API Proxy Deployment Guide

1. Scope

This guide describes how to deploy, upgrade, uninstall, and troubleshoot the API Proxy service on management nodes. The release package supports deployment on either a single management node or two management nodes. Remote scripts can start from either management node as the entry node and, when required, automatically detect the other management node and relay the operation.

2. Release Package Contents

The current release package is distributed as a `tar.gz` archive:

- `api-proxy-<version>.tar.gz`

After extraction, the directory mainly contains:

```
api-proxy.jar
build-info.properties
README.txt
DEPLOYMENT.zh-CN.pdf
DEPLOYMENT.en-US.pdf
deploy-api-proxy.sh
collect-api-proxy-logs.sh
cleanup-api-proxy-logs.sh
install-api-proxy.sh
deploy/api-proxy-common.sh
```

Included files:

File	Purpose
`api-proxy.jar`	Prepared executable service package.
`deploy-api-proxy.sh`	Remote deployment, upgrade, and uninstall entry script.
`install-api-proxy.sh`	Local install and uninstall script for a target management node.
`collect-api-proxy-logs.sh`	Collects service logs, systemd journals, management node logs, and dual-management-node diagnostics.
`cleanup-api-proxy-logs.sh`	Cleans service logs and journals.
`deploy/api-proxy-common.sh`	Shared deployment logic, including SSH, relay, and dual-management-node topology detection.
`build-info.properties`	Version, build commit, and build time.
`DEPLOYMENT.zh-CN.`	Chinese deployment guide PDF.

pdf	
DEPLOYMENT.en-US.pdf	English deployment guide PDF.
README.txt	Quick reference bundled in the release package.

3. Environment And Access Requirements

3.1 Deployment Host / Execution Environment

The deployment host is optional. It is only needed when you want to run the remote deployment, uninstall, log collection, or log cleanup commands described below. In practice, it is simply a Linux shell environment from which those scripts are executed against the target management nodes.

- It does not have to be a separately prepared VM.
- It can be a temporary Linux VM, a jump host, an operations workstation, or any management node itself.
- If you already copied the full release directory to the target management node and plan to run `install-api-proxy.sh` locally there, you do not need a separate deployment host.

If you use the remote scripts, the deployment host must have:

- Linux shell environment.
- `bash` , `ssh` , `sshpass` , `scp` , and `curl` .
- `tar` for extracting the release package.
- Root SSH access to the entry management node.

3.2 Target Management Nodes

Each target management node must have:

- Java 8 runtime, or an executable configured through `JAVA_BIN` / `JAVA_HOME` .
- systemd and `systemctl` .
- `curl` for local health checks.
- `qemu-img` installed in advance and reachable through `PATH` during service startup. If it is installed in a non-standard location, configure `adapter.backup.qemu-img-path` before starting the service.
- `mysql` client only when uninstall needs to clean owned service tables; if it is missing, the script skips database cleanup and continues the uninstall.
- Port `16443/tcp` reachable from the backup client or other upstream callers.

4. Default Names And Paths

Item	Default

systemd service	`api-proxy.service`
Install directory	`/opt/api-proxy`
Service jar	`/opt/api-proxy/api-proxy.jar`
Log directory	`/opt/api-proxy/logs`
Main log	`/opt/api-proxy/logs/api-proxy.log`
Upload workspace	`/opt/api-proxy/uploads`
Runtime workspace	`/tmp/api-proxy`
TLS directory	`/etc/api-proxy/tls`
TLS keystore	`/etc/api-proxy/tls/server-keystore.p12`
HTTPS port	`16443`
Local health check URL	`https://127.0.0.1:16443/ovirt-engine/api`
Certificate endpoint	`https://<AnyManagementNodeIP-or-VIP>:16443/ovirt-engine/services/pki-resource`

5. Extract The Release Package

Only the `tar.gz` package format is provided:

```
tar -xzf api-proxy-1.0.0.tar.gz -C /tmp
cd /tmp/api-proxy-1.0.0
```

6. Deployment, Installation, Upgrade, And Uninstall

Two workflows are supported:

- Remote deployment: run `deploy-api-proxy.sh` on the deployment host and let the script install through SSH.
- Local install: copy the full release directory to the target management node and run `install-api-proxy.sh` there.

6.1 Remote Deployment

Use the remote deployment entry point included in the release package:

```
./deploy-api-proxy.sh --install root@<node-ip>
```

Notes:

- In a single-management-node deployment, `<node-ip>` is that management node IP.

- In a two-management-node deployment, `<node-ip>` can be either management node IP; the script detects the other management node and completes deployment on both nodes.
- If `API_PROXY_SSH_PASSWORD` is not set, the script prompts for the SSH password without echoing it.
- If an existing deployment is detected, the script prompts for confirmation before overwriting it.
- The installer keeps a backup of the old jar on the target node, refreshes the systemd unit, and restarts the service.
- If a VIP is detected, post-install health checks prefer the VIP endpoint.

Non-interactive example:

```
export API_PROXY_SSH_PASSWORD='<ssh-password>'
export API_PROXY_ASSUME_YES=1
./deploy-api-proxy.sh --install root@<node-ip>
```

6.2 Local Install

If the full release directory is already on the target management node, run:

```
sudo ./install-api-proxy.sh --install
```

Notes:

- Local install must run as root.
- Local install does not require a separate deployment host and does not rely on `sshpass`.

Common local override example:

```
sudo INSTALL_DIR=/data/api-proxy LOG_DIR=/data/api-proxy/logs ./install-api-proxy.sh --install
```

6.3 Post-Deployment Checks

Check the API and certificate endpoint:

```
curl -k https://<AnyManagementNodeIP-or-VIP>:16443/ovirt-engine/api
curl -k https://<AnyManagementNodeIP-or-VIP>:16443/ovirt-engine/services/pki-resource
```

Check the service status:

```
systemctl status api-proxy --no-pager -l
journalctl -u api-proxy -n 100 --no-pager
tail -n 100 /opt/api-proxy/logs/api-proxy.log
```

6.4 Uninstall

Remote uninstall:

```
./deploy-api-proxy.sh --uninstall root@<node-ip>
```

Local uninstall:

```
sudo ./install-api-proxy.sh --uninstall
```

Default uninstall behavior:

- Stops and removes `api-proxy.service`.
- Removes the install directory, log directory, upload workspace, and runtime workspace.
- Drops owned service database tables.
- Does not remove the database or schema itself.
- In two-management-node deployments, database-table cleanup runs only once on the last node.
- If the target node does not have the `mysql` client, the script skips database cleanup and continues the uninstall.

Keep selected data during uninstall:

```
sudo CLEAN_LOGS=0 CLEAN_UPLOADS=0 ./install-api-proxy.sh --uninstall
```

7. Veeam Integration And Worker Deployment

7.1 Worker Role

In oVirt / RHV scenarios, Veeam uses Worker nodes to perform data transfer, backup reads, and restore writes. In this environment, Veeam still deploys and manages Workers in the oVirt model, while API Proxy translates relevant oVirt API calls into requests understood by the backend management platform.

From an operational perspective, users still add a virtualization platform, deploy Workers, create backup jobs, and run restore jobs in the Veeam console. The only difference is that the underlying target is the backend management platform exposed through API Proxy rather than native oVirt.

7.2 Veeam Connection Parameters

When adding a virtualization platform in Veeam using the oVirt / Red Hat Virtualization type, use the following parameters.

Veeam field	Recommended value
Platform type	oVirt / Red Hat Virtualization
Address	Any management node IP or the VIP. For two-management-node deployments, the VIP is recommended
Port	API Proxy default is `16443`. If your Veeam version only supports `443`, change API Proxy to `443` or

	use port forwarding
Username	Management-platform account, for example `admin`. Do not use `admin@internal` unless that is the actual account in your environment
Password	Plaintext password of the above management-platform account
Certificate	Use the certificate automatically generated by API Proxy. If Veeam prompts for certificate trust, confirm in the UI or import the Root CA

You can use the following command to verify whether credentials can obtain an oVirt-style token through API Proxy:

```
read -r -s MGMT_PASSWORD
curl -k -X POST "https://<AnyManagementNodeIP-or-VIP>:16443/ovirt-engine/sso/oauth/token" \
  -H "Content-Type: application/x-www-form-urlencoded" \
  --data-urlencode "grant_type=password" \
  --data-urlencode "username=<ManagementPlatformAccountName>" \
  --data-urlencode "password=${MGMT_PASSWORD}"
```

7.3 Critical Prerequisite: Worker Must Run On Intel Hosts

Veeam Worker must run on compute nodes with Intel CPUs. If a Worker is scheduled to AMD compute nodes, Worker VM startup failures, abnormal exits after startup, or persistent Worker unavailable status in Veeam may occur.

Check the compute host CPU vendor before deployment:

```
lscpu | grep 'Vendor ID'
```

Expected output:

```
Vendor ID: GenuineIntel
```

If the output is `AuthenticAMD`, do not use that host for Veeam Worker.

In mixed CPU clusters, prepare Intel-only Worker placement in advance:

- Use clusters, host groups, or dedicated resource pools containing only Intel hosts.
- Use scheduling labels, affinity policies, or operations processes to ensure Workers are never scheduled to AMD hosts.
- If auto-migration policies exist, also ensure Workers cannot migrate to AMD hosts during runtime.

7.4 Pre-Checks Before Worker Deployment

Before deploying Worker, verify:

- Veeam Backup & Replication Server can reach the API Proxy address and port.
- The Worker network can reach Veeam Server, the backup repository, and any management node IP or the VIP.

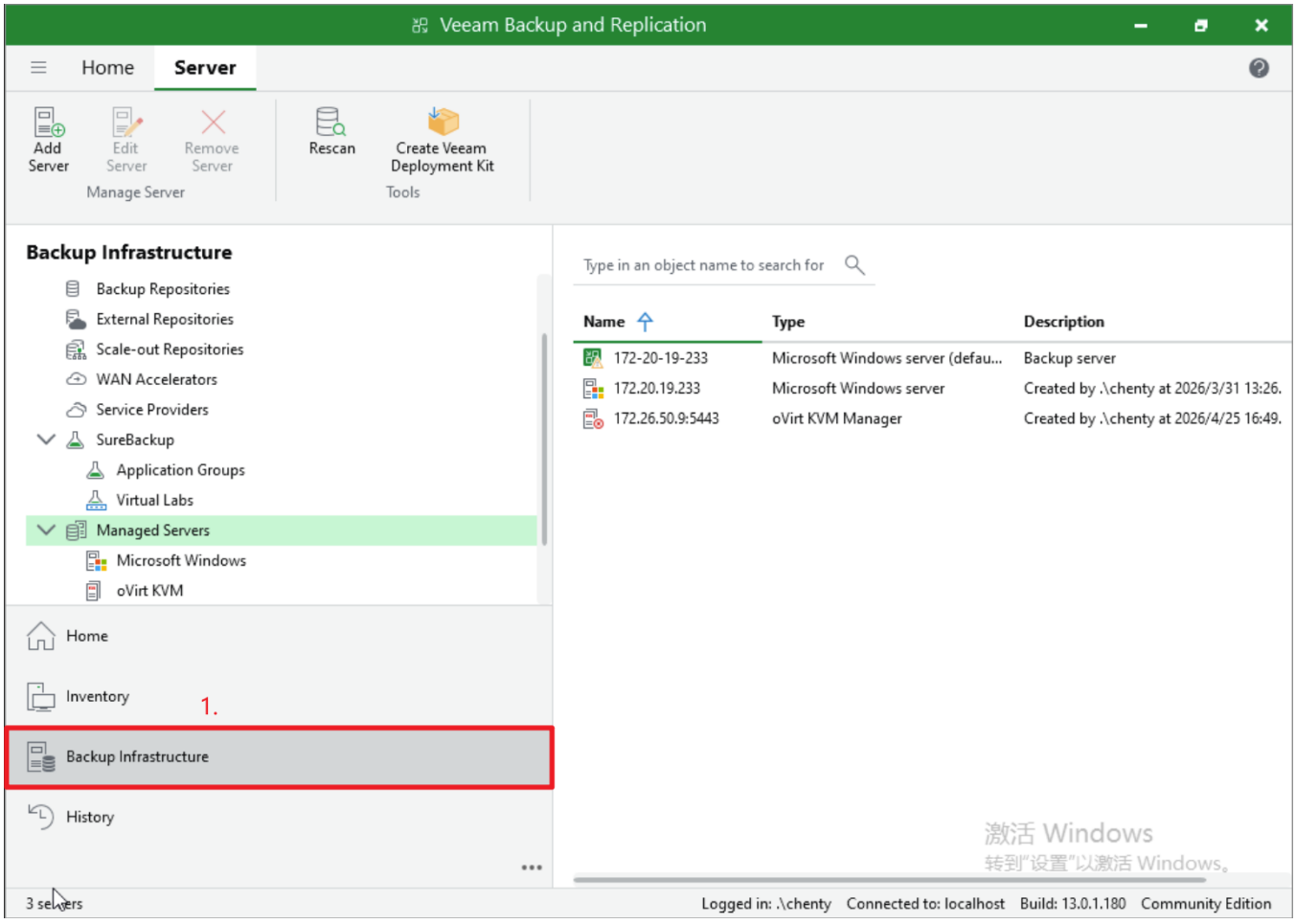
- Worker must be assigned a static IP manually; do not rely on DHCP auto-assignment.
- The Worker network can reach the image transfer endpoints used during backup and restore.
- Target primary storage has enough space for the Worker VM and temporary disks.
- The management-platform account has the permissions required for VM query, restore VM creation, disk create or attach, image transfer, and related operations. During implementation, use an admin account first for quick validation.
- Time synchronization is healthy across the Veeam Server, management nodes, and compute nodes.

7.5 Add API Proxy In Veeam And Deploy Worker

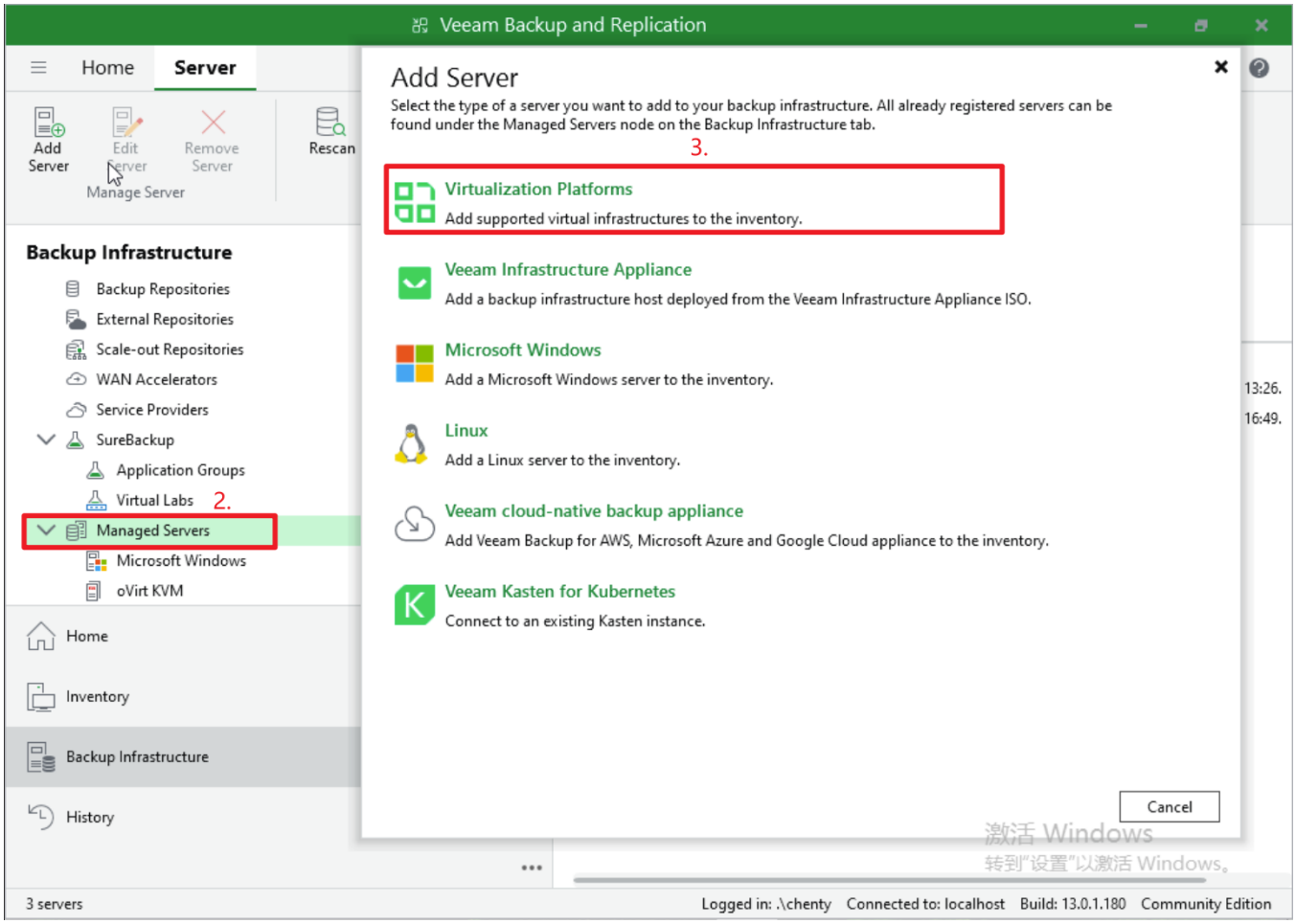
The following steps are based on Veeam Backup & Replication 13.0.1.180. UI labels may vary by version; use the equivalent oVirt / Red Hat Virtualization entry points where necessary.

7.5.1 Add The Virtualization Platform

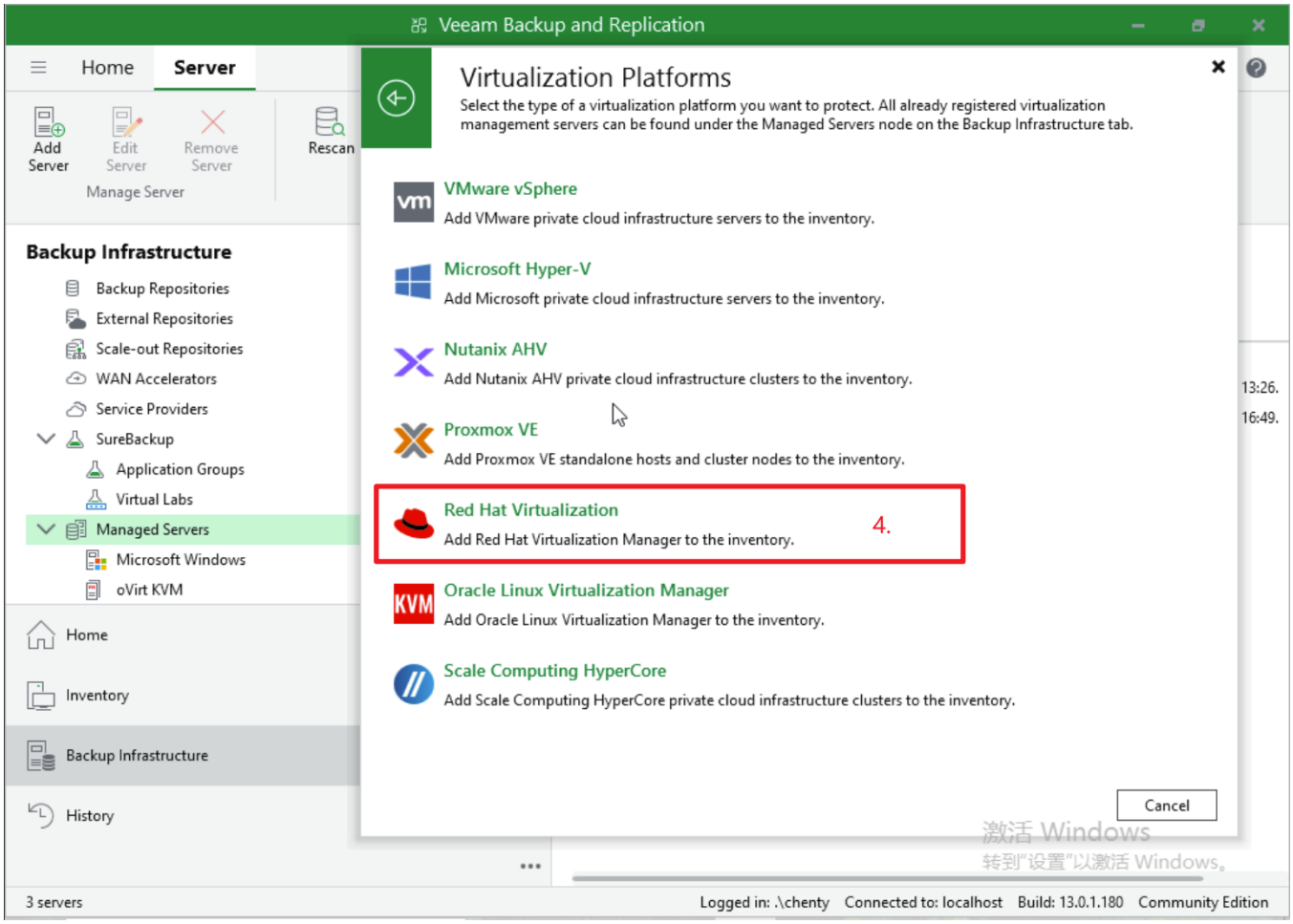
1. In Veeam console, open the virtualization infrastructure management page.



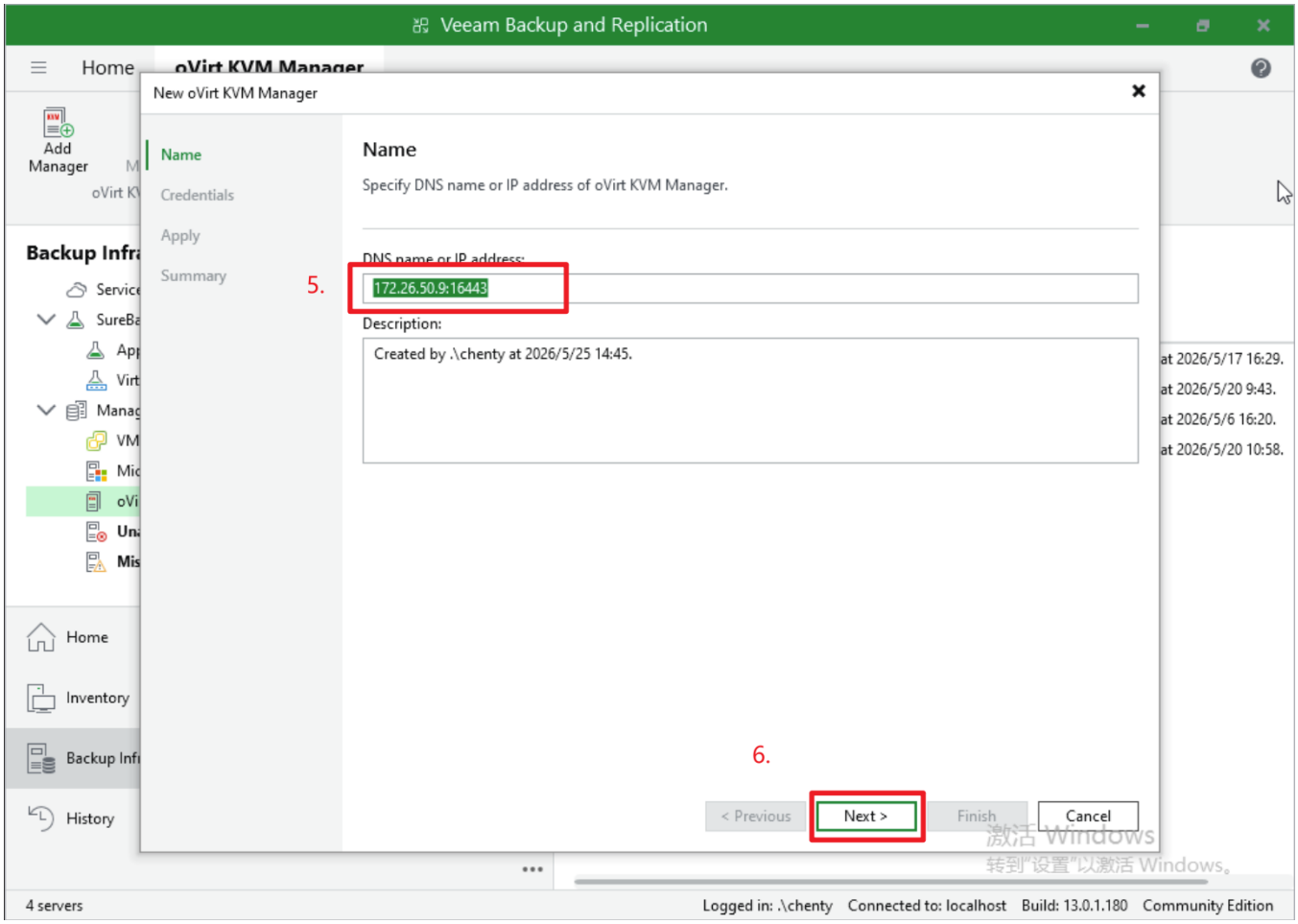
2. Choose to add a virtualization platform.



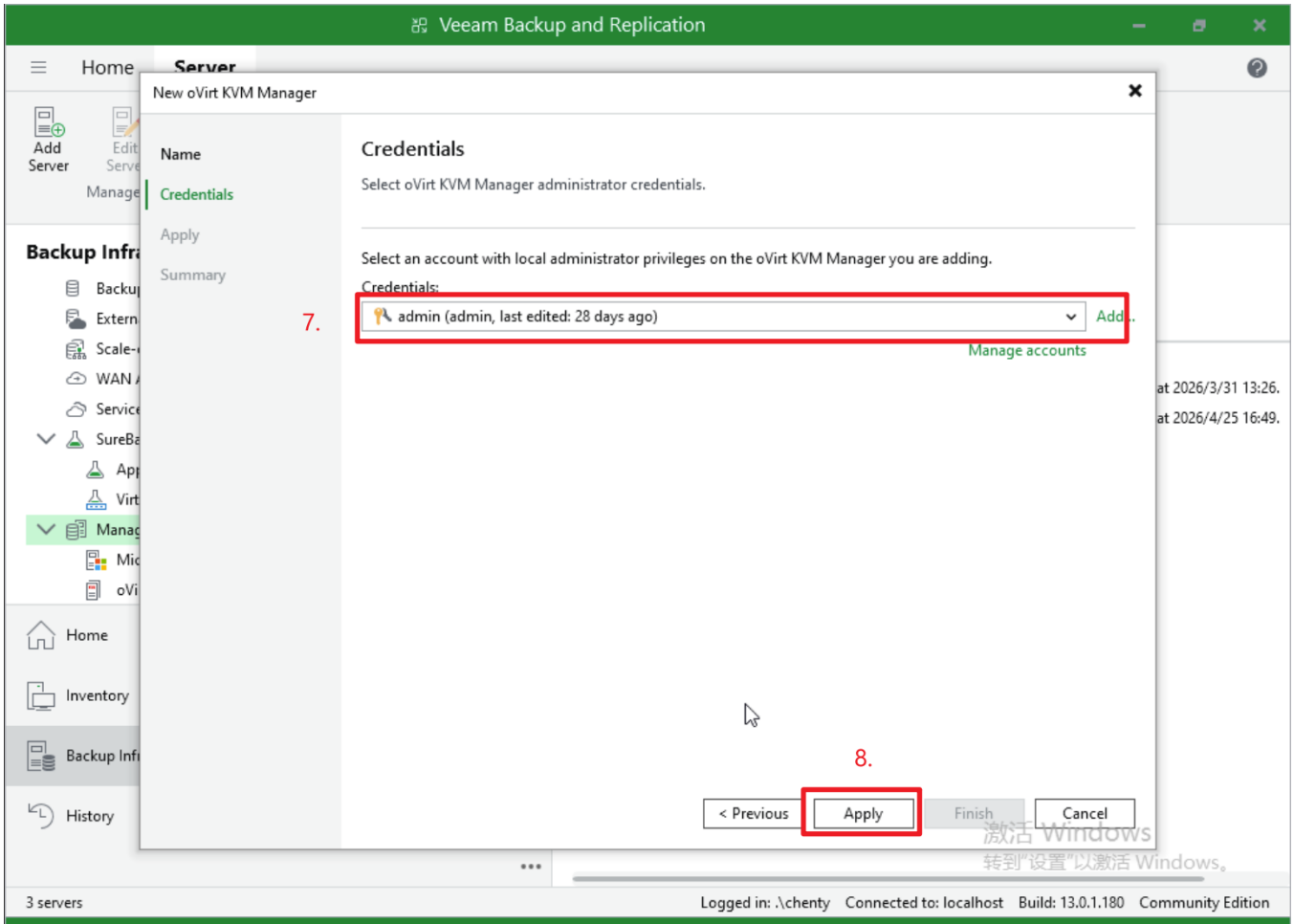
3. Select Red Hat Virtualization / oVirt type.

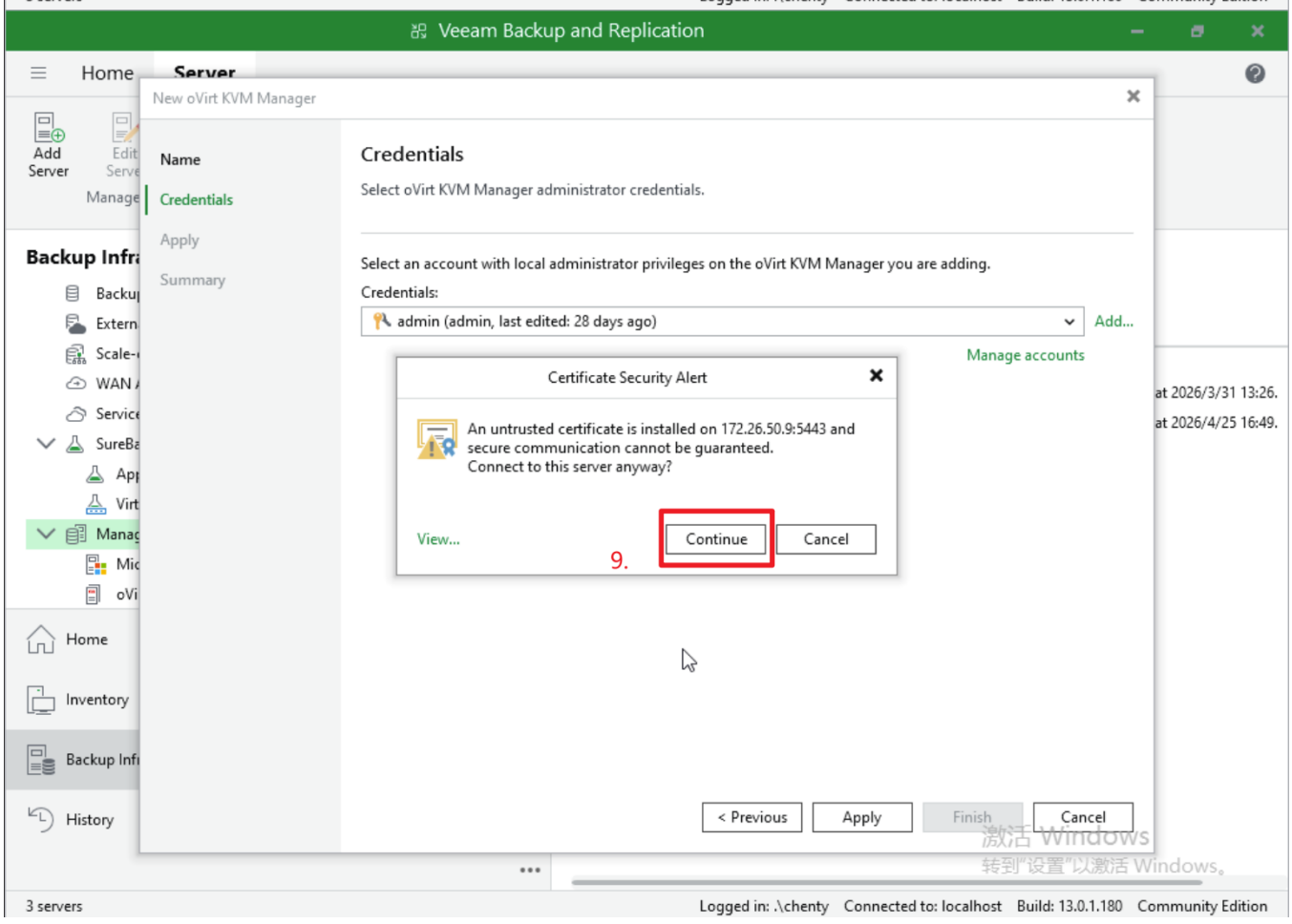


4. Create a new oVirt KVM Manager and fill in the API Proxy address and port.

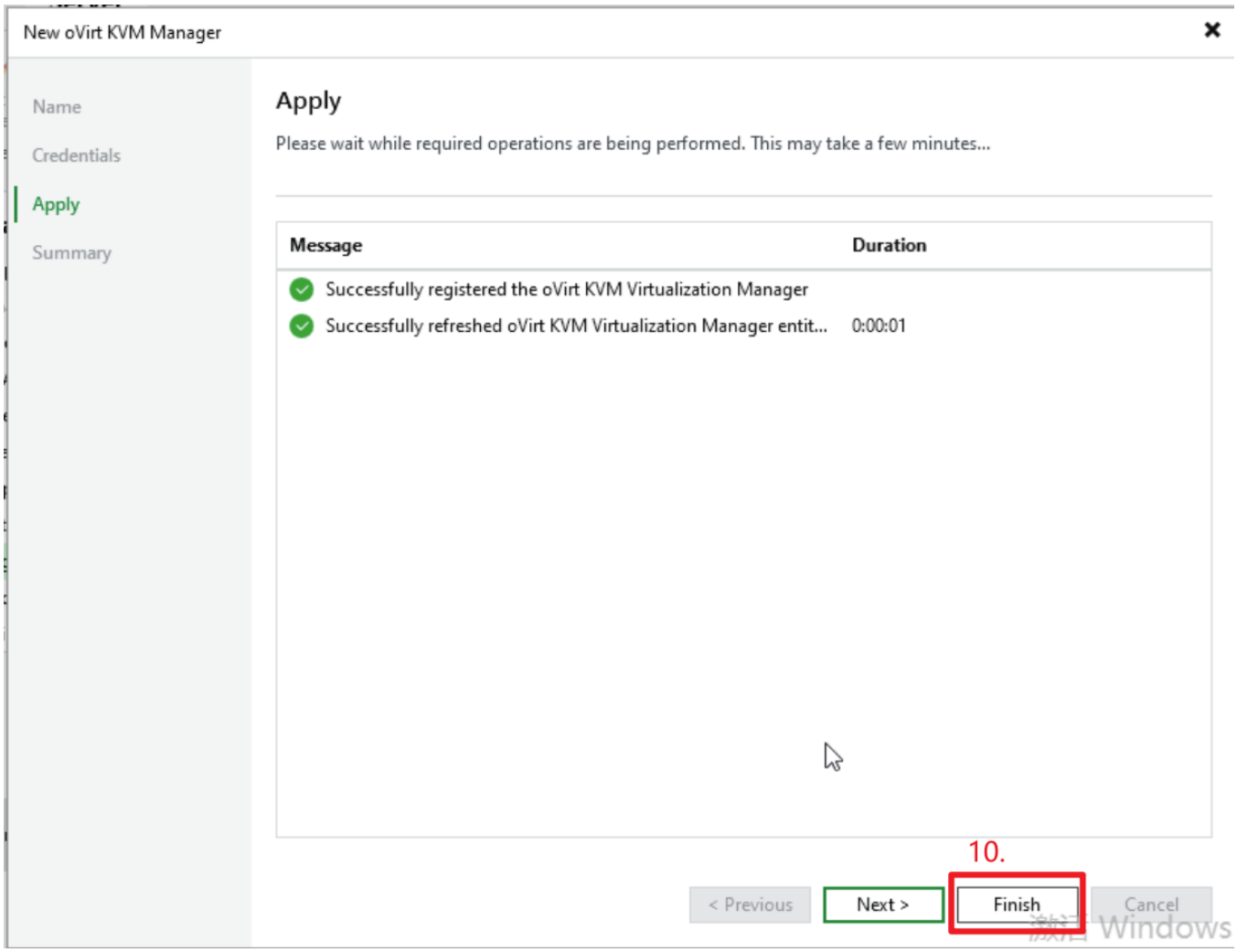


5. Fill in the management-platform account and password, then trust the API Proxy certificate.


Veeam Backup and Replication
Home
Server
Add Server
Edit Server
Manage
Backup Infra
New oVirt KVM Manager
Name
Credentials
Apply
Summary
Credentials
Select oVirt KVM Manager administrator credentials.
Select an account with local administrator privileges on the oVirt KVM Manager you are adding.
Credentials:
7.
admin (admin, last edited: 28 days ago)
Add...
Manage accounts
8.
< Previous
Apply
Finish
Cancel
Home
Inventory
Backup Infr
History
3 servers
Logged in: .\chenty
Connected to: localhost
Build: 13.0.1.180
Community Edition

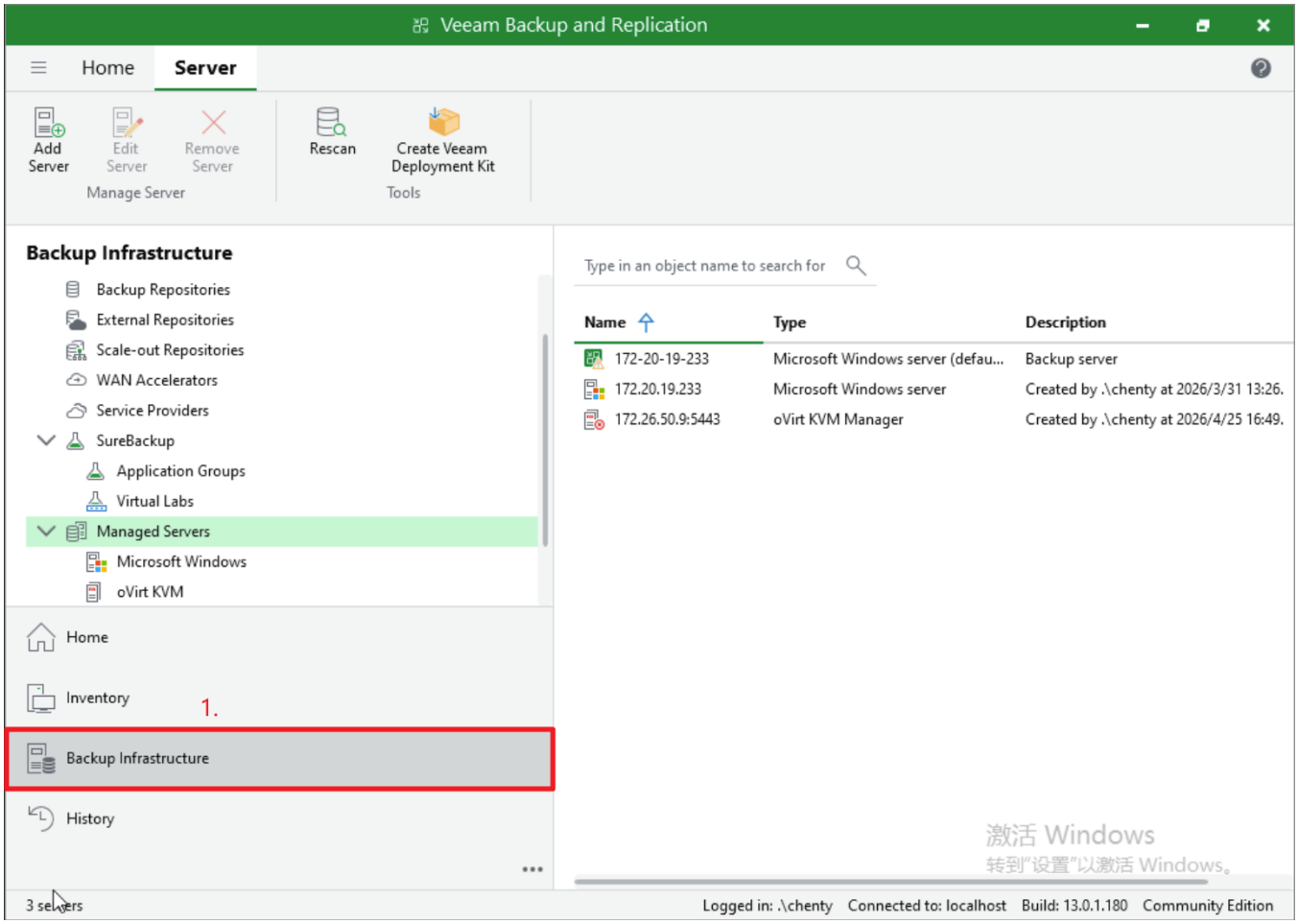

Veeam Backup and Replication
Home
Server
Add Server
Edit Server
Manage
Backup Infra
New oVirt KVM Manager
Name
Credentials
Apply
Summary
Credentials
Select oVirt KVM Manager administrator credentials.
Select an account with local administrator privileges on the oVirt KVM Manager you are adding.
Credentials:
admin (admin, last edited: 28 days ago)
Add...
Manage accounts
Certificate Security Alert
An untrusted certificate is installed on 172.26.50.9:5443 and secure communication cannot be guaranteed.
Connect to this server anyway?
View...
9.
Continue
Cancel
< Previous
Apply
Finish
Cancel
Home
Inventory
Backup Infr
History
3 servers
Logged in: .\chenty
Connected to: localhost
Build: 13.0.1.180
Community Edition

6. Click `Finish` to complete the oVirt KVM Manager creation.

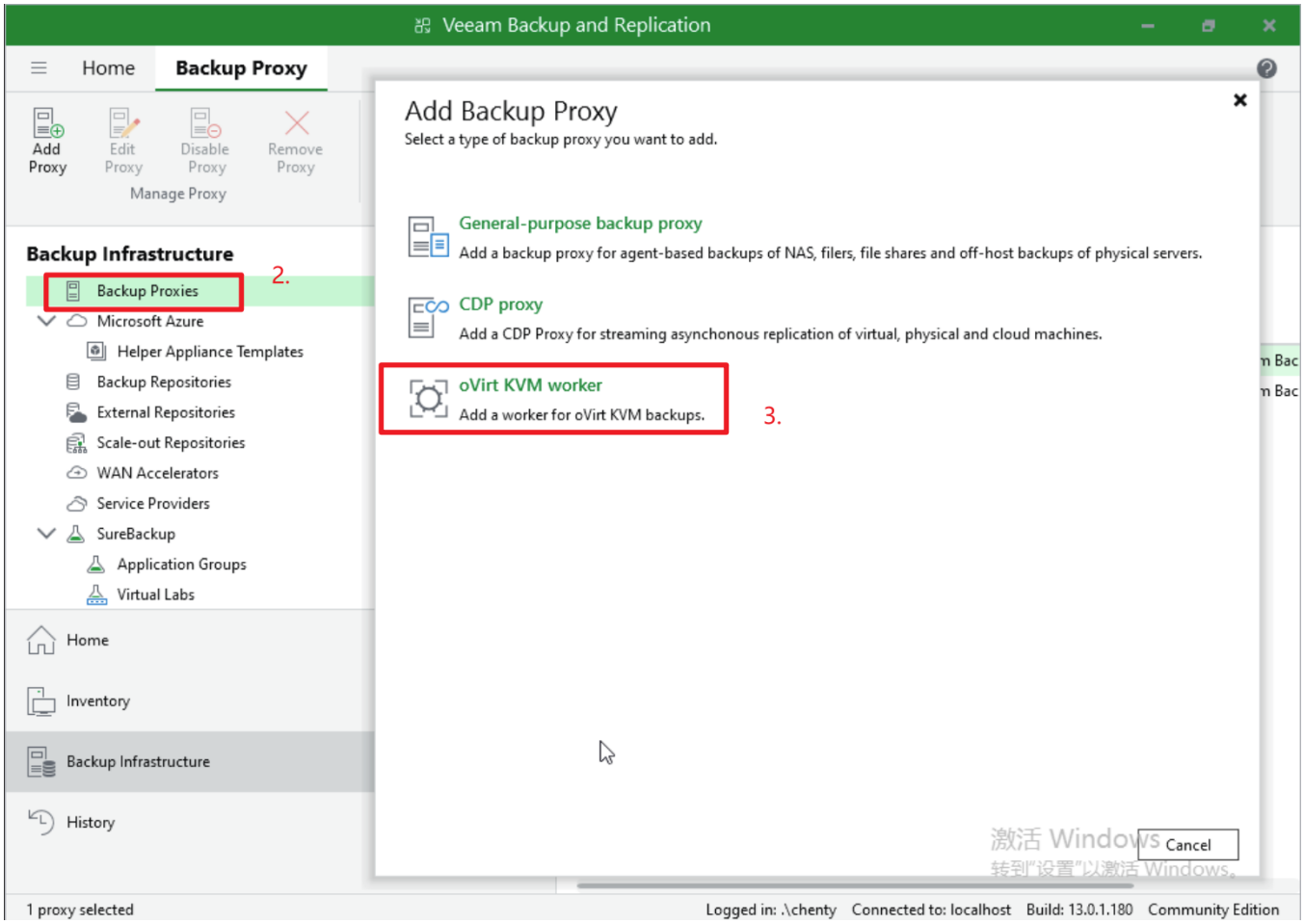


7.5.2 Create Worker

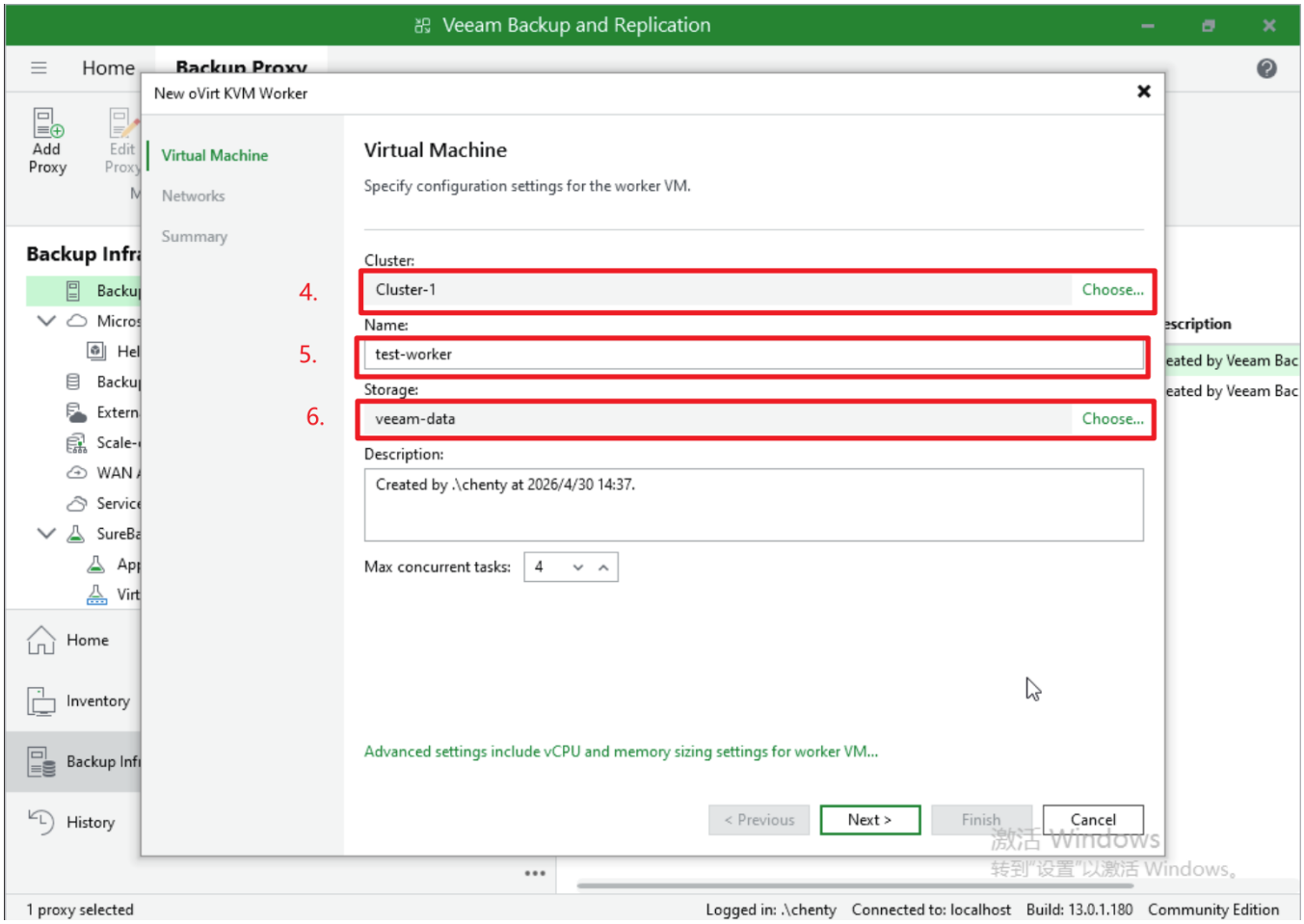
1. In Veeam console, open the virtualization infrastructure management page.



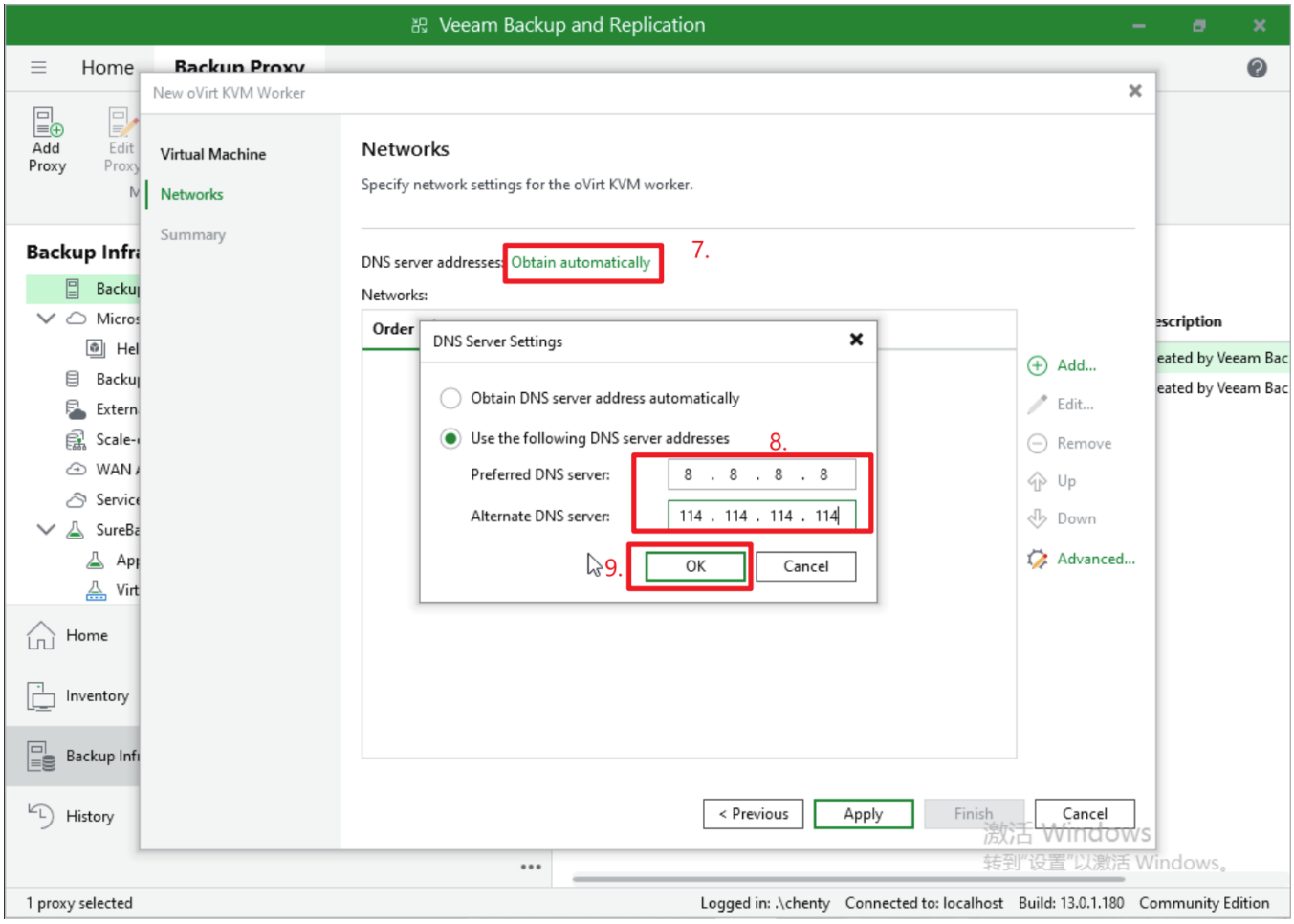
2. Under Backup Proxy, choose to add an oVirt KVM Worker.



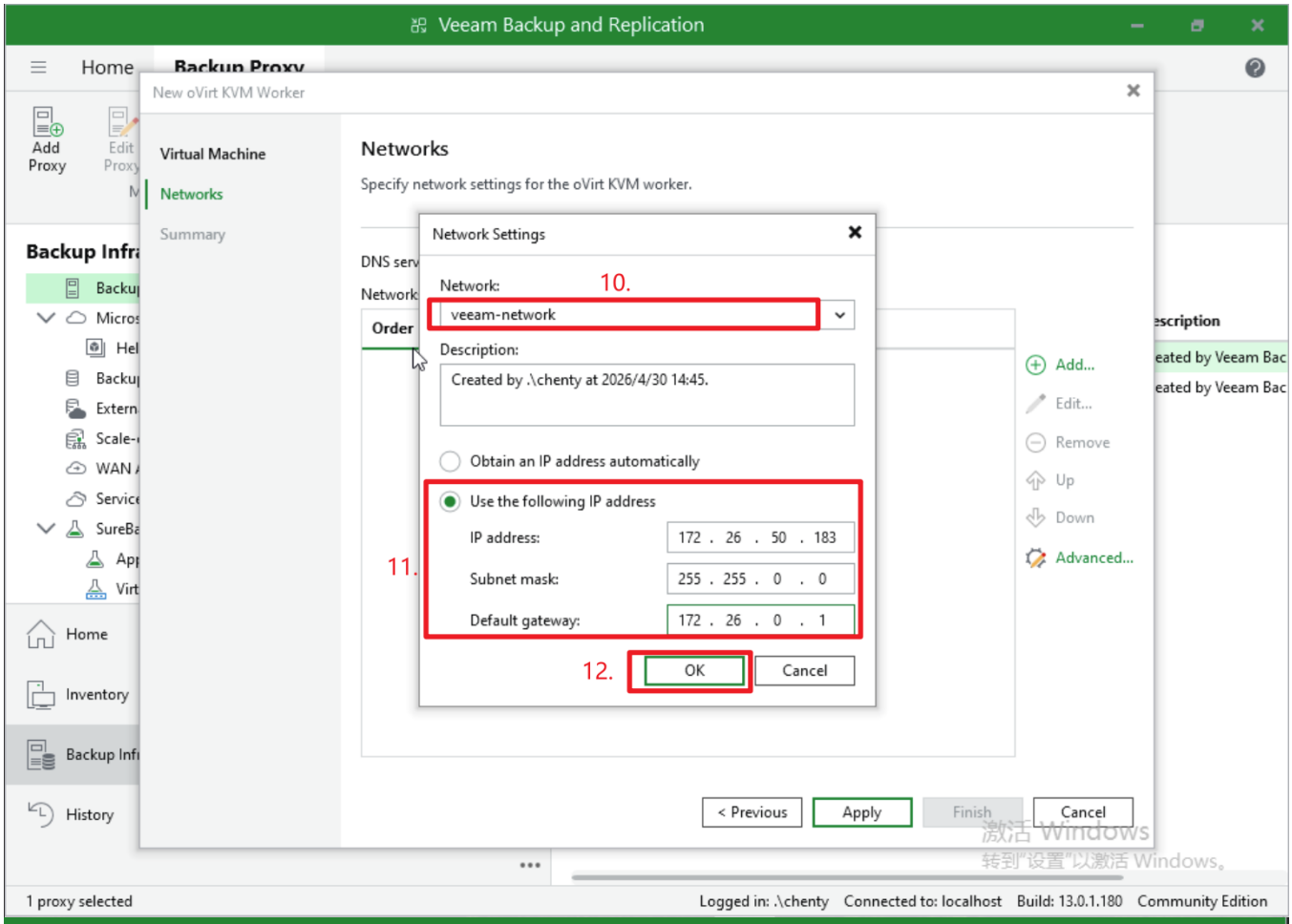
3. Configure Worker options including the cluster, Worker name, and storage pool. When selecting the Worker cluster, you must choose Intel hosts, an Intel-only cluster, or a resource scope restricted to Intel hosts by scheduling policy. CPU, memory, and concurrent task settings can start with the Veeam-recommended values and be tuned later based on backup windows.

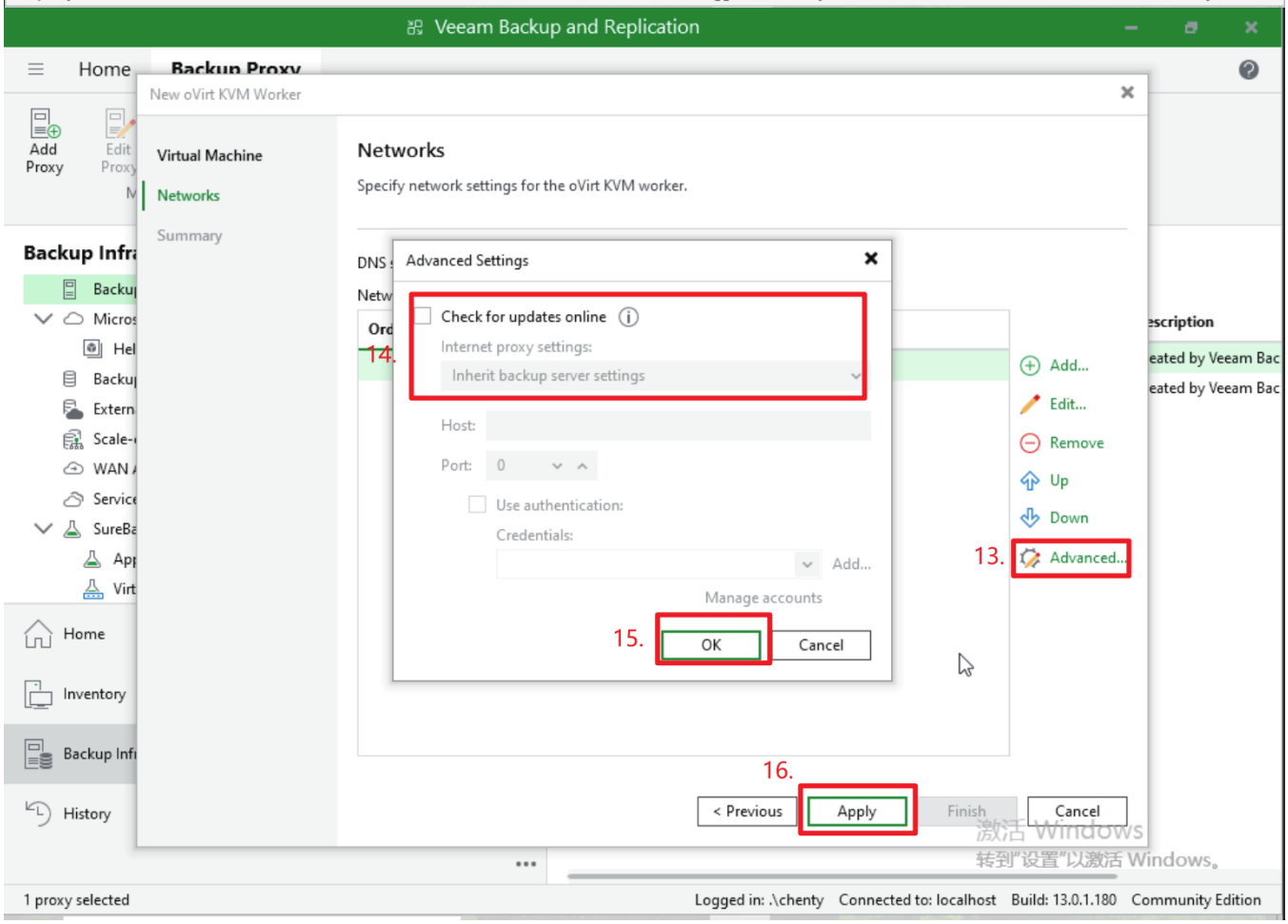


4. Select the Worker management network and ensure it can reach Veeam Server, the backup repository, and the API Proxy address.

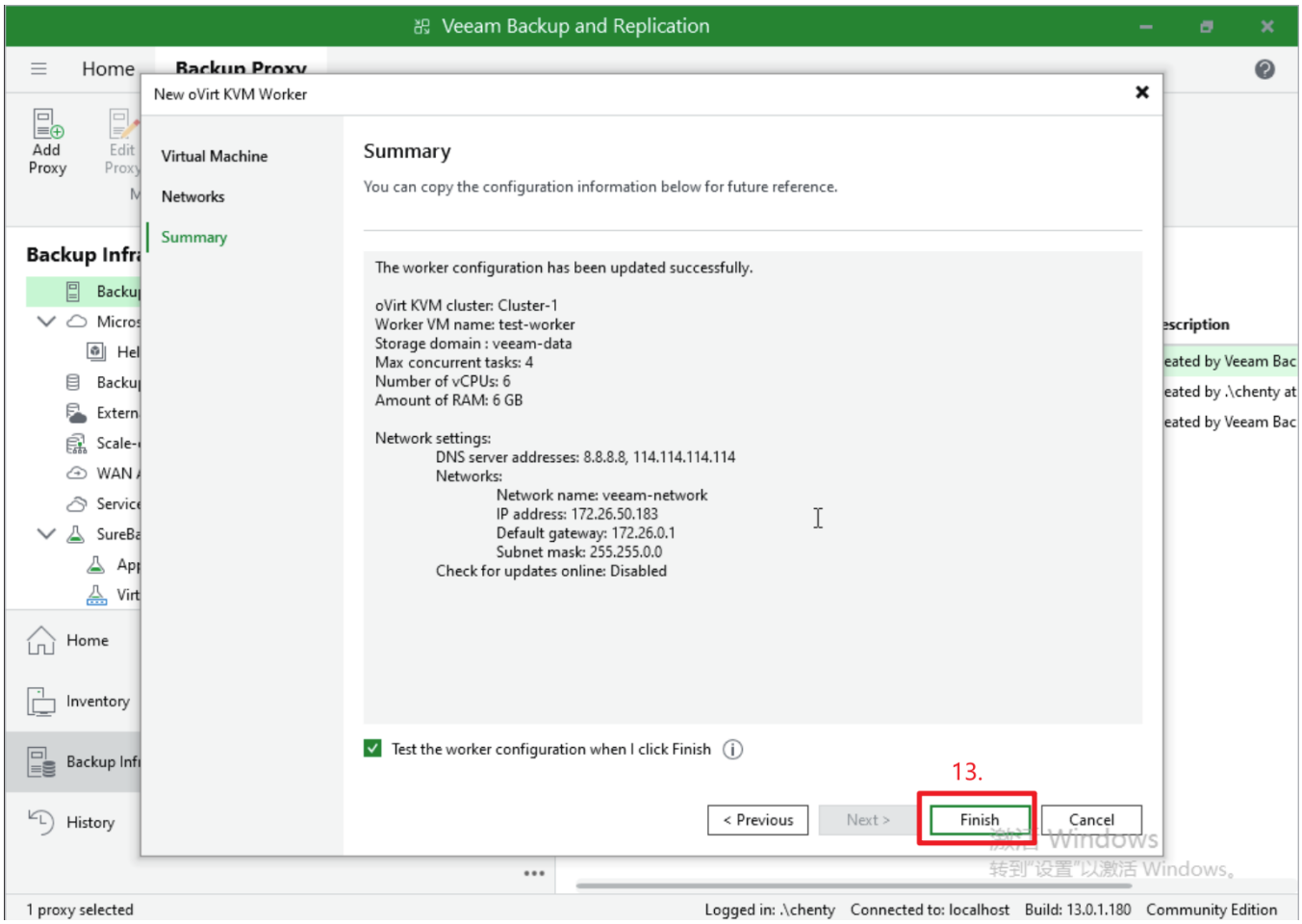


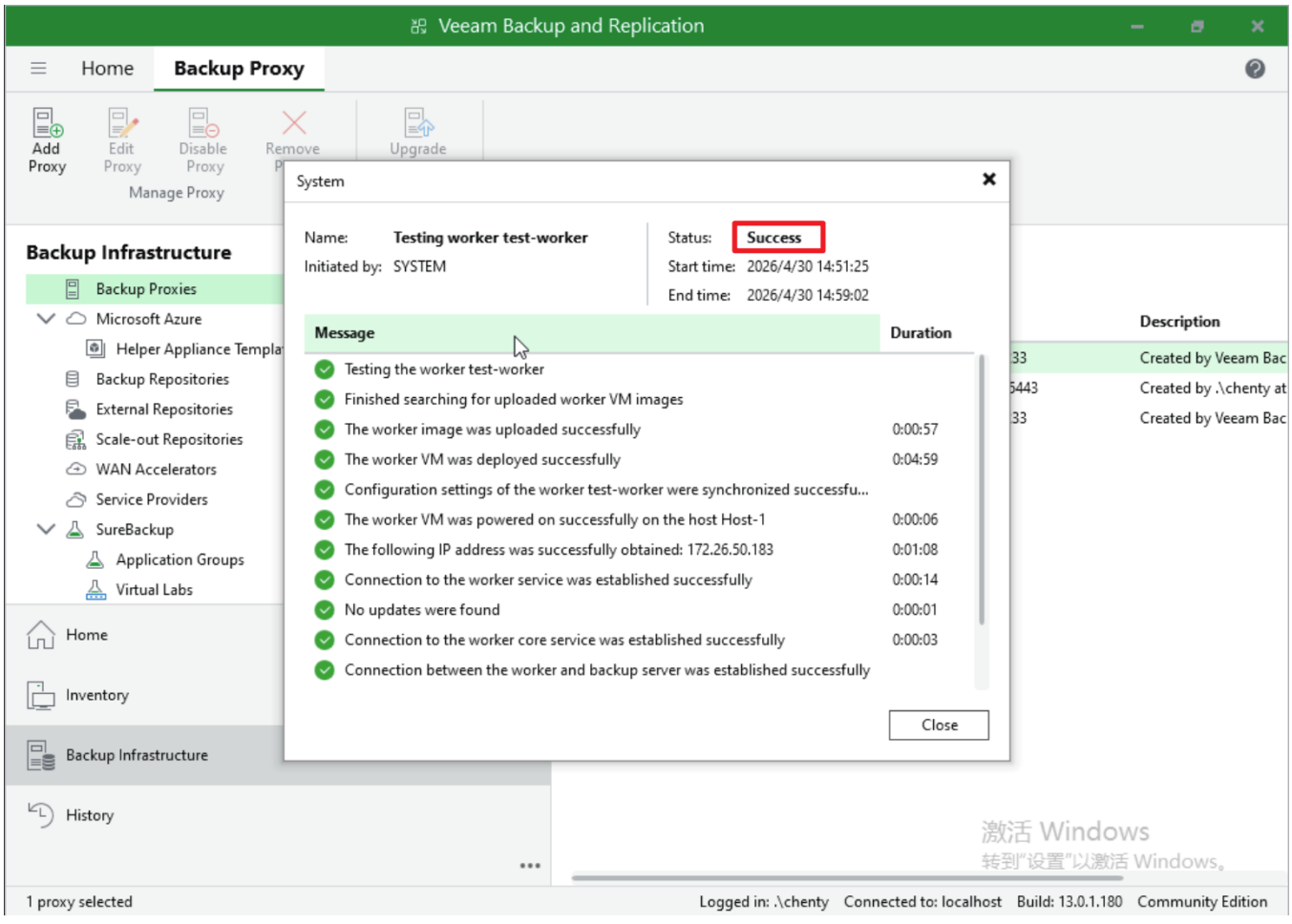
5. Assign a static IP manually for the Worker and ensure it is reachable on the management network and not duplicated by DHCP or other systems.


Veeam Backup and Replication
Home
Backup Proxy
New oVirt KVM Worker
Virtual Machine
Networks
Summary
Networks
Specify network settings for the oVirt KVM worker.
Network Settings
10.
Network:
veeam-network
Description:
Created by .\chenty at 2026/4/30 14:45.
Obtain an IP address automatically
Use the following IP address
11.
IP address: 172 . 26 . 50 . 183
Subnet mask: 255 . 255 . 0 . 0
Default gateway: 172 . 26 . 0 . 1
12.
OK
Cancel
Add...
Edit...
Remove
Up
Down
Advanced...
< Previous
Apply
Finish
Cancel
1 proxy selected
Logged in: .\chenty
Connected to: localhost
Build: 13.0.1.180
Community Edition

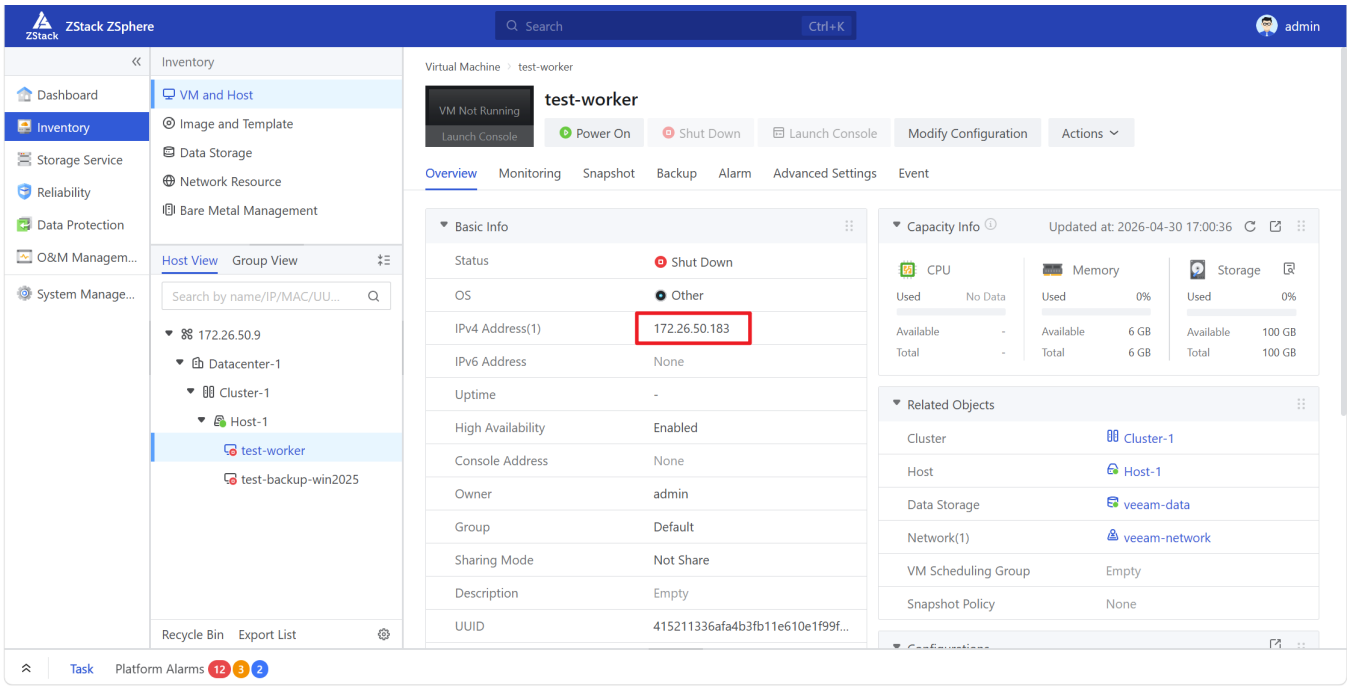

Veeam Backup and Replication
Home
Backup Proxy
New oVirt KVM Worker
Virtual Machine
Networks
Summary
Networks
Specify network settings for the oVirt KVM worker.
Advanced Settings
14.
Check for updates online
Internet proxy settings:
Inherit backup server settings
Host:
Port: 0
Use authentication:
Credentials:
Add...
Manage accounts
15.
OK
Cancel
13.
Add...
Edit...
Remove
Up
Down
Advanced...
16.
< Previous
Apply
Finish
Cancel
1 proxy selected
Logged in: .\chenty
Connected to: localhost
Build: 13.0.1.180
Community Edition

6. After completing the wizard, confirm in the management-platform console that the Worker VM is created successfully, its IP matches the expected static address, and the VM is running.


Veeam Backup and Replication
Home
Backup Proxy
New oVirt KVM Worker
Virtual Machine
Networks
Summary
Summary
You can copy the configuration information below for future reference.
The worker configuration has been updated successfully.
oVirt KVM cluster: Cluster-1
Worker VM name: test-worker
Storage domain : veeam-data
Max concurrent tasks: 4
Number of vCPUs: 6
Amount of RAM: 6 GB
Network settings:
DNS server addresses: 8.8.8.8, 114.114.114.114
Networks:
Network name: veeam-network
IP address: 172.26.50.183
Default gateway: 172.26.0.1
Subnet mask: 255.255.0.0
Check for updates online: Disabled
Test the worker configuration when I click Finish
13.
< Previous
Next >
Finish
Cancel
Add Proxy
Edit Proxy
Backup Infra
Home
Inventory
Backup Infr
History
1 proxy selected
Logged in: .\chenty
Connected to: localhost
Build: 13.0.1.180
Community Edition


Veeam Backup and Replication
Home
Backup Proxy
Add Proxy
Edit Proxy
Disable Proxy
Remove
Upgrade
Manage Proxy
System
Name: Testing worker test-worker
Initiated by: SYSTEM
Status: Success
Start time: 2026/4/30 14:51:25
End time: 2026/4/30 14:59:02
Message
Duration
Testing the worker test-worker
Finished searching for uploaded worker VM images
The worker image was uploaded successfully 0:00:57
The worker VM was deployed successfully 0:04:59
Configuration settings of the worker test-worker were synchronized successfu...
The worker VM was powered on successfully on the host Host-1 0:00:06
The following IP address was successfully obtained: 172.26.50.183 0:01:08
Connection to the worker service was established successfully 0:00:14
No updates were found 0:00:01
Connection to the worker core service was established successfully 0:00:03
Connection between the worker and backup server was established successfully
Close
Backup Infrastructure
Backup Proxies
Microsoft Azure
Helper Appliance Templa
Backup Repositories
External Repositories
Scale-out Repositories
WAN Accelerators
Service Providers
SureBackup
Application Groups
Virtual Labs
Home
Inventory
Backup Infrastructure
History
Description
Created by Veeam Bac
Created by .\chenty at
Created by Veeam Bac
1 proxy selected
Logged in: .\chenty
Connected to: localhost
Build: 13.0.1.180
Community Edition

7. Back in Veeam console, confirm the Worker status is available.



7.6 Troubleshooting Worker Deployment Failures

If Worker deployment or startup fails, prioritize the checks below:

Symptom	Check item	Recommended action
Worker VM cannot start	Whether it was scheduled to an AMD host	Delete the failed Worker and redeploy to an Intel host or Intel-only cluster
Worker created but Veeam connectivity is unstable	Whether the correct static IP was manually assigned	Reconfigure the fixed static IP and confirm accessibility from Veeam Server, API Proxy, and the repository
Veeam shows Worker unavailable	Worker cannot reach Veeam Server / API Proxy / repository	Check security groups, firewall, routing, and DNS
Veeam cannot add the virtualization platform	API Proxy address, port, certificate, or credentials are incorrect	Validate with `curl -k https://<Address>:16443/ovirt-engine/api` and the token endpoint first
Backup job fails after startup	Worker network or storage access issue	Check the Worker network, API Proxy logs, and Veeam job logs
Restore job fails to create a VM	Management-platform account permissions are insufficient, or the target cluster, network, or storage selection is incorrect	Re-test with an admin account and confirm the target restore resources are available

Recommended API Proxy-side log collection:

```
./collect-api-proxy-logs.sh root@<EntryNodeIP>
```

Also export the related Veeam task logs so Veeam request timestamps can be correlated with API Proxy logs.

7.7 Backup And Restore Recommendations

- When creating backup jobs in Veeam, select the VMs exposed through API Proxy by following the oVirt / RHV VM selection flow.
- For first-time integration, run an Active Full backup for one test VM to verify the Worker, network, and storage paths.
- During restore, follow the oVirt / RHV restore flow to choose the restore point, target cluster, target storage, and target network.
- In two-management-node deployments, Veeam should use the API Proxy VIP as the connection address to avoid reaching a non-active service entry after a management-node switchover.
- If the API Proxy VIP, management network, or certificate SAN changes, re-check the certificate and Veeam connection settings; regenerate the API Proxy TLS certificate if needed.
- In production, create a dedicated management-platform account for Veeam and grant only the permissions required for backup and restore. During integration testing, an admin account can be used first to shorten the troubleshooting path.

8. Entry IP Rules For Single-Management-Node And Two-Management-Node Deployments

Deployment, uninstall, log collection, and log cleanup all support single-management-node and two-management-node deployments.

`<node-ip>` denotes the entry node used by the script. It is not always the same as the address clients should use to access API Proxy.

Scenario	`<node-ip>` used in commands	Address clients should use to access API Proxy
Single management node	The only management node IP	The same management node IP
Two management nodes	Either management node IP; the script uses it as the entry node and automatically detects the other management node	VIP

- The entry node can be any management node.
- The scripts detect the other management node and the VIP from keepalived configuration.
- During remote operations, the script connects to the entry node first, then relays to the other management node.

- After installation, if a VIP is detected, the script prefers the VIP health endpoint.
- Log cleanup does not stop keepalived and does not intentionally move the VIP.
- If `api-proxy` is running, log cleanup stops the service briefly, performs cleanup, then starts it again and verifies health.

9. TLS

The service uses HTTPS by default:

- Listen port: `16443`
- Keystore: `/etc/api-proxy/tls/server-keystore.p12`
- Keystore type: `PKCS12`

If the keystore is missing, the service startup flow can generate a self-signed certificate through the built-in initialization logic. For production certificates, place the keystore in advance and override the path or password through start arguments.

Example:

```
sudo APP_OPTS="--server.ssl.key-store=file:/etc/api-proxy/tls/server-keystore.p12 --server.ssl.key-store-password=<password>" ./install-api-proxy.sh --install
```

10. Logging And Diagnostics

10.1 Enable API Request / Response Logging

By default, the service records normal runtime logs but does not record full API request and response bodies. Enable detailed API logging temporarily when troubleshooting protocol interactions:

```
./deploy-api-proxy.sh --install --enable-api-logging root@<node-ip>
```

For local install, use the equivalent command:

```
sudo ./install-api-proxy.sh --install --enable-api-logging
```

This writes `--adapter.api-logging.enabled=true` into the systemd start arguments. A later deployment or reinstall without this flag refreshes the unit and returns to the default disabled state.

10.2 Log Collection

Collect full logs from a single management node or a two-management-node deployment:

```
./collect-api-proxy-logs.sh root@<node-ip>
```

Use a specific output directory:

```
./collect-api-proxy-logs.sh --output-dir /tmp/api-proxy-logs root@<node-ip>
```

Common options:

```
./collect-api-proxy-logs.sh --keepalived root@<node-ip>
./collect-api-proxy-logs.sh --skip-keepalived root@<node-ip>
```

Notes:

- In two-management-node deployments, keepalived configuration and journal are collected by default.
- In single-management-node deployments, keepalived data is skipped by default unless you explicitly pass `--keepalived`.
- The default output directory is named `api-proxy-logs-YYYYMMDD-HHMMSS`.
- The top-level output directory also includes `collection-summary.txt` with the entry node, target nodes, VIP, and collection mode.

Each node directory usually contains:

File	Content
`summary.txt`	Node status, service state, VIP state, and log directory overview
`api-proxy-logs.tar.gz`	API Proxy file logs
`journal-api-proxy.log`	API Proxy systemd journal
`mn-logs.tar.gz`	Management node file logs
`journal-mn.log`	Management node service journal
`keepalived.conf`	Dual-management-node configuration, collected by default in two-management-node deployments
`journal-keepalived.log`	keepalived journal, collected by default in two-management-node deployments

10.3 Log Cleanup

By default, the script truncates the current service log, removes rotated compressed logs, and keeps systemd journal for the last 7 days:

```
./cleanup-api-proxy-logs.sh root@<node-ip>
```

Common options:

```
./cleanup-api-proxy-logs.sh --vacuum-time 7d root@<node-ip>
./cleanup-api-proxy-logs.sh --vacuum-size 1G root@<node-ip>
```

```
./cleanup-api-proxy-logs.sh --skip-journal-vacuum root@<node-ip>
./cleanup-api-proxy-logs.sh --delete-log-dir root@<node-ip>
```

Notes:

- If the service is running before cleanup, the script stops `api-proxy` briefly, performs cleanup, then starts it again and checks health.
- The script does not stop keepalived and does not move the VIP intentionally.
- `--delete-log-dir` deletes rotated application logs and truncates the active log.
- `--skip-failover` and `--force-active-cleanup` are still accepted for compatibility, but they only emit a warning now and do not trigger VIP failover or any keepalived action.

11. Common Environment Variables

Variable	Default	Description
`API_PROXY_SSH_PASSWORD`	empty	SSH password used by remote scripts; prompt if unset
`API_PROXY_ASSUME_YES`	empty	Set to `1` to skip replacement confirmation
`API_PROXY_ENABLE_API_LOGGING`	`0`	Set to `1` during remote install to match `--enable-api-logging`
`SERVICE_NAME`	`api-proxy`	systemd service name
`INSTALL_DIR`	`/opt/api-proxy`	Install directory
`LOG_DIR`	`/opt/api-proxy/logs`	Log directory
`UPLOAD_DIR`	`/opt/api-proxy/uploads`	Upload workspace
`RUNTIME_WORK_DIR`	`/tmp/api-proxy`	Runtime workspace
`JAVA_BIN`	auto-detect	Java executable path
`JAVA_OPTS`	`-Xms256m -Xmx2g`	JVM arguments
`APP_OPTS`	empty	Service start arguments
`AUTO_CREATE_UNIT`	`1`	Auto-create the systemd unit if it is missing
`SYSTEMD_UNIT_PATH`	`/etc/systemd/system/api-proxy.service`	systemd unit path
`HEALTH_CHECK_URL`	`https://127.0.0.1:16443/ovirt-engine/api`	Local health check URL
`BACKEND_CONFIG_PATH`	auto-detect	Backend configuration path; the install script reads

		this variable first
MYSQL_BIN	mysql	mysql client used during uninstall database cleanup
API_PROXY_DB_NAME	auto-detect	Database name used for owned-table cleanup during uninstall; defaults to the current backend database
API_PROXY_DB_HOST	auto-detect	Database host used for owned-table cleanup during uninstall
API_PROXY_DB_PORT	auto-detect	Database port used for owned-table cleanup during uninstall
API_PROXY_DB_USERNAME	auto-detect	Database user used for owned-table cleanup during uninstall
API_PROXY_DB_PASSWORD	auto-detect	Database password used for owned-table cleanup during uninstall
CLEAN_INSTALL_DIR	1	Remove install directory during uninstall
CLEAN_LOGS	1	Remove log directory during uninstall
CLEAN_UPLOADS	1	Remove upload directory during uninstall
CLEAN_RUNTIME_WORK_DIR	1	Remove runtime workspace during uninstall
CLEAN_DATABASE_TABLES	1	Drop owned service tables during uninstall

12. Health Checks And Troubleshooting

Service status:

```
systemctl status api-proxy --no-pager -l
systemctl is-active api-proxy
```

View logs:

```
journalctl -u api-proxy --no-pager -n 200
less /opt/api-proxy/logs/api-proxy.log
```

Check the port and endpoint:

```
ss -lntp | grep ':16443'
curl -k -i https://127.0.0.1:16443/ovirt-engine/api
```

Common issues:

Symptom	Check
SSH connection fails	root login permissions, password, network connectivity, firewall

Java missing	Install Java 8, or set `JAVA_BIN` / `JAVA_HOME`
Service start fails with `qemu-img not found at startup` in logs	Install `qemu-img`, ensure it is in `PATH`, or configure `adapter.backup.qemu-img-path`
Local health check times out	Inspect `journalctl -u api-proxy` and the main log, then verify port, TLS, and database connectivity
Database cleanup is skipped during uninstall	Verify whether the `mysql` client exists, or provide `API_PROXY_DB_*` settings explicitly
VIP health check fails	Verify VIP ownership, port reachability, and keepalived health
Veeam Worker remains unavailable	Verify Intel-host placement, static IP configuration, and network reachability to API Proxy and the repository

13. Minimum Acceptance Checklist

Before go-live, it is recommended to complete at least the following checks:

- `deploy-api-proxy.sh --install` completes successfully.
- `systemctl status api-proxy` shows the service running normally.
- `curl -k https://<AnyManagementNodeIP-or-VIP>:16443/ovirt-engine/api` returns success.
- Veeam can successfully add API Proxy as oVirt / RHV type.
- Veeam console can discover the clusters, hosts, storage, and VMs exposed through API Proxy.
- Veeam Worker is deployed on an Intel host and shown as available in Veeam.
- At least one test VM backup completes successfully.
- At least one test restore completes, and the restored VM is verified for boot, network, and disk behavior.

14. Security Recommendations

- Do not keep API request / response logging enabled longer than necessary; redeploy without `--enable-api-logging` after troubleshooting.
- Log bundles may contain request paths, resource IDs, stack traces, and environment details. Handle them according to internal procedures before sharing.
- Prefer short-lived environment variables for SSH passwords instead of hard-coding them into scripts or documentation.
- For production certificates, use a keystore issued by a trusted CA and manage the keystore password carefully.