The CrypTool 2.0 Development Cycle (16.09.2013)

Author: Nils Kopal (nils.kopal(AT)uni-kassel.de)

This document describes the life cycle which a Cryptool 2.0 component (also known as plugin) goes through during its development phases. Most Cryptool 2.0 components are the result of bachelor, diploma or master theses or projects and the time for development varies between three month and half a year. Nevertheless, the quality of a component is the most important aspect which should be kept in mind during the development. To assure this quality we defined several milestones which a component should reach before we decide to release it publicly. To get detailed information how to develop a component (Cryptool 2.0 plugin) have a look at the Development how-to [1].

I. Some important definitions:

a. Cryptool 2.0 SVN:

The "Cryptool 2.0 SVN" (Subversion) [4] is the centralized source management system. Every Cryptool 2.0 artefact (source code, templates, etc.) should be checked into the SVN.

b. Nightly Build:

The "Nightly Build" [5] is the automated Cryptool 2.0 build, which is generated each night automatically by the Cryptool 2.0 build server based on the "Core Solution" which is available in the "Cryptool 2.0 SVN". The compiling should never fail due compiling errors. The nightly build is available through the Cryptool 2.0 auto updater to every user who installed one of the previous nightly builds. It can be manually downloaded by the users using the Cryptool 2.0 website, too.

c. Public Solution:

The "Public Solution" is the Visual Studio solution which every developer may use to compile his own Cryptool 2.0. It contains all main components and plugins. There have to exist **no** compiling errors caused by checked in components (code or missing elements). Keep in mind that every developer works on this solution! So don't check in broken code or forget to check in new added files of your component.

d. Core Solution:

The "Core Solution" is the Visual Studio solution which only core developers have access to. It contains all main components and plugins, which should be release to the nightly build. Thus, it is the base for the nightly build. There have to exist **no** compiling errors caused by checked in components (code or missing elements) which are part of the "Core Solution".

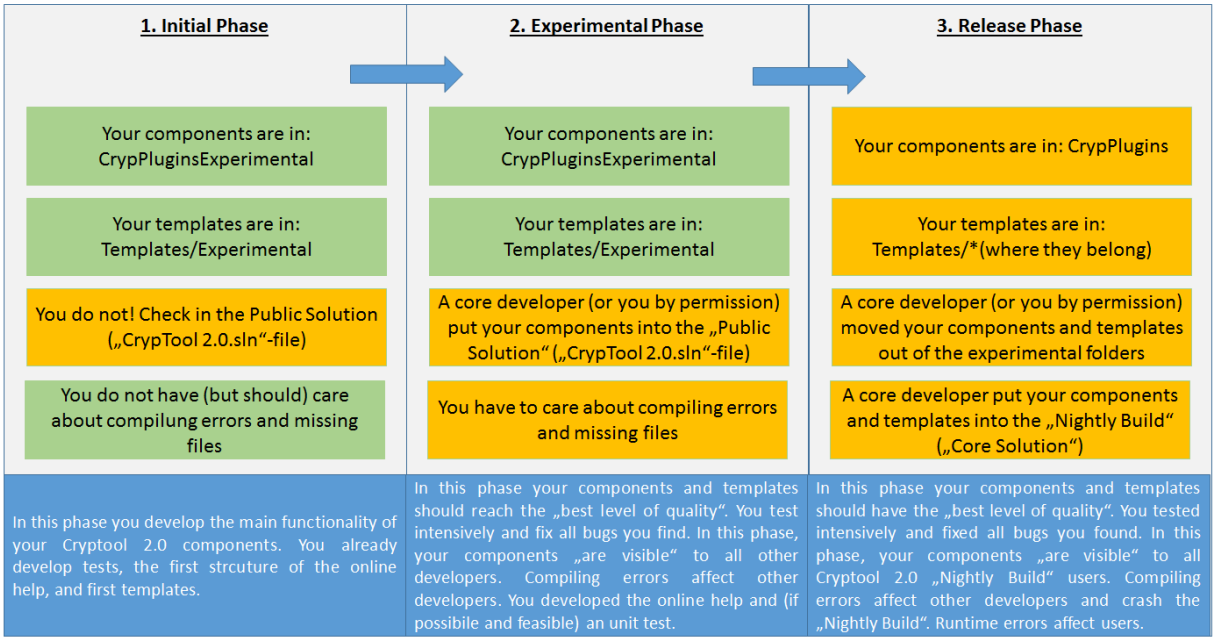
e. Templates:

The templates are special kind of Cryptool 2.0 cwm-files with extra meta information. They are delivered as part of the Cryptool 2.0 build. Users may open templates using the "Startcenter". A template should have its own icon and contain extra explanations for the user (text and/or images).

f. Unit Tests:

A "Unit Test" is a kind of test which tests the functionality of a single component. In Cryptool 2.0 we use unit tests to verify (for example) if an implemented cryptographic algorithm does what it should do. For the unit test, we use test vectors which are available publicly. Nearly every component should have its own unit test. The unit tests have to be placed in the "UnitTests" project.

II. The general development life cycle of a Cryptool 2.0 component:



Phase transitions are only possible if you have an advice of a core developer.

1. Initial phase:

In the initial phase your Cryptool 2.0 component start living. Use the Cryptool 2.0 development how-to to develop your component. Your new component should be placed into to “CrypPluginsExperimental” folder. In the initial phase, you are not allowed to check-in the public solution. Nevertheless, you do not have to take care about compiling errors or not checked in parts of your component (but you should take care). Have in mind to check in early and often to back up your work.

In this phase your Plugins are in: CrypPluginsExperimental
In this phase your Templates are in: Templates/Experimental

2. Experimental phase:

In the experimental phase, your component is put into the “Public Solution”. This is done by a core developer or by yourself if you have the permission of a core developer. At this point, your component is visible to all developers that updated the “Public Solution”. So your component must not contain any compiling errors or errors due to missing parts of your component (i.e. forgot to add files into the SVN). In this phase have to take care about compiling errors or not checked in parts of your component.

In this phase your Plugins are in: CrypPluginsExperimental
In this phase your Templates are in: Templates/Experimental

3. Release phase:

In the release phase, your component is put into the “Core Solution” by a core developer. Thus, your component is available to all users which use the “Nightly Build”. To get your component into the “Release phase” there have to exist the following:

- Your components are in a state of “high quality”. That means they contain as few as possible errors/bus/defects. You tested your components intensively. There exist no compiling errors etc.
- There exists at least one (or more) templates showing the functionality of your components
- There exists (if possible and feasible) a “Unit Test” which does not fail during testing
- There exists an xml online help (German and English) [7]
- Component is fully internationalized (German and English)
- For details have a look at [6]

In this phase your Plugins are moved to: CrypPlugins
In this phase your Templates are moved to: Templates/* (wherever it belongs to)

[1] Development how-to
https://www.cryptool.org/trac/CrypTool2/browser/trunk/Documentation/PluginHowTo/HowToDeveloper.pdf

[2] Development checklist
https://www.cryptool.org/trac/CrypTool2/wiki/HowToDevChecklist

[3] Plugin Template
https://www.cryptool.org/trac/CrypTool2/browser/trunk/Documentation/CrypPluginTemplate/CrypTool%202.0%20Plugin.zip

[4] Cryptool 2.0 SVN
https://www.cryptool.org/svn/CrypTool2

[5] Nightly Builds and Beta versions
http://www.cryptool.org/de/ct2-download-de

[6] Plugin Requirements
https://www.cryptool.org/trac/CrypTool2/wiki/PluginRequirements

[7] Documentation howto
https://www.cryptool.org/trac/CrypTool2/wiki/Documentation