

Runtime issues

First-time exceptions in Visual Studio

When the debug mode is enabled, the debugger gets the first chance to see all exceptions before the program does. If the debugger allows the program execution to continue, the program will see the exception as usual. If the program does not handle the exception, the debugger gets a second chance to see the exception, normally causing the program to crash if the debugger were not present.

When running an application with a debug mode enabled in Visual Studio, you may see the following message in the output window:

```
"First-chance exception at 0x76c1b727 in yourapp.exe: some_exception_type at memory location 0x0033411c"
```

The above message doesn't indicate that LM-X didn't handle an exception, but only notifies you that the exception occurred. If LM-X handles the exception gracefully, no second chance exception will appear.

Please note that LM-X guarantees exception safety and doesn't leave any exceptions unhandled, even if they occur internally.

Where does LM-X store its libraries?

Every LM-X protected application checks for the existence of dll files in the following order and uses the first path found:

On Linux:

1. The path specified by the [TMPDIR](#) environment variable.
2. The `/tmp` path.

On Windows:

1. The path specified by the `TMP` environment variable.
2. The path specified by the `TEMP` environment variable.
3. The path specified by the `USERPROFILE` environment variable.
4. The Windows directory.

USERPROFILE locks the licensing process

When re-defining the `USERPROFILE` environment variable on Windows, the license checkout may lock the licensing process: in such situations closing the application abruptly seems to be the only solution. One workaround to this problem is to restore the value *before* using any LM-X function and then set it again after making LM-X API calls to prevent this issue from occurring.