

MIS 5170

Operating System Security

Week 3

Windows

- Scripting
- Appropriate permissions

Tonight's Plan

2

- ❑ Questions from Last Week
- ❑ Review on-line posts
- ❑ In The News
- ❑ Scripting
- ❑ Appropriate Permissions
- ❑ Limit Services
- ❑ Shares
- ❑ Assignment 1 - Follow-up questions (Due Feb 7th)
- ❑ Next Week

Caution

3

- ❑ Some tools and techniques discussed and used in this course should only be used on systems you personally own, or have written permission to use.
- ❑ Some of the tools used have the potential to disrupt or break computer systems.

Wade Mackey's Advanced Penetration Testing

Questions from Last Week

4

- Questions?
 - ▣ Any follow-up questions about Snap Shots from last week?
 - Restore Snapshot...
 - Go back to before; remove changes and start recording changes again from this copy
 - Delete Snapshot...
 - Go Forward; delete a way back
 - ▣ General follow-up questions?



Scripting

❑ PowerShell

- ❑ PowerShell is a scripting language and command line interpreter replacement for Windows OSs.
 - On-line definition is – an object-oriented programming language and interactive command line shell for Microsoft Windows. PowerShell was designed to automate system tasks, such as batch processing, and create systems management tools for commonly implemented processes.

❑ Python

- ❑ Python is a interpretive scripting language with support for most OSs, which is helping it become one of the natural choices for Pen Tester.

Scripting (cont)

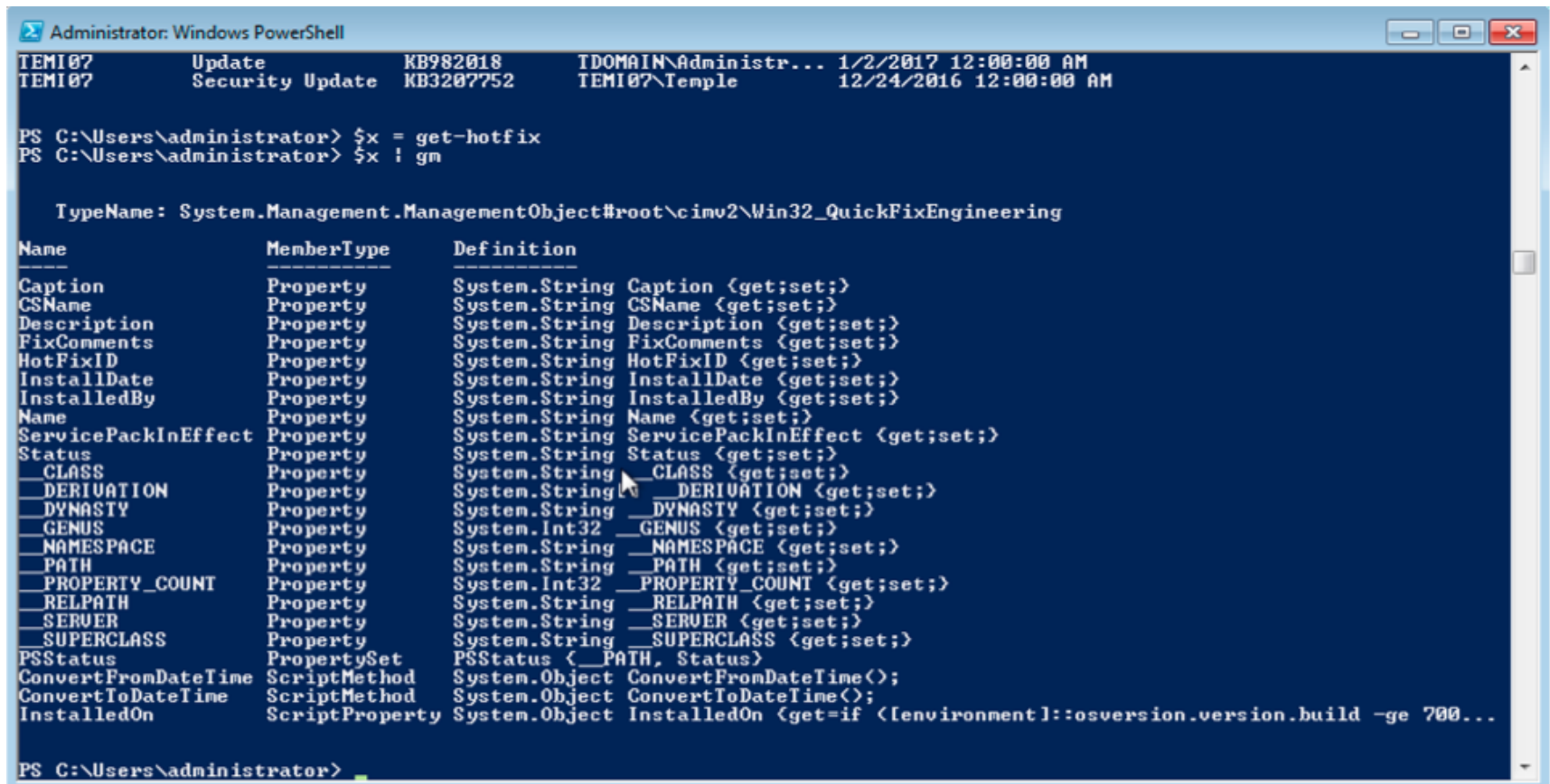
10

- ❑ PowerShell Commands to remember
 - ❑ Help
 - Help get-*
 - Help about_*
 - ❑ Get-Alias
 - ❑ <object> | gm = Get-Member

Scripting (cont)

11

PowerShell Demo



```
Administrator: Windows PowerShell
TEMI07 Update KB982018 TDOMAIN\Administr... 1/2/2017 12:00:00 AM
TEMI07 Security Update KB3207752 TEMI07\Temple 12/24/2016 12:00:00 AM

PS C:\Users\administrator> $x = get-hotfix
PS C:\Users\administrator> $x | gm

TypeName: System.Management.ManagementObject#root\cimv2\Win32_QuickFixEngineering

Name MemberType Definition
-----
Caption Property System.String Caption {get;set;}
CSName Property System.String CSName {get;set;}
Description Property System.String Description {get;set;}
FixComments Property System.String FixComments {get;set;}
HotFixID Property System.String HotFixID {get;set;}
InstallDate Property System.String InstallDate {get;set;}
InstalledBy Property System.String InstalledBy {get;set;}
Name Property System.String Name {get;set;}
ServicePackInEffect Property System.String ServicePackInEffect {get;set;}
Status Property System.String Status {get;set;}
__CLASS Property System.String __CLASS {get;set;}
__DERIVATION Property System.String __DERIVATION {get;set;}
__DYNASTY Property System.String __DYNASTY {get;set;}
__GENUS Property System.Int32 __GENUS {get;set;}
__NAMESPACE Property System.String __NAMESPACE {get;set;}
__PATH Property System.String __PATH {get;set;}
__PROPERTY_COUNT Property System.Int32 __PROPERTY_COUNT {get;set;}
__RELPATH Property System.String __RELPATH {get;set;}
__SERVER Property System.String __SERVER {get;set;}
__SUPERCLASS Property System.String __SUPERCLASS {get;set;}
PSStatus PropertySet PSStatus (__PATH, Status)
ConvertFromDateTime ScriptMethod System.Object ConvertFromDateTime();
ConvertToDateTime ScriptMethod System.Object ConvertToDateTime();
InstalledOn ScriptProperty System.Object InstalledOn {get-if <[environment]::osversion.version.build -ge 700...
```

Scripting (cont)

12

❑ PowerShell program review:

```
$filename = "C:\Windows\WindowsUpdate.log"
```

```
$reader = new-object System.IO.StreamReader(New-Object IO.FileStream($filename,  
[System.IO.FileMode]::Open, [System.IO.FileAccess]::Read, [IO.FileShare]::ReadWrite))
```

```
#start at the end of the file
```

```
$lastMaxOffset = $reader.BaseStream.Length
```


Scripting (cont)

13

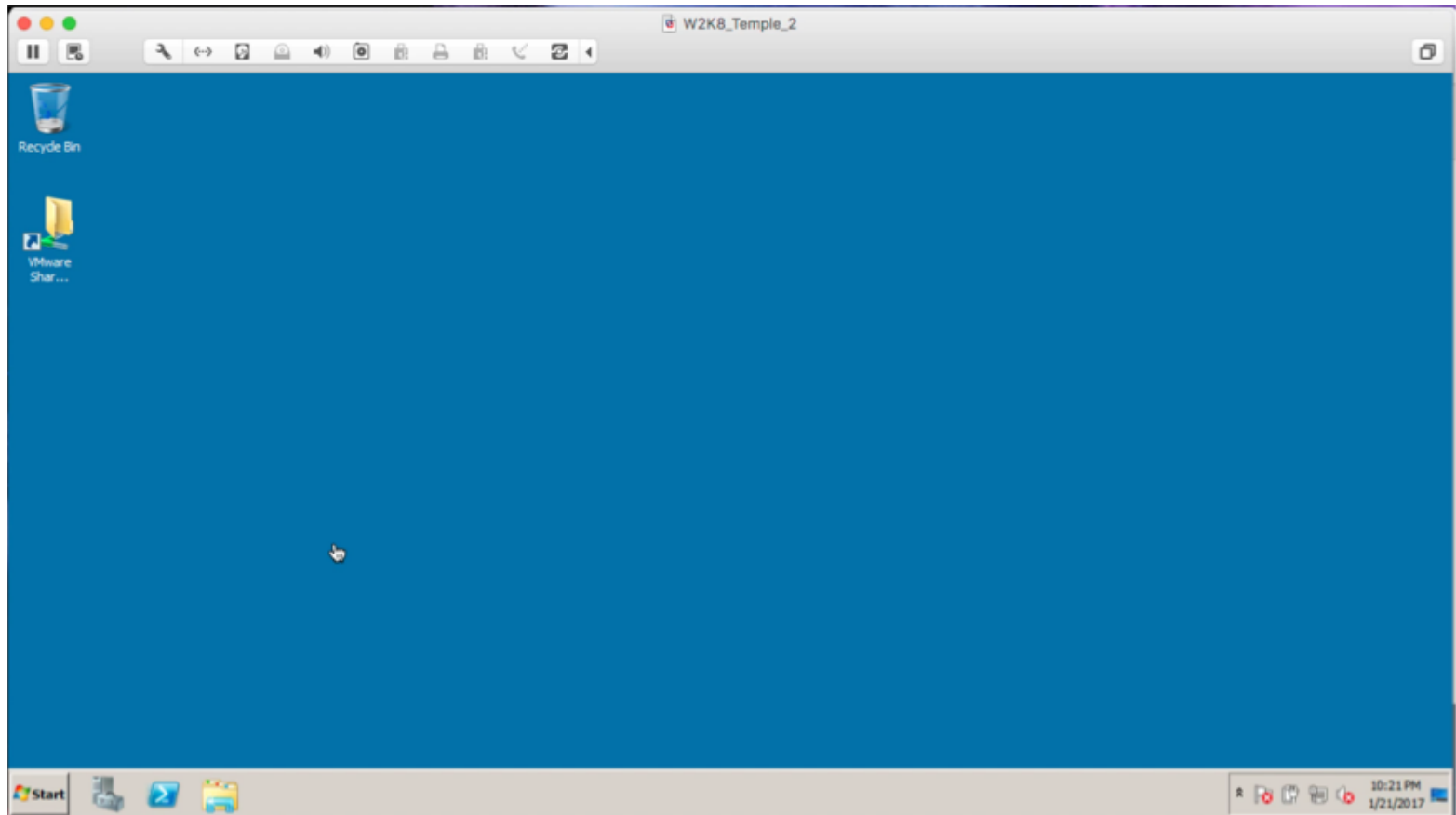
❑ PowerShell program review:

```
while ($true){
    Start-Sleep -m 100
    if ($reader.BaseStream.Length -eq $lastMaxOffset) { #if the file size has not changed, idle
        continue;
    }
    $reader.BaseStream.Seek($lastMaxOffset, [System.IO.SeekOrigin]::Begin) | out-null #seek to the last
max offset
    $line = ""
    while (($line = $reader.ReadLine()) -ne $null) { #read out of the file until the EOF
        write-output $line
    }
    $lastMaxOffset = $reader.BaseStream.Position #update the last max offset
}
```

Scripting (cont)

14

❑ PowerShell Demo



Scripting (cont)

15

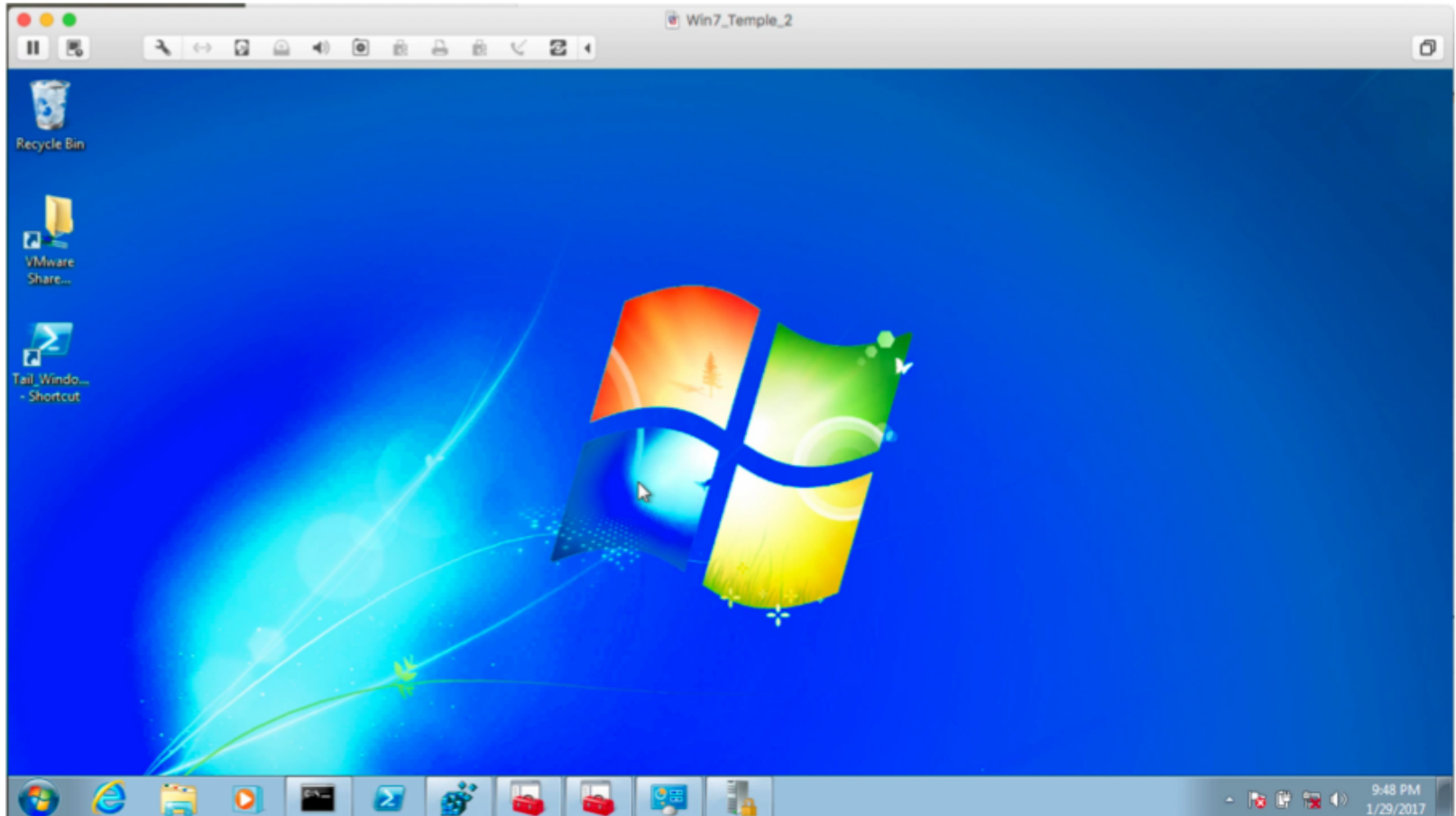
❑ Python (Similar example of Tail Command)

```
import time, os
filename = 'c:\Windows\WindowsUpdate.log'
file = open(filename,'r')
#Find the size of the file and move to the end
st_results = os.stat(filename)
st_size = st_results[6]
file.seek(st_size)
while 1:
    where = file.tell()
    line = file.readline()
    if not line:
        time.sleep(1)
        file.seek(where)
    else:
        print (line), # already has newline
```

Python (cont)

16

□ Tail Example



Scripting (cont)

17

- Questions?

Appropriate Permissions

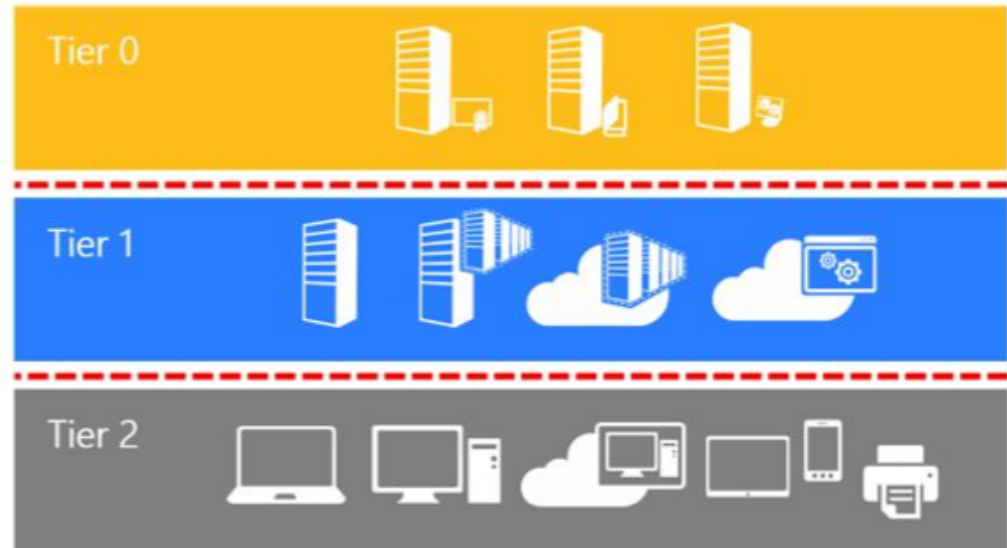
18

- Appropriate permissions
 - ▣ Appropriate permissions is a mind set to define a least privileged set of access. Examples are:
 - Users - should only be get the office products and their files
 - Helpdesk – should be to help users and their own files
 - Server Admins – Should be able to modify servers, but not user files
 - Domain Admins – Should be able to modify AD, but non of the below.

Appropriate Permissions (cont)

19

- Account Tiers
 - ▣ Domain Administrator – Tier 0
 - ▣ Server Admins – Tier 1
 - ▣ User Space – Tier 2



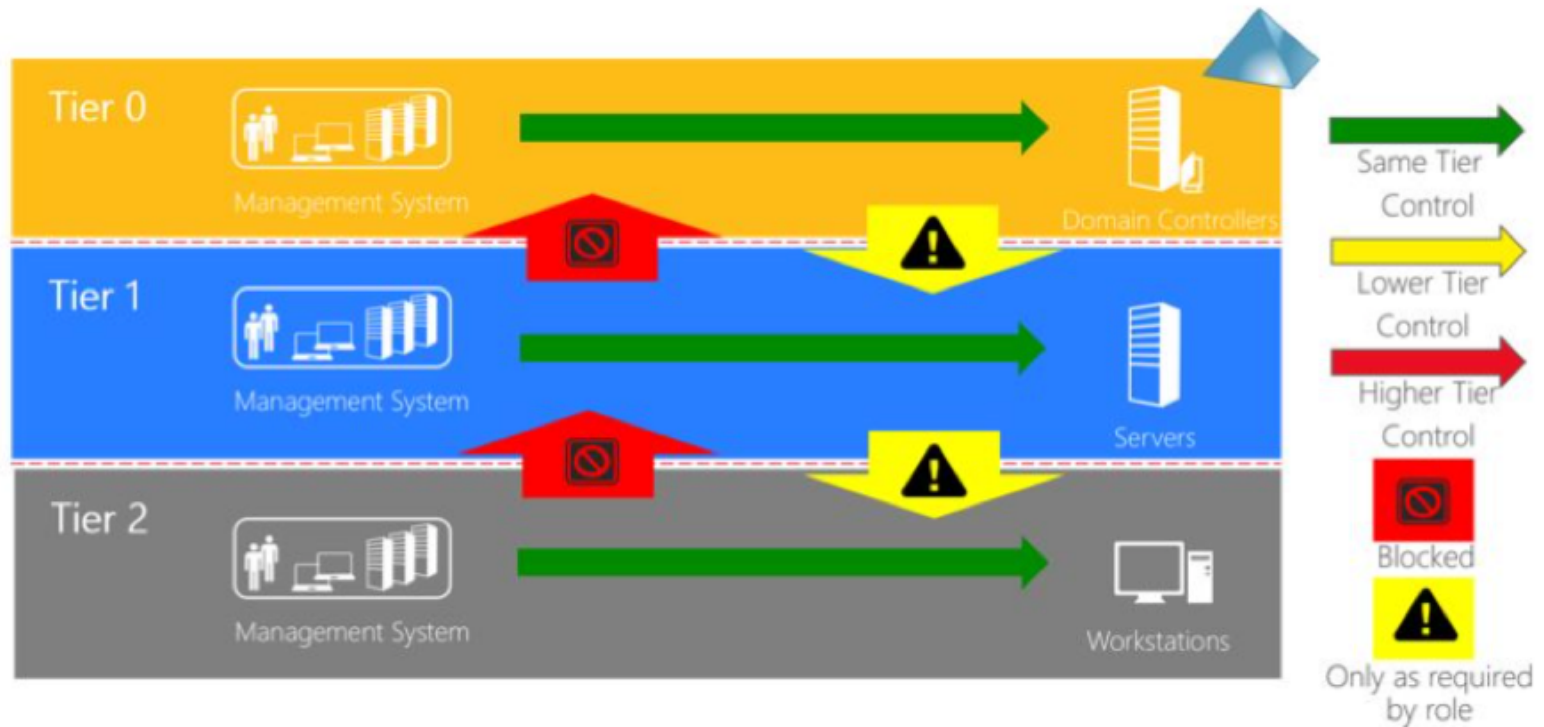
<https://technet.microsoft.com/en-us/windows-server-docs/security/securing-privileged-access/securing-privileged-access-reference-material>

Appropriate Permissions (cont)

20

Control restrictions

Control restrictions are shown in the figure below:

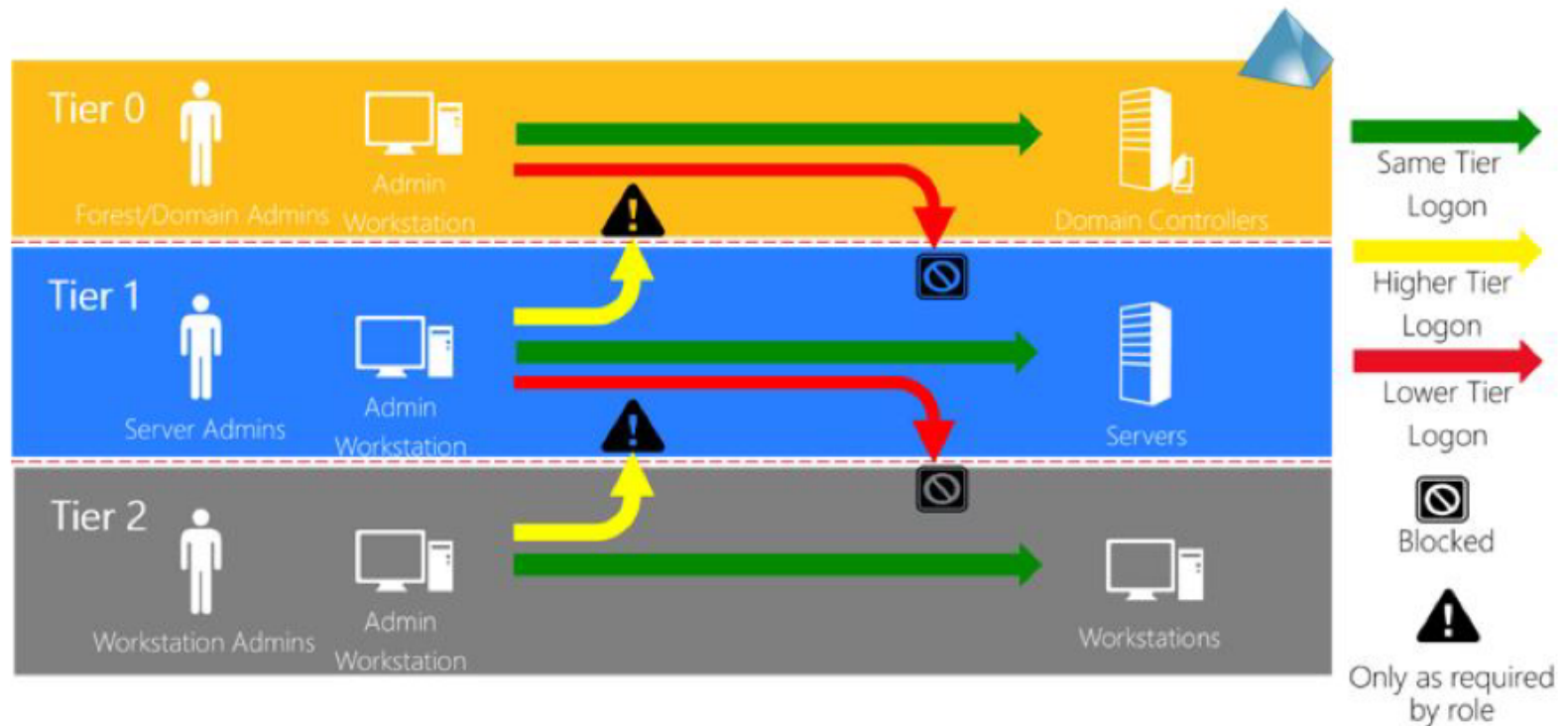


Appropriate Permissions (cont)

21

Logon restrictions

Logon restrictions are shown in the figure below:



Appropriate Permissions (cont)

22

- User Space – Tier 2
 - ▣ Tier 2 administrator – manage enterprise desktops, laptops, printers, and other user devices, and:
 - Can only manage and control assets at the Tier 2 level
 - Can access asset (via network logong type) at any level as required
 - Can only interactively log on to assets at Tier 2 level

Appropriate Permissions (cont)

23

- ❑ Server Space – Tier 1
 - ❑ Tier 1 administrator – manage enterprise servers, services, and applications, and:
 - Can only manage and control assets at the Tier 1 or Tier 2 level
 - Can only access assets (via network logon type) that are trusted at the Tier 1 or Tier 0 levels
 - Can only interactively log on to asset trusted at the Tier 1 level

Appropriate Permissions (cont)

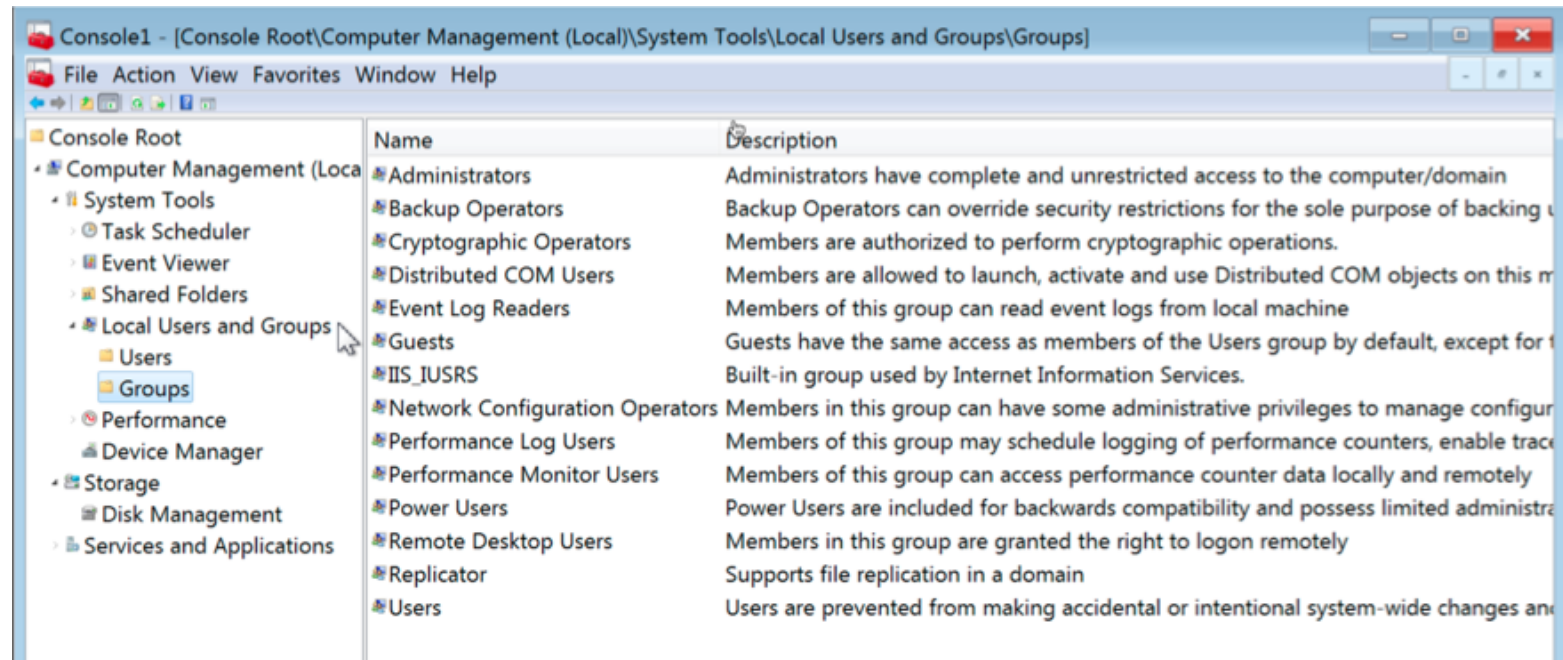
24

- Domain Admins – Tier 0
 - ▣ Tier 0 administrators – manage the identity store and a small number of systems that are in effective control of it, and:
 - Can manage and control assets and any level as required
 - Can only log on interactively or access assets trusted at the Tier 0 level

Appropriate Permissions (cont)

25

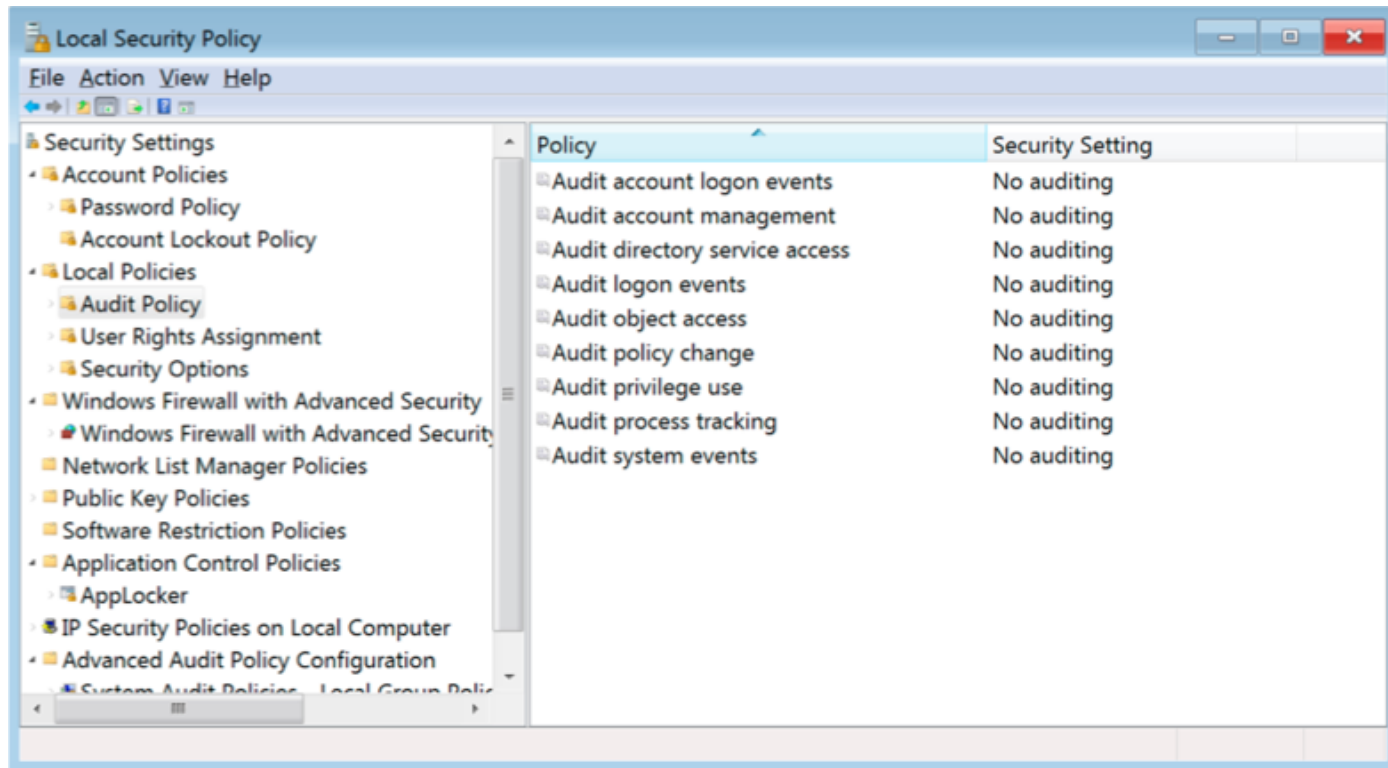
- Built-in Groups
 - ▣ Net localgroup
 - ▣ Demo



Appropriate Permissions (cont)

26

- ❑ Local Security Policy
 - ▣ Demo

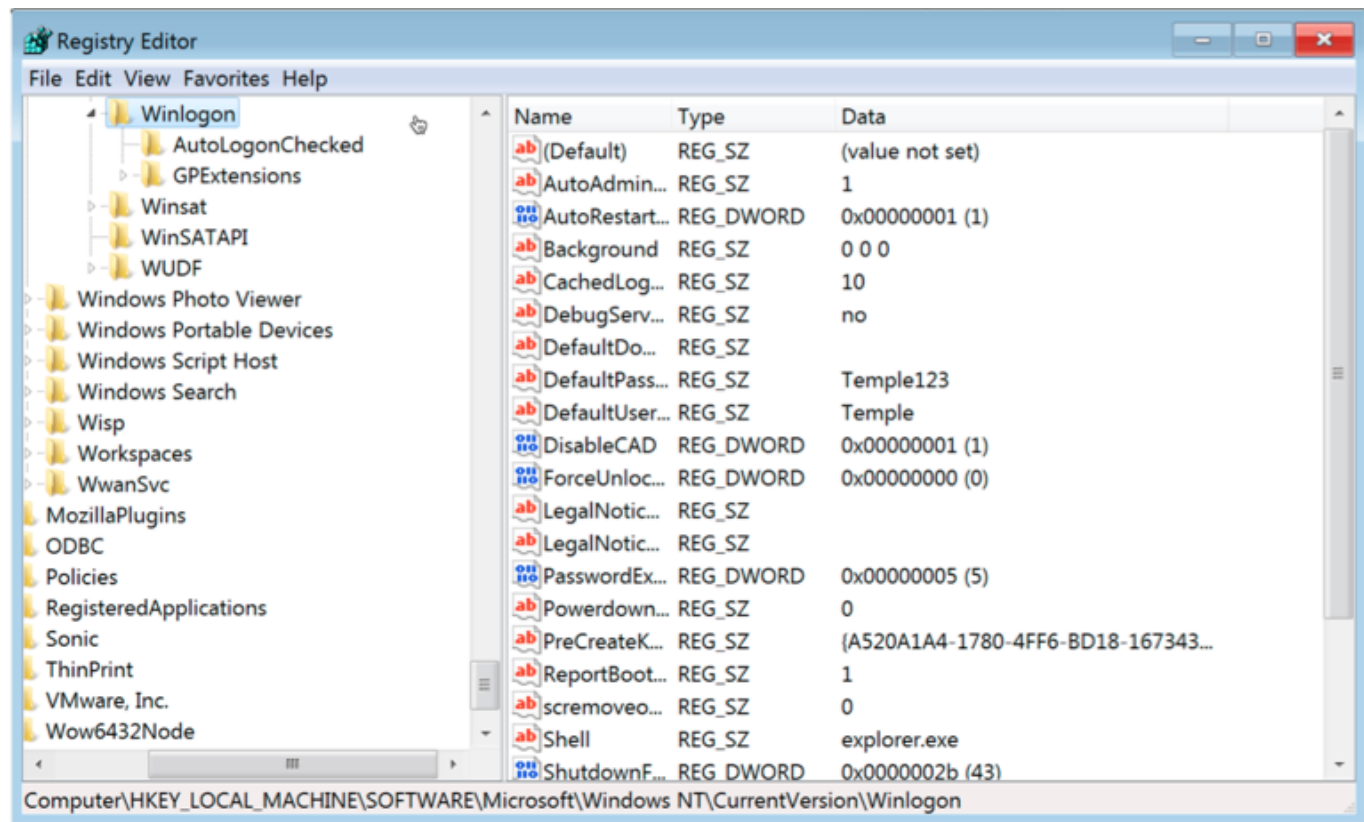


Appropriate Permissions (cont)

27

□ Registry – Windows Registry

▣ Demo



Appropriate Permissions (cont)

28

- Questions?

Limit Services

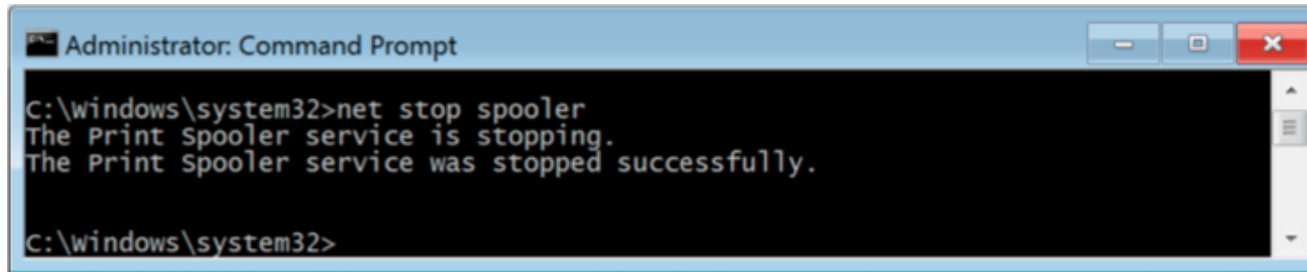
29

- ❑ How to stop services?
 - ❑ MMC
 - ❑ Command Line
- ❑ How to start services?
- ❑ How to change the account a service runs as?
- ❑ How to prevent services from starting?
- ❑ How to change services access?
 - ❑ Allow users to start them
 - Including the account that is running as
 - ❑ Allow a group of users to start them

Limit Services (cont)

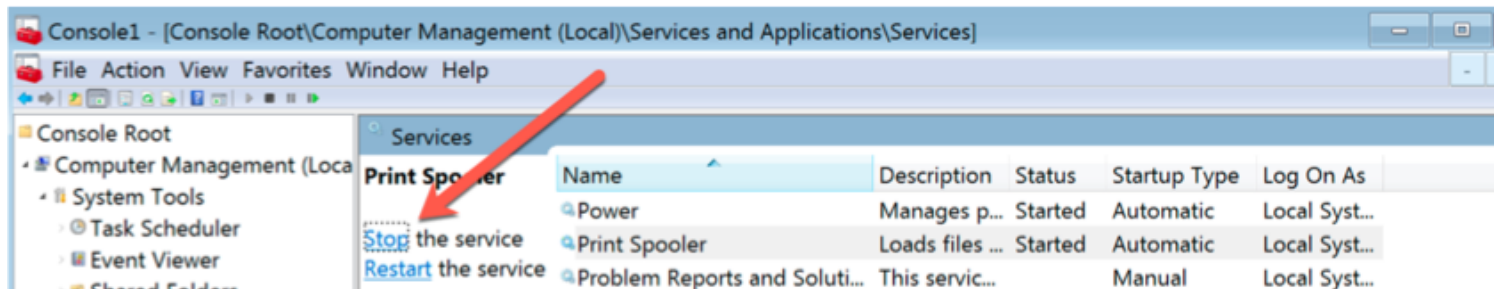
30

- How to stop services?
 - ▣ Command Line: net stop <service name>



```
Administrator: Command Prompt
C:\windows\system32>net stop spooler
The Print Spooler service is stopping.
The Print Spooler service was stopped successfully.
C:\windows\system32>
```

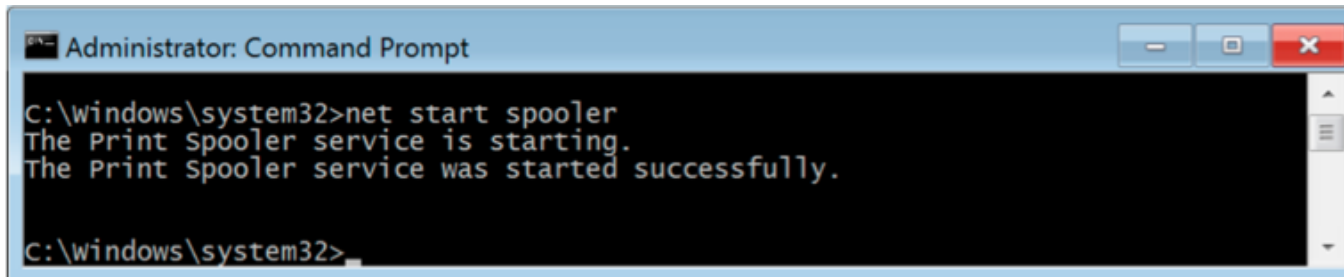
- ▣ GUI



Limit Services (cont)

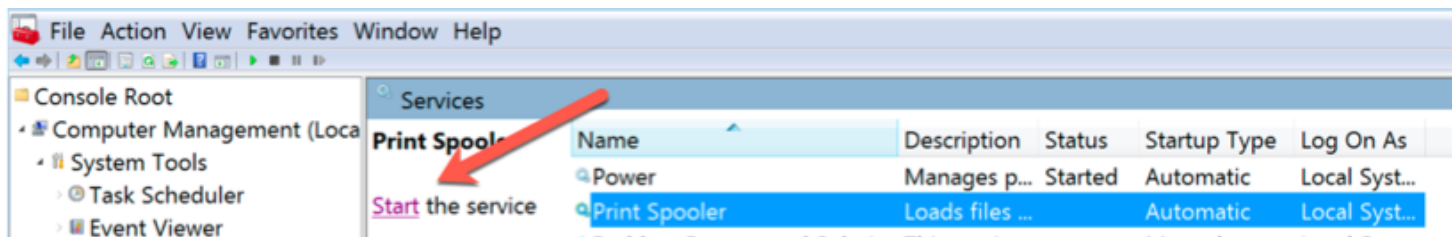
31

- How to start services?
 - ▣ Command Line: net start <service name>



```
Administrator: Command Prompt
C:\windows\system32>net start spooler
The Print spooler service is starting.
The Print spooler service was started successfully.
C:\windows\system32>
```

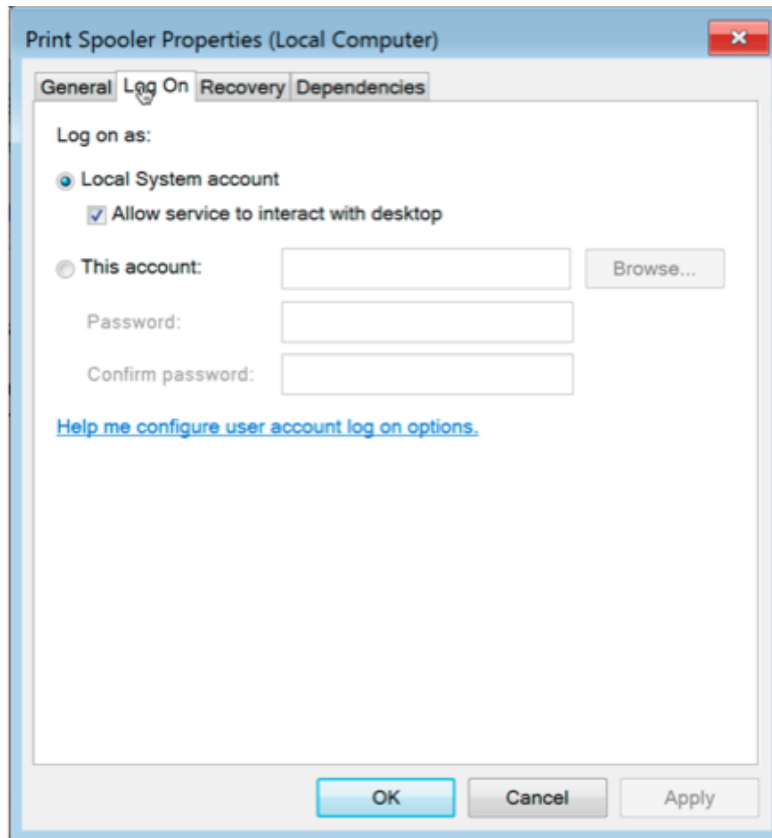
- ▣ GUI



Limit Services (cont)

32

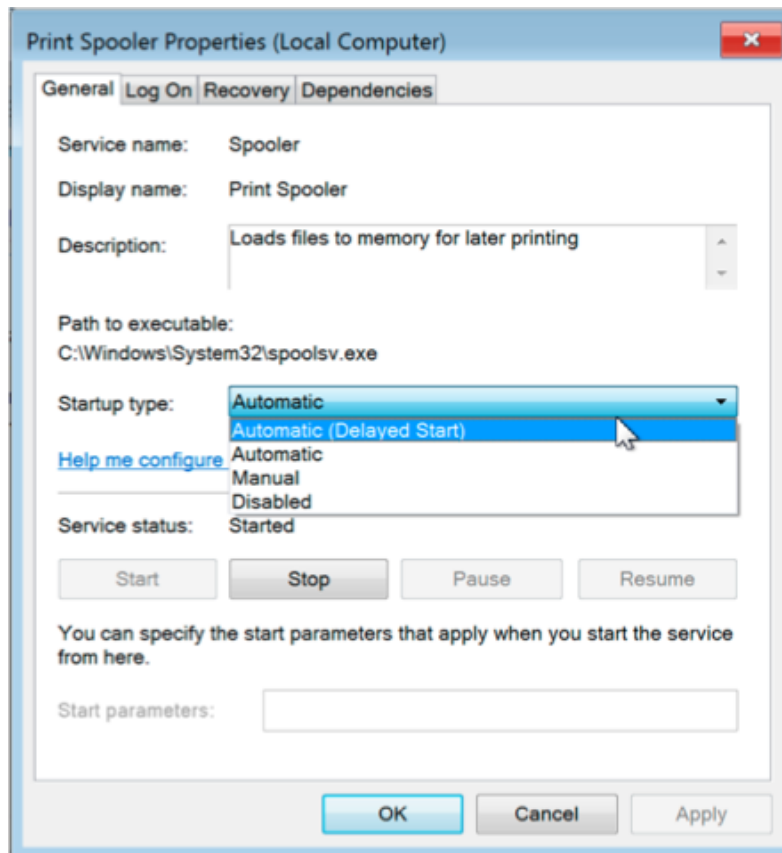
- How to change the account a service runs as?



Limit Services (cont)

33

- How to prevent services from starting?



Limit Services (cont)

34

□ How to change services access?

▣ How to list the current access:

■ sc sdshow spooler

- D:(A;;CCLCSWLOCRR;;;AU)(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;BA)(A;;CCLCSWRPWPDTLOCRRC;;;SY)S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)
- How to read these: <http://stackoverflow.com/questions/4436558/start-stop-a-windows-service-from-a-non-administrator-user-account>

▣ How to set access to add our user

■ subinacl /service spooler /grant=tuser=PTO

- D:(A;;CCLCSWLOCRR;;;AU)(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;BA)(A;;CCLCSWRPWPDTLOCRRC;;;SY)(A;;RPWPDT;;;S-1-5-21-3018343760-3943018779-3883650701-1003)S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)

▣ How to remove the access

■ subinacl /service spooler /deny=tuser=PTO

▣ Demo

Limit Services (cont)

35

- ❑ Including the account that is running as?
 - ❑ Similar to what we did in the previous slide; grant the account running the service the ability to start/stop it's own service
 - ❑ Grant files the service is running
 - ❑ Grant access to the registry the service needs

Limit Services (cont)

36

- ❑ Allow a group of users to start them?
 - ❑ How to list the current access:
 - `sc sdshow spooler`
 - ❑ How to set access to add our user
 - `subinacl /service spooler /grant=users=PTO`
 - ❑ Demo
 - ❑ How to remove the access
 - `subinacl /service spooler /deny=users=PTO`
 - ❑ Demo
 - ❑ Well Known SIDS [https://msdn.microsoft.com/en-us/library/windows/desktop/aa379649\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa379649(v=vs.85).aspx)

Limit Services (cont)

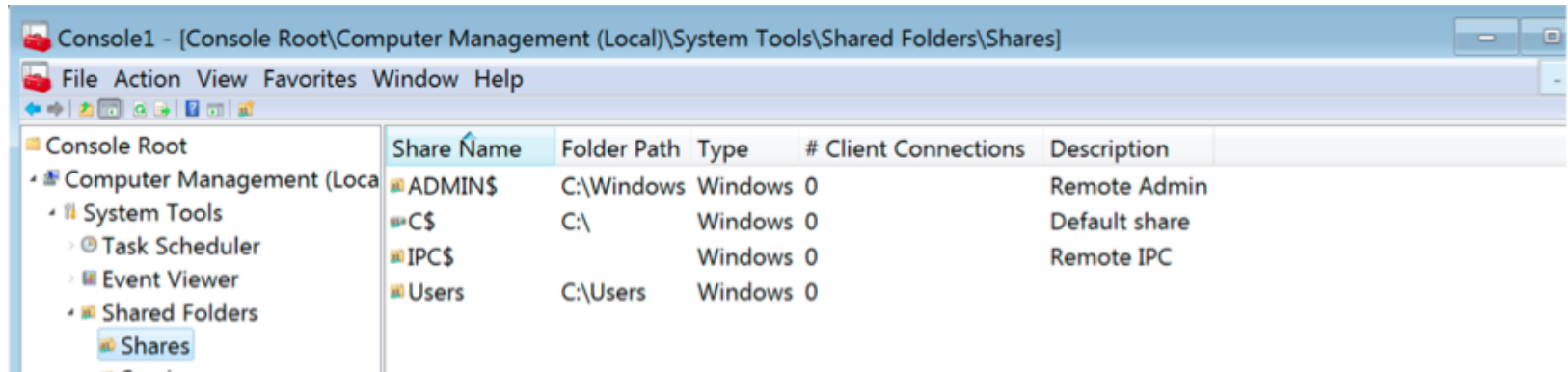
37

- Questions?

Shares

38

- Shares or File Shares
 - ▣ ACL – Access Control Lists
 - Demo



The screenshot shows a Windows console window titled "Console1 - [Console Root\Computer Management (Local)\System Tools\Shared Folders\Shares]". The window displays a table of shares with the following columns: Share Name, Folder Path, Type, # Client Connections, and Description. The table lists four shares: ADMIN\$, C\$, IPC\$, and Users.

Share Name	Folder Path	Type	# Client Connections	Description
ADMIN\$	C:\Windows	Windows	0	Remote Admin
C\$	C:\	Windows	0	Default share
IPC\$		Windows	0	Remote IPC
Users	C:\Users	Windows	0	

Shares(cont)

39

- Questions?

Assignment 1 Overview

40

- ❑ Requirements – a helpdesk style document and how-to video
 - ❑ Build a video of what you did; overview is fine
 - ❑ 1 – 2 pages on the main steps and sub-steps;
 - ❑ Create a patched Windows 7 Pro 64-bit OS using a type 2 hypervisor.
 - ❑ Create a Snap-Shot of patched windows 7 box for testing of installing software and show how to install and revert back to before software being installed. Note software is not important, but learning the interface of you hypervisor is what you want to show.
- ❑ Due Date: Feb 7th

Next Week

41

- ❑ Questions from previous week
- ❑ Configuration management practices
- ❑ System hardening
- ❑ Windows Group Policies
- ❑ Baselines
- ❑ Intrusion detection
- ❑ Intrusion prevention
- ❑ Questions about Assignment 1 (Due Feb 7)
- ❑ Assignment 2 Overview