Chapter 9 Lab

Once again, we’ve covered a lot of important concepts in a short amount of time. The best way to cement your new knowledge is to put it to immediate use. We recommend doing the following tasks in order, because they build on each other to help remind you of what you’ve learned and to help you find practical ways to use that knowledge.

To make this a bit trickier, We’re going to force you to consider the Get-ADComputer command. This command is installed on any Windows Server 2008 R2 or later domain controller – but you don’t need one! You only need to know two things:

* The Get-ADComputer command has a –filter parameter; running Get-ADComputer –filter \* will retrieve all computer objects in the domain.
* Domain computer objects have a Name property, which contains the computer’s host name.

With that in mind, complete these tasks:

Note You’re not being asked to run these commands. Instead, you’re being asked if these commands will function or not, and why. You’ve been told how Get-ADComputer works, and what it produces; you can read the help to discover what other commands expect and accept.

1. Would the following command work to retrieve a list of installed hotfixes from all domain controllers in the specified domain? Why or why not? Write out an explanation, similar to the ones we provided earlier in this chapter.

Get-Hotfix -computerName (get-adcomputer -filter \* |

Select-Object -expand name)  
  
This should work because the nested Get-ADComputer expression will return a collection of computernames and the –Computername parameter can accept an array of valies.

1. Would this alternative command work to retrieve the list of hotfixes from the same computers? Why or why not? Write out an explanation, similar to the ones we provided earlier in this chapter.

get-adcomputer -filter \* |

Get-HotFix  
  
This won’t work because Get-Hotfix doesn’t accept any parameters by value. Although it will accept –Computername by property name, but this command isn’t doing that.

1. Would this third version of the command work to retrieve the list of hotfixes from the domain controllers? Why or why not? Write out an explanation, similar to the ones I provided earlier in this chapter.

get-adcomputer -filter \* |

Select-Object @{l='computername';e={$\_.name}} |

Get-Hotfix  
  
This should work. The first part of the expression is writing a custom object to the pipeline that has a Computername property. This property can be bound to the Computername parameter in Get-Hotfix because it accepts pipeline binding by property name.

1. Write a command that uses pipeline parameter binding to retrieve a list of running processes from every computer in an AD domain. Don’t use parentheses.  
     
   get-adcomputer -filter \* |  
   Select-Object @{l='computername';e={$\_.name}} | Get-Process
2. Write a command that retrieves a list of installed services from every computer in an AD domain. Don’t use pipeline input; instead use a parenthetical command (a command in parentheses).   
     
   Get-Service –Computername (get-adcomputer -filter \* | Select-Object –expandproperty name)
3. Sometimes Microsoft forgets to add pipeline parameter binding to a cmdlet. For example, would the following command work to retrieve information from every domain controller in the domain? Write out an explanation, similar to the ones we provided earlier in this chapter.

get-adcomputer -filter \* |

Select-Object @{l='computername';e={$\_.name}} |

Get-WmiObject -class Win32\_BIOS

This will not work. The Computername parameter in Get-WMIObject doesn’t take any pipeline binding.