

**NAME**

remote-viewer – a simple remote desktop client

**SYNOPSIS**

**remote-viewer** [OPTIONS] — [URI]

**DESCRIPTION**

**remote-viewer** is a simple remote display client. The supported protocols are SPICE and VNC.

Starting remote-viewer without URI will open a simple dialog with an entry and a list of previously successfully accessed URI.

The URI can also point to a connection settings file, see the CONNECTION FILE section for a description of the format.

If URI is '-', then remote-viewer will read the standard input as a connection settings file and attempt to connect using it.

**OPTIONS**

The following options are accepted when running remote-viewer:

-h, --help

Display command line help summary

-V, --version

Display program version number

-v, --verbose

Display information about the connection

-z PCT, --zoom=PCT

Zoom level of the display window in percentage. Range 10–400.

-f, --full-screen

Start with the windows maximized to fullscreen.

If supported, the remote display will be reconfigured to match the physical client monitor configuration, by enabling or disabling extra monitors as necessary. This is currently implemented by the Spice backend only.

To specify which client monitors are used in fullscreen mode, see the CONFIGURATION section below.

-t TITLE, --title TITLE

Set the window title to **TITLE**

--debug

Print debugging information

-H HOTKEYS, --hotkeys HOTKEYS

Set global hotkey bindings. By default, keyboard shortcuts only work when the guest display widget does not have focus. Any actions specified in **HOTKEYS** will be effective even when the guest display widget has input focus. The format for **HOTKEYS** is <action1>=<key1>[+<key2>][,<action2>=<key3>[+<key4>]]. Key-names are case-insensitive. Valid actions are: toggle-fullscreen, release-cursor, secure-attention, smartcard-insert and smartcard-remove. The secure-attention action sends a secure attention sequence (Ctrl+Alt+Del) to the guest. Examples:

```
--hotkeys=toggle-fullscreen=shift+f11,release-cursor=shift+f12
```

```
--hotkeys=release-cursor=ctrl+alt
```

Note that hotkeys for which no binding is given are disabled. Although the hotkeys specified here are handled by the client, it is still possible to send these key combinations to the guest via a menu item.

`-k, --kiosk`

Start in kiosk mode. In this mode, the application will start in fullscreen with minimal UI. It will prevent the user from quitting or performing any interaction outside of usage of the remote desktop session.

Note that it can't offer a complete secure solution by itself. Your kiosk system must have additional configuration and security settings to lock down the OS. In particular, you must configure or disable the window manager, limit the session capabilities, use some restart/watchdog mechanism, disable VT switching etc.

`--kiosk-quit <never|on-disconnect>`

By default, when kiosk mode is enabled, `virt-viewer` will remain open when the connection to the remote server is terminated. By setting `kiosk-quit` option to "on-disconnect" value, `virt-viewer` will quit instead.

## HOTKEY

A key binding combination is described by a series of key strings separated by '+' that must be pressed together in order to activate the associated action.

It must be composed of modifiers (shift, ctrl or alt) and a non-modifier key. For example, "shift+f11".

## CONNECTION FILE

**remote-viewer** connection file is of INI file format, with a mandatory `[virt-viewer]` group and "type" key.

### Example

Opening a file with the following content will start `remote-viewer` in fullscreen and connect to the host "betsiboka" using the SPICE protocol:

```
[virt-viewer]
type=spice
host=betsiboka
port=5900
fullscreen=1
```

### Key list

`version` (string)

If `remote-viewer` version number isn't greater or equal to the required version, an error is raised with the expected version.

The version format accepted is a list of integers separated by '.'. It can be followed by a dash '-' and an additional build number with the same format.

Version comparison is done by comparing each integer from the list one by one. If any of the component is not a number, the version comparison will fail and consider that the 2 versions are considered to be the same.

`versions` (osid:version list)

This is a list of `osid:version` couples separated by ';'. `osid` is an arbitrary string, `version` is a version number in the same format as in the 'version' field. A given couple indicates that `remote-viewer` builds matching the given 'osid' (fedora22, debian7, ...) must be at least version 'version'. For consistency, it's recommended to use `libosinfo` OS shortids as the `osid`.

`newer-version-url` (string)

If specified, this field is an URL which will be displayed to the user when a version check fails.

`type` (string, mandatory)

The session type, either "spice", "vnc" or "ovirt".

`host` (string, mandatory)

The server host to connect to.

- `port` (integer)  
The server port to connect to.
- `tls-port` (integer)  
The server TLS/SSL port to connect to.
- `username` (string)  
The username for the session authentication.
- `password` (string)  
The password for the session authentication.
- `disable-channels` (string list)  
The list of session channels to disable.  
  
The current SPICE channels are: main, display, inputs, cursor, playback, record, smartcard, usbredir.
- `tls-ciphers` (string)  
Set the cipher list to use for the secure connection, in textual OpenSSL cipher list format. (see **ciphers**(1))
- `title` (string)  
String to present in the window title.
- `fullscreen` (boolean)  
Opens the client windows in fullscreen.
- `ca` (string)  
CA certificate in PEM format (using “\n” to separate the lines). This will be used to verify the SSL certificate used for SPICE TLS sessions.
- `host-subject` (string)  
Verify the certificate subject matches with the given subject.
- `toggle-fullscreen` (hotkey string)  
Key binding for entering and leaving fullscreen mode. (see **HOTKEY** for description of expected string)
- `release-cursor` (hotkey string)  
Key binding for releasing cursor grab. (see **HOTKEY** for description of expected string)
- `smartcard-insert` (hotkey string)  
Key binding for inserting emulated smartcard. (see **HOTKEY** for description of expected string)
- `smartcard-remove` (hotkey string)  
Key binding for removing emulated smartcard. (see **HOTKEY** for description of expected string)
- `color-depth` (integer)  
Set the color depth of the guest display (16 or 32).
- `disable-effects` (string list)  
A list of desktop effects to disable in the remote guest.  
  
The effects that can be disabled with SPICE are: wallpaper, font-smooth, animation or all.
- `enable-smartcard` (boolean)  
Set to 1 to enable client smartcard redirection.
- `enable-usbredir` (boolean)  
Set to 1 to enable client USB device redirection.
- `enable-usb-autoshare` (boolean)  
Set to 1 to enable client USB devices auto-sharing.
- `usb-filter` (string)  
Set a string specifying a filter to use to determine which USB devices to autoconnect when plugged in, a filter consists of one or more rules. Where each rule has the form of:

`class, vendor, product, version, allow`

Use `-1` for class/vendor/product/version to accept any value.

And the rules themselves are concatenated like this:

`rule1 | rule2 | rule3`

`secure-channels` (string list)

The list of session channels to secure.

The current SPICE channels are: main, display, inputs, cursor, playback, record, smartcard, usbredir.

`delete-this-file` (boolean)

Set to 1 for the client to remove this connection file (if it can't, it will fail silently)

`proxy` (string)

A proxy URL to tunnel the connection through.

At the time of writing this documentation, the only supported proxy method with Spice is HTTP CONNECT.

For example, to tunnel connection through foobar host HTTP proxy on port 8080, use the value `"http://foobar:8080"`.

### oVirt Support

The connection file can also carry some oVirt-specific options when oVirt support is compiled in. These options are used to interact with oVirt REST API. This is currently only used in order to show a menu allowing to change the CD image being used by the virtual machine from remote-viewer user interface. These options go in an optional [ovirt] group.

`host` (string, mandatory)

The oVirt instance to connect to. This corresponds to the hostname one would connect to access the oVirt user or admin portal.

`vm-guid` (string, mandatory)

GUID of the oVirt virtual machine to connect to.

`jsessionid` (string)

Value to set the 'jsessionid' cookie to. With oVirt 3.6, setting this authentication cookie to a valid value will allow to interact with the oVirt REST API without being asked for credentials.

`sso-token` (string)

Value to set the 'Authorization' header to. With oVirt 4.0 or newer, setting this authentication header to a valid value will allow to interact with the oVirt REST API without being asked for credentials.

`ca` (string)

CA certificate in PEM format (using `"\n"` to separate the lines). This will be used to validate the certificate used for the oVirt REST https session remote-viewer will establish.

## CONFIGURATION

A small number of configuration options can be controlled by editing the settings file located in the user configuration directory:

`<USER-CONFIG-DIR>/virt-viewer/settings`

This file is a text file in INI format, with application options in the [virt-viewer] group and per-guest options in a group identified by the guest's UUID. The application options should not be edited manually. There is also a special [fallback] group which specifies options for all guests that don't have an explicit group.

For each guest, the initial fullscreen monitor configuration can be specified by using the **monitor-mapping** key. This configuration only takes effect when the `-f/--full-screen` option is specified.

The value of this key is a list of mappings between a guest display and a client monitor. Each mapping is separated by a semicolon character, and the mappings have the format

<GUEST-DISPLAY-ID>:<CLIENT-MONITOR-ID>.

For example, to map guest displays 1 and 2 to client monitors 2 and 3 for the guest with a UUID of e4591275-d9d3-4a44-a18b-ef2fbc8ac3e2, use:

```
[e4591275-d9d3-4a44-a18b-ef2fbc8ac3e2]
monitor-mapping=1:2;2:3
```

The monitor-mapping must contain ids of all displays from 1 to the last desired display id, e.g. “monitor-mapping=3:3” is invalid because mappings for displays 1 and 2 are not specified.

## EXAMPLES

To connect to SPICE server on host “makai” with port 5900

```
remote-viewer spice://makai:5900
```

To connect to VNC server on host “tsingy” with port 5900

```
remote-viewer vnc://tsingy:5900
```

To connect to a virtual machine named “toliara” on an oVirt server at example.org

```
remote-viewer ovirt://[username@]example.org/toliara
```

## BUGS

Report bugs to the mailing list  
<http://www.redhat.com/mailman/listinfo/virt-tools-list>

## COPYRIGHT

Copyright (C) 2012–2014 Red Hat, Inc., and various contributors. This is free software. You may redistribute copies of it under the terms of the GNU General Public License <https://www.gnu.org/licenses/gpl-2.0.html>. There is NO WARRANTY, to the extent permitted by law.

## SEE ALSO

virt-viewer(1), spice-client(1), the project website <http://virt-manager.org>