

# ViewStation License Manager

Best Practice



The software programs described in this document and the information contained in this document are confidential and proprietary products of KISTERS or its licensors. KISTERS waives copyright for licensed software users to print out parts of the documentation in hard copy for their own use only. This documentation may not be transferred, disclosed, or otherwise provided to third parties. In duplicating any part of this document, the recipient agrees to make every reasonable effort to prevent the unauthorized use and distribution of the proprietary information.

No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

KISTERS reserves the right to make changes in specifications and other information contained in this publication without prior notice.

KISTERS makes no warranty of any kind with regard to this material including, but not limited to, the implied warranties or merchantability and fitness for a particular purpose.

KISTERS shall not be liable for any incidental, indirect, special or consequential damages whatsoever (including but not limited to lost profits) arising out of or related to this documentation, the information contained in it or from the use of programs and source code that may accompany it, even if KISTERS has been advised of the possibility of such damages.

Any errors found in any KISTERS product should be reported to KISTERS where every effort will be made to quickly resolve the problem.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

Copyright 2024 KISTERS  
Internet: [viewer.kisters.de](http://viewer.kisters.de)  
E-mail: [support-viewer@kisters.de](mailto:support-viewer@kisters.de)  
Phone: +49 (0) 2408 9385-360

Author: KISTERS  
Date of print of current edition: 16/05/2024  
Current software version: 2022.x

The logo for KISTERS, featuring a stylized 'K' symbol followed by the word 'KISTERS' in a bold, sans-serif font.

# Table of Contents

Part I	License Server Installation	4
Part II	Server Configuration	6
Part III	Privilege Rule Configuration	10
Part IV	License Monitor	16
Part V	Troubleshooting	18



Chapter I:

# License Server Installation

# 1 License Server Installation

To install the VS License Manager, follow these steps:

- **Download** the VS License Manager ZIP archive.
- Move the contents of the ZIP archive to a directory of your choice, for example, `C:\Kisters\VSLM`.
- Copy the `VSLM.lic` license file you received by e-mail to the directory of your choice next to `VSLM.exe`.
- To run the VS License Manager, double-click `VSLM.exe` with administrator privileges. This opens a console with a log of all VS License Manager events. Closing the console stops VS License Manager.
- If you see error messages that mention ports, open the required local ports in the Windows Firewall. The default values are 6200 and 6280. The protocol is TCP. If you are using a third-party firewall, you can ignore these messages. Check the ports accordingly.
- Configure the VS License Manager as needed, see [Server Configuration](#). For example, set an admin password for the License Monitor.
- Remove the underscore from `_ServerConfig.json` to activate the configuration file.
- Restart the VS License Manager for the new configuration to take effect.

## Run as Windows Service

- To install and run the VS License Manager as a Windows service, run `windows_service_install.bat` with administrator privileges. This will configure the service to start automatically after a system restart.
- After updating `VSLM.lic` or `ServerConfig.json`, run `windows_service_restart.bat` with administrator privileges to apply the changes to the VS License Manager service.
- To remove the Windows service, run `windows_service_uninstall.bat` with administrative privileges.

**Note:** If you are using the VS License Manager with a 3DViewStation WebViewer on the same host system, you must add it as a dependency to the 3DViewStation WebViewer Windows service. This will ensure a correct startup sequence when the system is restarted.



Chapter II:

# Server Configuration



## 2 Server Configuration

The behavior of the VS License Manager can be configured using a json file. In addition to general server settings, usage privileges and access permissions to features can also be regulated. To change the configuration, simply edit the `ServerConfig.json` file using a text editor as described below. Restart the VS License Manager for the changes to take effect.


**Note:** If special characters are used in file paths or usernames, the encoding of the JSON file must be UTF-8.

### ► General Settings

#### Ports

The communication between the VS License Manager and the 3DViewStation clients uses the TCP network protocol. The `TCP_PORT` setting defines the port used:

```
"TCP_PORT": 6200,
```

The [License Monitor](#) , the web interface of the VS License Manager, can be accessed via a web browser: `<License Server Address>:<HTTP_PORT>`. The `HTTP_PORT` setting determines the HTTP port used:

```
"HTTP_PORT": 6280,
```

#### Note:

- The specified local ports must be opened in the server's firewall for the VS License Manager to function properly. The required protocol is TCP for both ports.
- If you are using ports other than the default, please let us know so we can keep your license up to date. To do so, send an email to [orders-viewer@kisters.de](mailto:orders-viewer@kisters.de) with your server name, the TCP port you are using, and the serial number of your license.

#### Passwords

To access the Admin view of the License Monitor (including logs, configuration overview, and checkout history), a password must be configured:

```
"ADMIN_PASSWORD": "SuperUser123",
```

**Note:** The password is sent in base64 encoding only, so please choose a unique password that will be used only for the VS License Manager.

Optionally, it is possible to configure a main password. With this setting enabled, even the basic License Monitor will prompt for a password before displaying any information:

```
"MAIN_PASSWORD": "",
```

#### Logging

The location of the VS License Manager log files can be changed using the `LOG_PATH` parameter. The path can be absolute or relative with a leading `./`:

```
"LOG_PATH": "./Logs",
```

The log format can be specified with the eponymous parameter:

```
"LOG_FORMAT": "NORMAL",
```

The alternative log format to `NORMAL` is `RESTRICTED`, which suppresses writing usernames, host names and IP addresses to the log file and displaying them in the License Monitor. For support cases, it is possible to convert recent log files from normal to restricted format using a button in the License Monitor.

It is possible to configure the number of days before old log files are removed:

```
"LOG_ARCHIVE_COUNT": 90,
```

The default is 90 files, which is equivalent to 90 days or 3 months. If you never want log files to be removed, set the value to zero.

## ▶ Advanced Settings

### High Availability Licensing (HAL)

To ensure fail-safe licensing (e.g. server failure or updates), it is possible to set up redundant servers that can all serve the same licenses and features. The server addresses are defined like the value `LicenseServerAddress` in the client license `3DVS.lic`:

```
"HAL_SERVERS": "6200@server01;6200@server02;6200@server03",
```

The order of the servers defines the priority of availability, so the first server should be the main server with the highest availability. If one server goes down, the next server in the list will take over as long as the other server is unavailable. The Borrow functionality is only available on the main server and will be unavailable during the downtime of the main server. Our HAL configuration supports two or more servers, the default is three.

#### Note:

- The list must be the same for all server configurations and client licenses. Otherwise, undefined behavior occurs.
- Please inform us about changes in your HAL configuration to keep your license up to date. To do this, send an email to [orders-viewer@kisters.de](mailto:orders-viewer@kisters.de) with your server names, selected ports, the serial number of your license.

### Internet Protocol Version

The VS License Manager supports both versions of the Internet Protocol (IPv4 or IPv6). Which one is preferred can be defined here:

```
"IP_VERSION": "IPv4",
```

### History Settings

The history data (maximum usage and denials) is updated frequently, but only every eight hours by default to reduce server load. This can be changed in the preferences:

```
"HISTORY_UPDATE_HOURS": 8,
```

Similar to the log file archive cleanup, it is possible to automatically remove old history data. This includes only real-time data, which occupies most of the history data space. This setting is disabled by default:

```
"HISTORY_ARCHIVE_COUNT": 0,
```

### Heartbeats and Timeout

To ensure continued use of checked-out features, the VS License Manager uses heartbeats.

- Use `HEARTBEAT_SECONDS` to configure how often heartbeats are sent from clients. The default is 2 minutes and 20 seconds.
- `TIMEOUT_SECONDS` allows you to configure when a client should timeout after the last heartbeat has been successfully received. The default is 5 minutes.



- Use `HEARTBEAT_CONTROL_SECONDS` to configure how often the server checks for client heartbeats. The default is 3 seconds. If you have a large number of clients, this value can be increased slightly to reduce server load.

In the configuration file, these values will look like the following:

```
"HEARTBEAT_SECONDS": 140,  
"TIMEOUT_SECONDS": 300,  
"HEARTBEAT_CONTROL_SECONDS": 3,
```



Chapter III:

# Privilege Rule Configuration



### 3 Privilege Rule Configuration

The VS License Manager's usage privileges can be configured by editing and saving the `ServerConfig.json` file. This includes allowing, denying, restricting, and reserving checkout and borrowing of features for specific users. Restart the VS License Manager for the changes to take effect.

#### When to use which privilege rule?

- If you want most of your users to be able to use all features and only a few to be restricted, use [Deny Rules](#)<sup>13</sup>.
- If you want to allow a few privileged users to use certain features and restrict most of your users, use [Allow Rules](#)<sup>12</sup>.
- If you want to control the number of features available to different user groups, use [Limit Rules](#)<sup>14</sup> and [Reserve Rules](#)<sup>15</sup>.

#### How to specify users?

In each rule, users can be specified as usernames, host names, or IP addresses (IPv4 or IPv6, [see setting IP\\_VERSION](#)<sup>8</sup>) and must be separated by a semicolon:

```
"USERS": "UserOne;UserTwo;UserThree;..."
```

If the list becomes large, or if the same list is used in multiple privilege rules, it is useful to define `GROUPS` at the end of the configuration file. The format for `MEMBERS` is the same as for `USERS`. Curly braces `{...}` define a group within the group list `[...]`, separated by commas.

```
"GROUPS": [
  {
    "NAME": "Group 1",
    "MEMBERS": "UserOne;UserTwo;UserThree;...;UserN"
  },
  {
    "NAME": "Group 2",
    "MEMBERS": "Host01;Host02;Host03;...;HostX"
  }
]
```

You can now specify users as follows:

```
"USERS": "Group 1;Group 2"
```

Linking a group within a group is supported.

#### Note:

- When restricting many host names or IP addresses, you can use wildcards like `"4.3.2.*"` or `"PC*"` or IP ranges such as `"4.3.2.10-50"`. Usage examples can be found below.
- On Windows, you can use your local groups. If you have a local group `"My Users"` that contains a list of domain users or a domain user group, you can use it like a user or group name in the configuration file by simply prefixing it with `'&': "USERS" : "&My Users"`. Usage examples can be found below.
- The default `ServerConfig.json` contains two empty presets per rule, which can be edited or removed and are ignored by the VS License Manager.

#### Important syntax rules:

- The server configuration file uses standard JSON syntax.
- Square brackets `[...]` define the list of privilege rules for a privilege setting, separated by commas.
- Curly braces `{...}` define a single privilege rule, separated by commas.
- Multiple key-value pairs within a privilege rule must also be separated by commas.
- All users, features, and serial numbers must be separated by a semicolon.
- You can use the following examples as templates.

## Allow Rules

The `ALLOW_CHECKOUTS` and `ALLOW_BORROWS` allow rules restrict the use of features to all but the specified users. So if you have privileged users, use these settings. They consist of three keywords:

- `USERS`: required, list of usernames or host names, IP addresses or group names
- `FEATURES`: optional, list of the features to be restricted
- `SERIALNUMBERS`: optional, the affected licenses / serial numbers (only needed if using multiple licenses on one VS License Manager)

**Note:** The optional `FEATURES` and `SERIALNUMBERS` keywords can be removed, left blank, or set to '\*' if not used. The rule will then apply to all `FEATURES` / `SERIALNUMBERS` loaded by the VS License Manager.

### Examples

A rule that allows functionality features only for selected users would look like this:

```
"ALLOW_CHECKOUTS": [
  {
    "FEATURES":
    "FUNCTIONALITY_ADVANCED_ANALYSIS;FUNCTIONALITY_TECHDOC",
    "USERS": "Some user; Some other user; Another user"
  }
],
```

Allowing borrowing of `EXPORT_ALL` only for the `MEMBERS` defined in a `GROUP` with the `NAME` "Borrow Users" would look like this:

```
"ALLOW_BORROWS": [
  {
    "FEATURES": "EXPORT_ALL",
    "USERS": "Borrow Users"
  }
],
...
"GROUPS" : [
  {
    "NAME": "Borrow Users",
    "MEMBERS": "Some user; Some other user; Another user"
  }
]
```

The same rule, but restricting the borrowing of all features for all users except the members of the defined group:

```
"ALLOW_BORROWS": [
  {
    "USERS": "Borrow Users"
  }
],
```

Multiple allow rules are separated by a commas:

```
"ALLOW_CHECKOUTS": [
  {
```

```

    "FEATURES":
"FUNCTIONALITY_ADVANCED_ANALYSIS;FUNCTIONALITY_TECHDOC",
    "USERS": "Functionality Users"
  },
  {
    "FEATURES": "KAS",
    "USERS": "KAS Users"
  }
],

```

Allowing a feature only for a specific windows user group using the prefix & would look like this:

```

"ALLOW_CHECKOUTS": [
  {
    "FEATURES": "EXPORT_ALL",
    "USERS": "&WindowsGroup"
  }
],

```

Allowing a feature for a specific hostname pattern would look like the following. This would allow all PC's with names like PC001 and PC456 or WS12CAD and WS34CAD.

```

"ALLOW_CHECKOUTS": [
  {
    "FEATURES": "EXPORT_ALL",
    "USERS": "PC*;WS*CAD"
  }
],

```

Allowing one of the multiple licenses only for a specific IP domain would look like this:

```

"ALLOW_CHECKOUTS": [
  {
    "SERIALNUMBERS": "2024-0101-123456",
    "USERS": "4.3.2.*"
  }
],

```

Allowing a license only for a specific IP range would look like this:

```

"ALLOW_CHECKOUTS": [
  {
    "USERS": "4.3.2.10-50"
  }
],

```

## Deny Rules

The DENY\_CHECKOUTS and DENY\_BORROWS deny rules restrict the use of features to the specified users, so most users will still be able to use them. The syntax and keywords are the same as for allow rules.

## Examples

A rule that denies features to selected users would look like this:

```
"DENY_CHECKOUTS": [
  {
    "FEATURES":
    "FUNCTIONALITY_ADVANCED_ANALYSIS;FUNCTIONALITY_TECHDOC",
    "USERS": "Some user; Some other user; Another user"
  }
],
```

A rule that denies the borrowing of `EXPORT_ALL` for the `MEMBERS` defined in a `GROUP` with the `NAME` "Borrowing denied" would look like this:

```
"DENY_BORROWS": [
  {
    "FEATURES": "EXPORT_ALL",
    "USERS": "Borrow denied"
  }
],
...
"GROUPS" : [
  {
    "NAME": "Borrow denied",
    "MEMBERS": "Some user; Some other user; Another user"
  }
]
```

### Note:

- At the same time, checkout rules also restrict borrowing actions, but borrowing rules do not affect checkout actions.
- Deny rules override all allow rules. If a user is listed in both an allow rule and a deny rule for the same license and feature, the deny rule is applied and the allow rule is ignored.

## Limit Rules

The `LIMIT_CHECKOUT` and `LIMIT_BORROW` limit rules are useful for controlling the number of features that should be available to different user groups. They are also used to limit the borrowing duration of features. Their syntax is based on the allow and deny rules, with additional keywords:

- `COUNT`: The number of features that should be available to the specified `USERS`.
- `EXCEPT_USERS`: It is possible to add exceptions to a limit rule. So if you have privileged users within a larger group that you want to restrict, add them here.

Only for `LIMIT_BORROW`:

- `HOURS`: The maximum number of hours the specified features can be borrowed.
- `USERS` can be optional for this rule type. If it is removed or left blank, the borrowing limit is applied to all users.

## Examples

Restrict the use of a feature within a group of users:

```
"LIMIT_CHECKOUTS": [
```

```

    {
      "FEATURES": "IMPORT_ALL",
      "USERS": "User 1;User 2; User 3;...;User 30",
      "COUNT": 15
    }
  ],

```

Restrict the borrowing period of any feature to one week for all users, except for a privileged user who is allowed to borrow for three months. This is accomplished by first increasing the default value to three months, and then restricting all users except the privileged user:

```

"LIMIT_BORROWS": [
  {
    "HOURS": 2160
  },
  {
    "EXCEPT_USERS": "Privileged User",
    "HOURS": 168
  }
],

```

## Reserve Rules

The reserve rule `RESERVE_CHECKOUTS`, unlike the [limit rules](#)<sup>[14]</sup>, can be used to ensure that a certain number of features are always available to certain users. The syntax is the same as for [allow rules](#)<sup>[12]</sup>, with the addition of the `COUNT` keyword.

### Examples

Reserve a feature for three privileged hosts:

```

"RESERVE_CHECKOUTS": [
  {
    "FEATURES": "IMPORT_2D_ADVANCED",
    "USERS": "HostX;HostY;HostZ",
    "COUNT": 3
  }
],

```



Chapter IV:


# License Monitor






## 4 License Monitor

The License Monitor, the web interface of the VS License Manager, can be accessed via a web browser: `<License Server Address>:<HTTP_PORT>`. It displays license information, feature availability and usage, and connected clients. The License Monitor is automatically refreshed every 60 seconds to keep it up to date.

With the configured `Admin password`  it is possible to access an extended version of the License Monitor with the latest log messages, the configuration settings and a section to visualize the checkout history in charts. These tabs do not reload automatically.

If required, access to the basic License Monitor can be completely restricted by setting a `main password` .

### Checkout History

The VS License Manager stores data locally about how many features are used over time and by which users. In addition to a real-time view of feature usage for the current day, there are preconfigured charts for hourly usage for the last 24 hours, daily usage for the last 30 days, monthly usage for the last 12 months, and daily denied users for the last 30 days. It is also possible to create custom charts for individual time periods.

Usage data is stored in the installation directory under `/data/History/`. The data is stored in CSV format and can be easily used for further analysis. The columns are `time` (seconds since 01.01.1970), `count`, `feature`, `username` and `errorcode`.




Chapter V:

# Troubleshooting



## 5 Troubleshooting

### "No allow rule found in windows firewall for tcp <port>, license checkout might not be accessible"

- Check that you have allowed the [configured ports in your firewall](#) .
- If you are using a third-party firewall (or no firewall at all), you can ignore this message. We can only check this successfully, if the Windows firewall is active.