Windows安全基础下

Windows交互

图形化

RDP的工作原理基于客户端–服务器架构。用户在本地计算机上运行RDP客户端，连接到运行Windows操作系统的远程服务器。通过RDP，用户可以看到远程计算机的桌面，并与之交互，就像直接使用该计算机一样，它的默认端口号是3389。

RDP使用多种网络协议进行数据传输，其中包括TCP/IP。它采用加密机制保护数据传输的安全性，通常使用TLS（传输层安全性）进行加密，确保数据的保密性和完整性。

CMD

在Windows安全中，命令提示符（CMD）是一个强大的工具，允许用户以文本方式与操作系统交互。CMD可以执行多种任务，包括系统配置、文件管理、网络操作等。熟悉CMD的使用对系统管理员和安全专家而言至关重要，因为它可以用来直接访问系统功能并进行安全审计。

使用 `help` 或 `<command> /?` 可以查看命令帮助

常用命令：

命令	功能描述	示例
dir	列出当前目录下的文件和文件夹	dir
cd	更改当前工作目录	cd C:\Users\YourName\Documents
mkdir	创建新目录	mkdir MyFolder
rmdir	删除空目录	rmdir MyFolder
del	删除指定文件	del example.txt
copy	复制文件	copy file.txt D:\Backup\file.txt
move	移动文件	move file.txt D:\Documents\file.txt
ipconfig	显示网络适配器的IP配置	ipconfig
ping	检查与目标IP或域名的连接	ping google.com
tasklist	显示当前运行的进程列表	tasklist
taskkill	结束指定的进程	taskkill /PID 1234
cls	清屏	cls
exit	退出命令提示符	exit
echo	显示一行文本	echo Hello, World!
set	设置或显示环境变量	set PATH

find	在文件中查找指定字符串	find "text" example.txt
type	显示文本文件的内容	type example.txt

Powershell

PowerShell是微软推出的一种强大的命令行界面和脚本语言，专为系统管理和自动化任务设计。与传统的CMD相比，PowerShell不仅支持基本的命令操作，还可以调用.NET框架中的对象和类，实现更高级别的操作。它在Windows管理员和开发者中非常受欢迎，因为它能简化复杂的管理任务，并提供更强的灵活性和自动化支持。

在PowerShell中，变量使用 `$` 符号表示，例如 `$myVar = "Hello"`，这是非常直观的变量赋值方式。PowerShell的命令被称为 "cmdlets"（发音为 *command-lets*），每个cmdlet的命名遵循动词-名词的格式，例如 `Get-Process`，这使得命令更加易于理解和记忆。

PowerShell可以将一系列命令保存为脚本文件（扩展名为 `.ps1`），通过执行该脚本，用户可以实现自动化任务。在 PowerShell 中使用脚本的一种常见方法是将其导入，使用 `Import-Module .\PowerView.ps1` 或者 `. .\PowerView` 即可将其导入。

执行策略

PowerShell的执行策略（Execution Policy）是一种安全功能，用于控制和管理脚本执行的方式。它决定了脚本和配置文件是否可以在系统上运行，以及是否需要满足特定的安全要求。执行策略并不是一个权限模型，而是帮助用户防止不小心运行不受信任的脚本。

常见的执行策略如下：

- **Restricted**
  默认设置。禁止运行任何脚本，只允许交互式命令执行。此策略用于最大限度地保护系统，适合对PowerShell不熟悉的用户。
- **AllSigned**
  允许运行脚本，但所有脚本必须由受信任的发布者签名。运行未经签名的脚本时，系统会阻止，并要求用户进行交互确认。
- **RemoteSigned**
  允许本地创建的脚本运行，但从互联网或其他非本地来源下载的脚本必须经过数字签名。这是一个常见的选择，可以防止运行来自不受信任来源的脚本。
- **Unrestricted**

允许所有脚本运行，无论是否签名。对于从非本地来源下载的脚本，系统会给出安全警告，用户可以决定是否继续执行。

- **Bypass**
  完全绕过执行策略检查。这种策略不推荐在生产环境中使用，因为它完全忽略了执行策略的安全性。
- **Undefined**
  当执行策略设置为 "Undefined" 时，它意味着该作用域没有指定的执行策略。Windows系统默认作用域的策略为 "Restricted"。

可以使用 `Get-ExecutionPolicy` 命令来查看系统的当前执行策略 ，使用 `Get-ExecutionPolicy -List` 列出所有范围的执行策略。

```
PS C:\Users\Say> Get-ExecutionPolicy
Restricted
```

可以使用 `Set-ExecutionPolicy` 命令来更改执行策略，如：

```
Plain Text
1  Set-ExecutionPolicy Bypass -Scope Process
```

在不改变系统执行策略的情况下，可以使用 `-ExecutionPolicy` 参数临时绕过。例如，运行一个脚本时绕过策略：

```
Plain Text
1  powershell -ExecutionPolicy Bypass -File "C:\path\to\script.ps1"
```

用户和网络管理

用户管理

Windows 操作系统的用户管理功能允许管理员创建、修改和删除用户账号，设定权限以及管理用户组。系统中的每个用户都有一个唯一的 **SID（安全标识符）**，用于标识该用户并控制对系统资源的访问。使用 `whoami /user` 可以查看当前用户的SID。

```
C:\Users\any>whoami /user

用户信息
----------------

用户名       SID
============ ============================================
win10-pc\any S-1-5-21-671532499-4235418652-855860158-1001
```

该SID具体解释：

- **S**：代表安全标识符（SID）的前缀，表示这是一个 SID。
- **1**：表示 SID 的修订级别（版本号），通常为 `1` 。
- **5**：代表权威机构， `5` 代表 NT Authority（即 Windows 操作系统）。
- **21**：代表计算机或域 ID，确保唯一性。
- **671532499-4235418652-855860158**：这是一个由多个子部分组成的标识符，用于唯一标识特定的 Windows 安装。这些数字是基于计算机或域的标识符。
- **1001**：表示用户在该系统中的相对标识符（RID，Relative Identifier）。 `1000` 以上的 RID 通常表示这是一个本地用户账户， `1001` 是这个特定用户的唯一 ID。

SID

SID（Security Identifier，安全标识符） 是由一系列数字和标识符组成的结构，确保 Windows 中的每个用户、组或其他安全主体具有唯一的身份。SID 的格式遵循一个特定的结构，分为多个部分。

SID格式如下:

```
S-R-I-SA-SA-SA-RID
```

1. **S**：

- 表示 **Security Identifier** 的前缀。所有 SID 都以 `S` 开头，标识这是一个安全标识符。

2. **R（Revision Level）**：

- 这是 **修订号**，通常为 `1` ，表示当前 SID 的版本号。

3. **I（Identifier Authority）**：

- **标识机构**，用来表明该 SID 由哪个权威机构分配。常见的值包括：
  - `5`：NT Authority，表示 Windows 操作系统自身。
  - `1`：World Authority，表示所有用户（Everyone）。
  - `16`：AppContainer Authority，应用容器标识。

4. **SA（Sub-Authority）**：

- 这是 **子权限**，用于进一步细化 SID，确保在系统或域内唯一标识特定的用户、组或设备。
- SID 中可能会包含多个子权限，具体数量和长度取决于具体环境。这部分通常由多个以连字符分隔的数字组成。
  - **21** 是常见的子权限，用于标识系统或域的独特安装。

5. **RID（Relative Identifier）**：

- **相对标识符**，用于唯一标识用户或组。每个用户或组的 RID 是唯一的，通常是该 SID 的最后一部分。
  - 例如，`1001` 通常是本地用户的 RID。
  - 一些特殊的 RID 值表示系统账户，如：
    - `500`：Administrator 用户
    - `501`：Guest 用户

用户账号管理

Windows 用户账户有三类：

- **本地账户**：仅限于本地计算机的用户，通常用于单机使用。
- **Microsoft账户**：与微软在线服务关联的账户，允许用户通过网络登录多台设备。
- **域账户**：用于企业网络，由域控制器管理，允许用户在多个设备上通过网络访问资源。

在管理本地账户时，常用的命令行工具是 `net user`，如：

```
Plain Text
1  net user username password /add
```

命令行工具 `net localgroup` 可以用于管理用户组。添加用户到管理员组的示例如下：

```
net localgroup administrators username /add
```

使用 `net user` 命令可以列出当前机器的用户，常见的用户如下：

账户名称	功能	账户名标识	特点
Administrator	系统超级管理员账户，具有最高权限，可执行所有系统管理任务。	Administrator，SID 末尾 500	默认禁用，具有最高权限，可安装软件、修改系统配置、管理其他用户等。
Guest	提供有限权限的临时账户，适用于不需要创建个人账户的访客使用。	Guest，SID 末尾 501	权限非常有限，无法安装软件或更改系统设置，默认禁用。
普通用户账户	用户创建的标准账户，具有基本权限，适合日常使用。	用户指定，SID 从 1000 开始递增	可以使用软件、创建和修改个人文件，但不能更改系统级别设置或安装需要管理员权限的软件。
System	操作系统核心账户，用于执行系统级任务，拥有比 Administrator 更高的权限。	System，SID 末尾 18	用户无法直接登录，系统用来执行关键操作，无法删除或禁用。
LocalService	运行不需要网络访问的服务的账户，具有低权限，无法访问网络资源。	LocalService，SID 末尾 19	用于运行低权限的服务，适合本地操作，不适合访问网络资源，能提高系统安全性。
NetworkService	运行需要网络访问的服务的账户，具有较低权限，但可访问网络资源。	NetworkService，SID 末尾 20	适合需要网络连接但不需要本地高权限的服务，权限仍低于 Administrator，降低攻击风险。

网络管理

Windows 提供了 `ipconfig` 命令来查看和管理网络适配器的状态。通过它可以查看IP地址、子网掩码、默认网关等信息。

```
Plain Text
1  ipconfig /all
```

Windows 防火墙是系统安全的关键部分。管理员可以通过图形界面的“Windows Defender 防火墙”来进行防火墙规则的配置，或通过命令行工具 `netsh` 来执行相关操作。

```
Plain Text
1  netsh advfirewall set allprofiles state on
```

常见的网络诊断工具包括 `ping` 、 `tracert` 和 `netstat` ：

- `ping` 用于测试目标主机是否在线：

```
Plain Text
1  ping www.example.com
```

- `tracert` 用于跟踪数据包的路径，诊断网络路径中的瓶颈：

```
Plain Text
1  tracert www.example.com
```

- `netstat` 列出当前网络连接和端口使用情况，帮助管理员了解哪些应用程序正在占用网络资源：

```
Plain Text
1  netstat -an
```

Windows注册表

Windows注册表是一个分层的数据库，存储着系统、硬件、用户设置等信息。它是Windows操作系统的核心组成部分，用于配置系统和应用程序的运行参数。

注册表的基本结构

Windows注册表由多个“根键”（Root Key）组成，根键下面还有不同的子键和值，形成了类似文件夹的分层结构。每个键可以有不同的值，对应系统或软件的不同配置项。

常见的根键包括：

- **HKEY_CLASSES_ROOT (HKCR)**：存储文件关联和类注册信息。
- **HKEY_CURRENT_USER (HKCU)**：包含当前用户的配置信息，比如桌面设置、应用程序配置。
- **HKEY_LOCAL_MACHINE (HKLM)**：包含整个系统的配置，包括硬件、操作系统和软件的设置。
- **HKEY_USERS (HKU)**：存储计算机上所有用户的配置信息。
- **HKEY_CURRENT_CONFIG (HKCC)**：包含当前硬件配置文件的系统信息。

注册表键和值

注册表中每个“键”（Key）类似于文件夹，存储在不同的根键下。键里面包含“值”（Value），它们定义了系统或软件的配置。常见的注册表值类型有：

- **String (REG_SZ)**：一个字符串，通常用来表示路径、名称等。
- **Binary (REG_BINARY)**：二进制数据，存储设备配置等低级信息。
- **DWORD (REG_DWORD)**：32位数值，用于存储简单的数值配置，如开关、权限等。
- **QWORD (REG_QWORD)**：64位数值，用于存储更大的数值。
- **Multi-String (REG_MULTI_SZ)**：多字符串，用于存储多个独立的字符串。
- **Expandable String (REG_EXPAND_SZ)**：带有变量引用的字符串，通常是路径信息，可以根据系统环境变化。

注册表编辑器

Windows提供了一个工具**“注册表编辑器（regedit）”**来查看和编辑注册表。用户可以通过运行“regedit”命令来打开注册表编辑器。

注册表相关命令

- **regedit**：打开注册表编辑器。
- **reg query**：查询注册表键和值。

- **reg add**：添加新的注册表项或值。
- **reg delete**：删除注册表项或值。
- **reg export/import**：导出或导入注册表文件，用于备份或恢复。

示例命令：

- 查看注册表项：

```
Plain Text
1  reg query HKEY_CURRENT_USER\Software\Microsoft
```

- 添加注册表值：

```
Plain Text
1  reg add HKEY_CURRENT_USER\Software\MyApp /v Path /t REG_SZ /d "C:\Program F
   iles\MyApp"
```

- 删除注册表值：

```
Plain Text
1  reg delete HKEY_CURRENT_USER\Software\MyApp /v Path
```

Windows安全

SAM (Security Accounts Manager)

Security Accounts Manager (SAM) 是 Windows 操作系统中的一个重要组件，它用于管理和存储本地用户账户和安全相关的信息。具体来说，SAM 保存了系统中所有本地用户的详细资料，包括用户名、密码的哈希值、以及权限设置等。所有这些数据都存放在一个受保护的数据库文件中，该文件位于 `%SystemRoot%\system32\config\SAM` 路径下。

SAM 的主要作用在于保护和管理用户的登录凭证，它不会直接存储用户密码，而是保存密码的哈希值，哈希值无法直接逆推出密码本身。这种设计确保了即使攻击者获取了 SAM 文件，也无法轻易获取用户的明文密码。

SAM 的安全性非常关键，系统在运行时会加锁该文件，使其无法被复制或直接访问。即使是管理员账户也无法在系统运行过程中直接读取它。然而，在某些特殊情况下（比如通过物理访问或使用

某些漏洞），攻击者可能会尝试绕过这些保护措施获取 SAM 文件，进而进行密码破解。针对 SAM 文件的攻击手段包括提取 NTLM 哈希并尝试暴力破解，或使用现有哈希进行其他攻击。

UAC (User Account Control)

用户账户控制（UAC）是 Windows 中的一项安全功能，旨在防止未经授权的操作对系统做出修改。它通过权限提升提示来增强系统的安全性。当用户或程序尝试对系统级别的设置进行更改时，UAC 会要求确认或提供管理员凭证，从而确保这些更改得到了用户的明确授权。

UAC 的设计原则是减少恶意软件或无意的操作带来的安全威胁。日常操作中，用户一般以标准用户权限运行任务，这样即使某个程序被恶意修改，也很难对系统文件进行更改。只有当需要对系统进行更改时，用户才会通过 UAC 提示授予临时的管理员权限。这种设计最大限度地减少了对系统的风险。

当 UAC 提示出现时，用户必须明确同意或输入管理员密码，才能继续操作。不同的系统级别操作可能会触发不同的提示，包括安装新软件、更改网络设置、修改用户账户信息等。用户可以根据实际需求在控制面板中调整 UAC 的提示频率。

拓展资源

https://learn.microsoft.com/zh-cn/windows-server/administration/windows-commands/windows-commands