
Linux服务注册

Linux添加注册服务的两种方式

```
ln -s                  // 在/etc/rc.d/rc*.d目录中建立/etc/init.d/服务的软链接(*代表0～6七个运行级别之一）
chkconfig              //命令行运行级别设置
```

ln -s 建立软连接启动

在Linux中有7种运行级别（可在/etc/inittab文件设置），每种运行级别分别对应着/etc/rc.d/rc[0~6].d这7个目录



Tips：/etc/rc[0~6].d其实是/etc/rc.d/rc[0~6].d的软连接，主要是为了保持和Unix的兼容性才做此策

这7个目录中，每个目录分别存放着对应运行级别加载时需要关闭或启动的服务

由详细信息可以知道，其实每个脚本文件都对应着/etc/init.d/目录下具体的服务

K开头的脚本文件代表运行级别加载时需要关闭的，S开头的代表需要执行

因此，当我们需要开机启动自己的脚本时，只需要将可执行脚本丢在/etc/init.d目录下，然后在/etc/rc.d/rc*.d中建立软链接即可

我们建立了一个启动项sh：

Timing

```
#!/bin/bash

# 获取当前日期和时间，格式为 年-月-日-时-分-秒
current_time=$(date +"%Y-%m-%d-%H-%M-%S")

# 创建文件路径
file_path="/root/$current_time.txt"

# 写入当前时间到文件中
echo "Current time: $current_time" > "$file_path"

# 确认文件已创建
echo "File created: $file_path"
```

创建后放到/etc/init.d/目录下

建立软连接

```
ln -s /etc/init.d/Timing /etc/rc.d/rc3.d/S100Timing
```

此处Timing是具体服务的脚本文件，S100Timing是其软链接，S开头代表加载时自启动

如果需要在多个运行级别下设置自启动，则需建立多个软链接

这种方式比较繁琐，适用于自定义的服务脚本

chkconfig

如果需要自启动某些服务，只需使用chkconfig 服务名 on即可，若想关闭，将on改为off

```
[root@localhost ~]# chkconfig sshd on
[root@localhost ~]# chkconfig --list sshd
sshd            0:关闭  1:关闭  2:启用  3:启用  4:启用  5:启用  6:关闭
[root@localhost ~]#
```

在默认情况下，chkconfig会自启动2345这四个级别，如果想自定义可以加上--level选项

```
[root@localhost ~]# chkconfig sshd off
[root@localhost ~]# chkconfig --level 35 sshd on
[root@localhost ~]# chkconfig --list sshd
sshd            0:关闭  1:关闭  2:关闭  3:启用  4:关闭  5:启用  6:关闭
[root@localhost ~]#
```

Tips：--list选项可查看指定服务的启动状态，chkconfig不带任何选项则查看所有服务状态

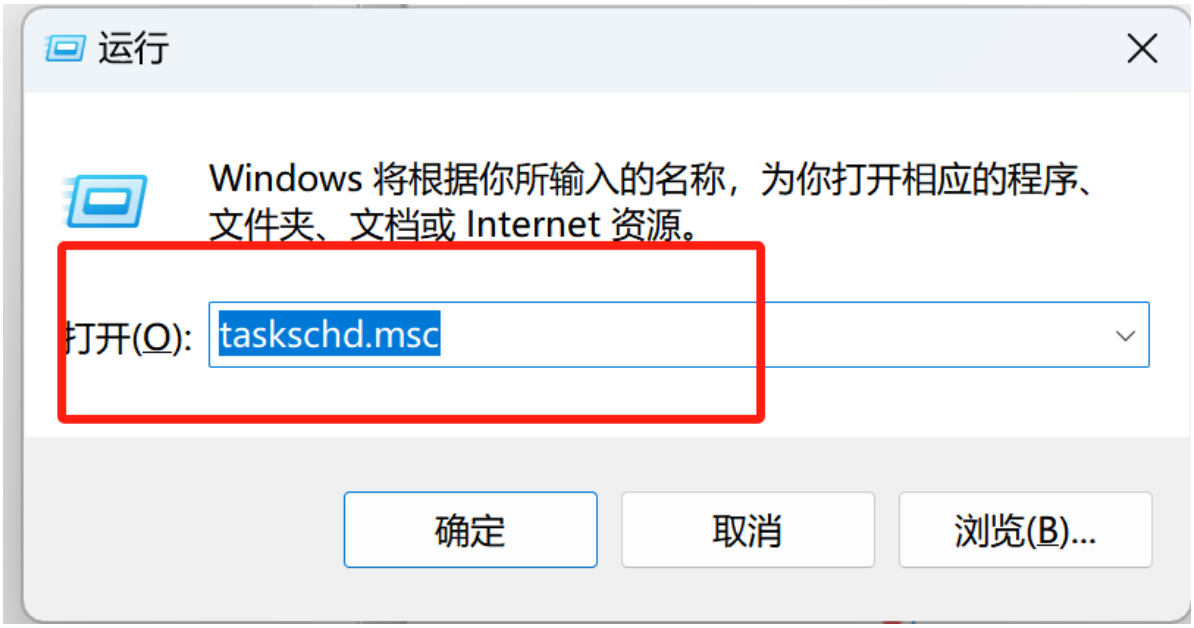
计划任务

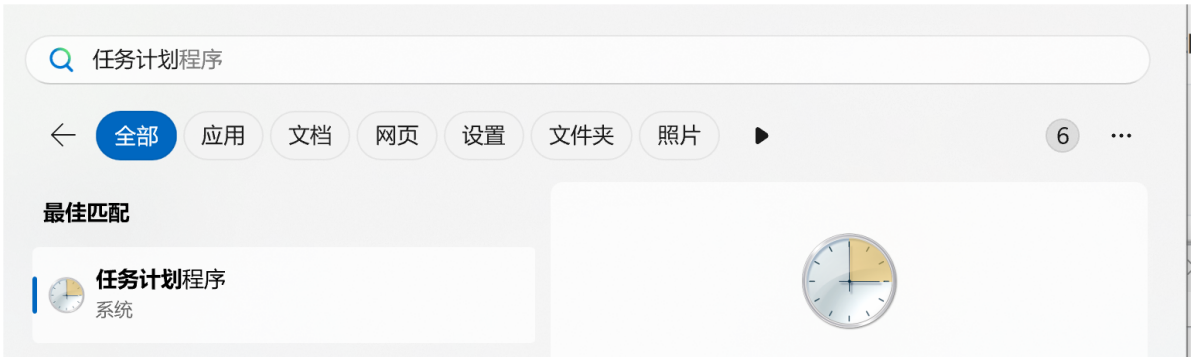
Windows计划任务

简介

任务计划程序是 Windows 内置的工具，可以用于创建、编辑和管理计划任务。可以按照以下步骤查看已设置的计划任务：

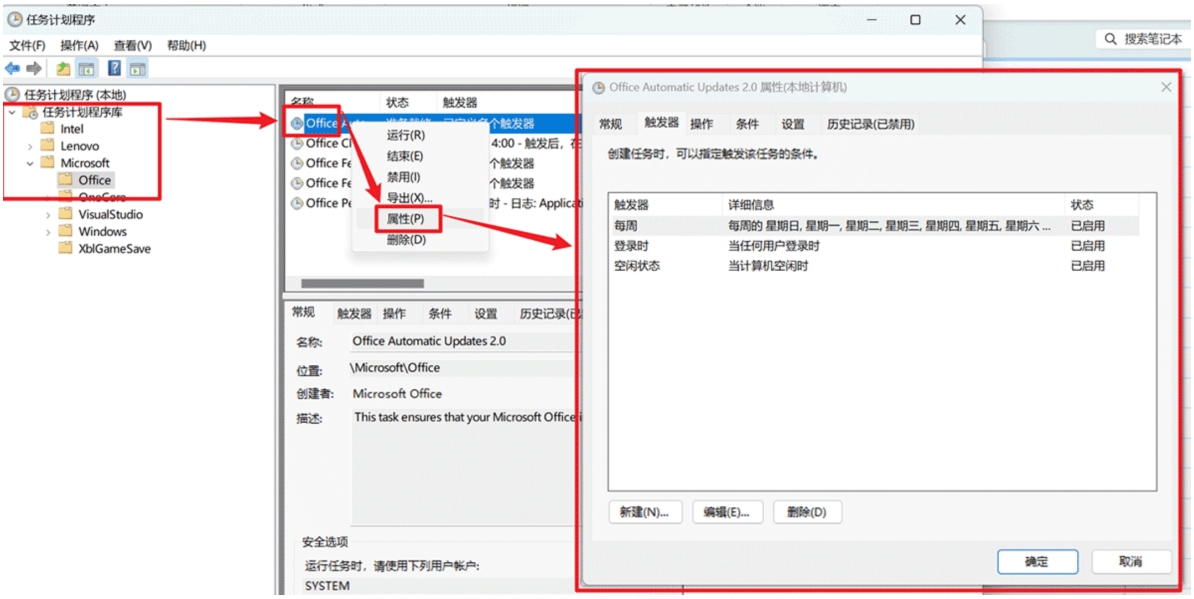
```
1、win+R 执行 taskchd.msc

2、搜索框中输入"任务计划"
```



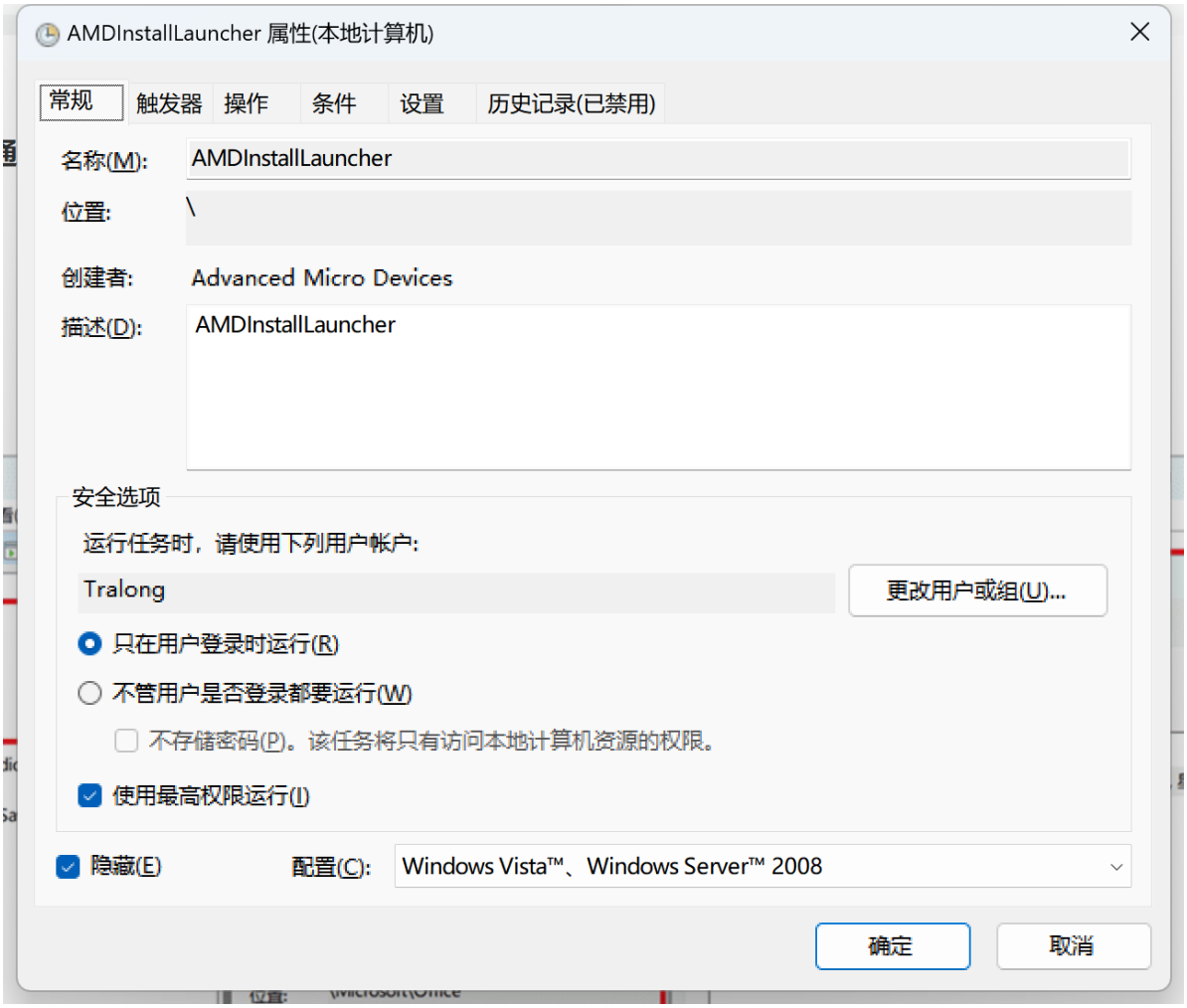


一个计划任务通常由几个内容构成

- **常规**
- **触发器**
- **操作**
- **条件**
- **设置**

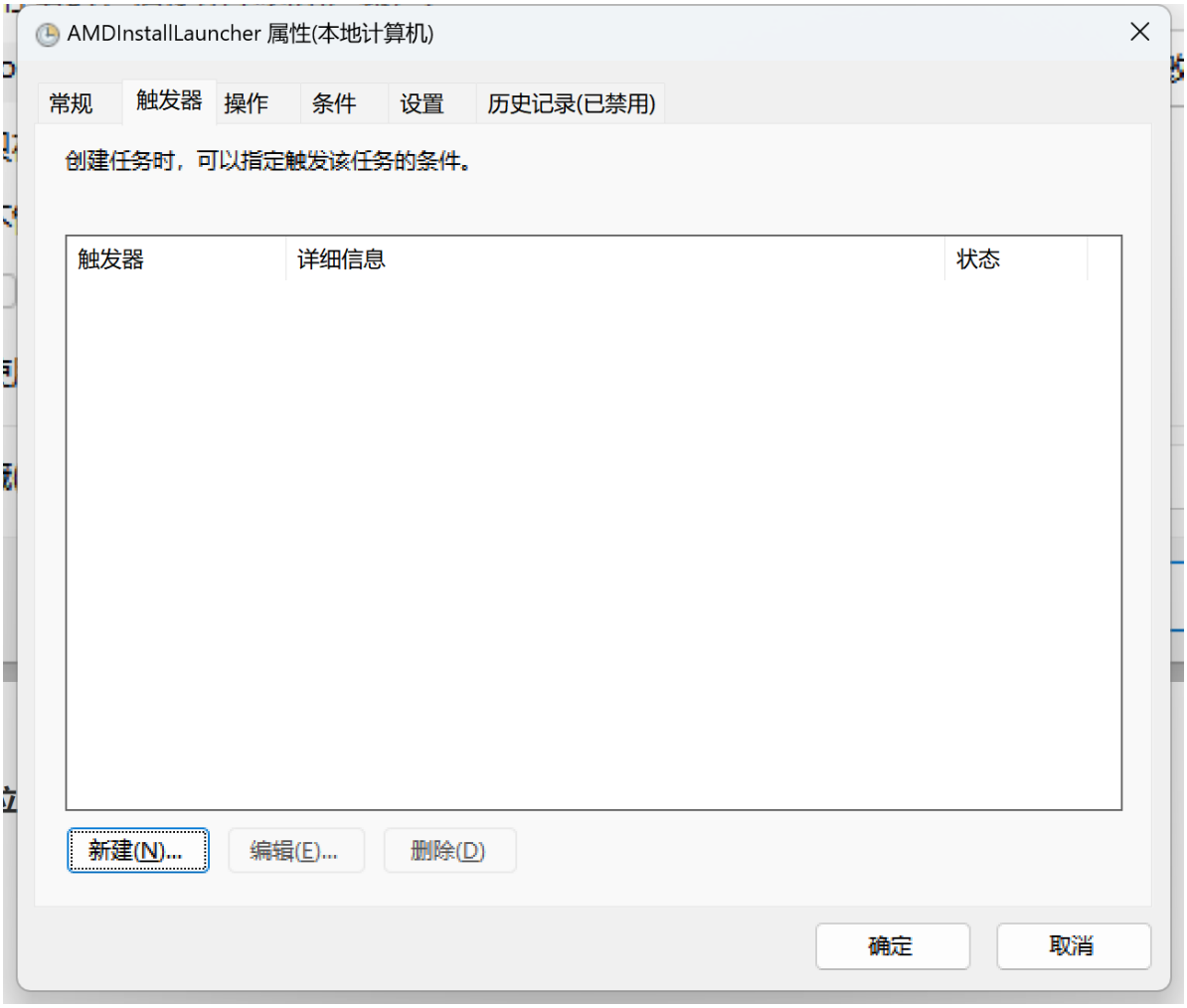


常规有以下内容

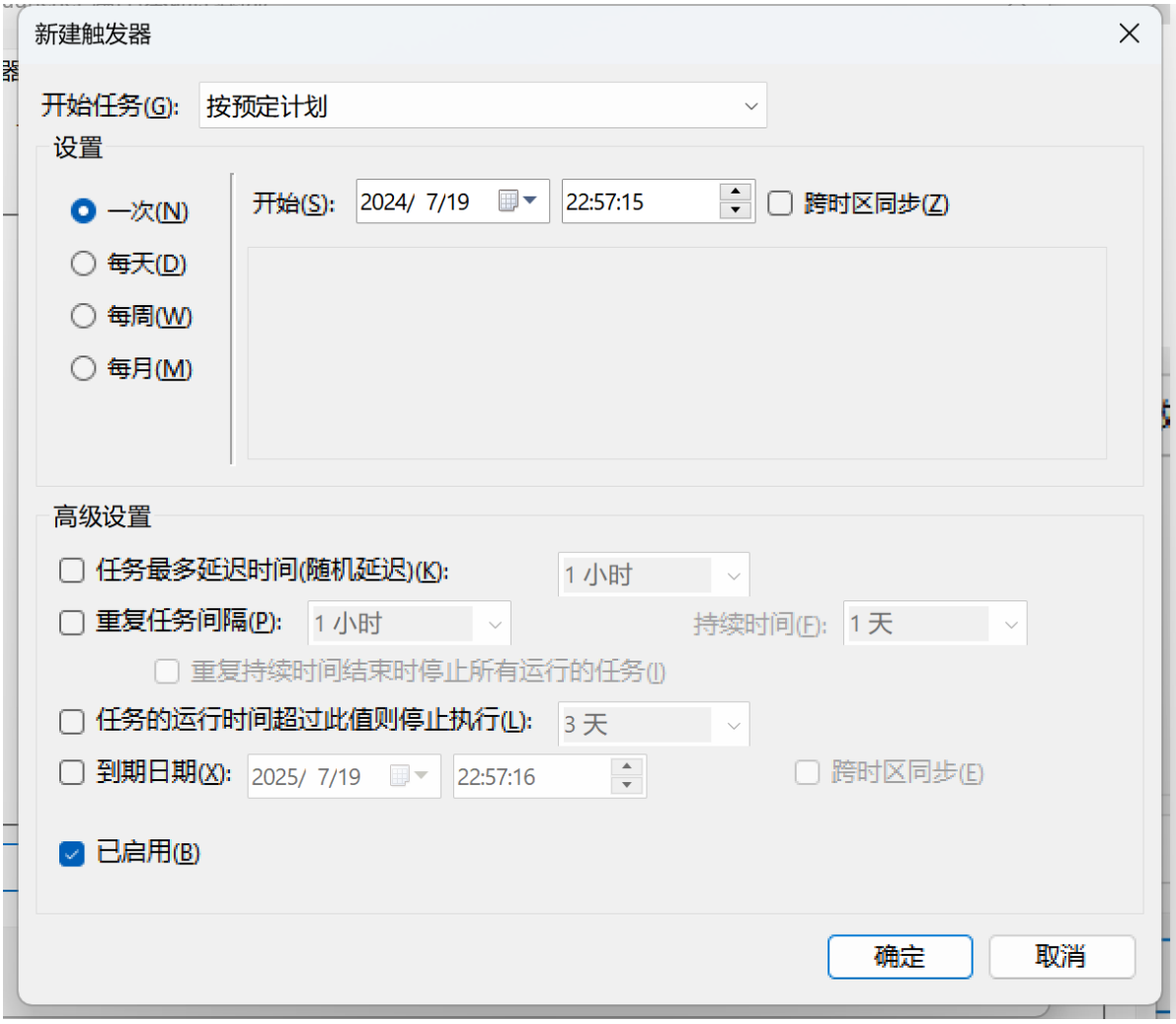


该任务计划的位置、以什么权限运行、在什么条件运行、以什么用户运行

触发器代表在什么情况下，会执行该任务计划，例如每天的8点、每个月的1号此类

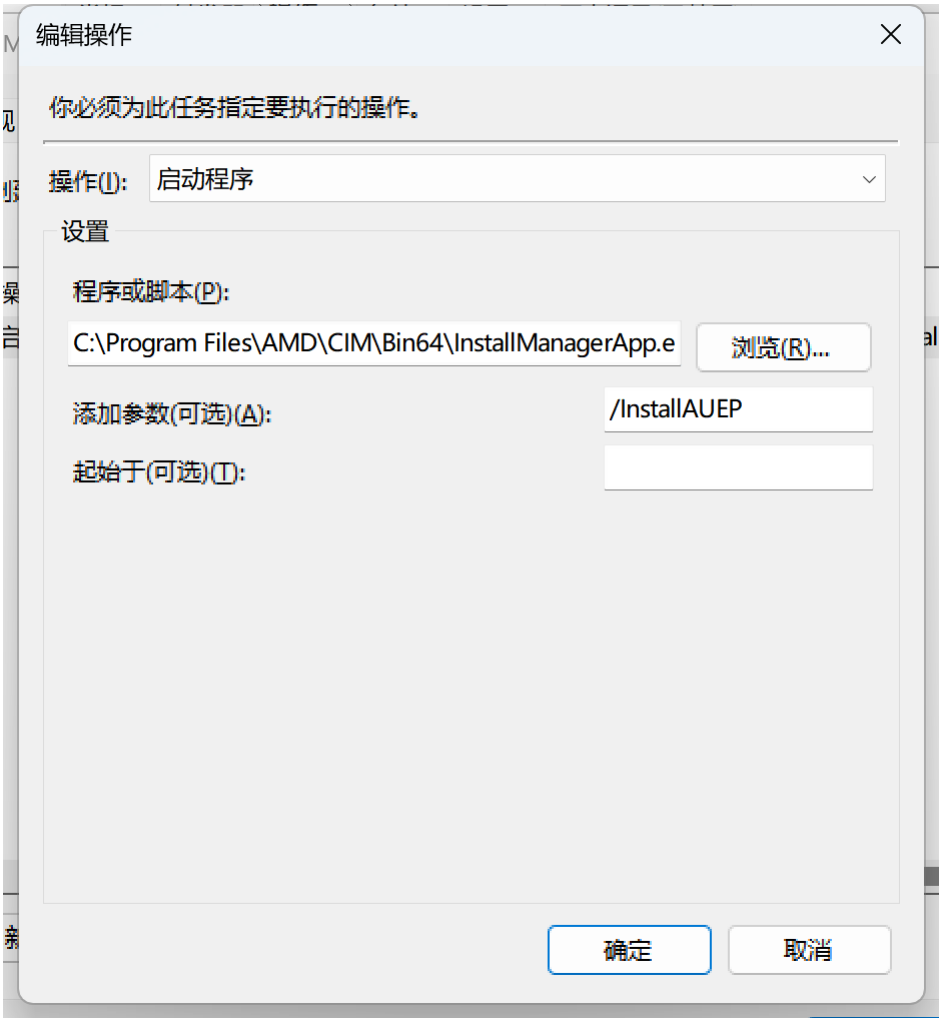


新建触发器，可以看到我们的条件有哪些

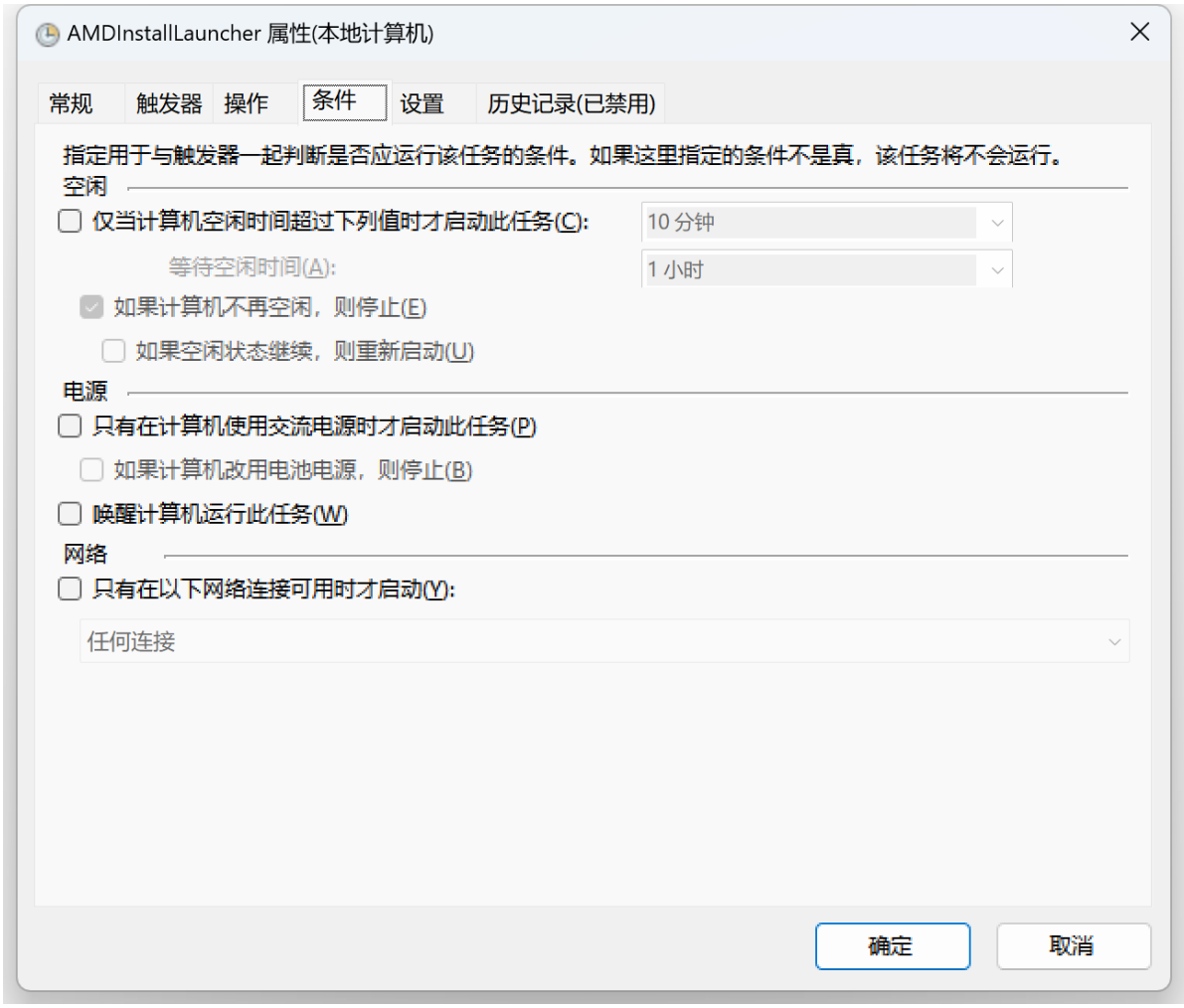


操作，也就是我们该任务计划会做出的动作

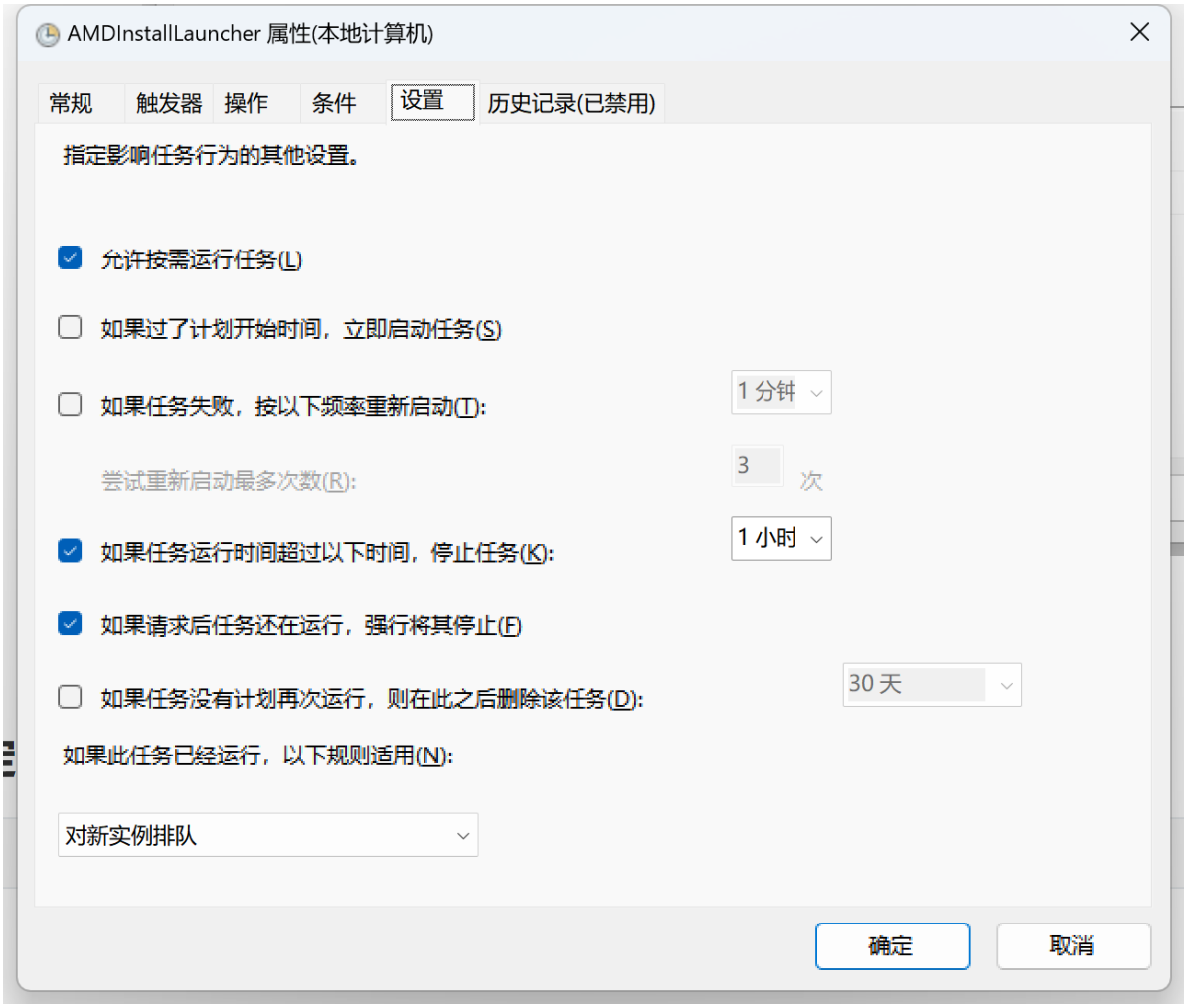
这里触发器的动作就是，执行C盘里指定路径的这个InstallManagerApp.exe 并带上参数 /InstallAUEP



条件，也就是指定用于触发器一起判断是否运行该任务的条件，如果条件这里不满足，就不会运行该任务计划



设置，也就是对任务计划的整体设置



其余方式操作计划任务

```
schtasks /query  //查询计划任务
```

```
C:\Users\Tralong>schtasks /query

文件夹: \
任务名                                  下次运行时间           模式
====================================== ====================== ===============
AMDInstallLauncher                      N/A                    就绪
AMDLinkUpdate                           N/A                    就绪
AMDRyzenMasterSDKTask                   N/A                    正在运行
OneDrive Per-Machine Standalone Update T 2024/7/20 10:33:48     就绪
OneDrive Reporting Task-S-1-5-21-2313981 2024/7/20 10:07:04     就绪
RtkAudUService64_BG                     N/A                    就绪
StartAUEP                               N/A                    正在运行
StartCN                                 N/A                    就绪
StartCNBM                               N/A                    就绪
StartDVR                                N/A                    就绪
User_Feed_Synchronization-{9AEE0FBD-B148 2024/7/20 5:14:14      就绪
WpsUpdateLogonTask_Tralong              N/A                    就绪
WpsUpdateTask_Tralong                   2024/7/20 0:09:14      就绪

文件夹: \Agent Activation Runtime
任务名                                  下次运行时间           模式
====================================== ====================== ===============
S-1-5-21-2313981345-260168748-3831032899 N/A                    已禁用
```

```
schtasks /query /tn "任务名称" /v   //获取指定计划任务详细信息
```

```
C:\Users\Tralong>schtasks /query /tn "StartCN" /v

文件夹: \
主机名          任务名                          下次运行时间           模式             登录状态
 上次运行时间              上次结果 创建者          要运行的任务                                起始于
                        注释                                                        计划任务状态
         空闲时间                              电源管理                                    作为用户运行
                  删除没有计划的任务             如果运行了 X 小时 X 分钟，停止任务       计划
                                                计划类型                  开始时间     开始日期    结束日期    天
                                        月                                    重复: 每                  重复: 截
止: 时间     重复: 截止: 持续时间          重复: 如果还在运行，停止
=============== ====================================== ====================== =============== ========================
 ====================== ============ ================ ========================================== ===============
======================== ========================================================================== =============
========= ====================================== ========================================== ====================
=================== ============================ ================================= ============ ========== ========== ==
======================================== ======================================== ====================== =========
=========== ============================ ==================================
DESKTOP-OM7PKM0  StartCN                         N/A                    就绪           交互方式/后台方式
 2024/7/19 8:50:59              1 Advanced Micro D "C:\Program Files\AMD\CNext\CNext\cncmd.exe" start N/A
                      N/A                                                                       已启用
         已禁用                                                                          Users
                  已禁用                          72:00:00                              计划数据在此格式中不可用。
                                                  登陆时                  N/A          N/A        N/A        N/
A                                       N/A                                    N/A                  N/A
          N/A                             N/A
```

```
schtasks /delete /tn "任务名称" /f    //删除指定计划任务
schtasks /run /tn "任务名称"          //启动指定计划任务
```

创建计划任务

```
/sc    计划任务类型，可选值为MINUTE、HOURLY、DAILY、WEEKLY、ONCE、ONSTART、ONLOGON、ONIDLE、MONTHLY、ONEVENT
       当使用了/sc参数为MINUTE、HOURLY、DAILY、WEEKLY时，我们需要指定/mo运行的间隔时间


/tn    计划任务名称，后续查询、修改、删除、执行时使用
/tr    需要运行的程序或命令，传入的命令中间如果有空格会被截断为程序和参数，因此需要将双引号转义并传入。
/ru    运行任务的用户账户名，不使用此参数的话使用执行schtasks命令的账户运行计划任务
/rp    运行任务的用户账户密码
```

```
/mo    指定任务在计划类型中的运行间隔
/d     指定任务在一个月或者星期的某一天运行，只适用于MONTHLY和WEEKLY类型。
/m     指定任务在某个月运行，只适用于MONTHLY类型。
/i     当计划任务类型为ONIDLE时，运行任务前计算机处于空闲状态的分钟数。
/st    当计划任务类型为MINUTE、HOURLY、DAILY、WEEKLY、MONTHLY时使用，指定任务的开始时间，默认为本地计算机的当前时间。
/ri    指定计划任务的重复间隔，以分钟为单位。不适合计划类型：MINUTE、HOURLY、ONSTART、ONLOGON、ONIDLE
/et    指定计划任务的结束时间，适用于计划类型：MINUTE、HOURLY， 在指定的结束时间之后，schtasks 不会再次启动任务，除非当前系统时间调回开始时间。默认情况下，没有结束时间。
/du    指定任务计划的持续时间，与/et类似，默认情况下没有持续时间。
/k     在指定计划任务的结束时间或持续时间后停止任务，如果不加此参数，则在时间到了会继续运行或者重启该任务。
/it    只在用户登录时运行
/z     在任务计划完成后删除任务计划
/f     在创建任务时如果任务已存在不显示警告
/RL    为作业设置运行级别。有效值为LIMITED 和 HIGHEST。默认值为 LIMITED。
/F     如果指定的任务已经存在，则强制创建任务并抑制警告。

    MINUTE:  1 到 1439 分钟。
    HOURLY:  1 - 23 小时。
    DAILY:   1 到 365 天。
    WEEKLY:  1 到 52 周。
    ONCE:    无修改者。
    ONSTART: 无修改者。
    ONLOGON: 无修改者。
    ONIDLE:  无修改者。
    MONTHLY: 1 到 12，或
    FIRST, SECOND, THIRD, FOURTH, LAST, LASTDAY。
    ONEVENT:  XPath 事件查询字符串。
```

例如

```
schtasks /create /tn "任务名称" /tr "C:\Program Files\MyProgram\myprogram.exe" /sc daily /st 09:00
```

创建一个每天9:00启动myprogram.exe的计划任务，名为任务名称

开机运行

- **无需登录，开机既可(onstart)**

```
schtasks /create /tn test /tr C:\Users\Administrator\Desktop\91.exe /sc onstart /ru                    不运行（计划任务显示准备就绪）
schtasks /create /tn test /tr C:\Users\Administrator\Desktop\91.exe /sc onstart /ru administrator  不运行（计划任务显示准备就绪）
```

- **需要登录，输入密码进入系统后执行(onlogin)**

```
schtasks /create /tn test /tr C:\Users\Administrator\Desktop\91.exe /sc onlogon
如添加计划任务时使用的账号，登录成功后即上线
```

实操

目标1：添加一个任务计划（每隔一分钟执行calc.exe），普通权限和system权限都要

```

```

目标2：创建计划任务"system_update"，触发程序为桌面的91.exe，运行级别为高级别，以system权限每隔三个小时运行一次

```

```

```
schtasks /create /sc minute /mo 1 /tn test /tr C:\WINDOWS\system32\calc.exe

#以system权限运行
schtasks /create /sc minute /mo 1 /tn test /tr C:\WINDOWS\system32\calc.exe /ru
system
```

```
schtasks /create /tn "system_update" /tr C:\Users\Administrator\Desktop\91.exe
/rl highest /F /sc hourly /mo 3 /RU system
```

Linux定时任务

简介

cron 是一个用于在预定时间执行命令的服务。它使用 crontab 来管理定时任务。每个用户都有一个对应的 crontab 文件，允许该用户设置定时任务

1、cron进程是linux中的守护进程，在系统后台运行，它会（默认每分钟）持续地检查 `/etc/crontab` 文件、`/etc/cron.*/` 目录、`/var/spool/cron/` 目录，读取调度任务并执行。

2、所有用户创建的crontab文件都保存在 `/var/spool/cron/` 目录，被cron服务定时检查

查看现有的定时任务

```
crontab -l
```

编辑定时任务

```
crontab -e
这将打开一个文本编辑器，让你编辑定时任务。每行一个定时任务，格式如下：

分钟 小时 日期 月份 星期 命令或脚本
例如，以下是一个在每天凌晨 3 点执行脚本的定时任务的例子：
0 3 * * * /path/to/your/script.sh

: 表示所有可能的值。例如， * * * * 表示每分钟都运行一次。
/n: 表示每 n 个时间单位运行一次。例如，/5 * * * * 表示每 5 分钟运行一次。
n: 表示特定的时间单位。例如，30 3 * * 1 表示在星期一的凌晨 3 点 30 分执行任务。
```

```
    # 每隔 5 分钟运行一次 Genshin_impact 脚本 ##
    */5 * * * * /root/Genshin_impact.sh

    ### 每天的凌晨 1 点运行 Genshin_impact 脚本 ##
    0 1 * * * /root/Genshin_impact.sh

    ### 每月的第一个凌晨 3:15 运行 Genshin_impact 脚本 ##
    15 3 1 * * /root/Genshin_impact.sh

    ### 每个工作日(Mon - Fri) 11:59 p.m 都进行启动。
    59 23 * * 1,2,3,4,5 /root/bin/Genshin_impact
    或者：
    59 23 * * 1-5 /root/bin/backup.sh

    ### 每周六、周日的3点10分执行充值原石
    10 3 * * 0,6 648.sh

    ### 晚上11点到早上8点之间每两个小时，及每天早上八点，输出信息到文件中
    0 23-7/2,8 * * * echo "have a good dream：）" >> /tmp/test.txt

    ### 每个月的4号与每个礼拜的礼拜一到礼拜三的早上11点执行命令
    0 11 4 * 1-3 command
```

删除定时任务

```
crontab -r
```

实操

目标1：添加一个定时任务，要求，每个月15号的17:30-23:59执行一个在/root/下的money.sh

目标2：添加一个定时任务，要求，每个月的1号与每个礼拜的礼拜一早上的8点执行poweroff

综合练习-复习

```
NS Domian Name System 域名系统  将域名转换成IP地址   方便记忆
CDN 内容分发网络 加速资源访问
ipconfig /displaydns
ipconfig /flushdns
ipconfig /registerdns

net user

net user admin
net user admin Esafe666 /add
net localgroup administrators admin /add
net localgroup administrators admin /del
net user admin /del
net localgroup
net user guest Esafe666
net user admin 1qaz!QAZ

administrators | Remote Desktop Users

Remote Destop Protocol 远程桌面协议

netstat -ano | findstr "3389"

TermService #远程桌面连接服务
tasklist /svc | findstr "TermService" //知道目标机器以开启RDP 但是不知道端口
tasklist /svc | findstr "TermService" #定位RDP的进程PID 1508
netstat -ano | findstr "1508" #通过PID定位RDP端口

net time \\192.168.1.1

net view #查看工作组网络内的其他机器名
net share
net session #查看连接本机的会话信息
sc query    #查看当前机器服务情况

net stop wuauserv
net start wuauserv
ping 192.168.179.1
ping -n 10 192.168.179.1
ping -l 65500 192.168.179.1
ping -t 192.168.179.1

ping -4 localhost #将主机名转换成IP地址发送ping数据包

ICMP Internet Control Message Protocol  互联网消息控制协议 检测主机存活

netstat -ano
netstat -r
copy 1.txt 2.txt
dir | cd | mkdir | rmdir | del | ren | type


IIS Internet Information Service
```

```
IT 信息技术 Information Technology

localhost #本机回环地址的主机名 127.0.0.1
0.0.0.0:80  #监听本机所有网卡IP的80端口  可以通过本机的所有网卡IP访问80端口
127.0.0.1:80 #监听127.0.0.1的80端口 仅本地能够访问80端口

默认web目录

日志路径
/var/log/access_log     #正常访问日志
/var/log/error_log      #错误日志
IIS C:\inetpub\wwwroot\     默认web目录

netsh firewall show state                    #查看防火墙状态信息
netsh advfirewall show allprofiles state  #查看防火墙状态信息
netsh firewall set opmode disable/enable #关闭/打开防火墙
netsh advfirewall set allprofiles state on/off #打开/关闭防火墙


netsh advfirewall firewall add rule name=demo action=allow protocol=tcp
localport=3389 dir=in
netsh advfirewall firewall delete rule name=demo
netsh advfirewall firewall add rule name=demo action=block protocol=tcp
localport=3389 dir=in
action = allow|block|bypass
dir = in | out
protocol = tcp|udp

Linux 防火墙
systemctl status firewalld
firewall-cmd --state
systemctl enable/disable firewalld
firewall-cmd --get-active-zones
--add-service={服务名}
--add-port={端口号/协议}
--remove-service={服务名}
--remove-port={端口号/协议}
--reload #重载
--permanent #永久设置
firewall-cmd --zone=public --add-port=80/tcp --permanent
firewall-cmd --reload #重载

firewall-cmd：firewalld命令行工具
--add-port：
--permanent：表示设置为永久
--zone：

firewall-cmd --zone=public --add-port=10-100/tcp --permanent
firewall-cmd --reload #重载
firewall-cmd --zone=public --add-service=http --permanent
firewall-cmd --zone=public --remove-service=http --permanent
```