Virtuozzo PowerPanel

Administrator's Guide

June 21, 2023

Virtuozzo International GmbH

Vordergasse 59

8200 Schaffhausen

Switzerland

Tel: + 41 52 632 0411

Fax: + 41 52 672 2010

https://virtuozzo.com

Contents

CHAPTER 1

About Virtuozzo PowerPanel

Virtuozzo PowerPanel is a solution for service providers that allows their customers to independently manage purchased virtual environments hosted on Virtuozzo Hybrid Server 7 nodes. Virtuozzo PowerPanel eliminates the need for service providers to develop their own VPS management portals, helping them save resources and focus on their primary goals.

In terms of architecture, Virtuozzo PowerPanel consists of a single controller node and multiple Virtuozzo Hybrid Server 7 compute nodes. The controller node runs the database, identity provider, API, VNC proxy, web server, and other required services while each compute node runs a compute service that interacts with Virtuozzo Hybrid Server 7.

Virtuozzo PowerPanel typically integrates with service provider's identity database via LDAP. However, the service provider may also choose to create users locally on the controller node.

Virtuozzo PowerPanel has two modes: admin and user. The admin mode provides means to manage assignment of virtual environments to users while the user mode offers virtual environment management tools. The administrator logged in the admin mode can also switch to the user mode and back for complete control over virtual environments and their assignments.

The next chapter explains how to log in to Virtuozzo PowerPanel and perform administration tasks.

CHAPTER 2

Installing Virtuozzo PowerPanel

To install Virtuozzo PowerPanel, you need to deploy one controller (management) node and after that as many compute nodes as required. The following sections describe these procedures in detail.

Note: Installation is performed by means of Ansible playbooks.

The following requirements that must be met to install Virtuozzo PowerPanel:

1. The controller can only be deployed on VzLinux 7 or CentOS 7 installed on a physical host or in a virtual machine or container. It cannot be installed on Virtuozzo Hybrid Server 7 or inside a Virtuozzo Automator container.
2. The controller's physical or virtual environment must have at least 2 CPU cores and 8GB of RAM.
3. The compute component can only be installed on a Virtuozzo Hybrid Server 7 host.
4. Operating systems on the controller and all compute nodes must be fully updated.
5. Each node in the PowerPanel cluster—the controller and all computes—must have a fully qualified domain name (FQDN) and be accessible by it.
6. Time must be synchronized on the controller and all compute nodes. For example, by running on each node `yum install ntpdate && ntpdate pool.ntp.org`.

It is also recommended to do the following:

- Connect the controller node to two networks: (a) internal for communication with compute nodes and (b) external for public access.
- Connect compute nodes to an internal network. Let only VMs and containers on compute nodes have

public access.

- Make sure that all compute nodes are configured to store backups locally. This is required for attaching backups to VEs to work.

2.1 Deploying the Controller

Note: If you need to change certificates used in deployment (e.g., replace self-signed ones with those from a trusted CA) or change controller hostname, rerun the `vzapi-installer controller` command with the new parameters. To change the controller hostname while keeping the same self-signed certificates, use the `--force-cert-gen` parameter.

To deploy the controller, do the following on the host chosen to be the controller:

1. (CentOS 7 only) Disable SELinux. To do this, set `SELINUX=disabled` in `/etc/sysconfig/selinux` file and reboot the system.
2. (CentOS 7 only) Import the key Virtuozzo PowerPanel packages are signed with:

   ```
   # rpm --import https://docs.virtuozzo.com/keys/VIRTUOZZO_GPG_KEY
   ```

3. Install the `pp-release` package. It will add the repository with PowerPanel packages.

   ```
   # yum install http://repo.virtuozzo.com/pp/releases/2.0.4/x86_64/os/\
   Packages/p/pp-release-2.0.4-3.vl7.noarch.rpm
   ```

4. Install the `vzapi-installer` package. It will provide the Ansible playbooks necessary to deploy PowerPanel and the `vzapi-installer` script that automates said deployment.

   ```
   # yum install vzapi-installer
   ```

5. Launch the deployment script. A self-signed certificate is generated and used by default, however, you are recommended to use your own certificate, private key, and certificate authority (CA) files. If you have one or more intermediate certificates, you can specify them as well to indicate a chain of trust. Every intermediate certificate will be handled by `openssl verify` as untrusted. The deployment script will automatically concatenate the certificates and use the resulting file during installation. The original CA file will added to `httpd` configs.

   - If you have an SSL certificate and key from a third-party trusted CA, run the deployment script as follows:

```
# vzapi-installer controller --ask-passwd --private-ip <IP_address> \
--ssl-ca-file <path_to_file> --ssl-cert-file <path_to_file> \
--ssl-key-file <path_to_file>
```

- If you have your own CA file that you used to generate an SSL key and a certificate that requires intermediate certificates for resolution, run the deployment script as follows:

```
# vzapi-installer controller --ask-passwd --private-ip <IP_address> \
 --ssl-ca-file <path_to_file> --ssl-cert-file <path_to_file> \
 --ssl-key-file <path_to_file> --ssl-intermediate-cert <path_to_file> \
 [--ssl-intermediate-cert <path_to_file> ...]
```

- If you want to deploy with a generated self-signed certificate, run the deployment script as follows:

```
# vzapi-installer controller --ask-passwd --private-ip <IP_address>
```

The `--private-ip <IP_address>` is the required parameter that sets the IP address of the network interface that the controller will use to communicate with compute nodes. Controller services used for internal communication will listen only on this network interface. This can be useful if you have a private network for the controller and compute nodes and do not want controller services to be accessible from other networks.

During setup, you will be asked to provide the administrator's password that will be required to log in to PowerPanel in the admin mode. If you need to specify the password in the command line, you can replace `--ask-passwd` with `-p <admin_passwd>`.

Note: Once set, the administrator's password is stored in the file `/var/lib/vzapi-installer/group_vars/all` alongside with automatically generated passwords for other services used by PowerPanel.

If you have a firewall enabled, the deploy script will create rules to open the required ports (see *Appendix A: Controller and Compute Ports* on page 31).

When deployment is completed successfully, you will see a recap from Ansible and information on how to access Virtuozzo PowerPanel web interface. For example:

```
PLAY RECAP *********************************************************************
ctrl.example.com         : ok=57   changed=49   unreachable=0    failed=0

Controller has been deployed successfully!
```

```
Virtuozzo PowerPanel web UI can be accessed at https://ctrl.example.com
with this username/password: admin/password
```

6. If you used a self-signed certificate generated during deployment, import its file `/var/lib/vzapi/vzapi_rootCA.crt` into the browser in which you will open Virtuozzo PowerPanel web interface.

If you later need to redeploy or update the controller without changing the admininstrator's password, skip password-related options and run

```
# vzapi-installer controller
```

Note: By default, each user is allowed to create 3 backups of each of their VEs. Any changes to this backup limit are applied only to VEs added to the Virtuozzo PowerPanel infrastructure after the change. If you need to change the default backup limit from the start, do so before deploying the compute nodes. For instructions, see *Setting the Backup Limit* on page 28.

2.2 Deploying Compute Nodes

To deploy compute nodes, that is, make your Virtuozzo Hybrid Server nodes manageable by the controller, do the following on the controller node:

1. Create a file, e.g. `nodes.lst`, with a list of Virtuozzo Hybrid Server nodes to be attached to the cluster. Specify each node, one per line, in the form `<user>:<password>@<node_hostname>`.

   Important: Hostnames are case-sensitive.

   For example:

   ```
   root:passwd1@vz_hostname1
   root:passwd2@vz_hostname2
   ```

   Use root credentials because deployment involves package installation, service setup, and other actions that require root privileges.

   Note: Keep this file if you plan to add more compute nodes later. For details, see *Adding Compute*

Nodes on page 6.

2. Set the required owner and file permissons to the nodes list file:

```
# chown root nodes.lst
# chmod 600 nodes.lst
```

3. Run the deploy script:

```
# vzapi-installer computes --nodes=nodes.lst
```

The `vzapi-installer` script will place SSH keys stored in `/var/lib/vzapi` on each node in the list and launch `ansible-playbook` to perform actual deployment.

Note the following:

- Once the compute component is deployed on a node, that node's name is added to the **computes** section of the file `/var/lib/vzapi-installer/inventory.json`.
- Virtuozzo does not have the firewall enabled by default. If, however, you have configured and started a firewall manually on a compute node, create rules to open the required ports on that compute node (see *Appendix A: Controller and Compute Ports* on page 31).

When deployment is completed successfully, you will see a recap from Ansible. For example:

```
PLAY RECAP *********************************************************************
compute1.example.com     : ok=15   changed=9    unreachable=0    failed=0
<...>
compute10.example.com     : ok=15   changed=8    unreachable=0    failed=0

Compute nodes have been deployed successfully!
```

If you later need to redeploy existing compute nodes listed in your `nodes.lst` file or update PowerPanel components on them, run

```
# vzapi-installer computes
```

2.2.1 Adding Compute Nodes

To add more compute nodes after installation, specify their information in `nodes.lst` in addition to compute nodes already deployed and run the deploy script:

```
# vzapi-installer computes --nodes=nodes.lst
```

Important: If invoked with the `--nodes=nodes.lst` option, the deploy script will only deploy compute nodes specified in `nodes.lst` and remove from Ansible configuration files all existing compute nodes that are not listed in this file. For information on how to remove a compute node from Virtuozzo PowerPanel completely, see *Removing Compute Nodes* on page 7.

2.2.2 Removing Compute Nodes

To remove a deployed compute node from Virtuozzo PowerPanel, do the following:

1. Stop and disable the `vzapi-compute` service on the compute node:

```
# systemctl stop vzapi-compute
# systemctl disable vzapi-compute
```

2. Remove the `vzapi-compute` package from the compute node:

```
# yum remove vzapi-compute
```

3. Remove information on the compute node and its VEs from the Virtuozzo PowerPanel database. Do the following on the controller node:

   3.1. Find out the ID of the node to delete by means of the `vzapi host list` command. For example:

```
# vzapi host list
    <...>
    {
        "created_at": "2017-03-06T12:18:27.000000",
        "hostname": "compute4.example.com",
        "id": "6ddbb2c1632a42ce9d2d8a83a7c93e04"
    },
    <...>
```

   3.2. Using the compute node ID, delete the information about the node and its VEs from the database. For example:

```
# vzapi host delete 6ddbb2c1632a42ce9d2d8a83a7c93e04
```

Note: If you remove host information from the database without stopping and removing the `vzapi-compute` service, said information will be restored in the database on next `vzapi-compute` restart.

2.3 Post-Installation Tasks

After installing Virtuozzo PowerPanel, the administrator needs to set up a user database on the controller node one of these ways:

- Integrate an LDAP database with Virtuozzo PowerPanel (see *Integrating Virtuozzo PowerPanel with a LDAP Database* on page 10).
- Create users locally on the controller node (see *Managing Users Locally* on page 13).

2.4 Upgrading Virtuozzo PowerPanel

Important: Make a complete backup of the controller before upgrading. If you have customized the configuration files `/etc/vzapi/vzapi.conf` and `/etc/keystone/keystone.conf`, copy them to a safe location and keep them at hand.

Note: Virtuozzo PowerPanel services may need to be stopped and restarted during upgrade, resulting in a certain downtime.

Typically, notifications are sent out about each Virtuozzo PowerPanel release, so you know when you can upgrade. In addition, you can manually check for updates at any time using `vzapi-installer check-upgrade --minimal` or `--full`. The latter option provides more ways to filter command's JSON output.

The output may look like this:

```
# vzapi-installer check-upgrade --minimal
{
  "ppcontroller.example.com": {
    "status": "updates-available",
    "updates": [
      "pp-ui-1.0.85-1.vl7"
    ]
  },
  "ppnode1.example.com": {
    "status": "updates-available",
    "updates": [
      "docker.io/virtuozzo/vzapi-compute:1.0.73"
```

```
    ]
  }
}
```

After confirming that an update is available, do the following:

1. Update the installer package:

   ```
   # yum update vzapi-installer
   ```

   YUM will back up the current configuration templates `/usr/share/vzapi-installer/roles/api/templates/vzapi.conf.j2` and `/usr/share/vzapi-installer/roles/keystone/templates/keystone.conf.j2` to `*.j2.rpmsave` and replace them with the updated ones.

2. If you have customized the `*.j2` templates, merge the customizations from the `*.j2.rpmsave` backups to the new templates.

   Note: For the upgrade, the identity driver must be set to SQL (`driver = sql`) in `/usr/share/vzapi-installer/roles/keystone/templates/keystone.conf.j2`.

   ---

3. Start the upgrade:

   ```
   # vzapi-installer upgrade <upgrade_package_URL>
   ```

   Where `<upgrade_package_URL>` is the URL of the `pp-release` package that installs a repository for the upgrade. The URL becomes available when an update is released. It is mentioned in *Deploying the Controller* on page 3 as well as the release notes.

   The controller and compute nodes will be redeployed. The current configuration files `/etc/vzapi/vzapi.conf` and `/etc/keystone/keystone.conf` will be deleted and created anew from the updated templates. If you have customized the `*.conf` files, merge the customizations from the previously saved copies to the new files.

Note: If compute nodes have issues connecting to the controller, try restarting their respective services with `systemctl restart vzapi-compute`.

CHAPTER 3

Setting Up User Database

3.1 Integrating Virtuozzo PowerPanel with a LDAP Database

This chapter describes how to integrate the Virtuozzo PowerPanel controller with an existing LDAP database (on the example of OpenLDAP).

Note: If your LDAP setup differs from this example, more configuration details are available here.

The following prerequisites need to be met prior to LDAP setup:

1. Virtuozzo PowerPanel must be deployed.
2. For remote LDAP, firewall must be disabled or necessary rules must be added.

To set up LDAP, do the following:

1. Decide on how to map LDAP attributes to Keystone user names and IDs. For example, assign names to the `sn` parameters, IDs to `cn` parameters.
2. In your LDAP database, create the `vzapi` and `admin` users.
   2.1. Find out the IDs and names of users `admin` and `vzapi` in the Keystone database on the controller. For example:

```
# openstack --os-cloud local user show admin
<...>
| id                  | 86921a8ec6a5497895ca07c5d6b738af |
```

```
<...>
# openstack --os-cloud local user show vzapi
<...>
| id                  | d8e4a93d60954c92b4239981c6c40707 |
<...>
```

2.2. Generate password hashes for the users `admin` and `vzapi`. Use controller admin's password for `admin`. Use the password in the `[keystone_authtoken]` section of `/etc/vzapi/vzapi.conf` for `vzapi`. For example:

```
# slappasswd
New password: <controller_admin_passwd>
Re-enter new password: <controller_admin_passwd>
{SSHA}E2qhe244kX8r+stFOb6mX2bfHYSpygTk
# slappasswd
New password: <vzapi_passwd>
Re-enter new password: <vzapi_passwd>
{SSHA}wBjzhGnmHl3hT9mZja9GLytOXBU4qHcS
```

2.3. Create the file `users.ldif` with the contents shown further. Specify the IDs of users `admin` and `vzapi` in the `cn` parameters, their password hashes in the `userPassword` parameters, and your domain name in the `dc` parameters. For example:

```
dn: cn=d8e4a93d60954c92b4239981c6c40707,ou=<PP_users_OU>,dc=ctrl,dc=example,dc=com
objectClass: person
cn: d8e4a93d60954c92b4239981c6c40707
sn: vzapi
userPassword: {SSHA}E2qhe244kX8r+stFOb6mX2bfHYSpygTk

dn: cn=86921a8ec6a5497895ca07c5d6b738af,ou=<PP_users_OU>,dc=ctrl,dc=example,dc=com
objectClass: person
cn: 86921a8ec6a5497895ca07c5d6b738af
sn: admin
userPassword: {SSHA}wBjzhGnmHl3hT9mZja9GLytOXBU4qHcS
```

Where `<PP_users_OU>` is the organizational unit with the list of users that need to be available in Virtuozzo PowerPanel.

2.4. Add the corresponding entry to the LDAP database:

```
# ldapadd -x -D cn=Manager,dc=ctrl,dc=example,dc=com -W -f users.ldif
Enter LDAP Password:
adding new entry "cn=d8e4a93d60954c92b4239981c6c40707,ou=<PP_users_OU>,dc=ctrl<...>"
adding new entry "cn=86921a8ec6a5497895ca07c5d6b738af,ou=<PP_users_OU>,dc=ctrl<...>"
```

3. On the controller node, edit `/etc/keystone/keystone.conf` according to your needs. You may need to do the following:

3.1. Specify LDAP server information. For example:

```
[ldap]
url = ldap://<ldap_server_address>
user = cn=Manager,dc=ctrl,dc=example,dc=com
password = <ldap_admin_password>
suffix = dc=ctrl,dc=example,dc=com
```

3.2. Specify the organizational units (OU) in the LDAP directory with information about users that will be managed in Virtuozzo PowerPanel. For example:

```
[ldap]
user_tree_dn = ou=<PP_users_OU>,dc=ctrl,dc=example,dc=com
user_objectclass = person
```

3.3. Switch to the LDAP identity driver:

```
[identity]
#driver = sql
driver = ldap
```

4. On the controller node, restart the Apache HTTP Server:

```
# systemctl restart httpd
```

5. On the controller node, create Keystone projects with names that match corresponding user names:

```
# vzapi user sync
```

This command needs to be run after creating or deleting users. It does not need to be run after editing user attributes or changing their passwords.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

---

6. On the controller node, make sure that Keystone sees LDAP users:

```
# openstack --os-cloud local user list
```

7. On the controller node, make sure that a project has been created for each user on step 5:

```
# openstack --os-cloud local project list
```

You should see a list of projects with names that match corresponding user names.

3.2 Managing Users Locally

If no identity provider is available to obtain user information from, an administrator with access to the controller's console can create users locally on the controller node. Created users appear in the PowerPanel web interface after creation and can be assigned virtual environments.

Note the following:

- You cannot create LDAP users with `vzapi user create`.
- If you switch to LDAP, existing local users will not be available in Virtuozzo PowerPanel anymore. They will, however, remain in the local identity database.
- Since the performance of specific actions in PowerPanel, such as password change in a virtual machine, etc., requires guest tools, install the guest tools in your virtual machines to ensure the smooth functioning of most PowerPanel operations.

3.2.1 Creating Users Locally

To create a local user, run the command `vzapi user create <username>` on the controller node. You will also be asked to create the user's password. For example:

```
# vzapi user create user1
Password:
Confirm:
```

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

To make sure that the user has been created, you can run the command `openstack --os-cloud local user list`. For example:

```
# openstack --os-cloud local user show user1
+---------------------+----------------------------------+
| Field               | Value                            |
+---------------------+----------------------------------+
| domain_id           | default                          |
| id                  | 315a8d0ee4d84474bd8c8c0574e9e987 |
| name                | user1                            |
| password_expires_at | None                             |
```

```
+--------------------+----------------------------------+
```

3.2.2 Changing Passwords of Local Users

To change the password of a user created locally, run the command `openstack --os-cloud=local user set <username> --password <new_passwd>`. For example:

```
# openstack --os-cloud=local user set user1 --password newpassword
```

3.2.3 Deleting Local Users

To delete a user created locally, run the command `vzapi user delete <username>`.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

For example:

```
# vzapi user delete user1
```

CHAPTER 4

Performing Administration Tasks

The main purpose of the admin mode is to provide an easy way for an adminstrator to assign virtual environments (VEs) hosted on compute nodes to end users from an identity provider, e.g., LDAP database. If such an identity provider is not available, users can be created locally on the controller node (see *Managing Users Locally* on page 13).

After having been assigned a VE, an end user can log in to Virtuozzo PowerPanel with the username and password set for the VE and manage it in the user mode.

4.1 Logging In

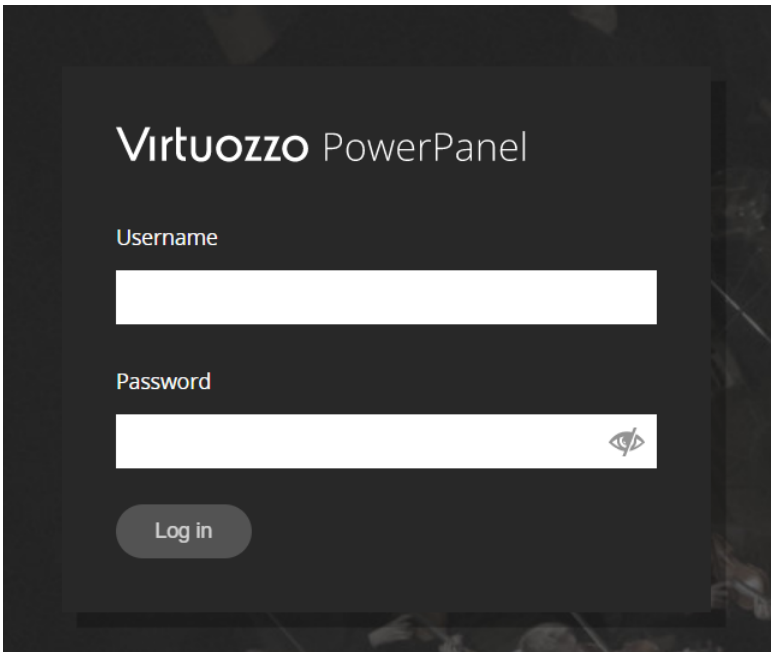
To log in to the Virtuozzo PowerPanel web interface, visit controller node's hostname or IP address in a supported web browser. The latest versions of all popular web browsers are supported: Chrome, Firefox, Edge, Opera, Safari.

Regular users can also log in to manage single VEs via an old-style welcome screen accessible at `https://<controller_hostname>/login/ve`. For details, see the Virtuozzo PowerPanel User's Guide.

If you deployed the controller with a self-signed certificate, accept it when prompted and add it to browser's exceptions.

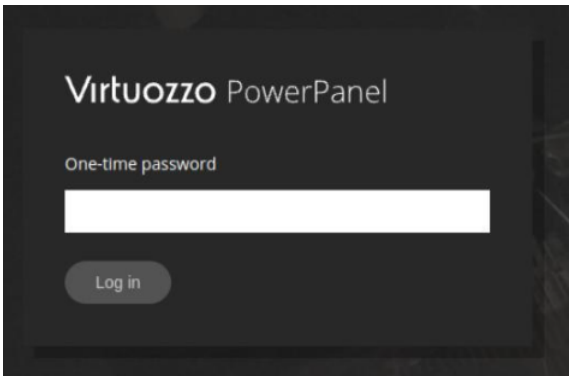
Note: Some browsers, e.g., Firefox, require exceptions to be added in the form of `<controller_address>:<port>` for ports 6556, 6557, and 35357.

You will be taken to the welcome screen where you will need to enter the username `admin` and the password created during controller setup (see *Deploying the Controller* on page 3).



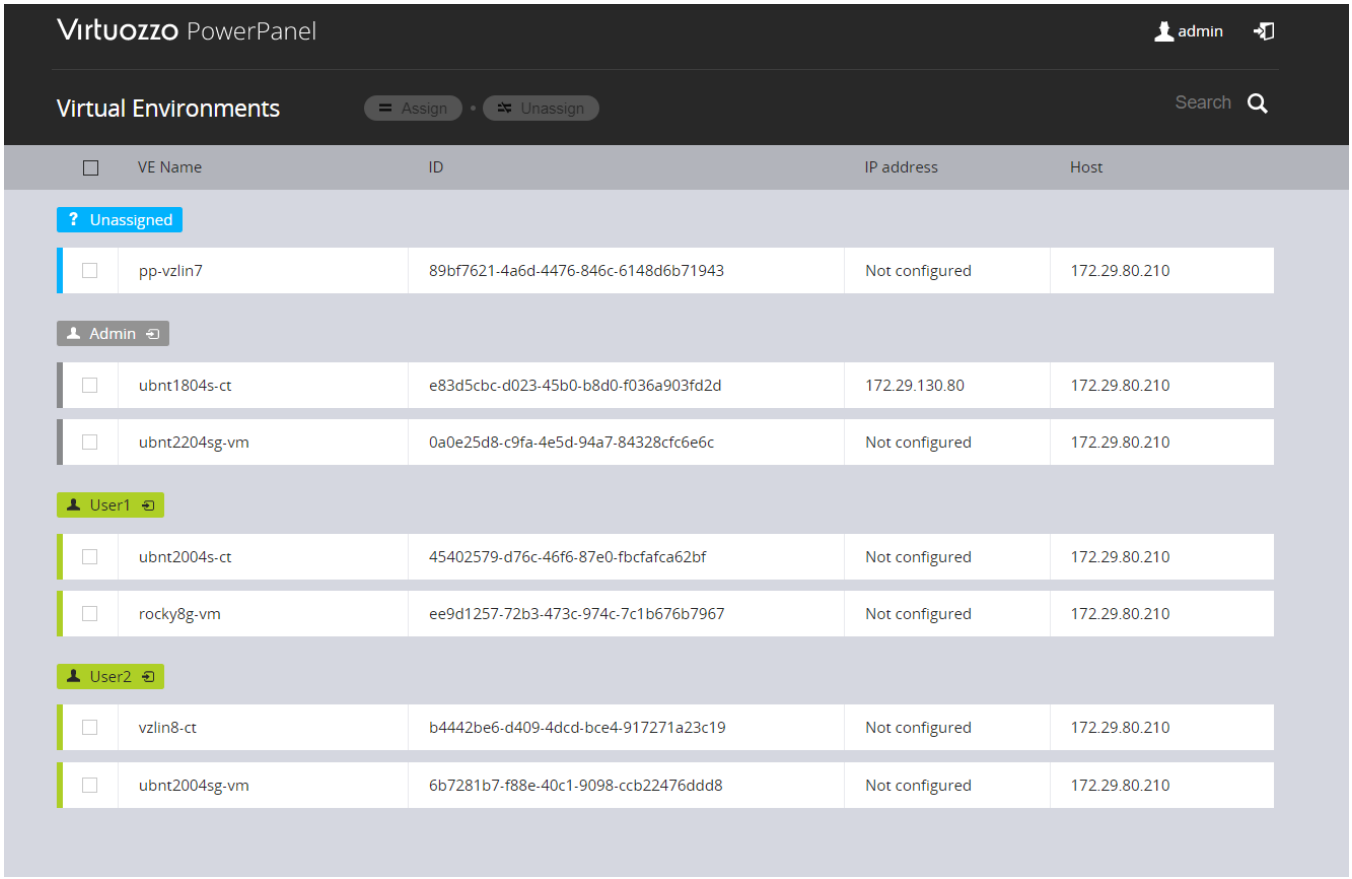
If multi-factor authentication (MFA) is enabled, you will need to scan the QR code and enter a one-time password to activate MFA. If you click **Cancel**, MFA will remain enabled but not activated. In that case, you will log in with the user name and password as before, but will see the QR code every time until MFA is activated.

If MFA has been activated, you will need to enter a one-time password.



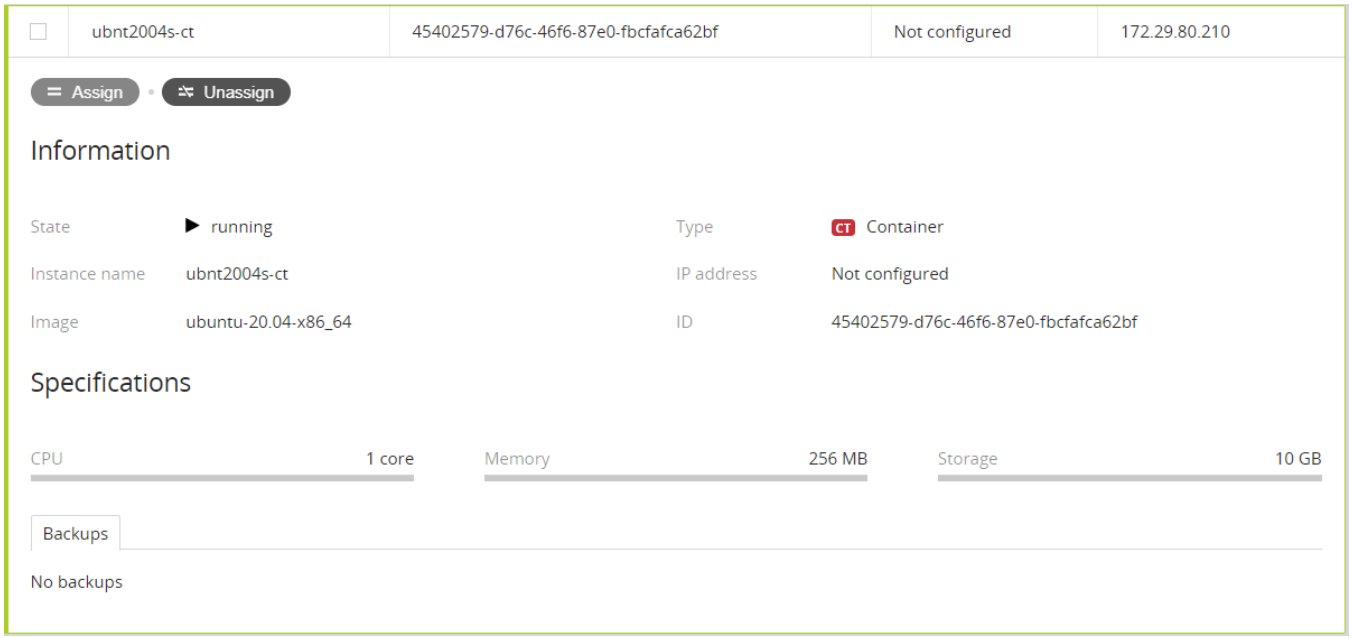
After logging in, you will be taken to the main screen of Virtuozzo PowerPanel.

Note: At any time, you can click the Virtuozzo PowerPanel logo in the top left corner to return to the main screen.



On the main screen, you can see the list of virtual environments hosted on all compute nodes. The VEs in the list are divided into groups based on what user they are assigned to, if any: **Unassigned**, **Admin**, and **<username>**.

Each VE area in the list expands on click, showing more details about the VE, including state, type, UUID, name, IP address (the one assigned with `prlctl`, not the one obtained from inside the guest OS), virtual hardware configuration, backups, and logs. In the VE area, you can also see the buttons for tasks available for this specific VE.



In the admin mode, the administrator can perform the following tasks:

- Assign VEs to users and unassign VEs from users, see *Managing Virtual Environment Assignments* on page 19.
- Impersonate users to perform user-specific VE management tasks, see *Impersonating Users* on page 21.

These tasks are described in more detail in the following sections.

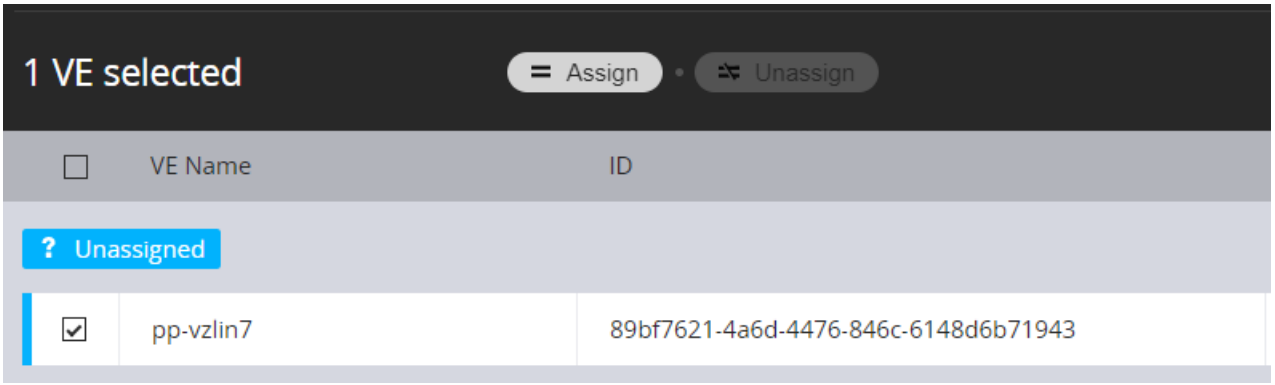
4.2 Managing Virtual Environment Assignments

Virtual environments listed in the **Unassigned** group can be assigned to users available to Virtuozzo PowerPanel.

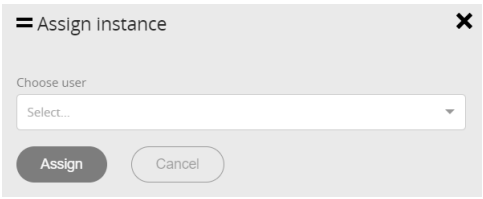
Note: Users that have no VEs assigned are not shown in the list on the main screen.

To assign one or more VEs to a user:

1. Check the boxes of the required VEs and click **Assign** above the VE list.



2. Select a user in the opened drop-down list and click **OK**.



To assign a single VE to a user, you can also click the VE area to expand it, click **Assign** in the VE area, select a user in the opened drop-down list, and click **OK**.

Unassigning VEs from users is done in a similar way by means of the **Unassign** button.

4.2.1 Assigning Virtual Environments to Users from Command Line

You can also assign virtual environments to users from command line as follows:

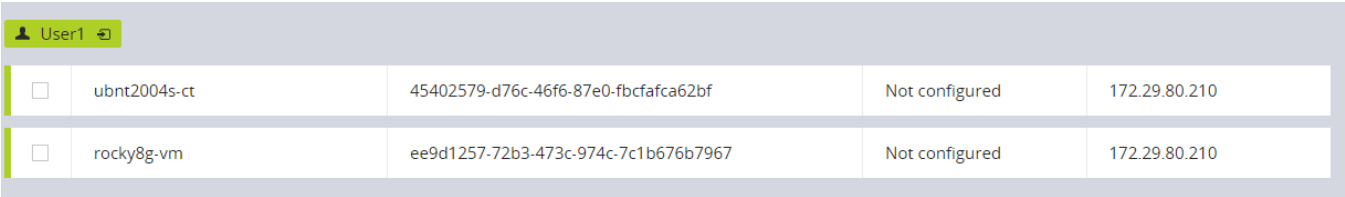
```
# vzapi instance update --new-user <username> --new-project <username> <VE_UUID>
```

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

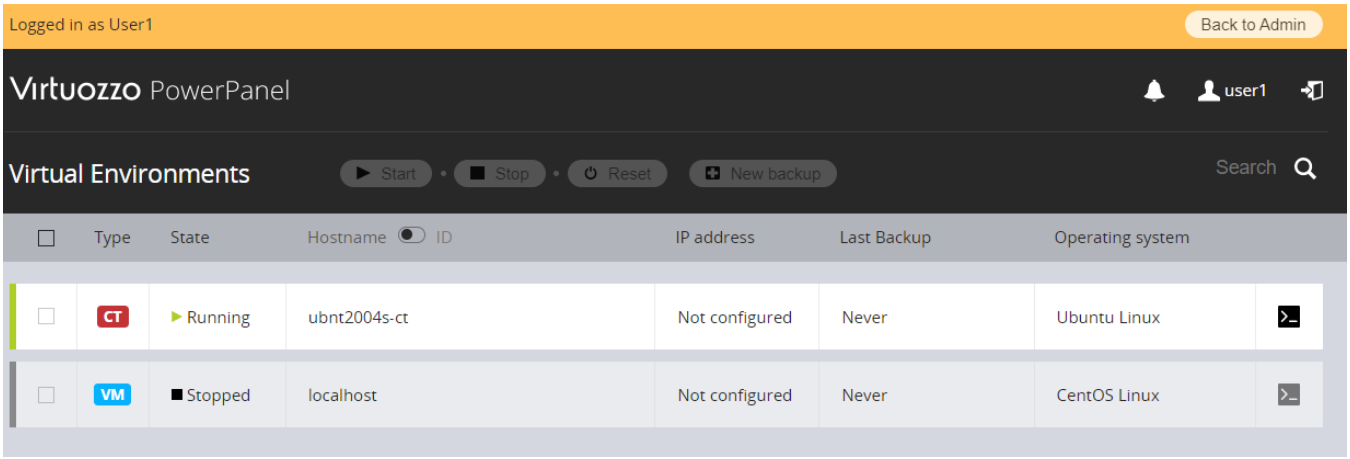
This can be useful if you need to automate the assignment process, for example.

4.3 Impersonating Users

By clicking a user name on the main screen (**Admin** included), the administrator can switch to the user mode, as if logged in to Virtuozzo PowerPanel under that user account, and manage that user's virtual environments.



An example of impersonation is shown below. Click **Back to Admin** in the top right corner to switch to the admin mode.



Impersonating the administrator differs from impersonating regular users, however, as the administrator still sees (and can manage) all VEs, not just those assigned to the user **Admin**.

Note: For details on tasks that can be performed in the user mode, see the Virtuozzo PowerPanel User's Guide.

4.4 Managing Settings

Administrators can create and manage settings, in particular, to toggle actions available to users, both in the web panel and the API.

Settings can be created and managed with the `vzapi setting` commands. Settings cannot be duplicated, however. Trying to create a setting that already exists will result in the HTTP status code 409 (conflict). You can either update the setting or delete it and create it anew.

4.4.1 Listing Settings

To list settings, use the `vzapi setting list` command.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

For example:

```
# vzapi setting list
[
    {
        "applies_to": {
            "instance_type": "all",
            "module_type": "all"
        },
        "created_at": "2021-03-11T14:35:58.000000",
        "param": "instance.action.change_password",
        "updated_at": null,
        "value": "0"
    },
    {
        "applies_to": {
            "instance_type": "ct",
            "module_type": "ui"
        },
        "created_at": "2021-03-11T14:36:30.000000",
        "param": "instance.action.reinstall",
        "updated_at": null,
        "value": "0"
    }
]
```

4.4.2 Creating Settings

To create a setting, specify a user action to toggle and the value of which `0` disables and `1` enables the setting. You can also narrow the setting down to affect only VMs or containers and only the web panel or API.

For a list of the currently supported user actions and other parameters, see Creating Settings.

To create settings, use the `vzapi setting create` command.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

For example, to completely disable the change password action, run

```
# vzapi setting create --param instance.action.change_password --value 0
{
 "applies_to": {
     "instance_type": "all",
     "module_type": "all"
 },
 "created_at": "2021-03-11T14:35:58.993419",
 "param": "instance.action.change_password",
 "updated_at": null,
 "value": "0"
}
```

To disable the reinstall action only in the web panel (i.e. hide the button), run

```
# vzapi setting create --param instance.action.reinstall --module-type ui --value 0
{
 "applies_to": {
     "instance_type": "all",
     "module_type": "ui"
 },
 "created_at": "2021-03-11T14:36:30.086787",
 "param": "instance.action.reinstall",
 "updated_at": null,
 "value": "0"
}
```

The action is only available for containers, so even though `instance_type` is set to `all` by default, only containers will be affected.

4.4.3 Showing Setting Details

To show the details of a setting, use the `vzapi setting show` command.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more

details, see *Managing Multi-Factor Authentication* on page 26.

For example:

```
# vzapi setting show instance.action.reinstall
{
    "applies_to": {
        "instance_type": "ct",
        "module_type": "ui"
    },
    "created_at": "2021-03-11T14:36:30.000000",
    "param": "instance.action.reinstall",
    "updated_at": null,
    "value": "0"
}
```

4.4.4 Updating Settings

To update a setting, use the `vzapi setting update` command.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

For example:

```
# vzapi setting update instance.action.reinstall --module-type api
{
    "applies_to": {
        "instance_type": "all",
        "module_type": "api"
    },
    "created_at": "2021-03-11T14:36:30.000000",
    "param": "instance.action.reinstall",
    "updated_at": null,
    "value": "0"
}
```

The next time you request this setting's details, the `updated_at` field will contain the timestamp of the update.

4.4.5 Deleting Settings

To delete a setting, use the `vzapi setting delete` command.

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

For example:

```
# vzapi setting delete instance.action.reinstall
```

4.5 Configuring Protection Against Brute-Force Attacks

To protect the system against brute-force (or password guessing) attacks, Virtuozzo PowerPanel can use device cookies as an extra authenticator for user devices. This allows the system to distinguish between trusted and untrusted clients and temporarily lock the latter out.

Protection is enabled by default.

To configure protection behavior, edit the following parameters in the `[device_cookie]` section of the `/etc/keystone/keystone.conf` configuration file:

`lockout_enable` **(default: boolean** `True`**)**
: Enables and disables protection. Set to boolean `False` to disable protection.

`lockout_failure_attempts` **(default: 5)**
: The maximum number of authentication attempts that a user can fail before the device cookie or untrusted client is locked for the number of seconds specified in `lockout_duration`.

`lockout_duration` **(default: 600)**
: The number of seconds a device cookie or an untrusted client is locked for after failing as many authentication attempts as specified in `lockout_failure_attempts`.

`jws_key_repository` **(default:** `/etc/keystone/device-cookie-keys/`**)**
: The directory with the public and private keys for validating JSON web signatures (JWS). Must be readable by Keystone's server process.

Restart the Apache HTTP Server on the controller to apply changes:

```
# systemctl restart httpd
```

Note: As blocked users are stored in the memory cache, restarting Keystone gives them an additional attempt to guess the password before lockout.

4.6 Managing Multi-Factor Authentication

Multi-factor authentication (MFA) adds a layer of security by additionally requiring timed one-time passwords (TOTP) generated by Google Authenticator.

Note: MFA only works with a local authentication database.

MFA is primarily intended to protect administrative accounts, but you can enable it for regular users as well. It is disabled by default.

To manage MFA, use the `vzapi --os-cloud local-credential mfa *` commands. In particular:

- To enable MFA for a specific user, e.g., admin, run `vzapi --os-cloud local-credential mfa enable <username>`. For example:

  ```
  # vzapi --os-cloud local-credential mfa enable admin
  {
      "activated": false,
      "blob": "MFWWO4TGMRYHA23RNJXWMZLXNY",
      "id": "a09896fe33f94490953602394a1bc59c",
      "name": "admin",
      "type": "totp",
      "user_id": "6fcd14baaa4b47f1a9de372037c2b68a"
  }
  The MFA credentials have been successfully enabled for the user admin
  ```

  On the next login, after entering the login and password, the user will be asked to scan the provided QR code with Google Authenticator and enter a one-time password. If the user does that, MFA becomes activated. The next time, the user will need to enter a one-time password without having to scan the QR code. If the user clicks **Cancel**, MFA remains enabled but not activated. In that case, the user will be able to log in with the user name and password as before, but will see the QR code every time until MFA

is activated.

The `blob` value in the output is the setup key the user will need to enter manually if they are unable to scan the QR code.

The `activated` value indicates whether the user has scanned the QR code, paired the mobile device, and thus activated MFA for their account.

Note: After activating MFA for the admin, use `vzapi --os-cloud local-credential <cmd>` instead of `vzapi --os-cloud local <cmd>` in commands run by the admin. The reason is that activating MFA forbids using plain password authentication and thus makes it impossible for the admin to use the `vzapi` tool the old way.

- To show users for which MFA is enabled, run `vzapi --os-cloud local-credential mfa list`. For example:

```
# vzapi --os-cloud local-credential mfa list
[
    {
        "activated": false,
        "blob": "MFWWO4TGMRYHA23RNJXWMZLXNY",
        "id": "a09896fe33f94490953602394a1bc59c",
        "name": "admin",
        "type": "totp",
        "user_id": "6fcd14baaa4b47f1a9de372037c2b68a"
    },
    {
        "activated": true,
        "blob": "NRWHE5D2NN4XO3DPOR3WK5DPPA",
        "id": "fdddab7c9b694b25b488e558840ae599",
        "name": "user1",
        "type": "totp",
        "user_id": "73b1aab1dcf546aea2b458fb24400bc5"
    }
]
```

- To disable MFA for a specific user, run `vzapi --os-cloud local-credential mfa disable <username>`. For example:

```
# vzapi --os-cloud local-credential mfa disable admin
The MFA credentials have been successfully disabled for the user admin
```

- To reset, that is disable and immediately enable, MFA for a specific user, run `vzapi --os-cloud local-credential mfa reset <username>`. For example:

```
# vzapi --os-cloud local-credential mfa reset admin
The MFA credentials have been successfully disabled for the user admin
{
    "activated": false,
    "blob": "MFWWO4TGMRYHA23RNJXWMZLXNY",
    "id": "a09896fe33f94490953602394a1bc59c",
    "name": "admin",
    "type": "totp",
    "user_id": "6fcd14baaa4b47f1a9de372037c2b68a"
}
The MFA credentials have been successfully enabled for the user admin
```

After that, the user will have to scan the QR code and activate MFA anew.

4.7 Miscellaneous Tasks

4.7.1 Setting the Backup Limit

By default, each user can create up to 3 backups of each of their virtual environments. To change the default backup amount for both new and existing VEs, do the following on the controller:

1. Set the `backup_default_max` parameter in the `[DEFAULT]` section of the configuration template `/usr/share/vzapi-installer/roles/api/templates/vzapi.conf.j2`. For example:

   ```
   [DEFAULT]
   <...>
   backup_default_max = 5
   <...>
   ```

   This will apply the change to new VEs on compute nodes that will be added to Virtuozzo PowerPanel in the future.

2. Run `vzapi-installer computes` to redeploy existing compute nodes and apply the change to new VEs that will be created on them. For more details on this command, see *Deploying Compute Nodes* on page 5.

3. Apply the change to existing VEs. For example:

   ```
   # mysql vzapi --execute="UPDATE instances SET backup_limit=5;"
   ```

Note: The backup directory (`/vz/vmprivate/backups` by default) can be changed with the `prlsrvctl set`

`--backup-path <path>` command.

To set the default backup limit for an individual VE with the ID `<VE_UUID>`, use the following command:

```
# vzapi instance update --backup-limit 5 <VE_UUID>
```

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

4.7.2 Setting the Default Backup Mode

By default, a full backup is created first and then only incremental backups are created (see `full_and_incremental` further). To change the default backup mode for both new and existing VEs, do the following on the controller:

1. Set the `backup_default_mode` parameter in the `[DEFAULT]` section of the configuration template `/usr/share/vzapi-installer/roles/api/templates/vzapi.conf.j2`.

   The following modes are available:

   - `always_full`, always create full backups.
   - `always_incremental`, create a full backup first, then only create incremental backups unless the full or the last incremental backup is deleted. If a mid-chain incremental backup is deleted, it is merged with the next one in the chain, increasing its size accordingly.
   - `full_and_incremental`, create a full backup first, then only create incremental backups unless the full backup is deleted. If a mid-chain incremental backup is deleted, the subsequent backups are deleted as well, and the next backup is a full one again.

   For example:

   ```
   [DEFAULT]
   <...>
   backup_default_mode = always_full
   <...>
   ```

   This will apply the change to new VEs on compute nodes that will be added to Virtuozzo PowerPanel in the future.

2. Run `vzapi-installer computes` to redeploy existing compute nodes and apply the change to new VEs that will be created on them. For more details on this command, see *Deploying Compute Nodes* on page 5.
3. Apply the change to existing VEs. For example:

```
# mysql vzapi --execute="UPDATE instances SET backup_mode='always_full';"
```

To set the default backup mode for an individual VE with the ID `<VE_UUID>`, use the following command:

```
# vzapi instance update --backup-mode always_full <VE_UUID>
```

Note: If MFA is enabled, replace `vzapi <cmd>` with `vzapi --os-cloud local-credential <cmd>`. For more details, see *Managing Multi-Factor Authentication* on page 26.

4.7.3 Setting the Idle Timeout

By default, users are automatically logged out of Virtuozzo PowerPanel after 24 hours of inactivity. You can change this default value by setting the `expiration` parameter (in seconds) in the `[token]` section in the file `/etc/keystone/keystone.conf`. For example:

```
[token]
<...>
expiration = 43200
<...>
```

Restart the Apache HTTP Server to apply the change:

```
# systemctl restart httpd
```

CHAPTER 5

Appendices

5.1 Appendix A: Controller and Compute Ports

This section lists the TCP ports that need to be open for Virtuozzo PowerPanel to operate properly.

5.1.1 Controller Ports

The following ports need to be open on the controller node:

- 80, 443 for public access
- 3306, 5671 for communication with compute nodes

If you have a firewall enabled on the controller, the deploy script will create rules to open the required ports automatically.

TCP port	Traffic	Description
5671	Incoming, outgoing	Used by the RabbitMQ message broker to exchange messages between PowerPanel components.
3306	Incoming, outgoing	Used to connect to the internal database.
443	Incoming, outgoing	Used to access the web panel itself, send problem reports to the support team.
80	Incoming, outgoing	Used for HTTP connections, e.g., to download Virtuozzo updates from remote repositories.

Continued on next page

Table 5.1.1.1 -- continued from previous page

TCP port	Traffic	Description
35357	Incoming, outgoing	Used to connect to the internal keystone from external applications for integration. Closed by default but can be opened manually via a firewall rule.

5.1.2 Compute Ports

Compute nodes do not have a firewall enabled by default. If you need to start a firewall service on a compute node, open all ports listed in the table to let it communicate with the controller node. For example, if you use `firewalld`:

```
# firewall-cmd --permanent --zone=public \
--add-port=5671/tcp \
--add-port=3306/tcp
```

TCP port	Traffic	Description
5671	Incoming, outgoing	Used by the RabbitMQ message broker to exchange messages between PowerPanel components.
3306	Incoming, outgoing	Used to connect to the internal database.