Access to Mac endpoint

- Parallels Secure Workspace

Resolution

Please note that this is not an officially supported solution from Parallels. This is a method to implement remote access, however usage of this method should be done at own risk. This article is designed to guide and assist with providing remote access to Mac client devices. This procedure can also be used for other VNC endpoint devices.

By default, connecting to a Mac device via Parallels Secure Workspace is not possible as Apple Macs do not make use of the standard RDP protocol. There are many possible solutions for connecting remotely to a Mac via Parallels Secure Workspace. Howeve, some are rather cumbersome and sometimes unreliable (xRDP, designed for Linux, but in some cases possible to use with Mac) and some are costly (third-party solutions for implementing the RDP protocol in a Mac device).

Mac does however make use of the slightly old-fashioned VNC protocol. It is possible to publish the (free) noVNC client on a Linux server, and implement this as a reverse proxy web app in Parallels Secure Workspace. This provides a secure and easy VNC access to the Mac endpoint devices, whilst still ensuring the environment and the users are fully protected by all of the security features Parallels Secure Workspace has to offer.

It's not necessary to install anything extra on the **client device** or the **Mac endpoint**.
This approach does require **1 additional Linux middle-man server** for handling the VNC connections.

This guide will explain the process for configuring VNC access on a Mac, installing NoVNC on a Linux server, configuring NoVNC to provide connectivity to a VNC endpoint, and publishing this as a reverse-proxied web application in Parallels Secure Workspace.

Mind that as of noVNC 1.4.0, a secure context (https) is now required.

1. Follow the instructions below on the **Mac Endpoint** to configure VNC connectivity on Mac.
    1. Go to the **System Preferences** and under the **Internet and Wireless** heading, click on **Sharing**.
    2. Enable the **Remote Management** checkbox.
    3. Click on **Computer Settings** and enable “VNC viewers may control screen with password:”
    4. Provide a password and click “OK”.
    5. (Optional) Click on **Options** and enable any other permissions needed.

2. On the **Linux middle-man server,** launch a terminal command line. Enter the command below and press Enter to install the NoVNC application on the Linux server.
   ```
   sudo snap install novnc
   ```

3. As of NoVNC 1.4.0, secure context (https) is required.
   This concern can be addressed by generating a self-signed certificate in this specific directory.
   Mind adjusting the IP address to the one of the Linux appliance.
   ```
   cd /snap/novnc
   openssl req -newkey rsa:2048 -nodes -keyout self.pem -x509 -days 365
   -out self.pem -addext "subjectAltName = IP:10.1.10.123"
   ```

4. Configure NoVNC as a service, so that when someone (or in this case the Parallels Secure Workspace appliance) connects to a specific port on the Linux server via the web browser, NoVNC will automatically connect to a specific VNC endpoint (the Mac for example).
   This needs to be done with a separate service/port for every VNC endpoint.

   In the terminal command line, enter this command and press [Enter]:
   `sudo snap set novnc services.n6082.listen=6082 services.n6082.vnc=172.22.2.154:5900` *(example command)* and press Enter.

   set novnc services.n6082.listen=6082 - This part of the command creates the service named **n6082** and configures it to listen to port **6082** of the Linux appliance. For every additional VNC service, use another port number.

   services.n6082.vnc=172.22.2.154:5900 - This part of the command configures the novnc service **n6082** to create VNC connections to the VNC host (in this example, the Mac device) **172.22.2.154** on port **5900** (default VNC port).

   The above details (the service name, port, VNC host IP address and port of the VNC host) will need to be adjusted if necessary.
5. It's possible to list the currently running NoVNC services with this command:
   `sudo snap get novnc services`

6. Log in to the **Parallels Secure Workspace environment** and open **System Settings.**
7. Go to **Manage > Applications**.
   Click **Add > Reverse Proxied Web Application. Name**: Add a name for this web app
   1. **Icon**: Upload an icon for this web app
   2. **Destination URL**: https://***<IP address of your Linux appliance>***:***<novnc connection port>***/vnc.html?port=**443**&host=***<source host header>***&autoconnect=1
      Example: https://10.1.10.123:6082/vnc.html?port=443&host=novncmac.example.com&autoconnect=1
      Important: The port here is configured as port **443** for connection. The **host** and **port** parameters in the URL above refer to the host header used to access this reverse proxied web application and the port number (nowadays port 443 (HTTPS) would be recommended).
   3. **Source Host Header**: The host header of the URL that will be used for accessing this reverse proxied web application.
   4. **User Labels**: Add labels of users who need to be able to see this specific app in Parallels Secure Workspace for the specific VNC host.
   5. **Rewrite Content**: Ensure this is set to **Enabled.**
8. Log out of and back into Parallels Secure Workspace. Launch the new web app. This app will open in another tab, and should connect immediately to the VNC endpoint. It may be necessary to first enter a password in NoVNC. On the left-hand side, there are extra controls for shortcut keys, the clipboard, full screen, display scaling etc.

For each VNC endpoint: use the steps above, but use a different port each time (so not 6082).

Useful links:

- https://novnc.com/info.html
- https://github.com/novnc/noVNC/blob/master/README.md#quick-start

---