

## Unit 7 Homework

### 1. Task 1

Name	GPO Status	WMI Filter
Account Lockout	Enabled	None
Default Domain Controller...	Enabled	None
Default Domain Policy	Enabled	None
No Control Panel	Enabled	None
No LLMNR	Enabled	None
PowerShell Logging	Enabled	None

### 2. Task 2

The screenshot shows the Group Policy Management console for the Forest: GOODCORP.NET. The left pane shows the hierarchy: Group Policy Management > Forest: GOODCORP.NET > Domains > GOODCORP.NET > Group Policy Objects > Account Lockout. The right pane shows the 'Account Lockout' policy configuration with tabs for Scope, Details, Settings, Delegation, and Status. The 'Settings' tab is active, showing the policy is 'Computer Configuration (Enabled)'. Below this, there are sections for 'Policies', 'Windows Settings', 'Security Settings', and 'Account Policies/Account Lockout Policy'. The 'Account Policies/Account Lockout Policy' section is expanded, showing a table of settings:

Policy	Setting
Account lockout duration	15 minutes
Account lockout threshold	10 invalid logon attempts
Reset account lockout counter after	15 minutes

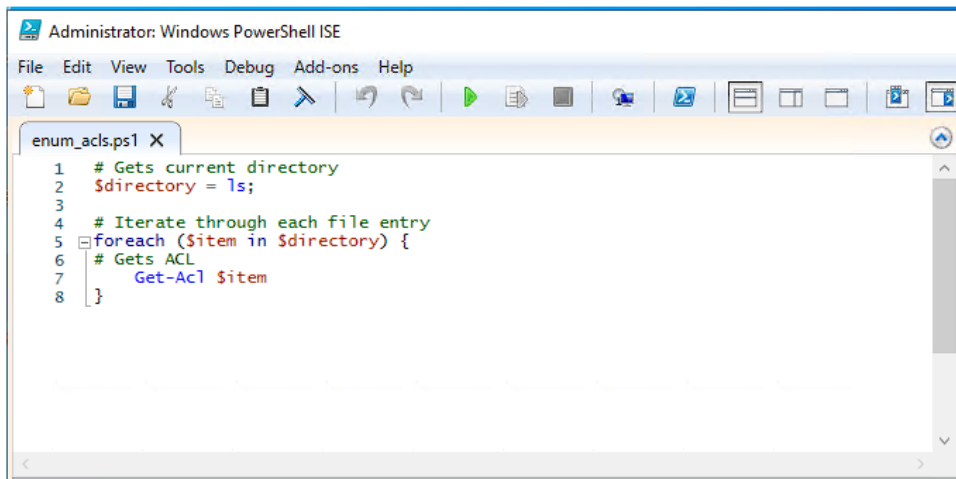
Below the table, there is a section for 'User Configuration (Enabled)' which shows 'No settings defined.'

### 3. Task 3

The screenshot shows the Windows PowerShell console with the command 'Select an item to view its description.' executed. The output is a table showing the status of various PowerShell settings:

Setting	State
Turn on Module Logging	Enabled
Turn on PowerShell Script Block Logging	Enabled
Turn on Script Execution	Enabled
Turn on PowerShell Transcription	Enabled
Set the default source path for Update-Help	Not configured

#### 4. Task 4



The screenshot shows the Windows PowerShell ISE interface. The title bar reads "Administrator: Windows PowerShell ISE". The menu bar includes "File", "Edit", "View", "Tools", "Debug", "Add-ons", and "Help". The toolbar contains various icons for file operations, editing, and execution. The script editor displays a file named "enum\_acls.ps1" with the following PowerShell code:

```
1 # Gets current directory
2 $directory = ls;
3
4 # Iterate through each file entry
5 foreach ($item in $directory) {
6     # Gets ACL
7     Get-Acl $item
8 }
```

The script is designed to list the current directory and then iterate through each file entry to retrieve its ACL (Access Control List) using the Get-Acl cmdlet.