

## How to use this sample

This sample shows how to connect a network drive using the "Run external application" action of our product FA-Panel, which can be used when the network connection is broken at OS startup.

- This sample works with Ver6 Rev3 or later.
- This sample uses the standard Windows Powershell.
- This sample runs on Windows 8 or later.

## File structure of the samples

The file structure of this sample is as shown in the table below.

server	
redundancy	
security	
make_passwordfile.ps1	Password creation shell
run.bat	Batch to launch shell
shell_networkdrive.ps1	Network drive connection shell
server.txt	Sample Server Configuration File

## About the sample server configuration file

In the sample server configuration file, the following settings are made.

- Setting "Environment Variable"
- Configuring the "Run external application" action
- Setting up events for action calls

### Setting "Environment Variable"

Define an environment variable if the setting values are different between the main PC and the sub PC. This time, what we will define in the environment variable is the network folder to be connected when duplexing. The network folder connected from the main PC is the sub PC's folder, and the network folder connected from the sub PC is the main PC's folder.

For example

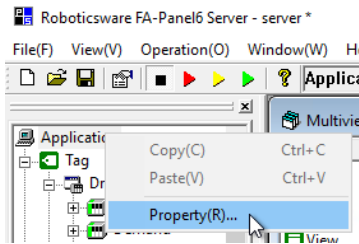
IP address of the main PC	192.168.1.1
IP address of the sub PC	192.168.1.2
Project folder	project

in this case

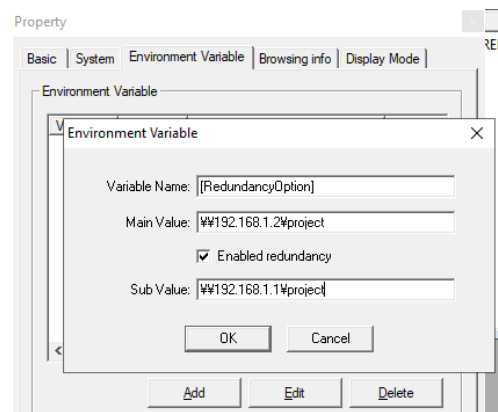
Network folders to be connected on the main	¥¥192.168.1.2¥project
Network folders to be connected by sub PCs	¥¥192.168.1.1¥project

Set the "Environment variable" as follows.

1. Right-click on "Application" to display its properties.

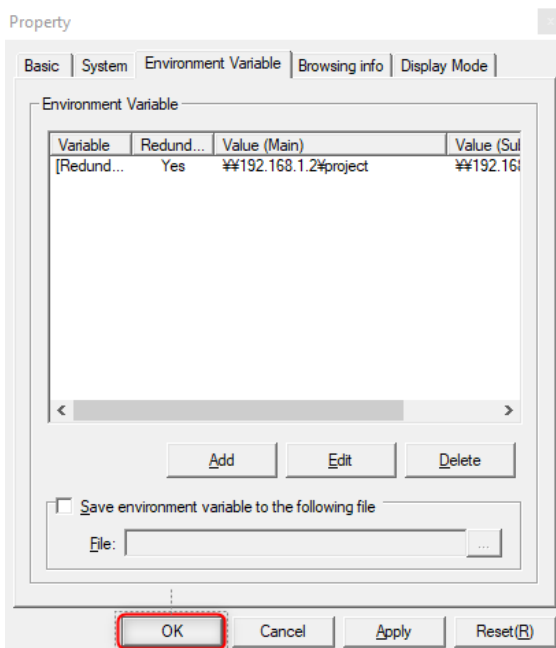


2. Select the "Environment Variable" tab, enter the following information using the Add button, and click the OK button.



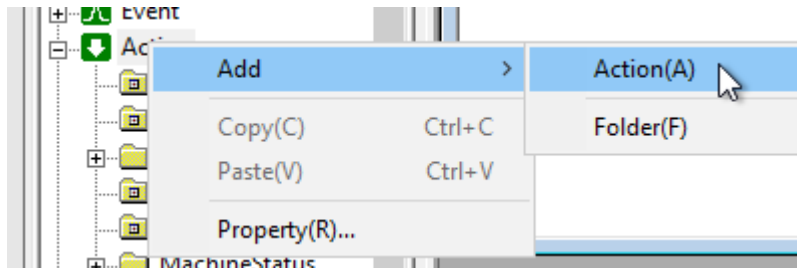
Variable Name	[RedundancyOption]
Main Value	¥¥192.168.1.2¥project
Enabled redundancy	Check the box
Sub Value	¥¥192.168.1.1¥project

3. Click the OK button.

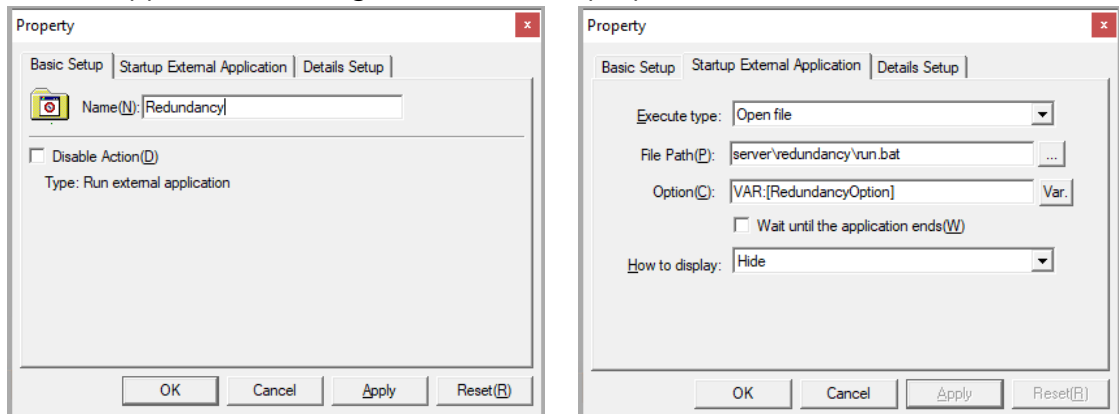


### Configuring the "Run external application" action

1. Right-click on "Action" and select "Add-Action".



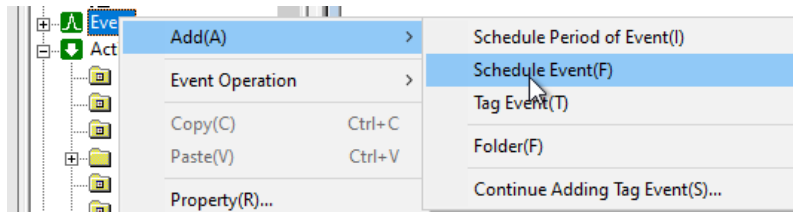
2. Add the "Run external application" action and configure the "Basic Setup" and "Launch External Application" setting from the action properties.



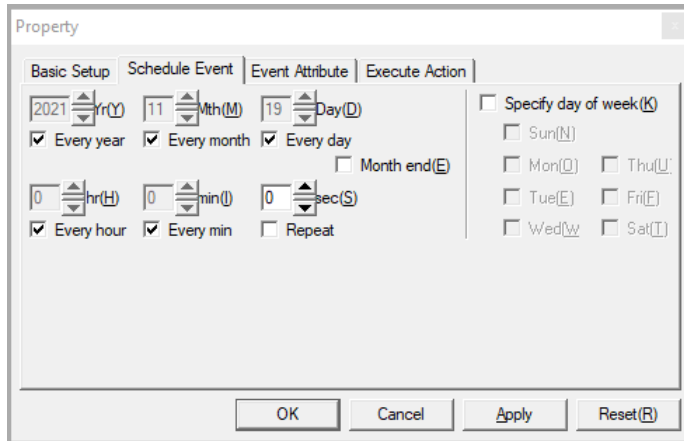
Name	Redundancy
Execute type	Select "Open file".
File Path	Specify the batch "run.bat" for launching the shell
Option	Click on the "Var." button and select the environment variable that you have registered. The value of the environment variable selected here will be passed as a parameter to the batch for launching the shell.
How to display	Select "Hide".

## Setting up events for action calls

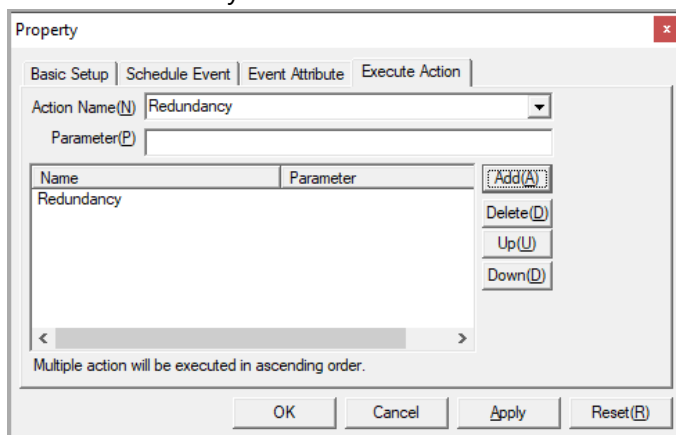
1. Right-click on "Event" and select "Add - Schedule Event".



2. Set an event for every minute as a timed event.

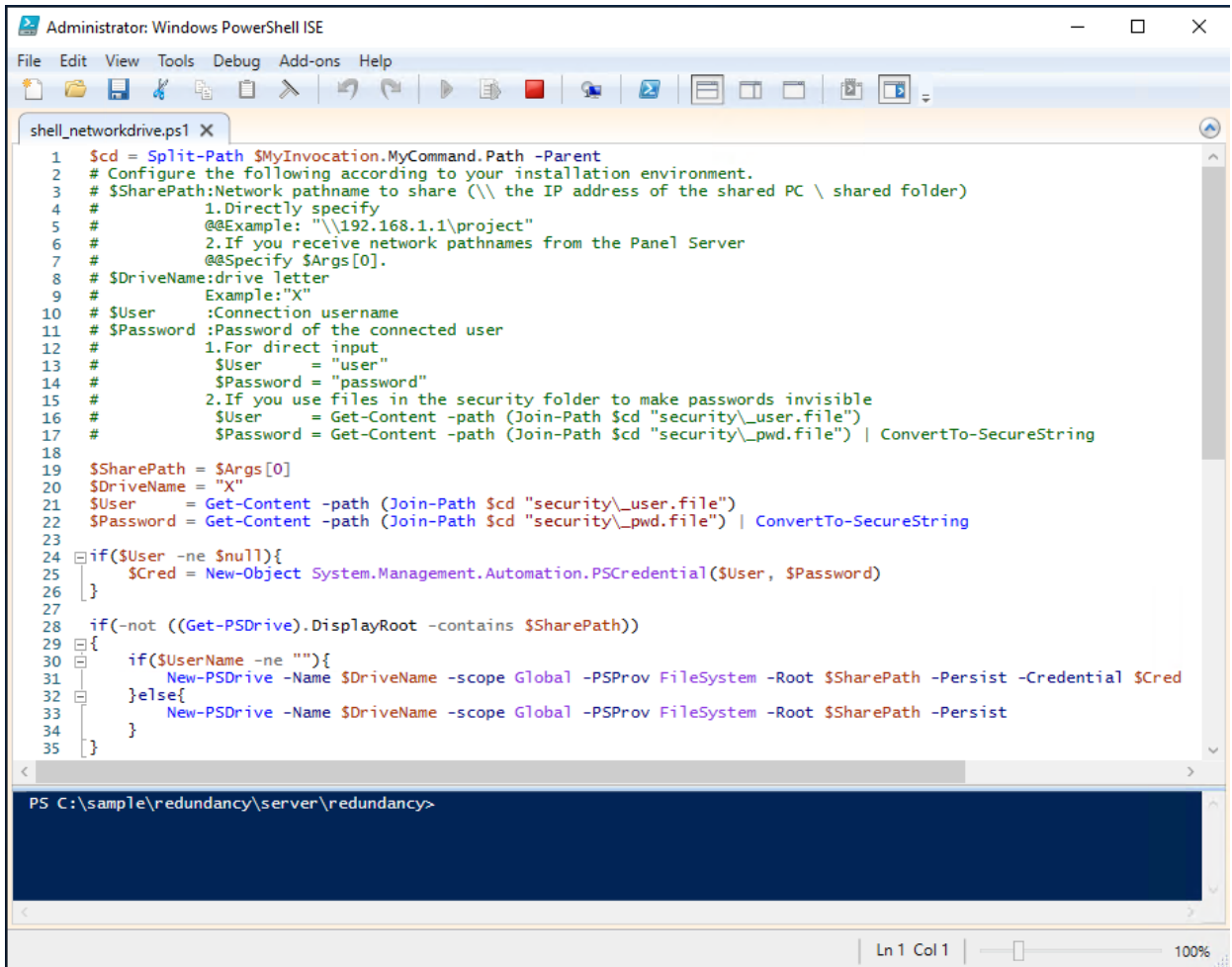


3. Add the action you created as the "Execution Action".



## About the Network Drive Connection Shell

Edit "shell\_networkdrive.ps1" to suit your environment. The changes to be made are described as comments.



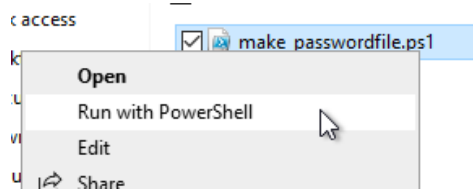
```
1 $cd = Split-Path $MyInvocation.MyCommand.Path -Parent
2 # Configure the following according to your installation environment.
3 # $SharePath: Network pathname to share (\\ the IP address of the shared PC \ shared folder)
4 # 1. Directly specify
5 #   @@Example: "\\192.168.1.1\project"
6 # 2. If you receive network pathnames from the Panel Server
7 #   @@Specify $Args[0].
8 # $DriveName: drive letter
9 #   Example: "X"
10 # $User : Connection username
11 # $Password : Password of the connected user
12 # 1. For direct input
13 #   $User = "user"
14 #   $Password = "password"
15 # 2. If you use files in the security folder to make passwords invisible
16 #   $User = Get-Content -path (Join-Path $cd "security\_user.file")
17 #   $Password = Get-Content -path (Join-Path $cd "security\_pwd.file") | ConvertTo-SecureString
18
19 $SharePath = $Args[0]
20 $DriveName = "X"
21 $User = Get-Content -path (Join-Path $cd "security\_user.file")
22 $Password = Get-Content -path (Join-Path $cd "security\_pwd.file") | ConvertTo-SecureString
23
24 if($User -ne $null){
25     $Cred = New-Object System.Management.Automation.PSCredential($User, $Password)
26 }
27
28 if(-not ((Get-PSDrive).DisplayRoot -contains $SharePath))
29 {
30     if($UserName -ne ""){
31         New-PSDrive -Name $DriveName -scope Global -PSProv FileSystem -Root $SharePath -Persist -Credential $Cred
32     }else{
33         New-PSDrive -Name $DriveName -scope Global -PSProv FileSystem -Root $SharePath -Persist
34     }
35 }
```

PS C:\sample\redundancy\server\redundancy>

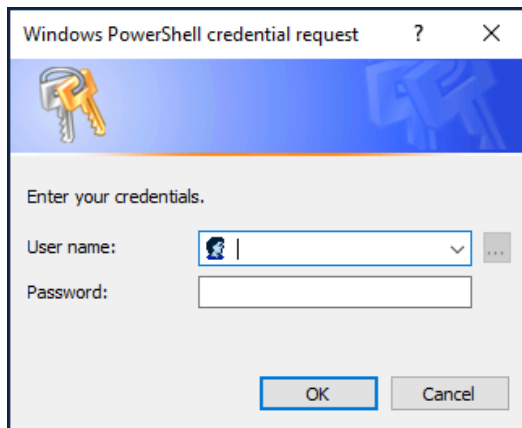
In the default description, the "X" drive is connected to the network folder defined in the environment variable. The user/password to be used in this case is the one in the authentication file. How to create an authentication file is explained in the next page.

## How to create an authentication file

1. Right-click on "make\_passwordfile.ps1" in this folder and select "Run with PowerShell".



2. Enter the user name and password to be used for authentication, and then click the "OK" button.



3. A user file "\_user.file" and a password file "\_pwd.file" will be created in this folder. The contents of the password file will be encrypted.

### Hint:

The authentication file for this procedure needs to be generated each time the computer to be deployed and the login user changes. For this reason, please execute this procedure once for the first time after logging in on the OS of the computer that will actually be used. An authentication file generated on a different computer cannot be used.