

# License your first application

Now that you have defined a license policy, you need a license template to integrate LM-X with your application.

## License file

To generate an XML license template:

1. Create a template of the license you are interested in.
2. Run the license generator provided with LM-X, `xmllicgen`, as described in [Generating licenses](#).

In the following example we assume the license will expire on January 1, 2018. This license defines one feature with version 1.0.

```
<?xml version="1.0" encoding="UTF-8"?>
<LICENSEFILE>
  <FEATURE NAME="feature">
    <SETTING MAJOR_VERSION="1" />
    <SETTING MINOR_VERSION="0" />
    <SETTING END="2018-01-01" />
  </FEATURE>
</LICENSEFILE>
```

Run `xmllicgen` to convert the above LM-X license template to a license file.

### On Unix:

```
# xmllicgen license.xml
```

### On Windows:

```
# xmllicgen.exe license.xml
```

The resulting file will be `license.lic`. `Xmllicgen` will automatically replace your `".xml"` extension with `".lic"` and save the license next to your template.

## Source code

### Header file

First of all, you must include the required header, `lmx.h`, located in the include directory of your SDK.

```
#include "lmx.h"
```

If your project and LM-X SDK are in separate directories, remember to specify the include directory for your compiler. For example:

### GCC:

```
gcc -I /usr/lmx-sdk-4.6.1/include/
```

### MSVC:

```
cl.exe /I "C:\Program Files\X-Formation\LM-X SDK v4.6.1 win64_x64\include\"
```

## Commonly used functions

In the following examples we will present five commonly used [LM-X API](#) functions.

### LMX\_Init

The [LMX\\_Init](#) function initializes the protection system. This function creates a handle needed to use other LM-X API functions. It returns `LMX_STATUS` variable that indicates initialization status.

```

LMX_HANDLE LmxHandle;

if (LMX_Init(&LmxHandle) != LMX_SUCCESS)
{
    printf("Unable to initialize!\n");
    return 1;
}

```

### LMX\_SetOption

The [LMX\\_SetOption](#) function sets up flags that change licensing behavior prior to license checkout. This function returns LMX\_STATUS variable that indicates the status of initialization.

In the following example LMX\_SetOption is used to set the license path to the current path.

```

LMX_SetOption(LmxHandle, LMX_OPT_LICENSE_PATH, ".");

```

### LMX\_Checkout

The [LMX\\_Checkout](#) function is one of the most important LMX API functions, because it checks out one or more licenses for a specific feature. This function requires that the feature name, version and the count of the features be defined as shown below.

```

if (LMX_Checkout(LmxHandle, "feature", 1, 0, 1) != LMX_SUCCESS)
{
    printf("Unable to checkout\n");
    LMX_Free(LmxHandle);
    return 1;
}

```

### LMX\_Checkin

The [LMX\\_Checkin](#) function returns the licenses for a single checked out feature or all checked out features.

```

LMX_Checkin(LmxHandle, "feature", LMX_ALL_LICENSES);

```

### LMX\_Free

The [LMX\\_Free](#) function, which has an inverse effect to [LMX\\_Init](#), frees any allocated memory used by the licensing system and closes any open connection to a license server.

```

LMX_Free(LmxHandle);

```

## Making it work

The following example illustrates a complete, compiled example that includes source code necessary to license your application using LM-X License Manager.

```

#include <stdio.h>

#include "lmx.h"

int main()
{
    LMX_HANDLE LmxHandle;

    if (LMX_Init(&LmxHandle) != LMX_SUCCESS)
    {
        printf("Unable to initialize!\n");
        return 1;
    }

    // Look for licenses in current directory.
    LMX_SetOption(LmxHandle, LMX_OPT_LICENSE_PATH, ".");
    if (LMX_Checkout(LmxHandle, "feature", 1, 0, 1) != LMX_SUCCESS)
    {
        printf("Unable to checkout!\n");
        LMX_Free(LmxHandle);
        return 1;
    }
    // Here you are safe to run your licensed features
    printf("Here you can run your features\n");

    LMX_Checkin(LmxHandle, "feature", LMX_ALL_LICENSES);
    LMX_Free(LmxHandle);
    return 0;
}

```

If you decide to copy and paste the above code block, save it as an `example.c`.

## Compilation

Let's assume you saved your program source code as an `example.c` and installed your LM-X SDK in the default directory.

To compile your first program run the following:

### GCC:

```

gcc -c -pthread -fPIC -Wall -Werror -fno-strict-aliasing -m64 -Wfatal-errors -Wno-unused-local-typedefs -Wno-
vla -Wno-attributes -O2 -c -O2 -I/usr/lmx-sdk-4.6.1/include/ example.c

gcc -static-libgcc -o example example.o /usr/lmx-sdk-4.6.1/linux_x64/liblmxclient.a -pthread -lrt -ldl

```

### MSVC:

```

cl /WX /MT /c /O2 -D_CRT_SECURE_NO_DEPRECATED /I "C:\Program Files\X-Formation\LM-X SDK v4.6.1
win64_x64\include\" example.c

link /WX /opt:noref example.obj "C:\Program Files\X-Formation\LM-X SDK v4.6.1
win64_x64\win64_x64\liblmxclient_mt.lib"

```

## Running your application

Your current directory should include the files listed in the table below.

File	Description
license.xml	Your license template.
license.lic	Ready to use license, generated with xmllcigen.
example.c	Your first program source code.
example/example.exe	Your program executable.

Now run your first LM-X licensed application.