# OSGi RFP-176 Malicious Bundle Test Framework - Readme

# Table of Content

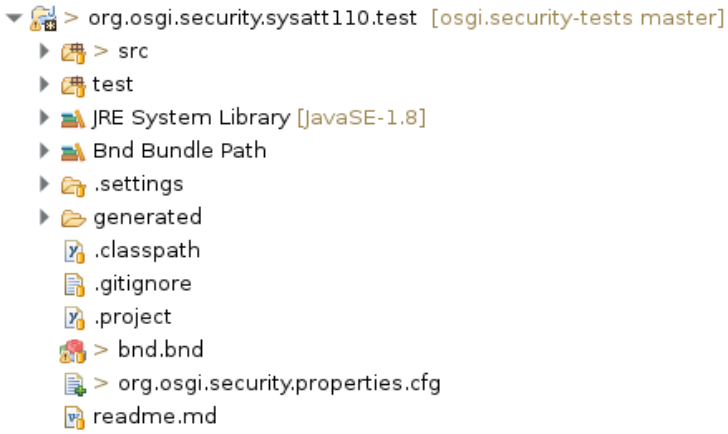
# 1 Test framework installation

This framework uses OSGi enRoute, a project which provides a programming model of OSGi applications. It includes a tool chain and libraries which permits to develop OSGi applications quickly. In this section, we will describe the installation procedure on a Linux target. All informations about this are available on enRoute website : prerequisites (http://enroute.osgi.org/tutorial_base/100-prerequisites.html) and enRoute environment setup (http://enroute.osgi.org/tutorial_base/200-workspace.html).

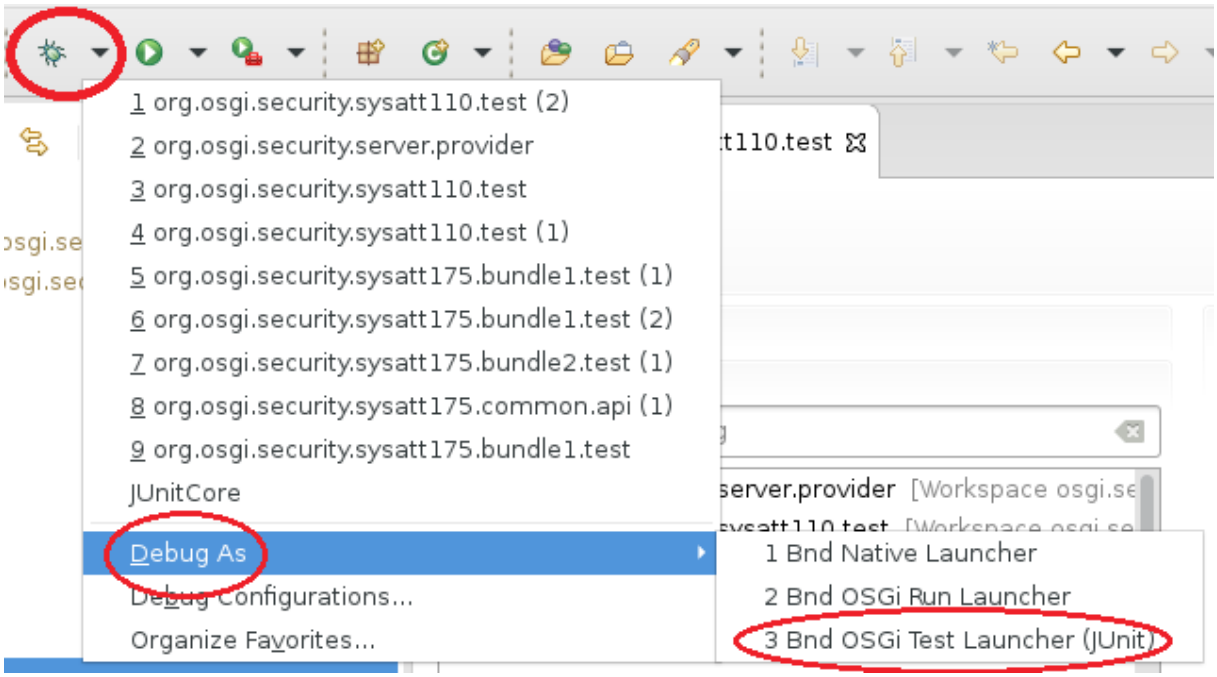
## 2 Test framework execution

In this part, we will detail how to use the test framework with OSGi enRoute.

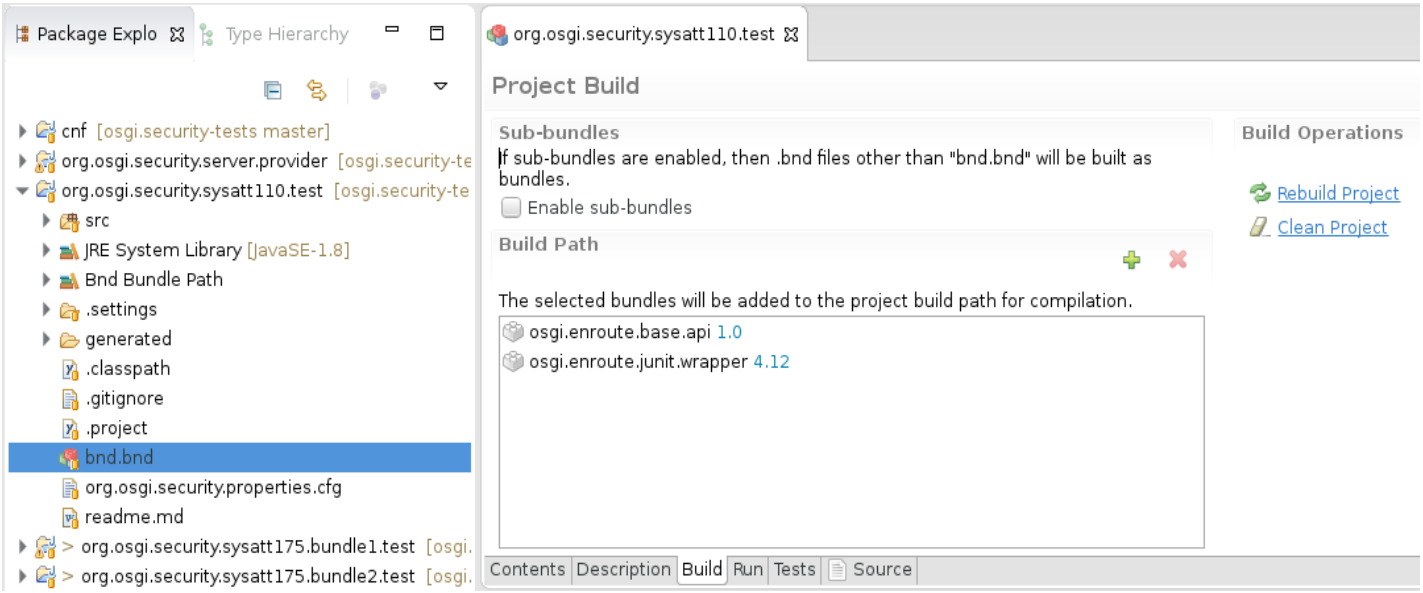
Each test has the following structure :



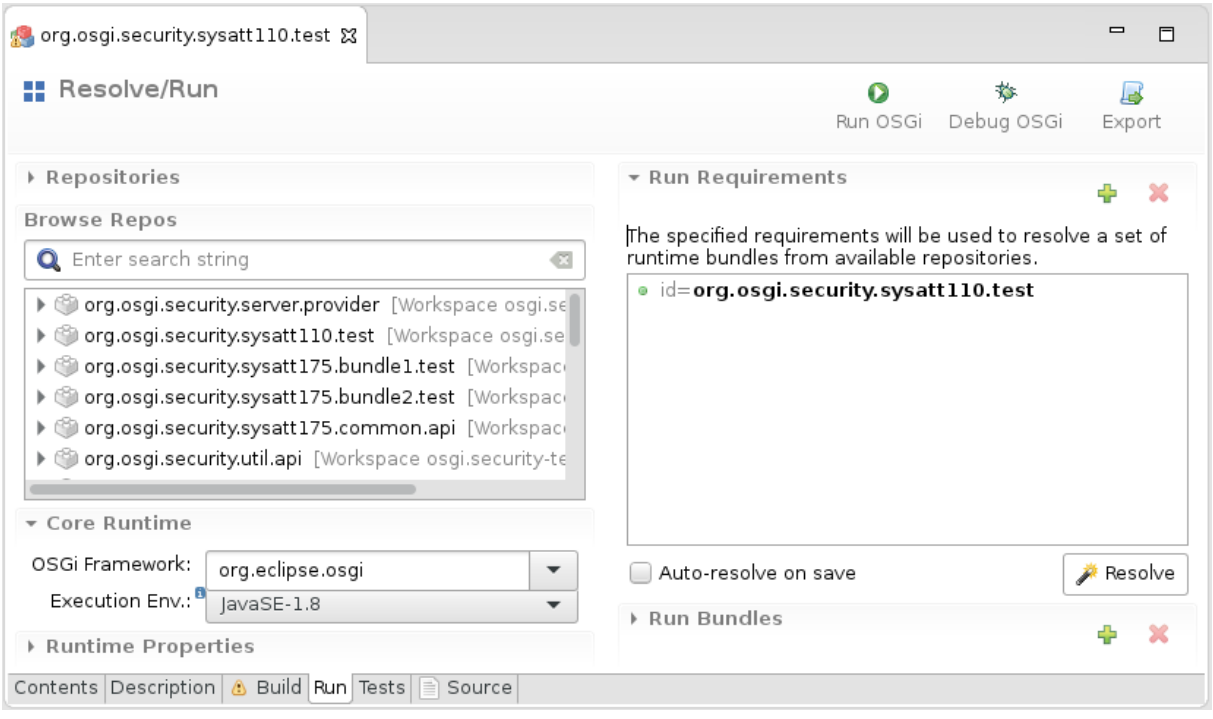

Test bundles are already generated, so you normally just need to run them without doing anything else. To do this, select the little arrow button just next to the « Debug As... » button. Then, select « Debug As » menu, and « 3 Bnd OSGi Test Launcher (JUnit) », as shown on the picture below :



If you need to rebuild all the bundles, you can do it with the **bnd** file. Open it, then select the « Build » tab :




Click on the « Rebuild project » button, then go in the « Run » tab :




Hit the « Resolv » button, click on « Finish » button, then run the bundle in the same way as described before.

# 3 OSGi Attack cases and countermeasures

Reader will find here below a detailed list of potential cyber-attacks that are achievable on OSGi platforms, through different attack targets:

- **OS** attacks, which are performed via OSGi bundles and target the OS.
- **OSGi Framework** attacks, which are performed via OSGi bundles and target the framework.

## 3.1 OS attacks

OS attacks group contains 9 attacks, which are divided in three types: **file/data leakage**, **code execution** and **file manipulation**. As the name suggests it, these attacks target the OS. The aim here is to hijack critical system data, manipulate system files or execute native code.

### 3.1.1 Files & Data leakage

| SYS-ATT-110: Log files hijacking | |
|---|---|
| Description | System log pertinent files hijacking, and sending of these files on the network to malicious distant server. This attack need a launched server. |
| Impact | Log files can contain some informations about system components/libraries/softwares (type, used version…). These informations can permit to perform attacks. |
| Recommendations | Set a sandbox system for bundle execution, in order to limit access to files and/or properties of the system. |

## 3.2 OSGi Framework attacks

OSGi attacks group is composed of 14 attacks, separated in four types: **data/information extraction badly formatted input data or files**, **Denial-of-Service** and **framework services manipulation**. Attacks of this group are more offensive than OS attacks group, as a large part of these attacks belongs to Denial-of-Service attack type. The goal here is to put the system out of service by Denial-of-Service attacks and manipulation of framework services.

### 3.2.1 Denial-of-Service attacks - Malicious service operations

| SYS-ATT-175: Infinite loop: mutually dependant services | |
|---|---|
| Description | Resource exhaustion by mutually dependant service subscriptions (ex: in a bilateral scenario, bundle A subscribes at bundle B services and conversely : bundle A state change provokes event sending, processed by bundle B which change its state, which provoke a event sending... etc, as an infinite loop plan). |
| Impact | The system may be not accessible when the infinite loop occurs. A restart may be necessary to access again to the system. |
| Recommendations | Resource Monitoring:<br>Setup a quota system and a runtime conflict resolution and monitoring system. |

# 4 Authors

For any suggestion or support, please contact :


| Julien Helmer | Romain Siche | Thibault Mainand |
|---|---|---|
| Cyber-security Technical leader | Cyber-security engineer | Cyber-security apprentice |
| julien.helmer@sogeti.com | romain.siche@sogeti.com | thibault.mainand@sogeti.com |