Tizen Content Security Framework

Proposal

Document version 1.0.2

Document Information

Document Details

Revision	1.0.0
Author	MMS Development Team

Revision Information

Revision	Revision Date	Author	Details
1.0.0	10/31/2012	MMS Development Team	Created
1.0.1	11/12/2012	MMS Development Team	Add concurrent scan requirement.
1.0.2	01/26/2013	MMS Development Team	Add license

Contents

Terms, Abbreviations, Definitions, Conventions

Items	Description
SDK	Software Development Kit
API	Application Programming Interface
Content Screening	Screening content for security consideration
Module	Program, service or any execution entity in the Tizen platform
Application	Executable provided by either system or third-party

Overview

This document is to propose the security framework on Tizen platform. Tizen content security framework will be responsible for passing the security API calls to security plug-in, which could be provided by either security vendor or dummy plug-in. Dummy plug-in here refers to place holder implementation of security plug-in, it will return not implemented errors to caller to indicate that no security plug-in is installed. The framework is also responsible for error handling when there is no security vendor plug-in installed.

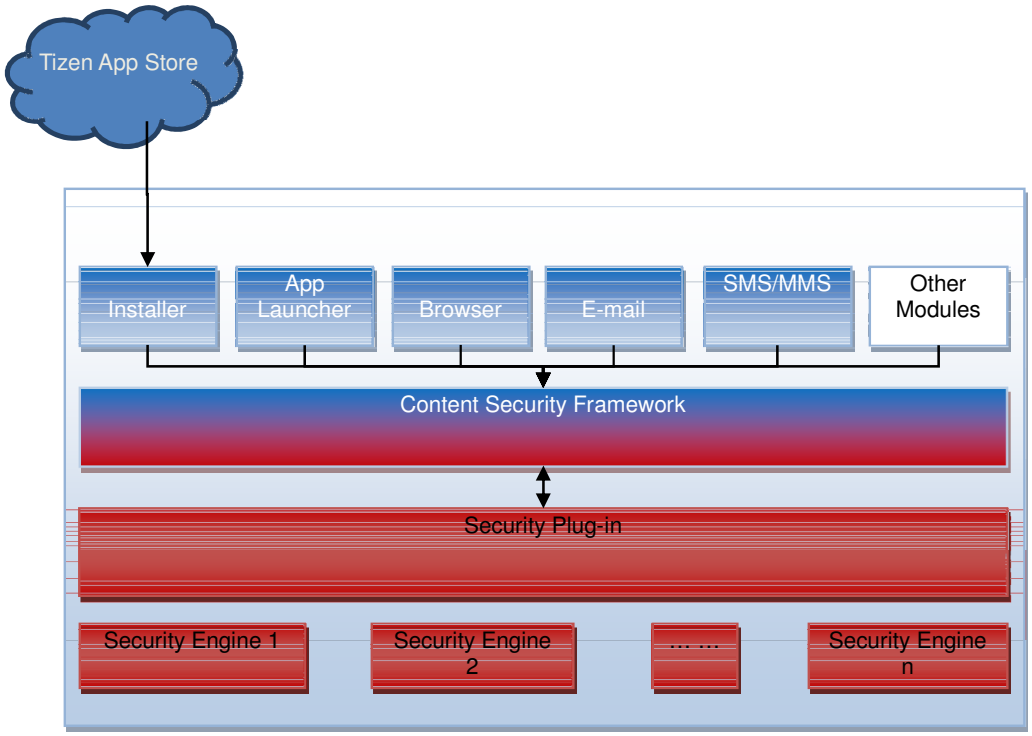

Figure 1 Overview of Content Security Framework

Solution Description

Both content security framework and security plug-in are shared libraries in Tizen case. All the code is loaded in the application memory space. The authentication of those libraries will be covered by Tizen certificate process.

Content security framework (libsecfw.so) will be linked directly to system component which is invoking the security API, while the security plug-in will be loaded in the runtime and installed along with security application package. The security application package should be signed with a trusted certificate which indicates that the package is authorized to carry security plug-in and is ready for use. And as a security consideration, Tizen installer will check this package whenever the application gets installed with following criteria:

Package format should conform to Tizen application format, it should be:

- Tizen application package is a simple zip file
- Package contains :
  - /bin/… (application executables)
  - /res/… (Icons, pictures, and other resource files)
  - /data/… (Data for the application)
  - /lib/plugin/libengine.so (This file will be copied into "/opt/usr/share/sec_plugin/ libengine.so" by installer)
  - /database/… (Data for the engine. E.g. Signature DB. This directory will be readable by anybody)
  - /info/manifest.xml (Containing proper app type. E.g. <category="sec-app" />)

Tizen installer enhancement:

- Check signature/certificate and should be certified by trusted party
- Copy /lib/plugin/libengine.so to /opt/usr/share/sec_plugin/ libengine.so
- Copy all files into /opt/usr/apps/[package id]/…
- Set smack label of /opt/usr/apps/[package id]/database to "sec_database"
- All apps which use content security framework already have rules to read "sec_database"
- Give permission for the security application (privilege for security check, like access to file system or other applications)

Content Security Framework

API standardizing

Content Security Framework will provide a set of APIs to other system modules with security features. Currently we have site engine and anti-virus engine API defined in this framework. Please refer to Tizen content screening and site engine API specification for detail.

Each security vendor who wants to add their plug-in to Tizen platform, need to provide a plug-in library which conforms to the Framework API which we defined in the framework above their own engines.

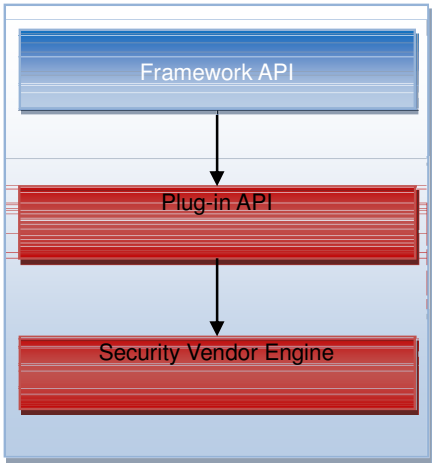

Figure 2 API Standardizing

Plug-in management

Content security framework is responsible for plug-in loading / reloading. It will always try to load the new plug-in from "/opt/usr/share/sec_plugin/libengine.so" when content security framework is reinitialized by library open API call, in this case it is TCSLibraryOpen(). This is saying that the newly installed security plug-in will be loaded only when TCSLibraryOpen() gets called. During the TCSLibraryClose() and TCSLibraryOpen(), the caller will keep using the old security plug-in until it close the library and reopen it.

Error handling

Content security framework will return not implemented error code to caller if there is no plug-in found at "/opt/usr/share/sec_plugin".

Concurrent Scan Support

The TCS security vendor plug-in must support concurrent scan in multi-tasking, so that Tizen component can have multiple threads to scan content concurrently.