

Model Obfuscation



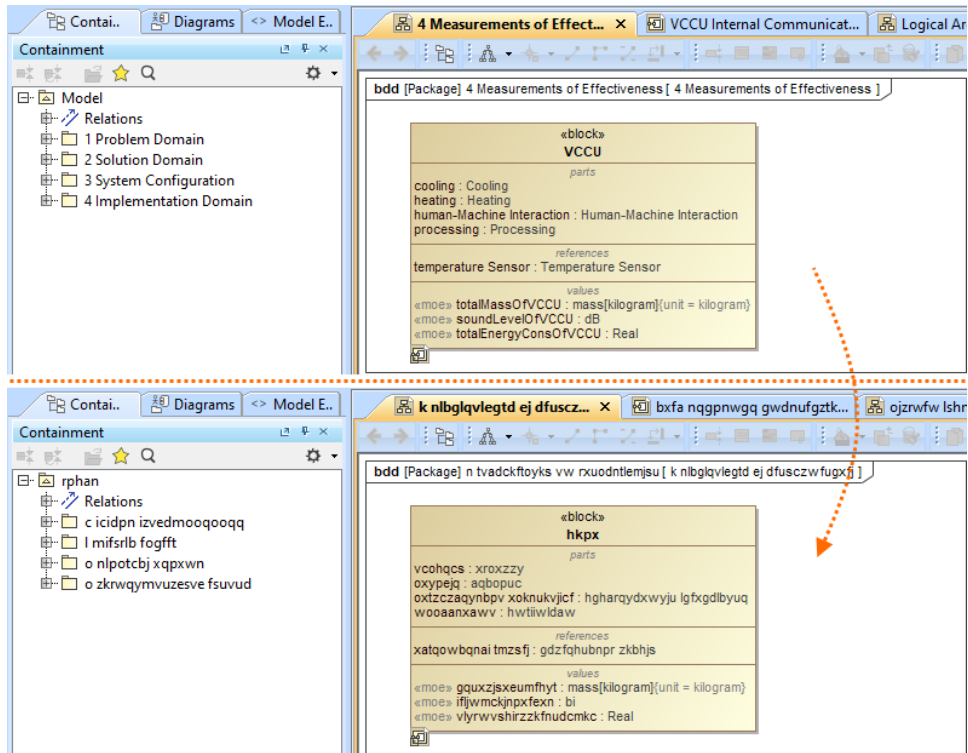
You need to have the Model Obfuscator plugin installed to use the obfuscation functionality.

Model Obfuscator plugin is used to obfuscate confidential data (texts or images) in the local project for sending it to No Magic support when reporting related issue. The obfuscation process changes user project data strings (e.g. class name "User_login" will be changed to "xcolnyrwx") and saves it as a new project.

Data obfuscation

To perform data obfuscation

1. In the main menu, click **File > Obfuscate Project**. The **Save** dialog opens.
2. Define the name and select the location in your file system for the obfuscated project. Click **Save**.
3. Once the obfuscated project is saved, the message "Obfuscation process is done" is displayed. Click **OK**.



An example of a project with obfuscated data.

Custom data obfuscation

You can create your own custom encryption algorithm instead of using the default one.

To create a custom obfuscation algorithm

1. Create a new plugin as described on the [Writing plugins](#) page.
 - 1.1. You can compile your plugin's main class by defining `<install.root>/lib/md_api.jar` in the classpath parameter.
 - 1.2. Create an additional class that extends `com.nomagic.magicdraw.modelobfuscator.WordObfuscator` in your plugin. For an example, see the code below:

```

package myplugin;

import com.nomagic.magicdraw.modelobfuscator.WordObfuscator;
import java.security.SecureRandom;

public class SecureWordObfuscator implements WordObfuscator
{
    private static final char STARTING_CHAR = 97;
    private static final int BOUND = 26;

    private final SecureRandom random = new SecureRandom();

    @Override
    public String obfuscateWord(String word)
    {
        int len = word.length();
        StringBuilder tokenBuilder = new StringBuilder(len);
        for (int i = 0; i < len; i++)
        {
            tokenBuilder.append(String.valueOf(Character.toChars(STARTING_CHAR + (char)
random.nextInt(BOUND))));
        }
        System.out.println("Original word: " + word + " obfuscated word: " + tokenBuilder.
toString());
        return tokenBuilder.toString();
    }
}

```

2. Compile it and include this class in your plugin's jar. When compiling the class, define `<install.root>/plugins/com.nomagic.magicdraw.modelobfuscator/modelobfuscator_api.jar` in the classpath parameters.
3. Add the Java option `-Dmo_word_obfuscator_impl=<full class name>` (e.g. `-Dmo_word_obfuscator_impl=myplugin.SecureWordObfuscator`) in the application's properties file (e.g. `magidraw.properties`) at the end of the `JAVA_ARGS=` line.
4. Open the modeling tool and obfuscate your project.