

Решение проблемы автоматического выхода из RDP-сессии в SCADA на Windows Server 2016

Описание проблемы

На сервере Windows Server 2016 работает SCADA-система для управления вентиляционным оборудованием. При бездействии пользователя в RDP-сессии программа автоматически завершает сеанс. Разработчики SCADA рекомендуют имитировать активность пользователя (движение мыши или нажатие клавиши) не реже одного раза в час.

Почему не подходит PowerToys?

PowerToys (в том числе версия 0.97.2) — это набор утилит для повышения продуктивности (FancyZones, PowerRename, поиск и т.п.), но он не предназначен для автоматизации ввода. Для имитации действий пользователя оптимально использовать встроенные средства Windows, например, PowerShell.

Решение: PowerShell-скрипты для имитации активности

Вариант 1. Имитация движения мыши (рекомендуется)

Скрипт перемещает курсор на 1 пиксель вправо и возвращает обратно. Это минимальное воздействие, не мешающее работе оператора, но достаточное для сброса таймера бездействия.

```
powershell
```

```
# keep-alive.ps1
```

```
Add-Type -AssemblyName System.Windows.Forms
```

```
while ($true) {
```

```
    # Сохраняем текущую позицию мыши
```

```
    $originalPos = [System.Windows.Forms.Cursor]::Position
```

```
    # Двигаем мышь на 1 пиксель вправо
```

```
    [System.Windows.Forms.Cursor]::Position = New-Object System.Drawing.Point(
```

```
        ($originalPos.X + 1), $originalPos.Y
```

```
)
```

```
    Start-Sleep -Milliseconds 100
```

```
    # Возвращаем мышь обратно
```

```
    [System.Windows.Forms.Cursor]::Position = $originalPos
```

```
    # Ждём 55 минут (3300 секунд) перед следующим циклом
```

```
    Start-Sleep -Seconds 3300
```

```
}
```

Вариант 2. Имитация нажатия клавиши (Scroll Lock)

Альтернативный способ – программное нажатие клавиши Scroll Lock, которая обычно не влияет на работу приложений.

```
powershell
# keep-alive-keys.ps1
Add-Type -TypeDefinition @"
using System;
using System.Runtime.InteropServices;
public class Keyboard {
    [DllImport("user32.dll")]
    public static extern void keybd_event(byte bVk, byte bScan, uint dwFlags, UIntPtr
dwExtraInfo);

    public static void PressScrollLock() {
        const byte VK_SCROLL = 0x91;
        const uint KEYEVENTF_EXTENDEDKEY = 0x0001;
        const uint KEYEVENTF_KEYUP = 0x0002;

        keybd_event(VK_SCROLL, 0, KEYEVENTF_EXTENDEDKEY, UIntPtr.Zero);
        System.Threading.Thread.Sleep(50);
        keybd_event(VK_SCROLL, 0, KEYEVENTF_EXTENDEDKEY | KEYEVENTF_KEYUP, UIntPtr.Zero);
    }
}
"@

while ($true) {
    [Keyboard]::PressScrollLock()
    Start-Sleep -Seconds 3300 # 55 минут
}
```

Развертывание решения

1. Создание файла скрипта

Сохраните один из приведённых выше скриптов в папку, например, C:\Scripts\keep-alive.ps1.

2. Разрешение выполнения скриптов PowerShell (если ещё не настроено)

Запустите PowerShell от имени администратора и выполните:

```
powershell
Set-ExecutionPolicy RemoteSigned -Scope LocalMachine -Force
```

3. Ручной запуск (для тестирования)

В открытой RDP-сессии выполните:

```
powershell
powershell.exe -NoProfile -WindowStyle Hidden -File "C:\Scripts\keep-alive.ps1"
```

Скрипт будет работать в фоне. Чтобы остановить – закройте окно PowerShell или убейте процесс.

4. Автоматический запуск при входе в систему (через Планировщик заданий)

Для надёжности настройте автоматический запуск скрипта после каждого входа пользователя в RDP-сессию.

Создайте задание с помощью PowerShell (замените ВАШ_ЛОГИН на имя пользователя, под которым выполняется RDP-сессия):

powershell

```
$action = New-ScheduledTaskAction -Execute "PowerShell.exe" `
-Argument "-NoProfile -WindowStyle Hidden -File C:\Scripts\keep-alive.ps1"
```

```
$trigger = New-ScheduledTaskTrigger -AtLogOn -User "ВАШ_ЛОГИН"
```

```
$principal = New-ScheduledTaskPrincipal -UserId "ВАШ_ЛОГИН" `
-LogonType Interactive -RunLevel Highest
```

```
Register-ScheduledTask -TaskName "SCADA KeepAlive" `
-Action $action -Trigger $trigger -Principal $principal
```

После этого задание появится в планировщике и будет запускать скрипт каждый раз при входе указанного пользователя.

Важные замечания и рекомендации

Аспект	Рекомендация
RDP-сессия	Скрипт должен запускаться внутри активной RDP-сессии того пользователя, который работает с SCADA. Не запускайте его как службу от SYSTEM – это не даст эффекта.
Безопасность	Используйте минимально необходимые привилегии. Запускайте скрипт от имени обычного пользователя, без прав администратора, если это возможно.
Надёжность	Планировщик заданий с триггером «При входе в систему» гарантирует автозапуск после перезагрузки сервера.
Тестирование	Перед внедрением проверьте, что скрипт действительно двигает курсор (или нажимает клавишу) и предотвращает выход из сессии. Рекомендуется временно уменьшить интервал ожидания до 1-2 минут для быстрой проверки.
Ресурсы	Скрипт потребляет минимум ресурсов (один цикл в час с короткой паузой).
Если .NET недоступен	В крайнем случае можно использовать сторонние утилиты, например, <code>nircmd.exe</code> (циклический вызов <code>nircmd.exe movecursor 1 0</code>).

Альтернативный вариант с nircmd.exe

Если по какой-то причине использование .NET нежелательно, можно применить утилиту nircmd (не входит в состав Windows, требует отдельной загрузки):

```
powershell  
while ($true) {  
    & "C:\tools\nircmd.exe" movecursor 1 0  
    Start-Sleep -Seconds 3300  
}
```

Заключение

PowerShell предоставляет простые и эффективные средства для имитации действий пользователя. Предложенный скрипт с движением мыши на 1 пиксель является оптимальным решением: он не **干扰** работу оператора, не требует дополнительных установок и легко автоматизируется через Планировщик заданий. Выбор конкретного варианта (мышь или клавиша) зависит от особенностей SCADA – проверьте, какой способ гарантированно сбрасывает таймер бездействия.