Assisted Lab: Configuring Controls

Scenario

In this lab, you will learn about several types of security controls, including preventive, detective, directive, and corrective.

As a cybersecurity analyst, you are working to discover weaknesses and vulnerabilities that your organization, Structureality Inc., needs to mitigate throughout its internal network. This lab focuses on ensuring you understand the nature of the various types of security controls by having you configure and use or test them. This will facilitate your recommendations as an analyst on what remediations to implement to resolve discovered security weaknesses.

Your cybersecurity analyst (CySA) workstation, running Windows Server 2019, is located in Structureality's server subnet.

Understand your environment

You will be working from a virtual machine named PC10 hosting Windows Server 2019, which is serving as a client in this lab environment.

Objectives

This activity is designed to test your understanding of and ability to apply content examples in the following CompTIA CySA+ objectives:

1.1 Explain the importance of system and network architecture concepts in security operations.
2.5 Explain concepts related to vulnerability response, handling, and management.

Configure and test preventive controls

A preventive control attempts to stop an unwanted activity from taking place. In this exercise, you will first perform an unwanted activity. Next, you will implement a preventive control to block that activity. And finally, you will attempt the unwanted activity again to test the preventive control.

1. Select the PC10 VM. Send Ctrl+Alt+Delete and, if needed, sign in as Rene using Pa$$w0rd as the password.
   Since *Jaime* may be set as the default account, you will need to select **Other user**, then enter Rene followed by Pa$$w0rd as the password.
   Select the Type Text icon to enter the associated text into the virtual machine.
   Since this will be the first time this account logs into this system, it may take a few moments for the Desktop to appear.
2. Select **Type here to search** from the taskbar, type file, then select **File Explorer** from the results.

3. In the File Explorer address bar enter \\10.1.16.1\CertEnroll.

   When an instruction reads "enter" it is informing you to type in the **bolded** and/or "quoted" item, then press **Enter** on your keyboard.
4. You should see the contents of the *CertEnroll* share.

   This is a problem as these files are for administrative use only. Rene is not an administrator and should not have access. You need to implement a prevention control so that Rene and other non-administrators cannot access this share.

   If a Networks display pop up, always type Yes to allow.

   Who should have access to a share containing sensitive data like CertEnroll?

   Everyone

   Administrators

   Authenticated Users

   Only users with a role requirement for such data

   Congratulations, you have answered the question correctly.
5. Close **File Explorer**.
6. Select the DC10 VM. Send Ctrl+Alt+Delete and, if needed, sign in as Structureality\Administrator using Pa$$w0rd as the password.
7. Minimize or close **Server Manager** if it appears. It will not be used in this lab.
8. Select **Type here to search** from the taskbar, type computer, then select **Computer Management** from the results.
9. Double-click **Shared Folders** from the left pane of Computer Management to expand its contents.
10. Select **Shares** from the expanded *Shared Folders*.
11. Right-click the **CertEnroll** folder from the right pane, then select **Properties**.
12. Select the **Share Permissions** tab on the *CertEnroll Properties* window.
13. Select **Everyone** in the *Group or user names:* area of the *Share Permissions* tab.
14. Select **Remove** to remove the *Everyone* group from the object's ACL entirely.

    The default privilege over objects in Windows is *no access*. Thus, without an explicitly defined *allow*, users will have a default or implicit *deny*.

    It is important not to implement an explicit *deny* at this juncture as it may have unintended consequences. For example, administrators are users, so they are automatically members of the Domain Users and Everyone groups. Setting *deny* for one of these groups would also deny access to the administrators who need access.

    Which of the following would be an effective method to directly block access to Rene and other similar users from accessing an admin-only resource?

    Create a new domain for non-administrative users

    Create the Nonadmin group and make all non-admins a member

    Set deny Full Control on the Everyone group

Create a clone object and set Deny on the Domain Users group
Congratulations, you have answered the question correctly.

15. Select **OK** to close the *CertEnroll Properties* window.
16. Switch back to the PC10 VM and, if needed, sign in as Rene using Pa$$w0rd as the password.
17. Select **Type here to search** from the taskbar, type file, then select **File Explorer** from the results.
18. In the File Explorer address bar enter \\10.1.16.1\CertEnroll.

19.
20. Sign out of PC10 by selecting the **Start** menu, then selecting **Rene** (which will be a circle at the top of the menu), then select **Sign out**. If prompted that there are open programs, select **Sign out anyway**.

You have successfully implemented a preventive control to block nonadministrative users from accessing resources that are for administrators only. In a real-world situation, you should compare any concerning issue to company security policy and configuration baselines. If you have discovered a variant or violation, it needs to be reported to the security team. This report may include recommendations for remediation.

Check your work

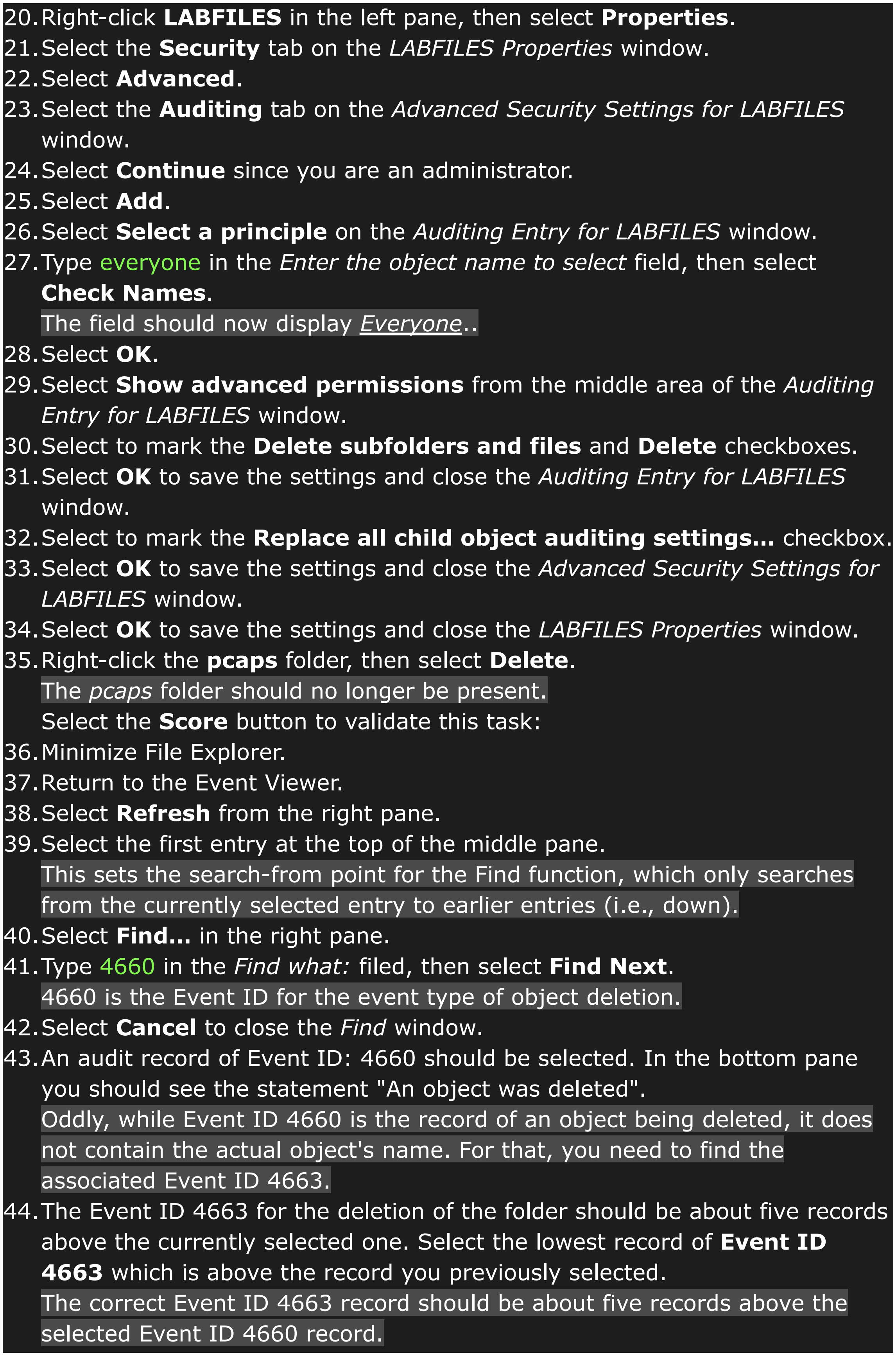
Confirm that you implemented a preventive control.

Confirm that you tested a preventive control.

Configure and test detective controls

A detective control records a log each time an event takes place, regardless of whether that activity is benign or malicious. In this exercise, you will first perform an activity that will not be logged. Next, you will configure logging to record that activity. Next, you will perform the activity again. Finally, you will review the log to confirm the record of the activity was created.

1. Connect to the PC10 virtual machine, send Ctrl+Alt+Delete, and sign in as jaime using Pa$$w0rd as the password.
   Jaime is a member of the Domain Admins group. So, this user account is an administrator on the PC10 system.
   Since the previous exercise was performed while logged in as Rene, that account may be the default when you access the PC10 VM again for this exercise. You will need to select **Other user**, then enter Jaime followed by Pa$$w0rd as the password.
2. Select **Type here to search** from the taskbar, type file, then select **File Explorer** from the results.
3. Select **LABFILES** from the *Quick access* area in the left pane of File Explorer.
4. Right-click the folder **empty** then select **Delete**.
   The *empty* folder should no longer be present.
   Select the **Score** button to validate this task:
5. Select **Type here to search** from the taskbar, type event, then select **Event Viewer** from the results.
6. Maximize the Event Viewer window.
7. Double-click **Windows Logs** to expand its contents.
8. Select **Security** from in the *Windows Logs* expanded contents.
9. Select **Find...** in the right pane.
10. Type empty in the *Find what:* filed, then select **Find Next**.
11. After a few moments of searching, a window will appear stating the search term was not found. Select **OK**.
    The results of the find operation indicate what?
    Jamie is an administrator
    Folder deletion is not being audited
    Users are unable to access empty folders
    User activity is being tracked
12. Select **Cancel** to close the *Find* window.
13. Select **Type here to search** from the taskbar, type local, then select **Local Security Policy** from the results.
14. Double-click **Local Policies** to expand its contents.
15. Select **Audit Policy** from the *Local Policies* expanded contents.
16. Right-click **Audit object access** in the right pane, then select **Properties**.
17. Select to mark both the **Success** and **Failure** checkboxes, then select **OK**.
    While the main switch for auditing object access activities is now on, auditing will not occur on most file objects until an on-object auditing setting is made.
18. Close the *Local Security Policy* window.
    The setting change should apply immediately. If the next steps do not result in a record of a folder deletion, then restart PC10 and repeat from here, but you will then need to delete the *MARKETING* folder.
19. Return to File Explorer.



20. Right-click **LABFILES** in the left pane, then select **Properties**.
21. Select the **Security** tab on the *LABFILES Properties* window.
22. Select **Advanced**.
23. Select the **Auditing** tab on the *Advanced Security Settings for LABFILES* window.
24. Select **Continue** since you are an administrator.
25. Select **Add**.
26. Select **Select a principle** on the *Auditing Entry for LABFILES* window.
27. Type everyone in the *Enter the object name to select* field, then select **Check Names**.

    The field should now display *Everyone*..
28. Select **OK**.
29. Select **Show advanced permissions** from the middle area of the *Auditing Entry for LABFILES* window.
30. Select to mark the **Delete subfolders and files** and **Delete** checkboxes.
31. Select **OK** to save the settings and close the *Auditing Entry for LABFILES* window.
32. Select to mark the **Replace all child object auditing settings...** checkbox.
33. Select **OK** to save the settings and close the *Advanced Security Settings for LABFILES* window.
34. Select **OK** to save the settings and close the *LABFILES Properties* window.
35. Right-click the **pcaps** folder, then select **Delete**.

    The *pcaps* folder should no longer be present.

    Select the **Score** button to validate this task:
36. Minimize File Explorer.
37. Return to the Event Viewer.
38. Select **Refresh** from the right pane.
39. Select the first entry at the top of the middle pane.

    This sets the search-from point for the Find function, which only searches from the currently selected entry to earlier entries (i.e., down).
40. Select **Find...** in the right pane.
41. Type 4660 in the *Find what:* filed, then select **Find Next**.

    4660 is the Event ID for the event type of object deletion.
42. Select **Cancel** to close the *Find* window.
43. An audit record of Event ID: 4660 should be selected. In the bottom pane you should see the statement "An object was deleted".

    Oddly, while Event ID 4660 is the record of an object being deleted, it does not contain the actual object's name. For that, you need to find the associated Event ID 4663.
44. The Event ID 4663 for the deletion of the folder should be about five records above the currently selected one. Select the lowest record of **Event ID 4663** which is above the record you previously selected.

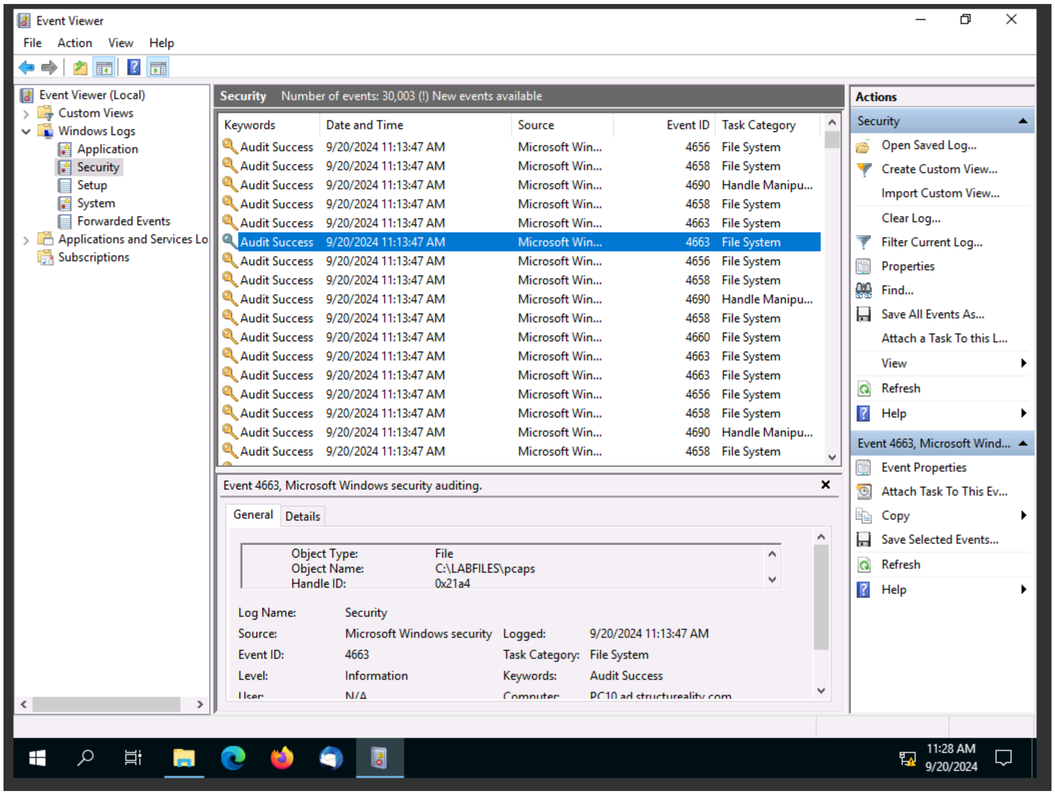
    The correct Event ID 4663 record should be about five records above the selected Event ID 4660 record.

45. Once you have selected the Event ID 4663 record, you can view the details in the bottom pane. On the *General* tab, there is a small scrollable sub-window with details. You should see a line of "Object Name: C\LABFILES\pcaps". This Event ID 4663 record confirms that the object deleted was the C\LABFILES\pcaps folder.
    You could also select the *Details* pane to see most of the same information.
    What is the purpose of a detective control?
    Deny access to an object
    Notify subjects about system policies
    Inform users of the proper steps to perform an activity
    Create a record of events and activities
46. Sign out of PC10 by selecting the **Start** menu, then selecting **Jaime** (which will be a circle at the top of the menu), then select **Sign out**. If prompted that there are open programs, select **Sign out anyway**.

You have successfully implemented a detective control to record object access activity.

Check your work

Confirm that you implemented a detective control.
Confirm that you tested a detective control.



Security Number of events: 30,003 (!) New events available

Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4656	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4658	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4690	Handle Manipu...
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4658	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4663	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4663	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4656	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4658	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4690	Handle Manipu...
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4658	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4660	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4663	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4663	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4656	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4658	File System
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4690	Handle Manipu...
Audit Success	9/20/2024 11:13:47 AM	Microsoft Win...	4658	File System

Event 4663, Microsoft Windows security auditing.

General | Details

```
        Object Type:            File
        Object Name:            C:\LABFILES\pcaps
        Handle ID:              0x21a4

Log Name:      Security
Source:        Microsoft Windows security   Logged:          9/20/2024 11:13:47 AM
Event ID:      4663                         Task Category:   File System
Level:         Information                  Keywords:        Audit Success
User:          N/A                          Computer:        PC10.ad.structureality.com
```

Configure and test directive controls

A directive control provides instruction to direct a user towards more compliant behavior. In this exercise, you will configure a directive control in the form of a login warning banner. Finally, you will test this directive control.

1. Connect to the PC10 virtual machine, send Ctrl+Alt+Delete and sign in as jaime using Pa$$w0rd as the password.
   Jaime is a member of the Domain Admins group. So, this user account is an administrator on the PC10 system.
2. Select **Type here to search** from the taskbar, type powershell, then right-click **Windows PowerShell** from the results, then select **Run as administrator**.
3. Select **Yes** on the User Account Control window.
4. Enter the following code into the *Administrator: Windows PowerShell* console:
   Be sure to press **Enter** on your keyboard after each entry appears in the PowerShell console. There will not be any confirmation.

   ```
   $BannerText = "This computer system is the property of the CySA+ Online Lab. It is for authorized use only. By using this system, all users acknowledge notice of, and agree to comply with, the Acceptable Use Policy (AUP). Unauthorized or improper use of this system may result in administrative disciplinary action, civil charges/criminal penalties, and/or other sanctions as set forth in the AUP. By continuing to use this system, you indicate your awareness of and consent to these terms and conditions of use. If you are physically located in the European Union, you may have additional rights per the GDPR. Visit the website gdpr-info.eu for more information."
   New-ItemProperty -Path "HKLM:\Software\Microsoft\Windows\CurrentVersion\Policies\System" -Name "legalnoticecaption" -Value "Authorized Use Only" -PropertyType "String" -Force | Out-Null
   New-ItemProperty -Path "HKLM:\Software\Microsoft\Windows\CurrentVersion\Policies\System" -Name "legalnoticetext" -Value $BannerText -PropertyType "String" -Force | Out-Null
   ```

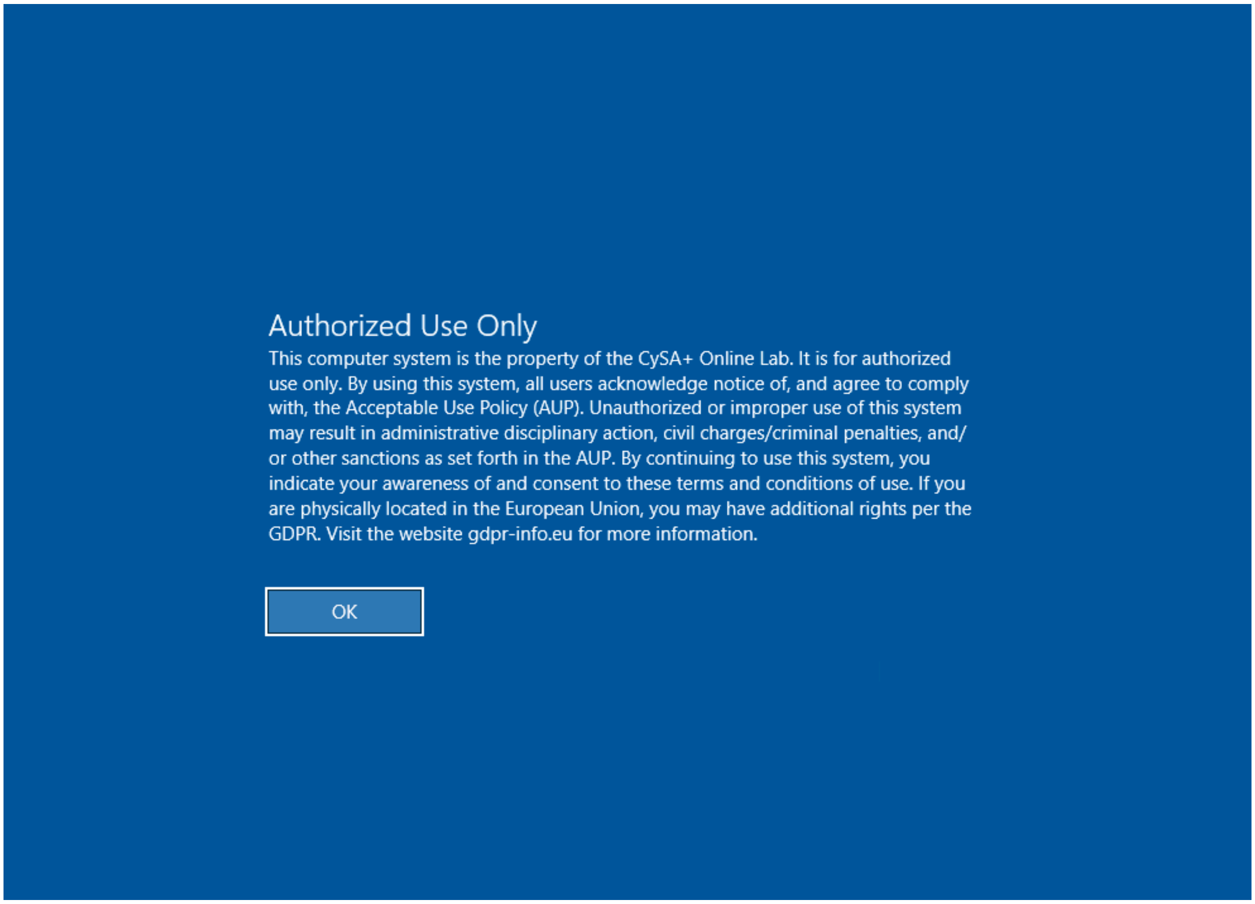
   The text of the warning banner in this exercise is an amalgamation of several banners used by various commercial and educational facilities. Be sure to consult with your own legal counsel before setting a warning banner to ensure it complies with laws and regulations.
   Select the **Score** button to validate this task:

5. Sign out of PC10 by selecting the **Start** menu, then selecting **Jaime** (which will be a circle at the top of the menu), then select **Sign out**. If prompted that there are open programs, select **Sign out anyway**.
6. Connect to the PC10 virtual machine, send Ctrl+Alt+Delete.
7. You should be presented with the login warning banner that was just defined.
   What is the goal of directive controls?
   Defense
   Compliance
   Prohibition
   Tracking
8. Read the warning banner, then select **OK**.
9. Complete the sign-in process as jaime using Pa$$w0rd as the password.

You have successfully implemented a directive control to inform personnel of the limitations and restrictions of a controlled system.

Check your work

Confirm that you implemented a directive control.
Confirm that you tested a directive control.



Authorized Use Only

This computer system is the property of the CySA+ Online Lab. It is for authorized use only. By using this system, all users acknowledge notice of, and agree to comply with, the Acceptable Use Policy (AUP). Unauthorized or improper use of this system may result in administrative disciplinary action, civil charges/criminal penalties, and/or other sanctions as set forth in the AUP. By continuing to use this system, you indicate your awareness of and consent to these terms and conditions of use. If you are physically located in the European Union, you may have additional rights per the GDPR. Visit the website gdpr-info.eu for more information.

Configure and test corrective controls

A corrective control is intended to detect when something is in a less secure or less desirable state, then attempts to return to the more secure or more desirable state. In some cases, the corrective control is able to repair minor damage to restore a system back to a more secure or desirable state.

In this exercise, you will first use a fault injection tool to trigger the existing correct control of Windows to trigger its native corrective control protection against misbehaving applications. Next, you will create and test a custom corrective control to protect the contents of a text file.

1. Connect to the PC10 virtual machine and, if needed, send Ctrl+Alt+Delete, select **OK**, then complete the sign-in process as jaime using Pa$$w0rd as the password.
2. Select **Type here to search** from the taskbar, type file, then select **File Explorer** from the results.

3. Select **SYSINTERNALS** from the *Quick access* area in the left pane.
4. Scroll to locate, then double-click **notmyfault64** to execute it.

   There is a CLI (command line interface) version of NotMyFault which has a c in the file name: notmyfaultc64. If a Command Prompt window flashes open and then disappears, you selected the CLI version, not the GUI version of NotMyFault64.

   Windows Sysinternals is a website that offers technical resources and utilities to manage, diagnose, troubleshoot, and monitor a Microsoft Windows environment. You can experiment with the Sysinternals tools in this lab environment or go directly to sysinternals.com to learn more and download the entire suite of nearly 75 tools onto your own system.
5. Select **Yes** on the User Account Control window.
6. Select the **Code overwrite** option, then select **Crash**.
7. The PC10 system should immediately experience a stop error (often called the BSOD (Blue Screen of Death)). The system will perform a partial memory dump (for potential analysis - which will not be done in this lab) and then reboot.

   What are the dual purposes of corrective controls? (Select two)
   Address an unwanted or less secure state or event
   Record evidence of user and event activities
   Return the system to a normal and generally secure condition
   Provide guidance on proper user behavior
8. Connect to the PC10 virtual machine and send Ctrl+Alt+Delete, select **OK**, then complete the sign-in process as jaime using Pa$$w0rd as the password.

You have verified that the Windows corrective control to protect the execution environment from misbehaving applications is active. While you might not prefer in-memory data to be lost, the stability of the Windows execution environment is protected by immediately ceasing all execution. You can be assured that once the system reboots, the offending application will not be running. This native Windows protective feature is the reason you should save early and save often when creating new content or media.

Next, you will create your own corrective control to simulate the correction functions of the SigVerif utility.

9. Select **Type here to search** from the taskbar, type powershell, then select **Windows PowerShell** from the results.

   In this portion of this exercise, you will create a corrective control to monitor the contents of a file. If the file contents change, then the control will restore the file back to its preferred content.

10. Enter "This is important" | Set-Content notes.txt.
    This command creates a text file containing the phrase "This is important".
    Select the **Score** button to validate this task:
11. Enter type notes.txt.
    This command displays the contents of the file.
12. Enter Get-FileHash ./notes.txt -Algorithm SHA256 | Select-Object -ExpandProperty Hash | Set-Content ./hash.txt.
    This command calculates a hash of the file and stores it in hash.txt for future use.
    The dot and slash (i.e., ./) in front of the filename are used to indicate the current working directory.
    Select the **Score** button to validate this task:
13. Enter echo blah >> notes.txt.
    This command injects new content into notes.txt, which changes the file.
    The use of double greater-than symbols (i.e., >>) performs an append rather than a replace function when capturing output into a file.
14. Enter type notes.txt.
    You should see different contents of the notes.txt file.
15. Enter if((Get-FileHash ./notes.txt -Algorithm SHA256).Hash -eq (Get-Content ./hash.txt)) {Write-Host "The file is correct."} else {Write-Host "The file has changed. Corrective action should be initiated."}.
    This command calculates the hash of notes.txt and compares it to the value stored in hash.txt. Since the file has changed, an error message is displayed.
16. Enter "This is important" | Set-Content notes.txt.
    This command is the corrective action to reset the contents of notes.txt back to the desired content.
17. Enter type notes.txt.
18. Enter if((Get-FileHash ./notes.txt -Algorithm SHA256).Hash -eq (Get-Content ./hash.txt)) {Write-Host "The file is correct."} else {Write-Host "The file has changed. Corrective action should be initiated."}.
    This command calculates the hash of notes.txt and compares it to the value stored in hash.txt. Since the file has been restored, a confirmation message is displayed.
    You have performed the corrective control manually. Now configure scripts to automate the process.
19. Enter notepad calchash.ps1.
20. Select **Yes** on the *Notepad* window about creating a new file.
21. Type the following into the new document: Get-FileHash ./notes.txt -Algorithm SHA256 | Select-Object -ExpandProperty Hash | Set-Content ./hash.txt.
22. Close **Notepad**, select **Save** when prompted.
    Select the **Score** button to validate this task:

23. Enter rm hash.txt.

    This command deletes the previous hash.txt file.

24. Enter ./calchash.ps1.

    This command executes the PowerShell script of calchash.ps1, which generates a new hash.txt file containing the hash of notes.txt.

    The dot and slash (i.e., ./) in front of the script name are essential for execution.

25. Enter type hash.txt.

    This command displays the contents of hash.txt

26. Enter notepad check.ps1.
27. Select **Yes** on the *Notepad* window about creating a new file.
28. Select the empty area of the Notepad window, then select the below to paste the script into the VM.

```
if((Get-FileHash ./notes.txt -Algorithm SHA256).Hash -ne (Get-Content
./hash.txt))
{
  "This is important" | Set-Content ./notes.txt
  Write-Host "The file has changed. Corrective action initiated."
}
else
{
  Write-Host "The file is correct. No corrective action needed."
```

29. }
30. Close **Notepad**, select **Save** when prompted.

    Select the **Score** button to validate this task:

31. Enter ./check.ps1.

    This command executes the PowerShell script of check.ps1, which calculates the hash of notes.txt and compares it to the value stored in hash.txt. If the file has not changed, a "No corrective action needed" message is displayed. If the file has changed, an "Corrective action initiated" message is displayed.

    The dot and slash (i.e., ./) in front of the script name are essential for execution.

    The result should display the "The file is correct. No corrective action needed." message since you previously restored the notes.txt file manually.

32. Enter type notes.txt.

    You should see the correct contents of the notes.txt file.

33. Enter echo blah >> notes.txt.

    This command injects new content into notes.txt, which changes the file.

    The use of double greater-than symbols (i.e., >>) performs an append rather than a replace function when capturing output into a file.

34. Enter type notes.txt.

    You should see the modified contents of the notes.txt file.

35. Enter ./check.ps1.

    This should display the "The file has changed. Corrective action initiated." message since the notes.txt file was modified.

    The dot and slash (i.e., ./) in front of the script name are essential for execution.

    What is the typical means (which was used in this exercise) to detect changes in a file?

    encryption
    authentication
    authorization
    hashing

36. Enter type notes.txt.

    You should see the corrected contents of the notes.txt file.

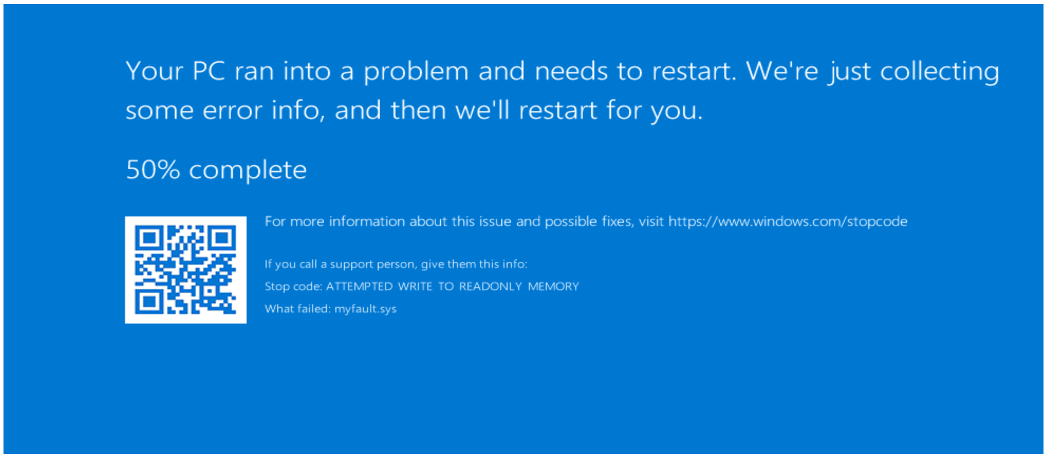
You have successfully implemented a corrective control to repair the contents of a file should it be modified.

This corrective action is similar to that performed by the Signature Verification (SigVerif) tool of Windows. SigVerif executes before each booting of Windows to ensure that the necessary files for a secure booting operation are present and meet a specific hash value. If any of those files are corrupted, they are removed and replaced with a valid file. The corrective actions you took manually can be automated to perform similarly. For example, you could schedule a boot task to run the check.ps1 script each time the system reboots. Also, you should run the calchash.ps1 script every time a valid change to notes.txt is performed. However, if you do elect to make a change to the contents of notes.txt, the correction action would need to be updated accordingly.

more...

Check your work

Confirm that you implemented a corrective control.
Confirm that you tested a corrective control.

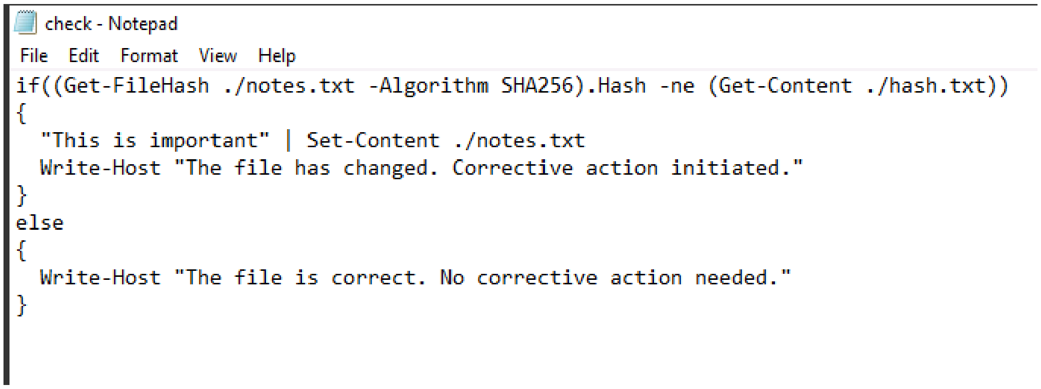


Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

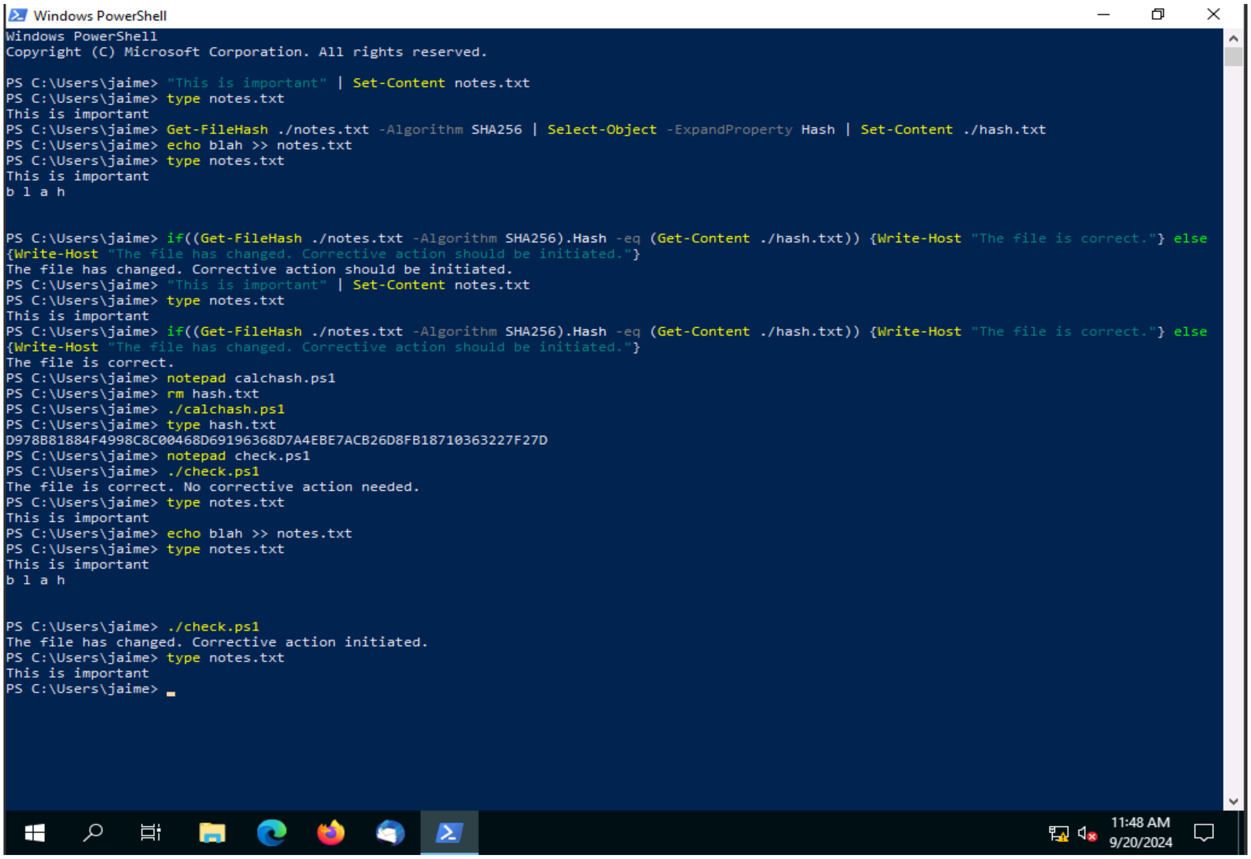
50% complete

For more information about this issue and possible fixes, visit https://www.windows.com/stopcode

If you call a support person, give them this info:
Stop code: ATTEMPTED WRITE TO READONLY MEMORY
What failed: myfault.sys



```
if((Get-FileHash ./notes.txt -Algorithm SHA256).Hash -ne (Get-Content ./hash.txt))
{
  "This is important" | Set-Content ./notes.txt
  Write-Host "The file has changed. Corrective action initiated."
}
else
{
  Write-Host "The file is correct. No corrective action needed."
}
```

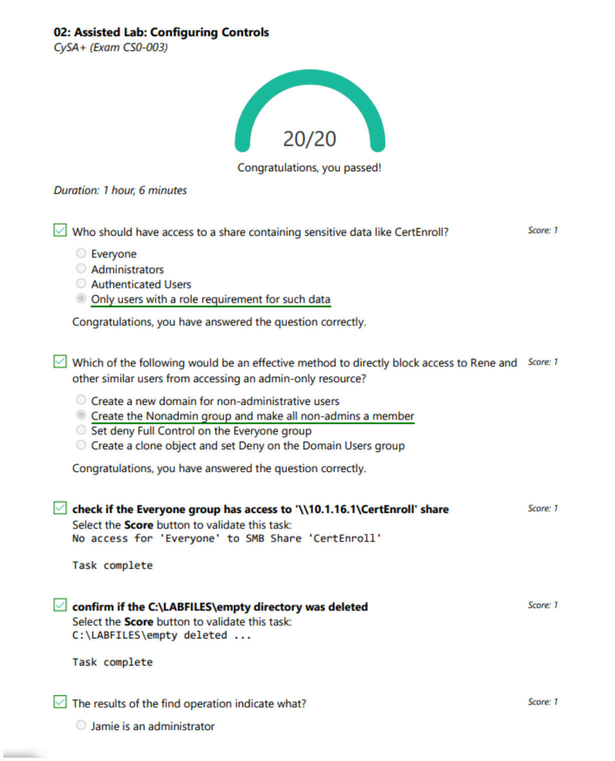


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\jaime> "This is important" | Set-Content notes.txt
PS C:\Users\jaime> type notes.txt
This is important
PS C:\Users\jaime> Get-FileHash ./notes.txt -Algorithm SHA256 | Select-Object -ExpandProperty Hash | Set-Content ./hash.txt
PS C:\Users\jaime> echo blah >> notes.txt
PS C:\Users\jaime> type notes.txt
This is important
b l a h


PS C:\Users\jaime> if((Get-FileHash ./notes.txt -Algorithm SHA256).Hash -eq (Get-Content ./hash.txt)) {Write-Host "The file is correct."} else
{Write-Host "The file has changed. Corrective action should be initiated."}
The file has changed. Corrective action should be initiated.
PS C:\Users\jaime> "This is important" | Set-Content notes.txt
PS C:\Users\jaime> type notes.txt
This is important
PS C:\Users\jaime> if((Get-FileHash ./notes.txt -Algorithm SHA256).Hash -eq (Get-Content ./hash.txt)) {Write-Host "The file is correct."} else
{Write-Host "The file has changed. Corrective action should be initiated."}
The file is correct.
PS C:\Users\jaime> notepad calchash.ps1
PS C:\Users\jaime> rm hash.txt
PS C:\Users\jaime> ./calchash.ps1
PS C:\Users\jaime> type hash.txt
D978B81884F4998C8C00468D69196368D7A4EBE7ACB26D8FB18710363227F27D
PS C:\Users\jaime> notepad check.ps1
PS C:\Users\jaime> ./check.ps1
The file is correct. No corrective action needed.
PS C:\Users\jaime> type notes.txt
This is important
PS C:\Users\jaime> echo blah >> notes.txt
PS C:\Users\jaime> type notes.txt
This is important
b l a h


PS C:\Users\jaime> ./check.ps1
The file has changed. Corrective action initiated.
PS C:\Users\jaime> type notes.txt
This is important
PS C:\Users\jaime> _
```



02: Assisted Lab: Configuring Controls
CySA+ (Exam CS0-003)

20/20

Congratulations, you passed!

Duration: 1 hour, 6 minutes

☑ Who should have access to a share containing sensitive data like CertEnroll? *Score: 1*

○ Everyone
○ Administrators
○ Authenticated Users
● Only users with a role requirement for such data

Congratulations, you have answered the question correctly.

☑ Which of the following would be an effective method to directly block access to Rene and other similar users from accessing an admin-only resource? *Score: 1*

○ Create a new domain for non-administrative users
● Create the Nonadmin group and make all non-admins a member
○ Set deny Full Control on the Everyone group
○ Create a clone object and set Deny on the Domain Users group

Congratulations, you have answered the question correctly.

☑ **check if the Everyone group has access to '\\10.1.16.1\CertEnroll' share** *Score: 1*

Select the **Score** button to validate this task:

```
No access for 'Everyone' to SMB Share 'CertEnroll'

Task complete
```

☑ **confirm if the C:\LABFILES\empty directory was deleted** *Score: 1*

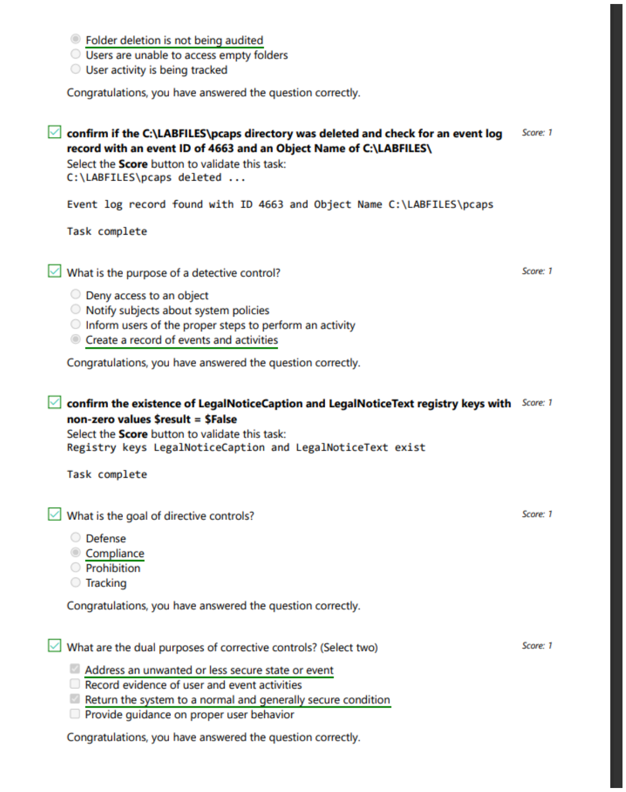
Select the **Score** button to validate this task:

```
C:\LABFILES\empty deleted ...

Task complete
```

☑ The results of the find operation indicate what? *Score: 1*

○ Jamie is an administrator



● Folder deletion is not being audited
○ Users are unable to access empty folders
○ User activity is being tracked

Congratulations, you have answered the question correctly.

☑ **confirm if the C:\LABFILES\pcaps directory was deleted and check for an event log record with an event ID of 4663 and an Object Name of C:\LABFILES\** *Score: 1*

Select the **Score** button to validate this task:

```
C:\LABFILES\pcaps deleted ...

Event log record found with ID 4663 and Object Name C:\LABFILES\pcaps

Task complete
```

☑ What is the purpose of a detective control? *Score: 1*

○ Deny access to an object
○ Notify subjects about system policies
○ Inform users of the proper steps to perform an activity
● Create a record of events and activities

Congratulations, you have answered the question correctly.

☑ **confirm the existence of LegalNoticeCaption and LegalNoticeText registry keys with non-zero values $result = $False** *Score: 1*

Select the **Score** button to validate this task:

```
Registry keys LegalNoticeCaption and LegalNoticeText exist

Task complete
```

☑ What is the goal of directive controls? *Score: 1*

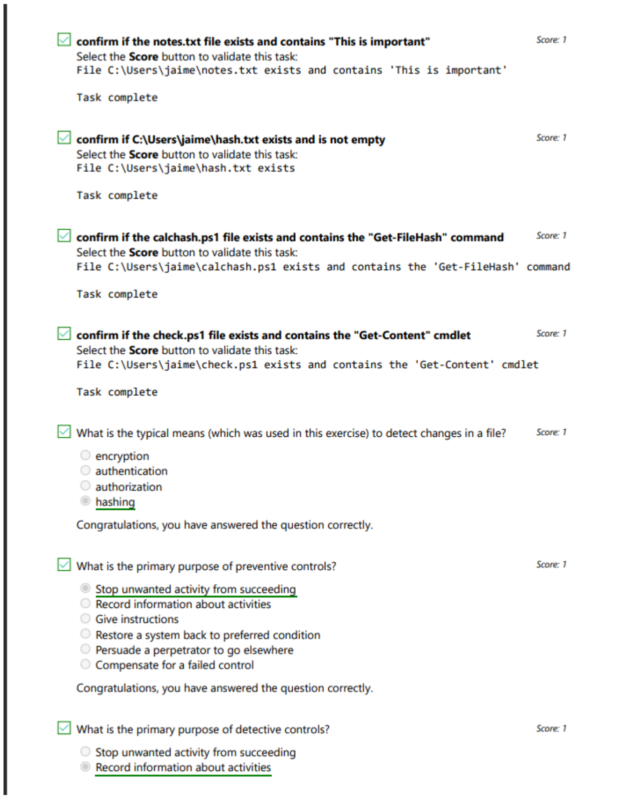
○ Defense
● Compliance
○ Prohibition
○ Tracking

Congratulations, you have answered the question correctly.

☑ What are the dual purposes of corrective controls? (Select two) *Score: 1*

☑ Address an unwanted or less secure state or event
☐ Record evidence of user and event activities
☑ Return the system to a normal and generally secure condition
☐ Provide guidance on proper user behavior

Congratulations, you have answered the question correctly.



☑ **confirm if the notes.txt file exists and contains "This is important"** *Score: 1*

Select the **Score** button to validate this task:

```
File C:\Users\jaime\notes.txt exists and contains 'This is important'

Task complete
```

☑ **confirm if C:\Users\jaime\hash.txt exists and is not empty** *Score: 1*

Select the **Score** button to validate this task:

```
File C:\Users\jaime\hash.txt exists

Task complete
```

☑ **confirm if the calchash.ps1 file exists and contains the "Get-FileHash" command** *Score: 1*

Select the **Score** button to validate this task:

```
File C:\Users\jaime\calchash.ps1 exists and contains the 'Get-FileHash' command

Task complete
```

☑ **confirm if the check.ps1 file exists and contains the "Get-Content" cmdlet** *Score: 1*

Select the **Score** button to validate this task:

```
File C:\Users\jaime\check.ps1 exists and contains the 'Get-Content' cmdlet

Task complete
```

☑ What is the typical means (which was used in this exercise) to detect changes in a file? *Score: 1*

○ encryption
○ authentication
○ authorization
● hashing

Congratulations, you have answered the question correctly.

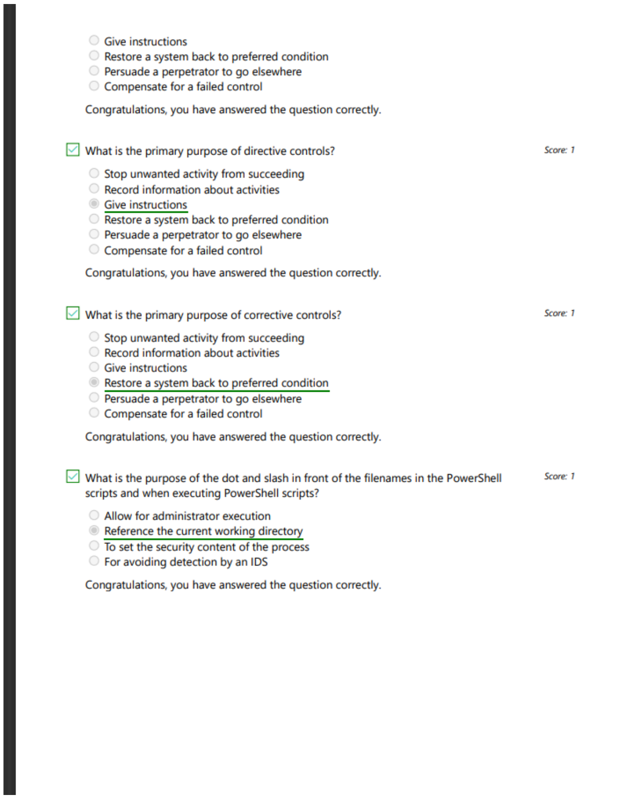
☑ What is the primary purpose of preventive controls? *Score: 1*

● Stop unwanted activity from succeeding
○ Record information about activities
○ Give instructions
○ Restore a system back to preferred condition
○ Persuade a perpetrator to go elsewhere
○ Compensate for a failed control

Congratulations, you have answered the question correctly.

☑ What is the primary purpose of detective controls? *Score: 1*

○ Stop unwanted activity from succeeding
● Record information about activities



○ Give instructions
○ Restore a system back to preferred condition
○ Persuade a perpetrator to go elsewhere
○ Compensate for a failed control

Congratulations, you have answered the question correctly.

☑ What is the primary purpose of directive controls? *Score: 1*

○ Stop unwanted activity from succeeding
○ Record information about activities
● Give instructions
○ Restore a system back to preferred condition
○ Persuade a perpetrator to go elsewhere
○ Compensate for a failed control

Congratulations, you have answered the question correctly.

☑ What is the primary purpose of corrective controls? *Score: 1*

○ Stop unwanted activity from succeeding
○ Record information about activities
○ Give instructions
● Restore a system back to preferred condition
○ Persuade a perpetrator to go elsewhere
○ Compensate for a failed control

Congratulations, you have answered the question correctly.

☑ What is the purpose of the dot and slash in front of the filenames in the PowerShell scripts and when executing PowerShell scripts? *Score: 1*

○ Allow for administrator execution
● Reference the current working directory
○ To set the security content of the process
○ For avoiding detection by an IDS

Congratulations, you have answered the question correctly.