

## Using Set-PrintConfiguration PowerShell Scripts Examples

In today's technological world, managing and configuring printers efficiently is crucial for any organization. Windows provides a powerful tool called PowerShell, which allows system administrators to automate various tasks, including printer management. One of the cmdlets available in PowerShell is Set-PrintConfiguration, which enables administrators to configure printer settings programmatically. This article will provide factual and instructive information on how to utilize the Set-PrintConfiguration cmdlet with examples adapted for the Windows environment.

### Examples:

1. **Changing the Default Printer Configuration:** To change the default printer configuration using PowerShell, you can use the Set-PrintConfiguration cmdlet along with the -PrinterName parameter. For example:

```
Set-PrintConfiguration -PrinterName "Printer1" -DuplexingMode OneSided
```

This command will set the printer named "Printer1" to use one-sided printing by default.

2. **Modifying Printer Preferences:** Printer preferences can be modified using the Set-PrintConfiguration cmdlet with the -PrinterName and -Preference parameter. For instance:

```
Set-PrintConfiguration -PrinterName "Printer2" -Preference "color"
```

This command will set the printer named "Printer2" to print in color as the default preference.

3. **Configuring Printer Permissions:** PowerShell allows you to configure printer permissions programmatically. The Set-PrintConfiguration cmdlet can be used with the -PrinterName and -PermissionSDDL parameters to set printer permissions. Here's an example:

```
Set-PrintConfiguration -PrinterName "Printer3" -PermissionSDDL "O:BAG  
:SYD:(A;;FA;;;SY)(A;;FR;;;BA)"
```

This command will set the printer named "Printer3" with the specified security descriptor definition language (SDDL) string, granting full access to the system and read access to built-in administrators.