

# VDI

Projetos e ideias relacionadas a Infraestrutura de Desktop Virtual

- [VDI Remoto com Raspberry Pi e VPN Forcepoint](#)

# VDI Remoto com Raspberry Pi e VPN Forcepoint

Versão do Hardware tratado no documento: Raspberry Pi 3B+

Versão do Software tratado no documento: Raspbian GNU/Linux 10 (buster)

## 1.0) Introdução

Diante da perspectiva crescente de trabalho remoto e de acesso a poder de computação, dados e serviços independente do local, iniciamos experimentos para proporcionar estes recursos através de técnicas de VDI ( Infraestrutura de Desktop Virtual) utilizando o Raspberry Pi e Firewall Forcepoint de forma a proporcionar um acesso barato e livre aos recursos internos da instituição.

Através da solução abaixo, podemos criar um acesso a uma estação Windows remota e segura utilizando apenas um Raspberry Pi e alguns softwares Livres pré configurados.

O acesso remoto a uma estação Windows exigirá a aquisição de licenciamento para tal com a Microsoft.

Os experimentos demonstrados abaixo fazem referencia a conexão de VDI com estação da Reitoria do IFSP e deve ser adaptada para outros ambientes.

## 2.0) Implantação em RP3B+ (Raspberry Pi 3B+) com interface gráfica pré instalada

Os procedimentos abaixo permitem a implantação do serviço em um RP3B+ com sistema operacional configurado **com** interface gráfica em sua instalação.

Esta solução possui como pré-requisito a implantação da VPN C2S com a reitoria, os procedimentos para tal podem ser encontrados [AQUI](#).

### 2.1) Instalação dos pacotes necessários

Instale o software necessário para realizar a conexão remota com Windows via protocolo RDP com os comandos abaixo:

```
sudo apt-get update && sudo apt-get install rdesktop
```

## 2.2) Configuração e inicialização do serviço

Agora vamos criar um serviço do SystemD que ficará encarregado de verificar se a conexão VPN já está pronto para utilização e iniciará o serviço de conexão Remote Desktop:

Crie e edite o arquivo **"/lib/systemd/system/remote.service"**:

```
sudo nano /lib/systemd/system/remote.service
```

Adicione o conteúdo a seguir no arquivo:

```
[Unit]
Description=Remote Desktop
Requires=network-online.target

[Service]
Type=idle
Environment=DISPLAY=:0
Environment=XAUTHORITY=/home/pi/.Xauthority
ExecStart=/usr/bin/rdesktop -k pt-br -n <HOSTNAME-DO-RASPBERRY-PI> -g 100% -z -r sound:local -x 0x80 -u
<USUARIO-REMOTO> -P -f <IP-DO-SERVIDOR-DE-DESTINO>
ExecStartPre=/bin/sh -c 'until ping -c1 <IP-DO-SERVIDOR-DE-DESTINO>; do sleep 1; done;'
Restart=always
RestartSec=3

[Install]
WantedBy=graphical.target
```

Não se esqueça de alterar os parâmetros para seu ambiente.

O serviço acima fará todas as checagens necessárias e também irá reiniciar o terminal remoto caso ele seja fechado pelo cliente.

Agora temos que instalar o serviço e inicializa-lo:

Reload das configurações do serviço:

```
sudo systemctl daemon-reload
```

Instalação do serviço:

```
sudo systemctl enable remote.service
```

Iniciar o serviço:

```
systemctl start remote.service
```

DICA: Você pode verificar o status do serviço trocando o "start" do comando anterior por "status"

Está feito! ao reiniciar o equipamento, se estiver tudo certo, o sistema irá inicializar e logo em seguida realizar a conexão RDP com a estação remota!

## 3.0) Implantação em RP3B+ (Raspberry Pi 3B+) sem interface gráfica pré instalada

Fonte: [SITE](#)

Até o presente momento esta solução não foi validada completamente pela nossa equipe.

Os procedimentos abaixo permitem a implantação do serviço em um RP3B+ com sistema operacional configurado **sem** interface gráfica em sua instalação.

### 3.1) Instalação dos pacotes necessários

Instale o software necessário para realizar a conexão remota com Windows via protocolo RDP com os comandos abaixo:

```
sudo apt-get update && sudo apt-get install xinit xserver-xorg xserver-xorg-video-fbdev rdesktop
```

### 3.2) Configuração e inicialização do serviço

Crie o arquivo **".xinitrc"** que irá servir para executarmos o Remote Desktop automaticamente, uma vez que o ambiente gráfico inicie:

```
nano /home/pi/.xinitrc
```

Adicione a seguinte linha neste arquivo e salve (Substitua o IP com o endereço IP do seu Desktop, lembre de configura-lo com um IP Fixo para que você não tenha que ficar mudando isso no futuro):

```
exec /usr/bin/rdesktop -k pt-br -n <HOSTNAME-DO-RASPBERRY-PI> -g <RESOLUCAO-DESEJADA> -z -r  
sound:local -x 1 -u <USUARIO-REMOTO> -f <IP-DO-SERVIDOR-DE-DESTINO>
```

Agora vamos criar um serviço do SystemD que ficará encarregado de verificar se a conexão VPN já está pronto para utilização e iniciará o serviço de conexão Remote Desktop:

Crie e edite o arquivo **"/lib/systemd/system/remote.service"**:

```
sudo nano /lib/systemd/system/remote.service
```

Adicione o conteúdo a seguir no arquivo:

```
[Unit]  
Description=Remote Desktop  
Requires=network-online.target  
  
[Service]  
Type=idle  
Environment=DISPLAY=:0  
Environment=XAUTHORITY=/home/pi/.Xauthority  
ExecStart=xinit /home/pi/.xinitrc  
ExecStartPre=/bin/sh -c 'until ping -c1 <IP-DO-SERVIDOR-DE-DESTINO>; do sleep 1; done;'  
Restart=always  
RestartSec=3  
KillMode=process  
TimeoutSec=infinity  
  
[Install]  
WantedBy=graphical.target
```

Não se esqueça de alterar os parâmetros para seu ambiente.

O serviço acima fará todas as checagens necessárias e também irá reiniciar o terminal remoto caso ele seja fechado pelo cliente.

Agora temos que instalar o serviço e inicializa-lo:

Reload das configurações do serviço:

```
sudo systemctl daemon-reload
```

Instalação do serviço:

```
sudo systemctl enable remote.service
```

Iniciar o serviço:

```
systemctl start remote.service
```

DICA: Você pode verificar o status do serviço trocando o "start" do comando anterior por "status"

Está feito! ao reiniciar o equipamento, se estiver tudo certo, o sistema irá inicializar e logo em seguida realizar a conexão RDP com a estação remota!

## 4.0) Aumento do tempo de espera por login na estação remota

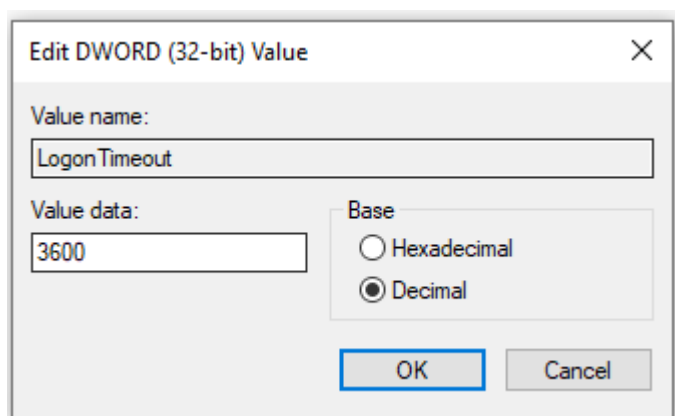
Fonte: [SITE](#)

Para que a tela de login fique esperando as credenciais por mais tempo (normalmente 1 minuto), você pode criar uma chave no registro do Windows.

Crie uma chave DWORD chamada **LogonTimeout** no container:

**Computer\HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp**

Coloque o valor da chave em segundos (3600 segundos = 1 hora):



Dessa forma quando o Raspberry Pi iniciar e a sessão RDP começar ele aguardará por 1 hora até que você faça o login.

Se você não logar em tempo, o rdesktop vai fechar, reinicie o Raspberry Pi ou o serviço remote.service:

```
sudo systemctl restart remote.service
```

## 5.0) Extras

### 5.1) Escala de tela

Se o seu Raspberry Pi não estiver preenchendo a tela inteira quando ligado a um monitor, exibindo barras pretas nas laterais, execute a seguinte alteração:

Altere o arquivo de configuração padrão do RP:

```
sudo nano /boot/config.txt
```

Em seguida descomente a linha abaixo e salve as alterações:

```
...  
# uncomment this if your display has a black border of unused pixels visible  
# and your display can output without overscan  
disable_overscan=1  
...
```

Reinicie o equipamento e a tela deve agora estar com a cobertura correta.


### 5.2) Login sem senha no RDP do Windows Server

Para que o login sem senha funcione no Windows server, de forma a apresentar a tela de login somente após a conexão RDP, execute as seguinte alteração no sistema:

Você deve **desabilitar** o "Network Level Authentication (NLA)" logo após ativar a permissão para acesso remoto via RDP:

## Advanced settings

### Configure Network Level Authentication

 ☐ Require computers to use Network Level Authentication to connect (recommended)

[Why allow connections only with Network Level Authentication?](#)

### External connections

[Learn how to allow remote connections from outside your local network](#)

### Remote Desktop port

Current Remote Desktop Port 3389

[Learn how to change the listening port for Remote Desktop](#)

## 5.3) Raspberry Pi reiniciando sem motivo

O superaquecimento pode ser uma das causas de reboot aleatório do equipamento, assim como uma fonte de energia insuficiente e/ou equipamentos que demandam muita energia ligados ao USB ou GPIO do equipamento, tais como HDs externos, etc.

Para verificar que houve throttling (redução de frequência do processador devido a temperatura), execute:

```
sudo vcgencmd get_throttled
```

Se o resultado for **0x0** está tudo bem neste quesito, caso não seja é melhor verificar a refrigeração do equipamento.