processrobot

Multi-Solobot hosting on one Server

Enterprise Robotic Process Automation 2019

Contents

Multi-Solobot hosting on one Server

High Density refers to ProcessRobot's ability to have multiple Robots executing processes on a single Windows Server.

In non-high-density environments, the flow of execution is the following:

On machine A, Solobot A logs in and executes Processes using User's A credentials:

SoloBot A → Machine A → User A (Autologin)

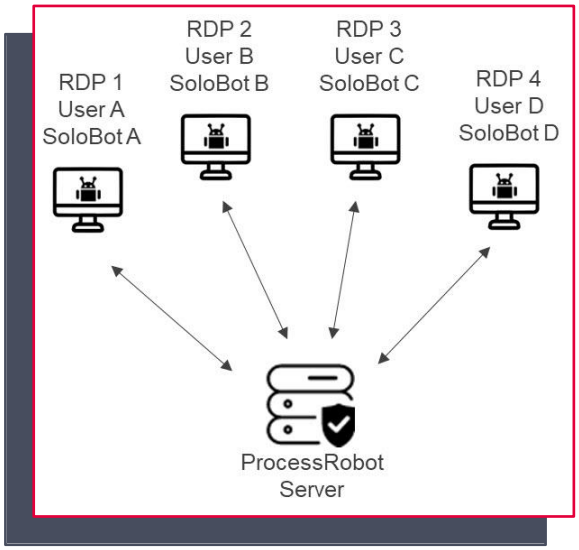
In a High-Density environment, it is possible to assign multiple SoloBots on a single Windows Server machine using different User credentials. Each User will launch an RDP session on the Server, as per the following scheme:

SoloBot A → Machine A → Remote Desktop session from User A

SoloBot B → Machine A → Remote Desktop session from User B

SoloBot C → Machine A → Remote Desktop session from User C

…and so on.

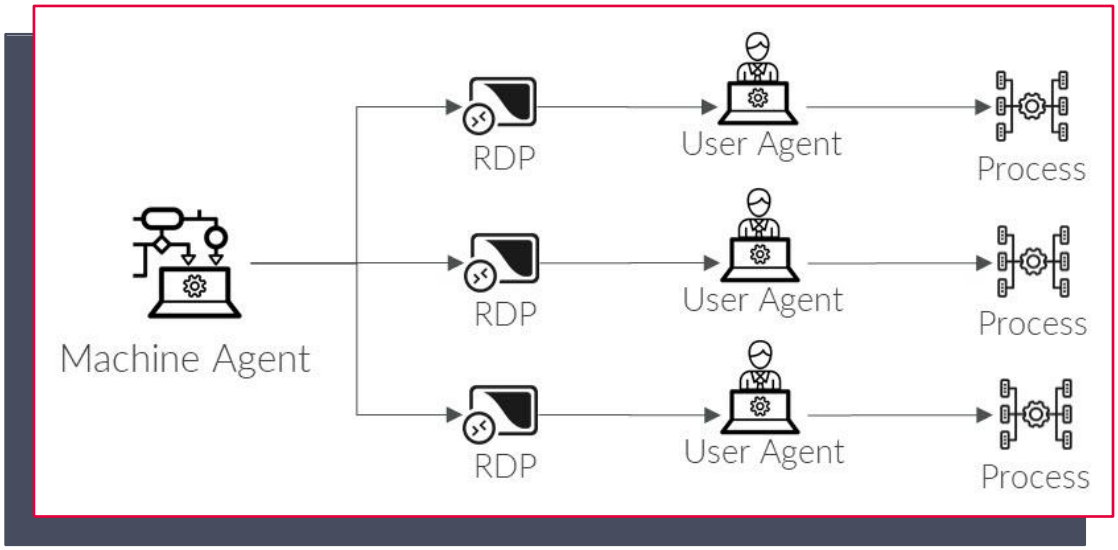


In this regard, ProcessRobot will allow multiple SoloBots on the same Server to be connected under different Users; in the same way as when one initiates a remote desktop session on a server.

The instances will be spawned from a single installation per server. A single installation of the SoloBot software is required on the workstation. The multiple Robots and Users will then be declared in the Control Desk.

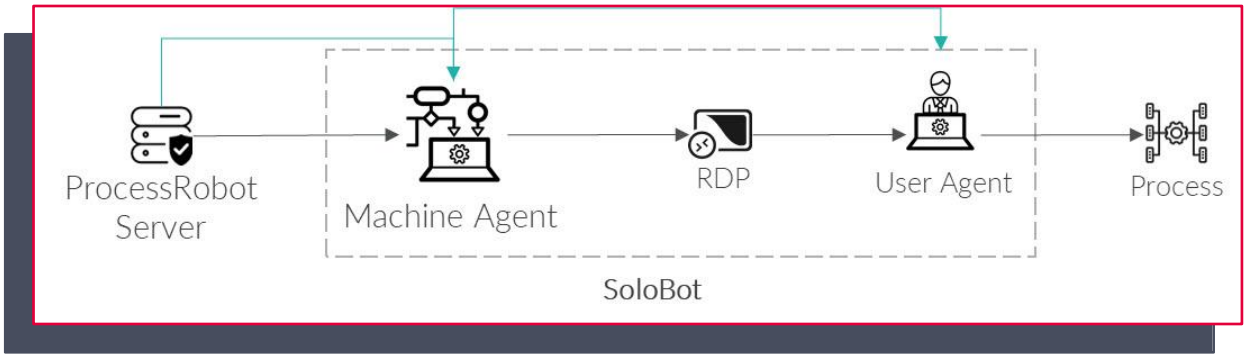
This new feature will utilize the machine in the best manner, in order to run multiple Processes on multiple SoloBots, taking maximum advantage of the available hardware resources.

High Level Overview



A single Machine Agent on the Windows Server SoloBot machine will be able to establish multiple RDP connections to itself, one connection per registered SoloBot. Login in each of these connections will be achieved by using the credentials of the Active Directory User account assigned to each SoloBot.

Process Flow Overview



The Process Flow of initiating the execution of a Process on a SoloBot may follow one of two possible paths, depending on whether the Active Directory User is already logged in.

If the User is logged in:

The ProcessRobot Server sends a signal to the Machine Agent, which in turn sends a signal to the User Agent, which runs the Process.

If the User is NOT logged in:

The ProcessRobot Server sends a signal to the Machine Agent, which initiates a RDP session (to machine it resides), logs in with the User's credentials, starts the User Agent, which then runs the Process.

NOTES

1. High density scenarios are supported only for different User instances,
2. In cases where Users are logged in, ProcessRobot supports the use of Sidebots as well

Use of Credentials

The technology used to establish the RDP connection is the .NET Microsoft Library: **IMsRdpClient**.

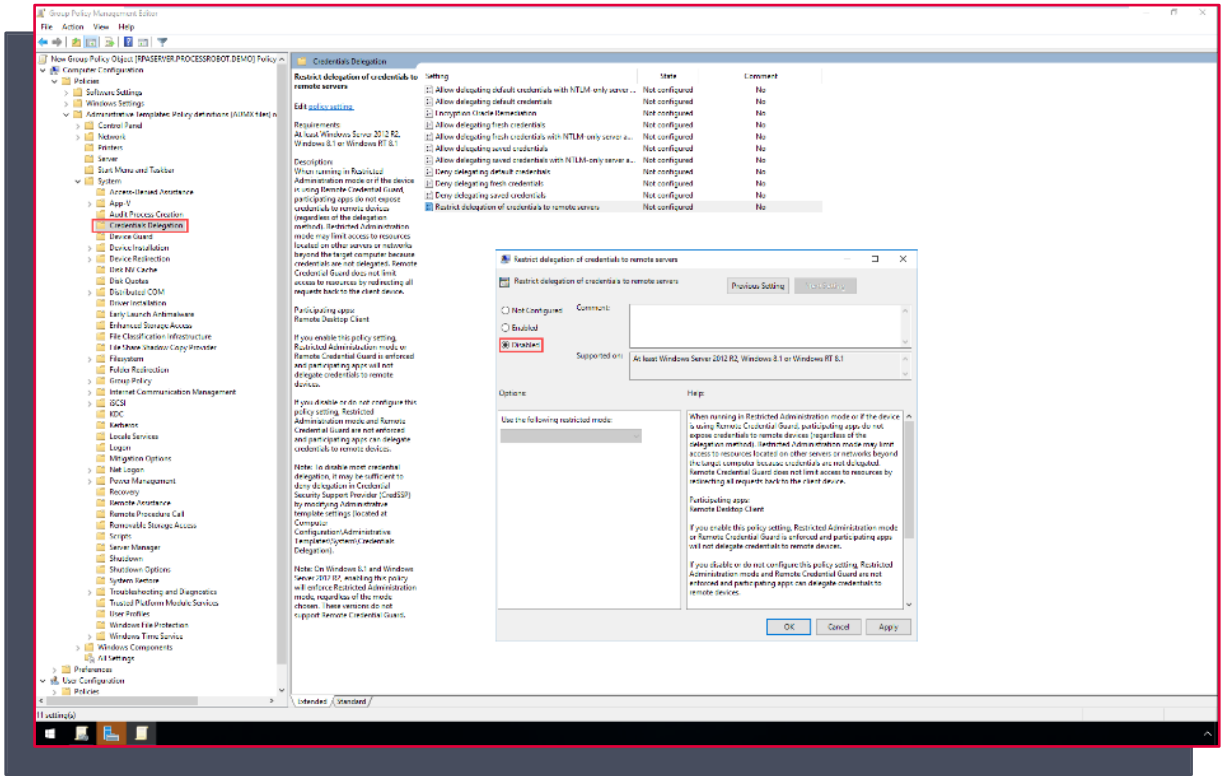
ProcessRobot does not store the Active Directory Users' credentials, neither username nor password via the Credential Manager. Instead, the credentials are supplied programmatically to the RDP library. Consequently, no credential file is created, as ProcessRobot does not store these credentials anywhere.

Group Policy Requirements

Enabling the multi-tenant approach requires changes to the following two Group Policies:

Policy 1 - "Restrict delegation of credentials to remote server"

The "Restrict delegation of credentials to remote server" policy must be ***disabled*** since the credentials are sent from the Machine Agent on the Window Server machine to the .NET Microsoft RDP library on the same machine in order to initialize RDP connections. In essence, credentials are delegated to the same machine that sends them.

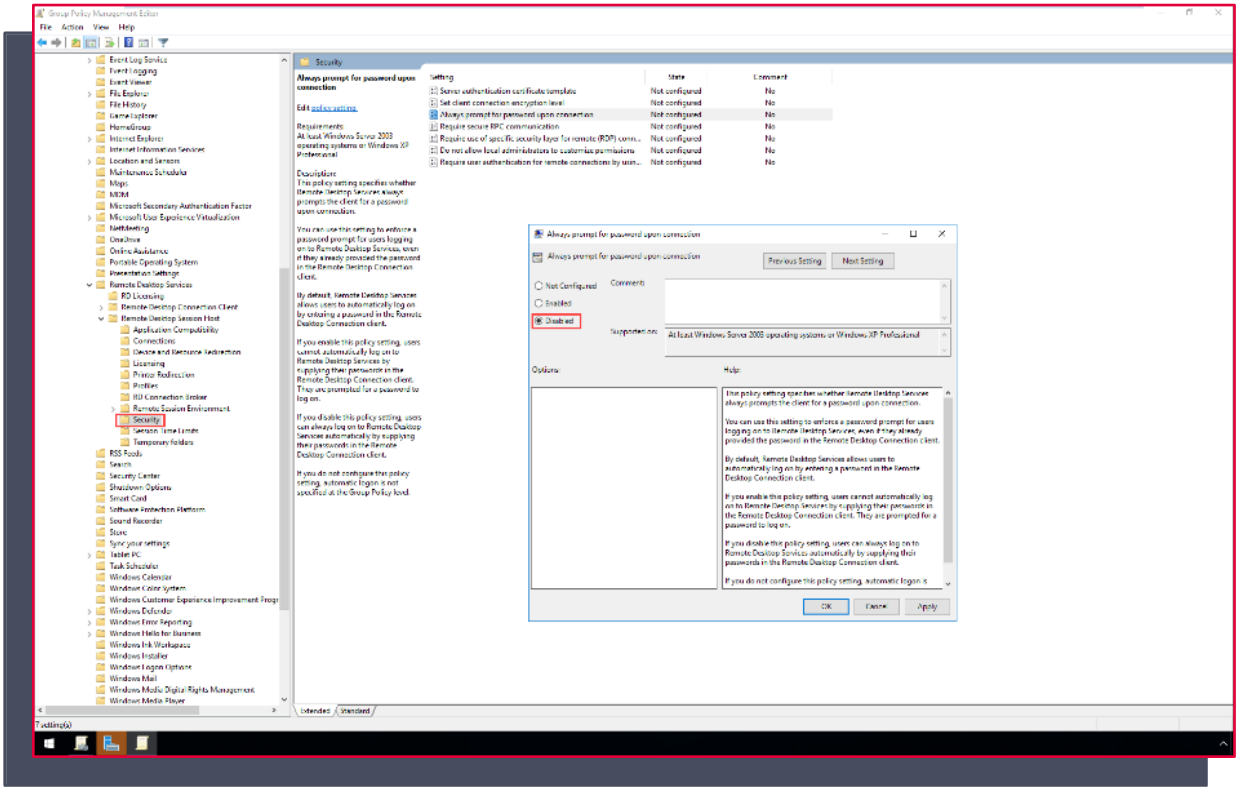


The path to disable the above policy is:

Gpedit.msc > Computer Configuration > Administrative Templates > System > Credentials Delegation > Restrict delegation of credentials to remote servers: **Disabled**

Policy 2 - "Always prompt for password upon connection"

The "Always prompt for password upon connection" policy must be ***disabled***. Otherwise, this policy will interfere with the SoloBots' autologin feature. As mentioned previously, the credentials are passed programmatically to the RDP library. If this policy is enabled, the system will again require the User's password, thus preventing the autologin.



The path to disable the above policy is:

Gpedit.msc > Computer Configuration > Administrative Templates > System > Windows Components > Remote Desktop Services > Remote Desktop Session Host > Security > Always prompt for password upon connection: **Disabled**