
SDE

Building an Ubuntu desktop gold template

Learn how to create an Ubuntu 24.04 LTS desktop template in Proxmox for a VDI environment, using xrdp for remote access. Ideal for engineers, this guide covers installation, configuration, and template management.

AUTHOR

Paul Flanders

PUBLISHED

26 Oct 2026

DOCUMENT

InfoHub — Knowledge, news, and updates from EssingtonITS

Building an Ubuntu desktop gold template

Paul Flanders · 26 Oct 2026

Build an Ubuntu 24.04 LTS desktop with Remote Desktop (xrdp) once in Proxmox, then turn it into a template that SDE VDI clones to fill a Linux desktop pool.

Applies to: Proxmox VE 8/9 (single node or cluster), Ubuntu 24.04 LTS · Audience: engineers building the VDI environment · Time: about 1½ hours

Contents

1. [How it fits together](#)
2. [Which desktop](#)
3. [Before you start](#)
4. [Choose where the template lives](#)
5. [Create the master VM](#)
6. [Install Ubuntu Server](#)
7. [Configure the desktop](#)
8. [Set up cloud-init](#)
9. [Release a template version](#)
10. [Test the template](#)
11. [Add the pool in SDE VDI](#)
12. [Updating the image later](#)
13. [Troubleshooting](#)
14. [Final checklist](#)
15. [Related pages](#)

1. How it fits together

You build and keep two things in Proxmox:

Item	Example name	What it is
Master VM	<code>ubu2404-xfce-master</code>	A normal VM where you install and update software. It keeps its identity, so you can keep using it to make new versions.
Gold template	<code>gold-ubu2404-xfce-v1</code>	A copy of the master with its machine identity removed, converted to a Proxmox template. SDE VDI makes linked clones of it. You pick this name in the SDE VDI admin page.

When SDE VDI creates a desktop from the template, it:

1. Makes a linked clone, named after the pool's pattern (for example `ubu-dsk-007`).
2. Starts it. On first boot, **cloud-init** reads the clone's name from Proxmox and makes it the hostname. It also gives the clone a new machine ID, new SSH keys and a new RDP certificate.
3. Waits, through the QEMU guest agent, until cloud-init has finished and the hostname matches the VM name. Then it waits for Remote Desktop to answer on port 3389 and marks the desktop as a spare.

After a user signs out, SDE VDI sees that their session has gone (it asks the guest agent for `logintl list-sessions`), deletes the desktop and clones a fresh one.

Why the machine ID mattersUbuntu uses `/etc/machine-id` for, among other things, the DHCP client ID. If every clone kept the master's ID, the DHCP server could give them all the **same IP address**. Section 9 removes it so each clone makes its own.

Pool settings for LinuxLinux desktops use xrdp, which doesn't support NLA, and they don't join the domain yet. When you pick **Linux** in the pool editor, it sets **RDP security** to TLS and ticks **Users sign in on the desktop**: users sign in with a local account such as `vdiuser`. Section 11 has the full list.

2. Which desktop

This guide installs **Ubuntu Server 24.04 LTS** and adds the **Xubuntu minimal desktop** (Xfce) on top. Don't use the Ubuntu Desktop ISO with GNOME for a pool.

GNOME works badly under xrdp on 24.04: its own Remote Desktop service (`gnome-remote-desktop`) competes for port 3389, a GNOME session over xrdp needs extra configuration to start at all, it shows colour-manager password pop-ups, and Firefox is a snap. Xfce starts reliably over xrdp, uses far less memory per desktop, and Xubuntu's own theme, panel and menu keep it looking like a finished Ubuntu desktop (section 7.5 makes Remote Desktop sessions use them).

3. Before you start

You need	Details
Ubuntu Server ISO	The current Ubuntu Server 24.04 LTS image (<code>ubuntu-24.04.x-live-server-amd64.iso</code>) from <code>releases.ubuntu.com/24.04/</code> .
The ISO in Proxmox	Upload it to an ISO storage: <i>Datacenter</i> → <i>node</i> → <i>storage (e.g. local)</i> → <i>ISO Images</i> → <i>Upload</i> , or <i>Download from URL</i> . On a cluster, use shared ISO storage or upload to the node you build on.
Internet access from the VM	The installer and <code>apt</code> download packages from the Ubuntu mirrors.
A VM ID range	Suggested: <code>8000-8099</code> for master VMs, <code>9000-9099</code> for gold templates. This guide uses <code>8020</code> and <code>9020</code> .
SSH client	Optional but recommended: Windows Terminal, PuTTY or any SSH client, so you can paste the commands below.

4. Choose where the template lives

Linked clones are created on the **same storage** as the template. Pick the storage first, because it decides which nodes can run the pool:

Template storage	Clones can run on	Use when
Shared (Ceph RBD, NFS with qcow2)	Any node in the cluster	A cluster with shared storage. This is best for scaling.
Local (LVM-thin, ZFS)	Only the node that holds the template	Single-node sites, or a cluster without shared storage.
Local <code>dir</code> with raw disks	Can't make linked clones	Don't use for templates.

Cluster with only local storage? Build the template on one node and give the pool only that node, or make an identical template on each node with the same name. SDE VDI treats same-named templates as one image.

5. Create the master VM

In the Proxmox web UI, select **Create VM** and use these settings. Anything not listed can stay at its default.

Tab	Setting	Value
General	VM ID / Name	8020 / ubu2404-xfce-master
OS	ISO image	The Ubuntu Server 24.04 ISO
	Guest OS	Linux, version 6.x - 2.6 Kernel
System	Machine / BIOS	q35 / default (SeaBIOS)
	SCSI Controller	VirtIO SCSI single
	Qemu Agent	Ticked
Disks	Bus / Size	SCSI, 32 GB (more if the pool's applications need it), on your template storage
	Discard / SSD emulation / IO thread	All ticked
CPU	Cores	2-4
	Type	x86-64-v2-AES (the default). Use host only if every node in the cluster has the same CPU.
Memory	Memory	4096 MiB, ballooning off

Network	Bridge / Model	The bridge (and VLAN tag) the desktops will use; model VirtIO (paravirtualized)
---------	----------------	--

After the wizard, open **Options → Boot Order** and put the disk (`scsi0`) first and the CD drive second. The empty disk can't boot, so the installer starts from the CD the first time, and after the install the VM boots from the disk even with the ISO still attached.

Screenshot: Proxmox, VM 8020 → Options → Boot Order dialog, with `scsi0` first and `ide2` second, both ticked

Or create it with one command

In the **Shell** of the node the template will live on. First fetch the installer straight onto the node (check the current file name on `releases.ubuntu.com/24.04/` and use it in both commands):

```
wget -P /var/lib/vz/template/iso/ https://releases.ubuntu.com/24.04/ubuntu-24.04.4-live-server-amd64.iso
```

Then create the VM, swapping in the ISO name, bridge and storage (`datastore01` here):

```
qm create 8020 --name ubu2404-xfce-master --ostype l26 --machine q35 \
--scsihw virtio-scsi-single --scsi0 datastore01:32,iothread=1,discard=on,ssd=1 \
--cpu x86-64-v2-AES --cores 2 --memory 4096 --balloon 0 \
--net0 virtio,bridge=vbr0 --agent enabled=1 \
--ide2 local:iso/ubuntu-24.04.4-live-server-amd64.iso,media=cdrom \
--boot order='scsi0;ide2' --vga std
```

6. Install Ubuntu Server

1. Start the VM, open the **Console** and choose **Try or Install Ubuntu Server**.
2. Language **English**. If offered, choose **Continue without updating** for the installer itself.
3. Keyboard: layout **English (UK)**, variant **English (UK)**.

4. Type of installation: **Ubuntu Server**. Don't choose *Ubuntu Server (minimized)*. Leave *Search for third-party drivers* unticked.
5. Network: leave it on DHCP. The network port (usually `ens18`) shows an address. Select **Done**.
6. Proxy: leave empty unless your network needs one. Mirror: keep the default and wait for the mirror test to pass, then **Done**.
7. Storage: keep **Use an entire disk**, and **untick Set up this disk as an LVM group**. Without LVM, cloud-init can grow the disk when a pool's clones get a bigger one. Select **Done**, check the summary, then **Done → Continue** to confirm.
8. Profile: your name `VDI admin` , server name `ubu2404-xfce-master` , username `vdiadmin` , and a strong password. Record it in the password manager.
9. Ubuntu Pro: **Skip for now**, then **Continue**.
10. SSH: tick **Install OpenSSH server**. Leave *Import SSH key* empty.
11. Featured server snaps: tick nothing. Select **Done**.
12. When it says **Installation complete!**, select **Reboot Now**. At "Please remove the installation medium", press Enter.
13. In Proxmox, open **Hardware → CD/DVD Drive → Edit** and choose *Do not use any media*.

Screenshot: Ubuntu Server installer, Guided storage configuration, with "Use an entire disk" ticked and "Set up this disk as an LVM group" unticked

On the first boot, wait until the console shows a login prompt and the cloud-init messages have stopped (a minute or two). Then sign in as `vdiadmin` , or connect with SSH from your workstation (`ssh vdiadmin@<ip>` ; `ip -br addr` at the console shows the address), which makes copying and pasting easier.

Shortcut: the prep script does most of section 7 and 8 Copy the repo's `scripts/vdi` folder to the Proxmox node. In the VM, start the guest agent first: `sudo apt install -y qemu-guest-agent && sudo systemctl start qemu-guest-agent` . Then run `scripts/vdi/pve-template.sh push 8020` on the node. It copies `template-prep.sh` into the VM over the guest agent. Then in the VM: `sudo bash /root/template-prep.sh` (add `--test-user vdiuser` on dev).

On Ubuntu it installs Xubuntu minimal and does 7.1, 7.2, 7.4, 7.6–7.9, 7.12 and the cloud-init file in section 8. The Ubuntu-only steps are still yours to do by hand: **7.3, 7.5, 7.10 (check), 7.11, and step 1 of section 8**. The manual steps below remain the reference.

7. Configure the desktop

Run every command in this section on the master, as `vdiadmin`.

7.1 Update and install the VDI packages

```
sudo apt update
sudo apt full-upgrade -y
sudo apt install -y xubuntu-desktop-minimal xrdp xorgxrdp pipewire-module-xrdp \
  qemu-guest-agent cloud-init cloud-guest-utils ssl-cert fuse3
sudo reboot
```

This takes 5–15 minutes. If apt shows a screen asking which services to restart, keep the selection and press Enter. After the reboot the console shows the Xubuntu sign-in screen (LightDM).

Package	Why
xubuntu-desktop-minimal	The Xfce desktop with Xubuntu's theme, panel, menu, LightDM and PipeWire sound, without the full set of Xubuntu applications.
xrdp, xorgxrdp	The Remote Desktop server. Users connect to it through SDE Remote.
pipewire-module-xrdp	Plays the session's sound on the user's PC. xrdp pulls it in anyway; naming it makes sure.
qemu-guest-agent	Lets Proxmox and SDE VDI see the IP address, run commands, check who is signed in, and shut the VM down cleanly.
cloud-init	Gives each clone its own name and identity on first boot. Already on Ubuntu Server; listed so apt marks it as wanted.
cloud-guest-utils	Lets cloud-init grow the disk if a pool's clones get a bigger disk.
ssl-cert	Makes the self-signed certificate xrdp uses for TLS.
fuse3	For mapped drives: with Drive redirection ticked in the pool's settings, the user's PC folder appears in the desktop as <code>~/thinclient_drives</code> . Desktops are VMs, so nothing else is needed in Proxmox.

7.2 Start the guest agent

```
sudo systemctl start qemu-guest-agent
systemctl status qemu-guest-agent --no-pager
```

It starts by itself on every boot as long as the VM's *QEMU Guest Agent* option is enabled. In Proxmox, the VM's **Summary** now shows its IP address.

7.3 Network: DHCP on any network port

The installer wrote a network file for the port by name. This replaces it with one that takes DHCP on whichever Ethernet port the VM has, so a change of port name (for example after moving the VM between hypervisors) can't leave a desktop offline. Do this at the Proxmox **Console**, not over SSH, because the network restarts.

```
ls /etc/netplan/
sudo rm -f /etc/netplan/*.yaml
sudo tee /etc/netplan/01-sde-vdi.yaml >/dev/null <<'EOF'
# SDE VDI: DHCP on any Ethernet port, whatever its name.
network:
  version: 2
  ethernets:
    lan:
      match:
        name: "en*"
      dhcp4: true
EOF
sudo chmod 600 /etc/netplan/01-sde-vdi.yaml
sudo netplan apply
ip -br addr
```

The port (for example `ens18`) shows **UP** with an IPv4 address. The network stays with Ubuntu Server's own network service (`systemd-networkd`); the desktop's NetworkManager leaves wired ports alone.

7.4 Remote Desktop (xrdp)

```
# Let xrdp read its TLS certificate
sudo adduser xrdp ssl-cert

# Make Xfce the session for every user
sudo update-alternatives --set x-session-manager /usr/bin/xfce4-session
```

```
sudo systemctl enable --now xrdp
sudo systemctl restart xrdp
```

Check that nothing else holds the Remote Desktop port and that the firewall is off or allows it:

```
sudo apt purge -y gnome-remote-desktop 2>/dev/null || true
sudo ss -ltnp | grep 3389
sudo ufw status
```

The `ss` line names `xrdp` . `ufw` says *inactive* on a new install; if it's active, run `sudo ufw allow 3389/tcp` .

Picture quality Ubuntu 24.04 ships the older xrdp 0.9 series, which has no GFX graphics mode. Leave **Quality** on **Auto** in SDE Remote: the app recognises xrdp and uses RemoteFX, which is the smooth choice here. No change to `/etc/xrdp/xrdp.ini` is needed.

7.5 Xubuntu's look in Remote Desktop sessions

At the console, LightDM starts the Xubuntu session, with Xubuntu's theme, panel layout and menu. xrdp starts plain Xfce instead, which looks unfinished and asks every new user how to set up the panel. This file gives Remote Desktop sessions the Xubuntu settings too:

```
sudo tee /etc/X11/Xsession.d/55sde-vdi-xubuntu >/dev/null <<'EOF'
# SDE VDI: Remote Desktop sessions get Xubuntu's settings, panel and theme.
# LightDM sets DESKTOP_SESSION itself; xrdp doesn't.
if [ -z "$DESKTOP_SESSION" ]; then
    DESKTOP_SESSION=xubuntu
    export DESKTOP_SESSION
fi
EOF
```

Ubuntu's own session scripts (`56xubuntu-session` and `60x11-common_xdg_path`) then add the Xubuntu folders for every session.

Want Ubuntu's own colours? Install the Yaru theme with `sudo apt install -y yaru-theme-gtk yaru-theme-icon` . Then, in a Remote Desktop session as `vdiuser` (7.12), open *Settings* → *Appearance* and choose **Yaru** on the **Style** and **Icons** tabs, and *Settings* → *Window Manager* to pick a matching title bar. Local users' settings are part of the image, so every desktop in

the pool starts with them.

7.6 Stop the "authentication required" pop-ups

Without this, every Remote Desktop session asks for a password to "create a color managed device".

```
sudo tee /etc/polkit-1/rules.d/45-sde-vdi-colord.rules >/dev/null <<'EOF'
// SDE VDI: allow colour management in remote sessions without a password prompt
polkit.addRule(function(action, subject) {
    if (action.id.indexOf("org.freedesktop.color-manager.") === 0) {
        return polkit.Result.YES;
    }
});
EOF
```

7.7 Screen locker

`light-locker` often leaves xrdp sessions on a black screen after reconnecting. Xubuntu 24.04 uses `xfce4-screensaver` instead, which still locks an idle session (users unlock with their password). Remove `light-locker` in case something pulled it in:

```
sudo apt purge -y light-locker 2>/dev/null || true
```

7.8 Session time limits

SDE VDI only recycles a desktop once the user has **signed out**. This makes sure an abandoned session ends 30 minutes after the user disconnects.

```
sudo sed -i -E \
-e 's/^KillDisconnected=.*\/KillDisconnected=true/' \
-e 's/^DisconnectedTimeLimit=.*\/DisconnectedTimeLimit=1800/' \
/etc/xrdp/sesman.ini
grep -E '^(KillDisconnected|DisconnectedTimeLimit|IdleTimeLimit)= ' /etc/xrdp/sesman.ini
```

Expect `KillDisconnected=true`, `DisconnectedTimeLimit=1800` and `IdleTimeLimit=0`.

7.9 No sleep or hibernation

A pool desktop must never sleep, or it drops out of the pool.

```
sudo systemctl mask sleep.target suspend.target hibernate.target hybrid-sleep.target
```

7.10 Sound

Sound needs nothing more than the packages from 7.1: PipeWire runs in each session, and `pipewire-module-xrdp` adds an **xrdp-sink** output when the session starts. You check it in 7.14. The pool's **Audio** setting is on by default.

7.11 Updates, snaps and Firefox

Desktops are short-lived copies of the template, so they shouldn't update themselves. You update the master instead (section 12). Turn off automatic updates, so clones don't spend their first minutes downloading:

```
sudo tee /etc/apt/apt.conf.d/20auto-upgrades >/dev/null <<'EOF'
APT::Periodic::Update-Package-Lists "0";
APT::Periodic::Unattended-Upgrade "0";
EOF
sudo systemctl disable --now apt-daily.timer apt-daily-upgrade.timer
```

Snaps. Check what's installed:

```
snap list
```

On a new install it says *No snaps are installed yet*. Keep it that way where you can: each desktop would otherwise check for snap updates. Leave `snapt` itself installed, because removing it also removes the `ubuntu-server-minimal` package. If an application exists only as a snap, install it and hold automatic refreshes:

```
sudo snap install <name>
sudo snap refresh --hold
```

Refresh snaps on the master with `sudo snap refresh` when you make a new version.

Firefox. In Ubuntu 24.04, `apt install firefox` installs the snap. Install Mozilla's own package instead, which starts faster and has no snap updates:

```
sudo install -d -m 0755 /etc/apt/keyrings
wget -q https://packages.mozilla.org/apt/repo-signing-key.gpg -O- \
  | sudo tee /etc/apt/keyrings/packages.mozilla.org.asc >/dev/null
echo "deb [signed-by=/etc/apt/keyrings/packages.mozilla.org.asc]
https://packages.mozilla.org/apt mozilla main" \
  | sudo tee /etc/apt/sources.list.d/mozilla.list >/dev/null
sudo tee /etc/apt/preferences.d/mozilla >/dev/null <<'EOF'
Package: *
Pin: origin packages.mozilla.org
Pin-Priority: 1000
EOF
sudo apt update
sudo apt install -y firefox
apt policy firefox | head -3
```

`apt policy` shows the installed version coming from `packages.mozilla.org`, not `1snap1`.

7.12 Test user (dev only, no Active Directory)

Linux desktops aren't joined to a domain, so users sign in with a local account built into the image.

```
sudo adduser vdiuser
```

Answer the password prompts; the other questions can be left empty. Don't sign in as `vdiuser` at the Proxmox console (see 7.14).

7.13 Install the pool's software

Install what this pool needs, for example `sudo apt install -y libreoffice`, or specialist applications. Install system-wide, not into a user's home directory: whatever `vdiuser` changes in a session is lost when that desktop is recycled.

7.14 Check Remote Desktop works

1. From a Windows PC on the same network, open **Remote Desktop Connection** to the master's IP. Accept the certificate warning (the certificate is self-signed).
2. At the xrdp sign-in screen, leave **Session** on *Xorg* and sign in as `vdiuser`. You get the Xubuntu desktop (Xubuntu wallpaper, panel at the top, Whisker menu) with no

password pop-ups and no panel set-up question.

3. Open a terminal (*Applications → Terminal Emulator*) and check sound:

```
pactl info | grep -E 'Server Name|Default Sink'
```

Expect *PulseAudio (on PipeWire ...)* and `Default Sink: xrdp-sink`. Play any sound (for example a video in Firefox) and you hear it on the PC.

4. Sign out with the **Log Out** button in the applications menu.

Screenshot: Remote Desktop session to the master as vdiuser, showing the Xubuntu desktop and a terminal with the pactl output

One session per user xrdp can't open a session for a user who is already signed in at the console. Test with `vdiuser`, not the account signed in at the Proxmox console.

8. Set up cloud-init

Proxmox's own cloud-init settings would make each clone create an extra user, set up the network its own way, and run a full package upgrade on first boot. This section limits cloud-init to giving each clone its own identity.

1. Remove the installer's cloud-init leftovers. `99-installer.cfg` tells cloud-init to ignore Proxmox's cloud-init drive, so clones would keep the master's name.

```
sudo rm -f /etc/cloud/cloud-init.disabled /etc/cloud/cloud.cfg.d/99-  
installer.cfg  
ls /etc/cloud/cloud.cfg.d/
```

Other files in that folder, such as `curtin-preserve-sources.cfg` (keeps cloud-init away from the apt sources) and `05_logging.cfg`, stay.

2. Add the SDE VDI settings. It's the same file as on Debian:

```
sudo tee /etc/cloud/cloud.cfg.d/99-sde-vdi.cfg >/dev/null <<'EOF'  
# SDE VDI: cloud-init only gives each clone its own identity.
```

```
# Proxmox's cloud-init drive (NoCloud); nothing else.
datasource_list: [ NoCloud, ConfigDrive ]

# The network file from section 7.3 does DHCP; cloud-init leaves the network
alone.
network:
  config: disabled

# Hostname = the VM name in Proxmox (set per clone by SDE VDI).
preserve_hostname: false
manage_etc_hosts: true

# Proxmox asks for the distro "default user". Make it a locked account with no
# shell and no sudo, so it can't be used to sign in.
system_info:
  default_user:
    name: ci-unused
    lock_passwd: true
    shell: /usr/sbin/nologin
    sudo: false
    groups: []

# Keep password sign-in for SSH (local accounts).
ssh_pwauth: true

# Give each clone its own RDP certificate (runs once per clone).
runcmd:
  - [ make-ssl-cert, generate-default-snakeoil, --force-overwrite ]
  - [ systemctl, restart, xrdp ]
EOF
```

3. Check the file parses:

```
sudo cloud-init schema --system
```

It should report *Valid schema*. A warning about user data not being present is expected on the master.

4. Shut the master down: `sudo poweroff` . This is the end of the master build.

9. Release a template version

Scripted release With the master shut down, run on its node: `scripts/vdi/pve-template.sh`
`release 8020 9020 gold-ubu2404-xfce-v1` . It does the work of steps 2–12 below: the full clone,
the clean-up and power-off over the guest agent (with the network left connected, as it takes
only a minute or two), the empty CD drive, the cloud-init drive, *Upgrade packages* off, the
`sde-gold`

tag, the conversion to a template and the clone right for SDE VDI. Take the snapshot (step 1) and write the Notes (step 10) yourself. Afterwards, `pve-template.sh check 9020` lists anything still missing.

To keep the master reusable, you remove the machine identity from a **copy** of it. Do this for v1 and for every later version.

1. With the master shut down, choose **Snapshots → Take Snapshot**. Name it `v1` and add a description of what's in this version.
2. Right-click the master → **Clone**.
 - Mode: **Full Clone**.
 - VM ID: `9020` ; Name: `gold-ubu2404-xfce-v1` .
 - Target storage: your template storage ([section 4](#)).
 - On a cluster, Target node: the node the template should live on.
3. On the new VM, open **Hardware → Network Device → Edit** and tick **Disconnect**, so this copy doesn't take a DHCP lease or clash with the master.
4. Start it, open the Console, sign in as `vdiadmin` (choose *Xubuntu Session* at LightDM, or press `Ctrl+Alt+F3` for a text console) and open a terminal.
5. Remove its identity, then power it off:

```
sudo apt clean
sudo cloud-init clean --logs --seed --machine-id
sudo rm -f /etc/ssh/ssh_host_*
ls -l /var/lib/dbus/machine-id      # must be a link to /etc/machine-id
cat /etc/machine-id                # must say "uninitialized" (or be empty)
history -c
sudo poweroff
```

If `/var/lib/dbus/machine-id` is a regular file rather than a link, run `sudo rm /var/lib/dbus/machine-id && sudo ln -s /etc/machine-id /var/lib/dbus/machine-id` before powering off.

6. **Don't start it again.** Starting it would give it a new identity, which every clone would then share.

7. Prepare its hardware:

- **Network Device → Edit:** untick **Disconnect**. Clones inherit this setting.
- **CD/DVD Drive → Edit:** *Do not use any media*.
- **Add → CloudInit Drive:** bus *IDE*, on your template storage.

8. Open the VM's **Cloud-Init** tab:

- **Upgrade packages:** set to **No**. Otherwise every clone runs a full upgrade on first boot.
- Leave *User*, *Password* and *SSH public key* empty.
- Ignore *IP Config*; the config from [section 8](#) leaves the network to the file from 7.3.

9. **Options:** check that *QEMU Guest Agent* is *Enabled*.

10. On its **Summary** page:

- Add the tag `sde-gold`.
- In *Notes*, record the version, the date and what changed.

11. Right-click it → **Convert to template**.

12. Allow SDE VDI to clone it, in the node's Shell:

```
pveum acl modify /vms/9020 --users sde-vdi@pve --roles PVETemplateUser
```

You can skip this if the template is in the Proxmox pool `sde-gold`, which already grants it.

Screenshot: Proxmox, template 9020 → Hardware tab, showing the CloudInit Drive (ide0), the empty CD/DVD drive and the connected network device

Screenshot: Proxmox, template 9020 → Cloud-Init tab, with Upgrade packages set to No and User, Password and SSH public key empty

10. Test the template

Do what SDE VDI will do, by hand, once. Make two clones, so you can check they get different identities.

1. Right-click the template → **Clone**. Mode: **Linked Clone**; Name: `ubu-test-001` . Repeat with `ubu-test-002` . Start both.
2. After about a minute, each clone's **Summary** shows an IP address. The two addresses must be **different**.
3. In the Proxmox **node Shell**, check the identities. Replace `<vmid>` with each clone's ID in turn:

```
qm guest exec <vmid> -- hostname
qm guest exec <vmid> -- cat /etc/machine-id
qm guest exec <vmid> -- cloud-init status
```

Each hostname matches its VM name, the two machine IDs are different, and cloud-init says `status: done` . SDE VDI waits for exactly this.

4. Connect with Remote Desktop to one clone's IP as `vdiuser` . You get the Xubuntu desktop with no password pop-ups. The certificate warning shows the clone's own name.
5. While you're connected, check that the guest agent can see the session. SDE VDI uses this to tell whether a desktop is in use:

```
qm guest exec <vmid> -- loginctl list-sessions
```

It lists a session for `vdiuser` .

6. Sign out. Then stop and remove both test clones.

Record the timings Note how long a clone takes from Start to accepting Remote Desktop. This sets how many spare desktops a pool needs.

11. Add the pool in SDE VDI

On the SDE VDI admin page, open **Templates** first: the new template is listed under the gold templates, and its checks (guest agent, network connected, no ISO, cloud-init drive) all pass. Then go to **Pools → New pool**:

Field	Value	Notes
Display name	<i>Ubuntu Desktop</i>	What users see in SDE Remote.
Pool ID	ubuntu	Lower-case letters, digits and dashes. Can't be changed later.
Desktop OS	Linux	Sets RDP security to TLS, ticks Users sign in on the desktop and clears the domain fields.
Gold template	gold-ubu2404-xfce-v1	The checks show underneath.
Desktop names	ubu-dsk-{nnn}	Exactly one {nnn} counter; letters, digits and dashes.
Spare desktops / Maximum desktops	1 / 2	Raise them once the pool is proven.
Grant to Guacamole groups	e.g. sde-vdi-users	Who sees the pool in SDE Remote.
Advanced → Nodes	The template's node, on local storage	None ticked = any node that can clone the template.
Advanced → Join domain: OU	Empty	Domain join is Windows-only for now; a Linux pool with an OU isn't saved.

Advanced → RDP security	TLS	xrdp doesn't support NLA; a Linux pool set to NLA isn't saved.
Advanced → Remote Desktop	Users sign in on the desktop (dev / local accounts) ticked	Tick Drive redirection for mapped drives; Audio and Clipboard are on by default.

Screenshot: SDE VDI admin page → New desktop pool, with Desktop OS set to Linux and the Advanced section open showing RDP security TLS and "Users sign in on the desktop" ticked. Select **Create pool**. On the **Desktops** tab, the first desktop shows **Preparing** with the steps *Cloning*, *Starting Windows/Linux*, *Windows setup* (on a Linux desktop this means cloud-init hasn't finished), *Waiting for network* and *Waiting for Remote Desktop*, then **Spare**.

Then test it end to end:

1. In SDE Remote, as a user in the granted group, the **Ubuntu Desktop** tile shows "1 ready". Select **Connect**.
2. Sign in as `vdiuser` when asked. You get the Xubuntu desktop. If **Drive redirection** is ticked, your PC's folder is in `~/thinclient_drives`.
3. Sign out with **Log Out**. Within a couple of minutes, the desktop is deleted on the **Desktops** tab and a new spare is prepared.

12. Updating the image later

1. Start the **master** (`ubu2404-xfce-master`), not a template.
2. Run `sudo apt update && sudo apt full-upgrade -y` (and `sudo snap refresh` if you use snaps), and add or update software.
3. Shut it down.
4. Follow [section 9](#) again with the next version: snapshot `v2`, full clone to `gold-ubu2404-xfce-v2`, clean, convert to template, grant the clone right.

5. In the SDE VDI admin page, edit the pool and pick the v2 template. Spare desktops are replaced at once; desktops in use are replaced as their users sign out.
6. Delete the v1 template once no pool uses it and no desktops are still based on it.

Moving to a later Ubuntu LTS Build a new master from the new ISO with this guide rather than running `do-release-upgrade` on the master. A fresh install gives a clean template.

13. Troubleshooting

Problem	Fix
All clones get the same IP	The machine ID wasn't reset. Check that <code>/etc/machine-id</code> in the template copy said <i>uninitialized</i> or was empty before power-off. Release a new version with section 9, step 5 done again.
Desktop error: hostname is "...", expected "...": check the template has a CloudInit drive	cloud-init didn't take the name from Proxmox. Check the template's Hardware has a CloudInit Drive, and that <code>/etc/cloud/cloud.cfg.d/99-installer.cfg</code> and <code>/etc/cloud/cloud-init.disabled</code> are gone on the master (section 8, step 1). Release a new version.
Desktop stays at "Windows setup"	On a Linux desktop this step means cloud-init hasn't reported <code>status: done</code> . In the node Shell run <code>qm guest exec <vmid> -- cloud-init status --long</code> . An error usually names the failing part, often a typo in <code>99-sde-vdi.cfg</code> .
Clone has no IP address	The network file is missing or names a port. Check <code>/etc/netplan/01-sde-vdi.yaml</code> on the master (7.3) and that it's the only file in <code>/etc/netplan/</code> .

First boot takes many minutes	<i>Upgrade packages</i> is still on in the Cloud-Init tab, or automatic updates are still on (7.11). Set Upgrade packages to No on the template.
Remote Desktop closes straight after the certificate prompt	xrdp can't read its key. <code>/var/log/xrdp.log</code> shows "Permission denied" for <code>/etc/xrdp/key.pem</code> . Run <code>sudo adduser xrdp ssl-cert</code> and <code>sudo systemctl restart xrdp</code> (7.4).
Connection refused on port 3389	xrdp isn't running, or something else holds the port. Check <code>sudo ss -ltnp grep 3389</code> , <code>systemctl status xrdp</code> and <code>sudo ufw status</code> .
Black screen, or the session closes after sign-in	The user is already signed in at the console, or <code>light-locker</code> is installed. Sign out of the console session; remove <code>light-locker</code> (7.7). Details are in <code>/var/log/xrdp-sesman.log</code> and the user's <code>~/.xorgxrdp.*.log</code> .
Plain grey Xfce, or "Welcome to the first start of the panel"	The session didn't get the Xubuntu settings. Check <code>/etc/X11/Xsession.d/55sde-vdi-xubuntu</code> (7.5). For a user who already chose a panel layout, delete their <code>~/.config/xfce4</code> folder.
"Authentication required to create a color managed device"	The polkit rule is missing or has a typo. Check <code>/etc/polkit-1/rules.d/45-sde-vdi-colord.rules</code> (7.6).

Can't unlock the screen after it locks	Disconnect and reconnect, then sign in again. If it keeps happening, turn locking off for the pool's users in <i>Settings</i> → <i>Screensaver</i> → <i>Lock Screen</i> as <code>vdiuser</code> on the master, or remove <code>xfce4-screensaver</code> .
No sound	Check the pool's Audio is ticked. In the session, <code>pactl info</code> should show <code>Default Sink: xrdp-sink</code> . If it doesn't, check <code>pipewire-module-xrdp</code> is installed and <code>/etc/xdg/autostart/pipewire-xrdp.desktop</code> exists, then sign out and in again.
Picture is laggy	In SDE Remote, set Quality back to Auto (it uses RemoteFX for xrdp), or pick RemoteFX by hand.
Summary says "Guest Agent not running"	Check that <i>Options</i> → <i>QEMU Guest Agent</i> is Enabled (this needs a full VM stop and start, not a restart) and that <code>qemu-guest-agent</code> is installed.
"Linked clone" option is greyed out	The template is on storage that can't do linked clones. See section 4 .

14. Final checklist

Before picking the template in a pool, check that:

- It is a **template** named `gold-<os>-<variant>-v<N>` and tagged `sde-gold`, and `sde-vdi@pve` may clone it.
- Its disks are on storage that supports linked clones.
- It has a CloudInit drive, and *Upgrade packages* is set to No.

- The QEMU Guest Agent option is enabled.
- The network device is connected and the CD drive is empty.
- The master had `99-installer.cfg` removed and `99-sde-vdi.cfg` added, and the schema check passed.
- The only netplan file is `01-sde-vdi.yaml` , matching `en*` .
- Two test clones got their own hostnames, machine IDs and IP addresses, and cloud-init said `status: done` .
- Remote Desktop gives the Xubuntu desktop with no pop-ups, and sound plays through `xrdp-sink` .
- The guest agent shows the signed-in session (`loginctl list-sessions`).
- Sleep is masked, automatic updates are off, and the xrdp disconnected-session limit is 30 minutes.
- The master VM is shut down, and has a snapshot for this version.

15. Related pages

- [Setting up SDE VDI desktop pools](#): setting up SDE VDI end to end
- [Building a Linux desktop gold template](#): the same template on Debian 13
- [Building the image in VirtualBox and moving it to Proxmox](#): building this image in VirtualBox and moving it to Proxmox
- [Building a Windows 11 gold template](#): building a Windows 11 gold template

SDE VDI knowledge base · Ubuntu desktop gold template · Design: `docs/design/vdi.md` §
Gold templates