Cendio ThinLinc®

Anatomy of a ThinLinc Session

HOW THINLINC WORKS

ThinLinc is made up of several tightly integrated open-source components, all held together by a small amount of proprietary "glue". Roughly 80% of ThinLinc is open-source, with the remaining 20% consisting of things like administrative tools, licensing, and automation. These proprietary and open-source components work together in order to create a functioning system.

This document provides an overview of how a ThinLinc remote desktop session is created, and the components involved in its operation.

DIFFERENT COMPONENTS

With the native client*, each component has a different role to play in providing a seamless remote desktop experience for ThinLinc users.

OpenSSH: This is the security layer of the ThinLinc connection. All traffic between the client and server is sent over an encrypted "secure shell" (SSH) tunnel.

TigerVNC: This is the performance-optimised implementation of the VNC protocol used in ThinLinc. VNC is used for transmitting visual data, i.e. the graphical component of the remote desktop environment.

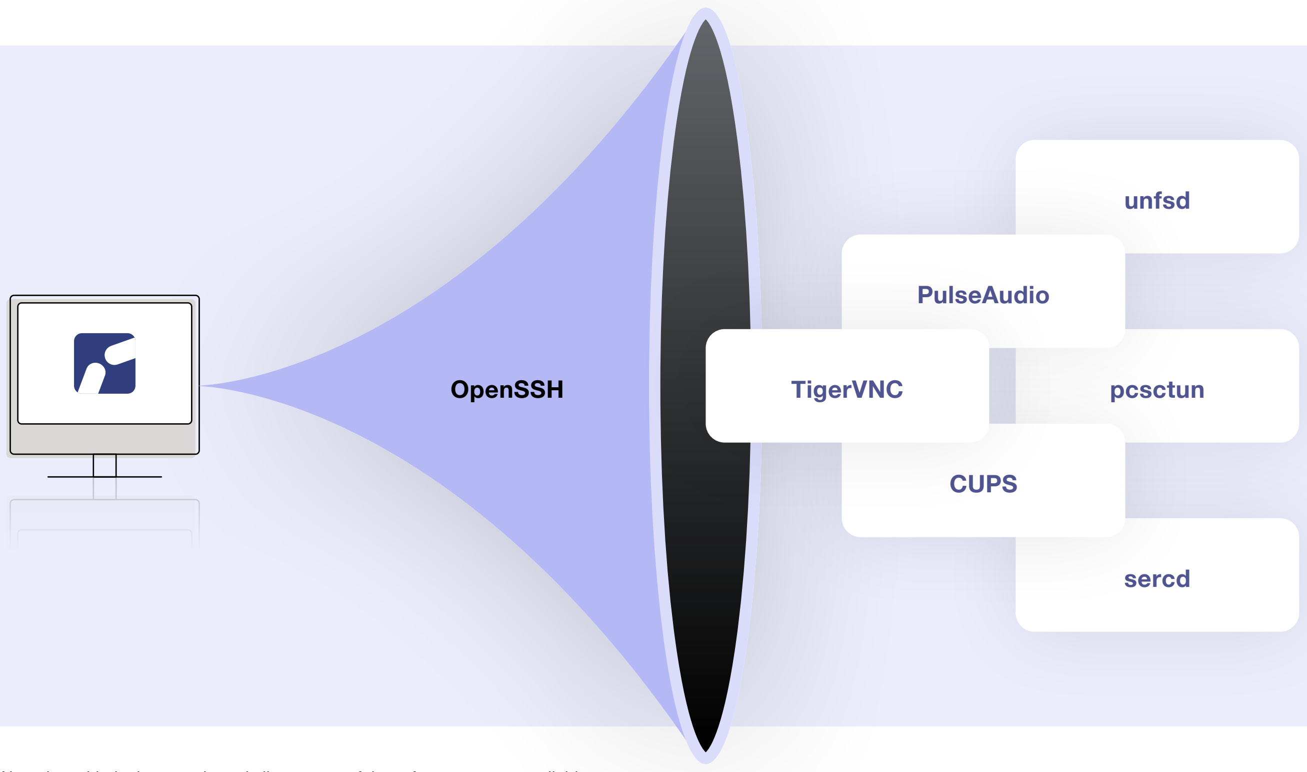
PulseAudio: This component is responsible for audio transport, providing full-duplex audio within the remote session.

CUPS: This is responsible for local printer access, allowing users to print to a locally-connected printer from within their remote desktop session.

unfsd: The "userspace NFS daemon" provides support for publishing local drives to the remote desktop session, using the NFS protocol.

pcsctun: This component publishes locally-attached smart-card readers into the ThinLinc session, enabling them to be accessed remotely.

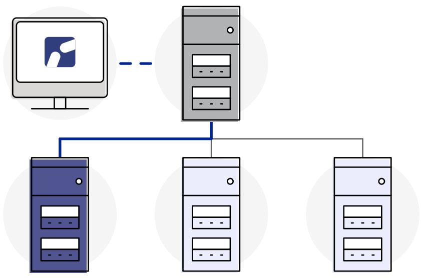
sercd: This redirects serial port devices (not USB) into the remote session, enabling them to be accessed remotely.



* Note that with the browser-based client, many of these features are unavailable

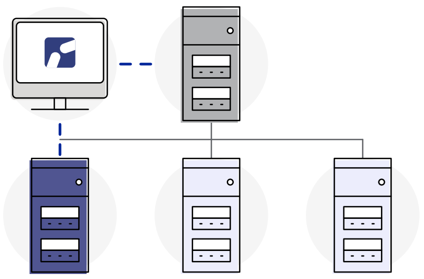
A THINLINC SESSION IS CREATED IN THREE STEPS:

1



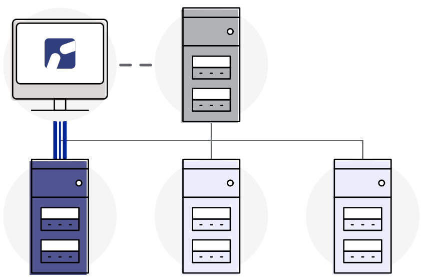
The client connects to the "master" server. If authentication is successful, and the user is allowed to start a session, a new session will be created on one of the "agents".

2



The client sets up an SSH tunnel to the agent with the newly created session. Using a technique called "local port forwarding", each protocol is sent over the encrypted SSH tunnel.

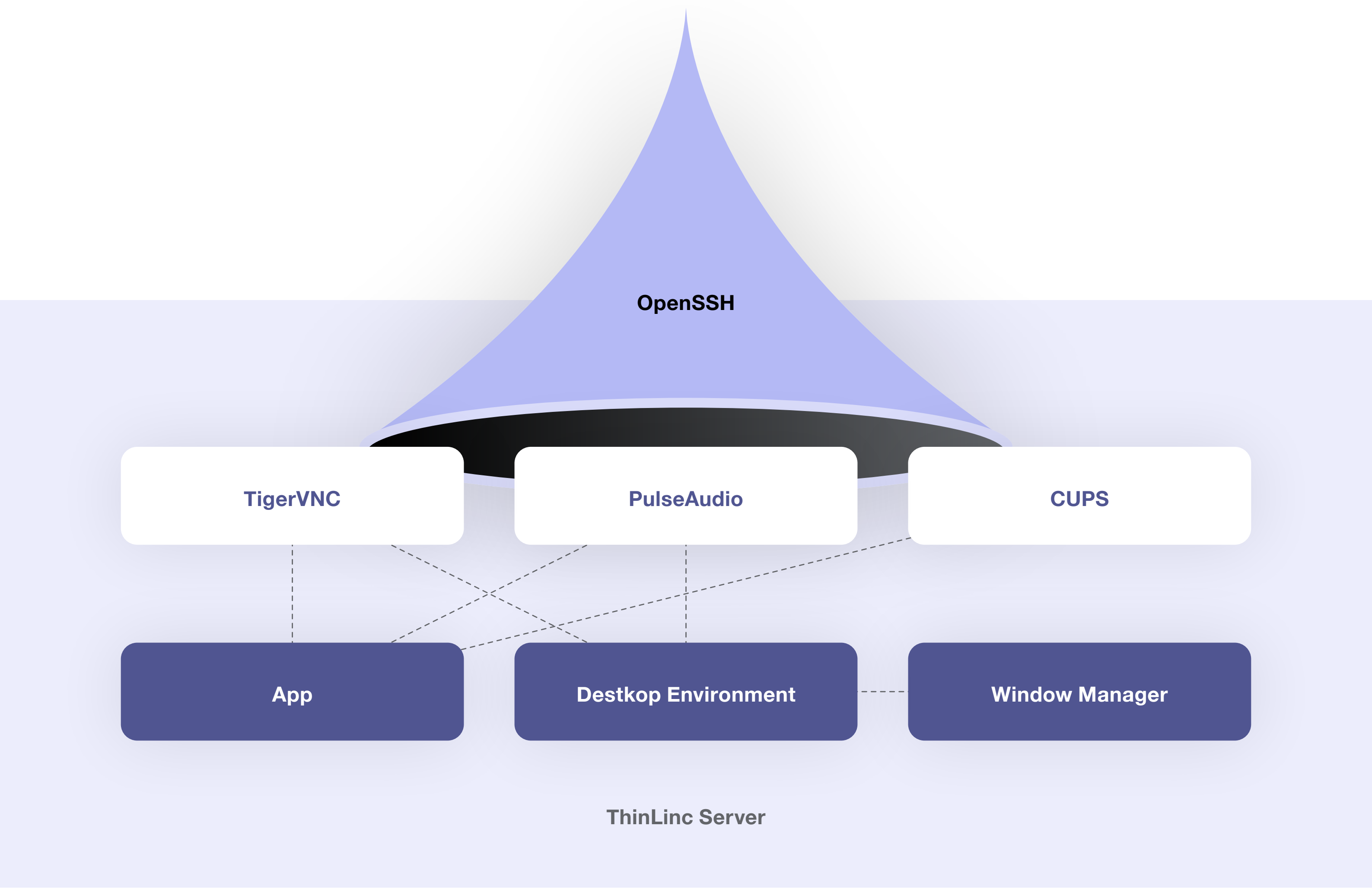
3



The client sets up communication between the client and remote session, by sending information over the forwarded ports, through the tunnel.

SERVER-SIDE COMPONENTS

On the server side, applications communicate with services which are launched as part of the session. These services have their input and output redirected over the forwarded ports, establishing communication between the client and server.



OpenSSH
TigerVNC
PulseAudio
CUPS
App
Destkop Environment
Window Manager
ThinLinc Server