SEMLA:

Standardized encryption of Modelica libraries and artifacts

Feb 25 2021

Modelon

WHY SEMLA?

- Need standard packaging of commercial closed source Modelica libraries
  - Portable across different Modelica tools (compilers and editors)
  - Cross-platform (Windows/Linux/OSx)

- Need to decouple IP protection and licensing of Modelica tools (compilers and editors) and Modelica libraries
  - Tool and library may use different licensing mechanisms

Modelon

CURRENT STATE

- Specification proposal & default implementation available online :
  https://github.com/modelon-community/SEMLA/blob/master/doc/SEMLA.md
- Issue on Modelica Association Github
  https://github.com/modelica/ModelicaSpecification/issues/1868

- Tool support:
  - Used in released version of Modelon OCT Compiler and Libraries
    - Modelon IMPACT, ANSYS TwinBuilder
    - Some Modelon customers for in-house libraries
  - Supported by OpenModelica
  - Interest from other vendors ESI, Wolfram and Maplesoft

Modelon

KEY CONCEPTS

- Modelica Library Container (MLC)
  - Zip file (*.mol*) with special structure that contains
    - Encrypted *.moc* files
    - *manifest.xml* meta-data file
    - Library Vendor Executables (LVE)
- Library Vendor Executable (LVE)
  - Library vendor provided executable that enables
    - Interface to the MLC
    - Library specific license check
    - One LVE can handle multiple tools (Modelon's does it)

test_library.mol

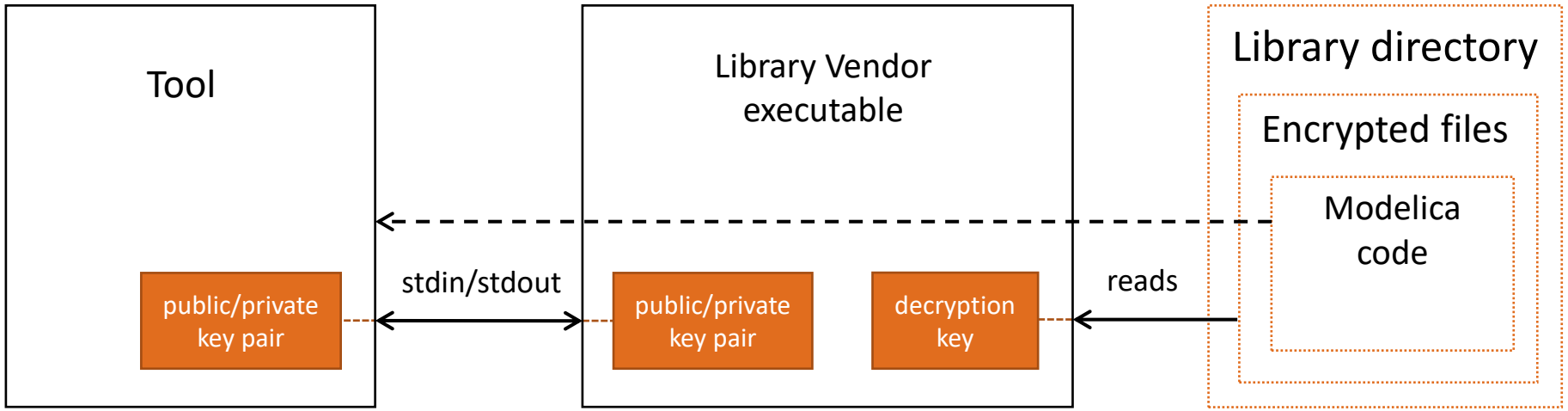
```
test_library/
 .library/
    lve_win64.exe
    lve_linux64
    manifest.xml
  Module/
    package.moc
    package.order
    testInPackage.moc
  binary.gif
  package.order
  testModel.moc
```

- Trusted (authorized) tool
  - SEMLA framework relies on trust between library vendors and tool developers
  - Correct handling of protection access and licensing annotations in the library remains tool responsibility
- SEMLA Protocol
  - Protocol for communication between LVE and Modelica tool

Modelon

COMMUNICATION CHANNELS AND KEYS

- Authorized Tool (Modelica compiler or editor) starts LVE (Library Vendor Executable) and establishes secure connection to it
- LVE is responsible for validating the tool and implements library side of the protocol including
  - Library license check
  - Library decryption
- Default LVE implementation does not parse or modify Modelica code in any way

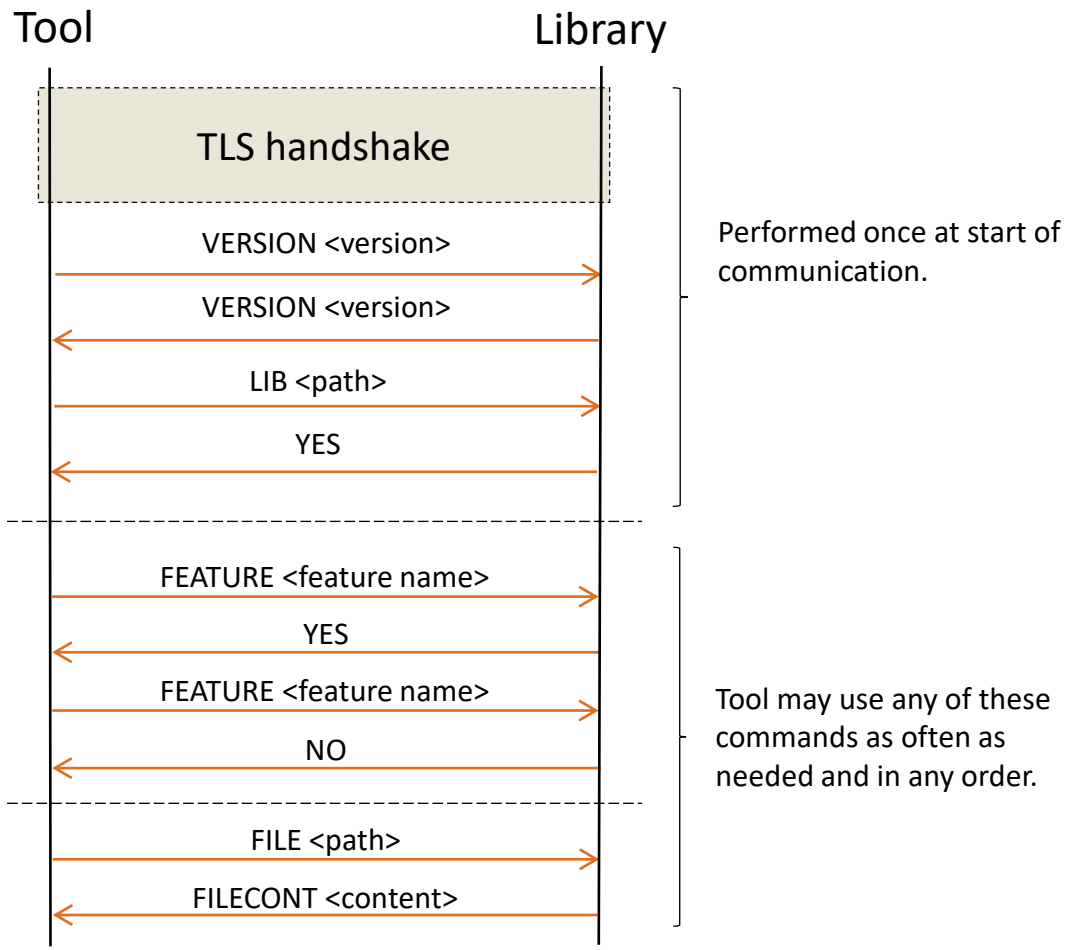


SEMLA PROTOCOL FLOW

Handshake:
- LVE checks that the tool is authorized
- Optional license check.

License check:
- tool request features based on annotations in Modelica
- LVE performs license check

Decryption



CURRENT STATUS SUMMARY

- SEMLA is tested with several Modelica-tools
- SEMLA is cross-platform
- Handles encryption of Modelica code
- Handles optional licensing
- Decouples licensing and IP-protection between tool and library
- Could be an enabler for much broader Modelica-community for model & library content

Modelon

NEXT STEPS & OPEN QUESTIONS

It has been unclear who should be responsible for this:

- This is not a core “language” issue, rather an API/protocol
- This is in the interest of MAP-LIB, but MAP-LIB members are no experts in the required technical knowledge

- Figure out whether other standards want to use it as well for their artifacts
  - FMI, SSP, eFMI if it becomes a project?
- Agree on forum to drive standardization forward – which MA project(s) should drive this?
- If multiple projects: which process/repositories should be used?
- Should this be organized as a short-term project?

Modelon