

- [Que es Black Duck Hub?](#)
- [Que es un componente?](#)
- [Que es y por que Black Duck Hub?](#)
- [Crear un proyecto en Black Duck Hub.](#)
- [Escanear el código del repositorio \(Bamboo\).](#)
- [Review de componentes.](#)
- [Obtener información necesaria para el review.](#)
- [Agregar componentes no detectados.](#)
- [Ignorar componentes.](#)
- [Manejo de Snippets.](#)
- [Monitorear componentes](#)

- [Script de monitoreo en Python](#)
- [Solicitar PDF de licencias y copyright](#)
- [Actualización de confluence.](#)
- [Tickets de Jira.](#)

# Que es Black Duck Hub?



## Descripción:

**Black Duck Hub es un sistema de chequeo de licenciamiento y propiedad intelectual para las librerías de terceros que usamos en nuestros proyectos.**

# Que es un Componente?



**Componente es como se conoce a las librerías dentro de Black Duck Hub.**

**Librería son esos archivos java que desplegamos en la carpeta LIB, son básicamente código reusable, en general se hacen con una tarea en mente.**

**Por ejemplo: Si hago una aplicación para un banco, en ella tendré varios métodos que podría reusar en el futuro para otras aplicaciones de bancos que desarrolle.**

**Puedo hacer una librería con estos métodos para facilitar su uso en otra aplicación y no tener que reescribir métodos.**

**NOTA: El acceso y uso a estas librerías depende de cada lenguaje, no me parece necesario explicar en mas detalle su significado.**

# Que es y por que Black Duck Hub?



## Que tiene que ver esto con Black Duck Hub?

Bueno, en una aplicación podemos usar librerías que hayamos creado nosotros, o librerías creadas por terceros.

La persona o personas que crearon dicha librería lo pueden haber hecho por amor al arte, o por dinero. Entonces cuando usamos la librería de un tercero tenemos que saber exactamente que conlleva hacerlo, si debemos pagar, o en algunos casos cumplir con ciertos requisitos, porque de no hacerlo debemos pagar multas.

Aquí entra Black Duck Hub, con esta herramienta nosotros enviamos al equipo legal las librerías que usa nuestra aplicación con la información legal que tenemos de las mismas, para que ellos comprueben que podemos usarla sin problemas, o en el caso de que debamos tomar acciones extras para usarla, asegurarse de que lo hacemos. Y evitar problemas graves en el futuro.



# Que es y por que Black Duck Hub?



**Estas librerías tienen una “Licencia” que fue elegida por el desarrollador, y debemos revisarla y cumplir con la misma.**

**Hay 3 tipos de licencia (hay muchos mas pero nos interesa estas 3):**

**Licencia Comercial:** Es una librería que fue echa específicamente para ganar dinero.

**Licencia Freeware:** Son librerías que se puede ejecutar, distribuir y estudiar en forma gratuita.

**Licencia Open Source Software:** También son librerías gratuitas pero a diferencia de Freeware, se puede hacer modificaciones, se tiene acceso al código fuente y se puede mejorar o adaptar a lo que uno necesita.

# Que es y por que Black Duck Hub?



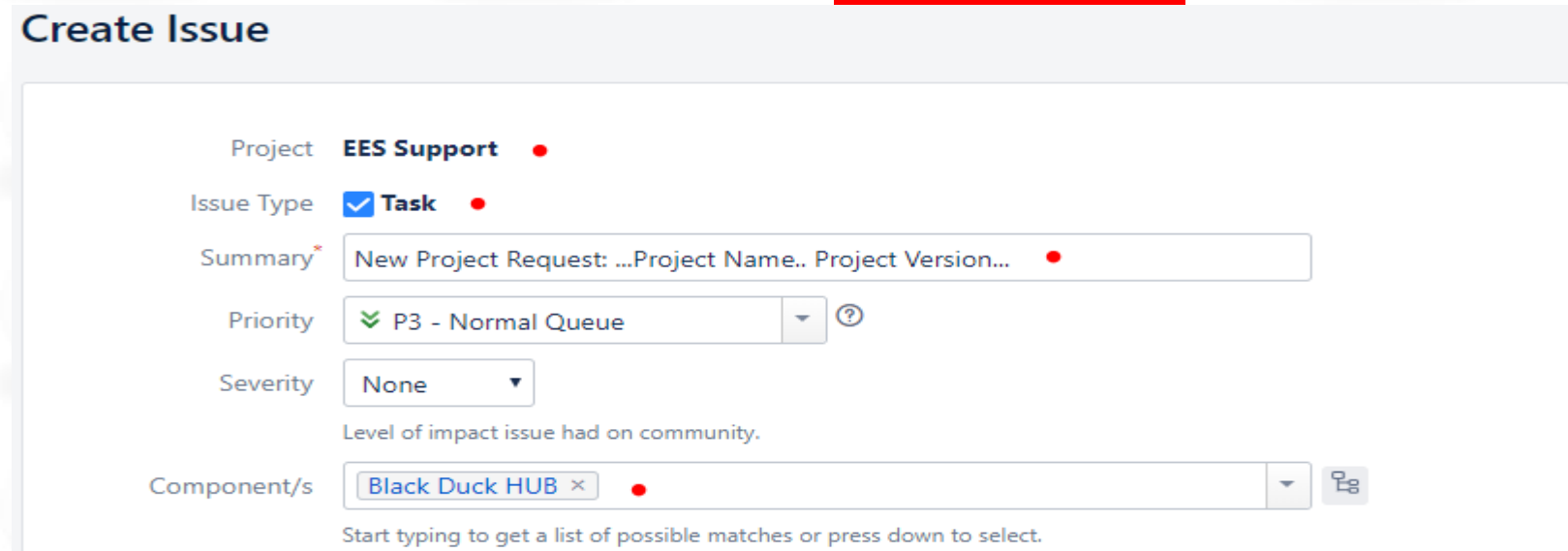
- Black Duck es una herramienta web de Avaya que se accede en: <https://blackduck.avaya.com/>
- Para el logueo en Black Duck Hub se utilizan las credenciales de Avaya GLOBAL.
- En Black Duck Hub las librerías se llaman “Componentes”

A screenshot of the Black Duck Login web form. The form is titled 'Black Duck Login' and contains two input fields: a username field with the text 'mmateos' and a password field with masked characters '.....'. Below the input fields are two buttons: 'Reset' and 'Login'.

# Crear un proyecto en Black Duck Hub.

La creación del proyecto se pide mediante un ticket de Jira.  
Es importante que la creación se pida únicamente después de que el plan Daily de bamboo este armado y funcionando correctamente.

Este enlace es a la creación del ticket: [Crear Ticket](#)

A screenshot of the 'Create Issue' form in Jira. The form is titled 'Create Issue' and contains several fields: 'Project' is set to 'EES Support'; 'Issue Type' is set to 'Task'; 'Summary' is 'New Project Request: ...Project Name.. Project Version...'; 'Priority' is 'P3 - Normal Queue'; 'Severity' is 'None'; and 'Component/s' is 'Black Duck HUB'. There are red error icons next to the Project, Issue Type, Summary, and Component/s fields. A help icon is next to the Priority dropdown. Below the Component/s field, there is a note: 'Start typing to get a list of possible matches or press down to select.'

Si se usa el enlace proporcionado mas arriba, solo deberá completarse Summary y Description.

# Pedir la creación del proyecto en Black Duck Hub.



## Summary:

- 🕒 **Project Name:** El nombre del proyecto debe ser igual al nombre en Forge. O sea, ir a Forge <https://forge.avaya.com/> buscar el proyecto y usar el mismo nombre.
- 🕒 **Versión:** La versión del proyecto puede ser cualquiera que uno decida, pero se intenta tener una nomenclatura razonable. Lo normal, es que la versión sea el nombre de la aplicación. Por lo que si el mismo cliente, por ejemplo DTV Project, tiene muchas aplicaciones, cada aplicación será una versión, como IVR Pagos, IVR futbol, etc. Si se hace cambios en una aplicación y se debe hacer un nuevo Black Duck en la misma, entonces se puede nombrar IVR Pagos V 1.0, IVR Pagos V 1.1, Etc. O poner el nombre de la app y la misma versión del Release.








# Pedir la creación del proyecto en Black Duck Hub.

## Descripción:

Description

Style

**B** *I* U A <sup>o</sup> <sub>p</sub>     + 

1. Project Name (Should exactly match with Forge Project) ...?:

2. Project Version?:

3. Do you have an older project which can be cloned to create the new project?:

Please furnish the following information also required for project creation:

Project Owner: :

Project Description:

What is your department / business unit? ..AGS/CC/DataSoln/CAE/DCA/EPT/Labs/GCS

Ops/GO/PS/SME/UC/UCP/VID:



What is the release date of expected version or release? ..YYYY/MM/DD:

What is the commit date of expected version or release? ..YYYY/MM/DD:

Do you want any component changes should be applicable across project? (Yes or No):

Visual

Text

# Pedir la creación del proyecto en Black Duck Hub.



## Descripción:

1. Project Name (Should exactly match with Forge Project) ...?: **Mismo que en el Summary.**
2. Project Version?: **Mismo que en el Summary.**
3. Do you have an older project which can be cloned to create the new project?: **Se puede usar cualquier aplicacion del cliente como base.**

**Esto ahorra tiempo al hacer el review, si el mismo cliente tiene otra aplicacion, usenla como base. Solo tomara los componentes que tiene la aplicacion que usan, los que no esten en la nueva aplicacion no se clonaran.**

**Please furnish the following information also required for project creation:**

**Project Owner:** **Martin Cespedes mcespedes (Siempre debe ser Tincho)**

**Project Description:** **Poner el nombre del Proyecto y de la aplicación.**

**What is your department / business unit?** **..AGS/CC/DataSoln/CAE/DCA/EPT/Labs/GCS**

**Ops/GO/PS/SME/UC/UCP/VID:** **EPT**

**What is the release date of expected version or release?** **..YYYY/MM/DD:**

**What is the commit date of expected version or release?** **..YYYY/MM/DD:**

**Pongan una o dos semanas en el futuro, no es necesario poner fechas exactas. Porque dependera de cuanto tarden en hacer review y cuanto tarde legales en aprobar los components, ni se fijan en esto.**

**Do you want any component changes should be applicable across project? (Yes or No):** **NO**

**En el template del ticket no esta esta ultima parte pero es importante:**

**Please create Bamboo Plan. Link: [http://bamboo.forge.avaya/\\*\\*\\*\\*\\*](http://bamboo.forge.avaya/*****)**

**Por ultimo Agregar enlace al plan Daily del proyecto en Bamboo.**

# Pedir la creación del proyecto en Black Duck Hub.

Finalmente hacer click en "Create":

Blocks None

Sprint

Jira Software sprint field

Create Cancel

Ejemplo



AVAYA EES EES Support / EDS-42286

## New Project Request: TSYS Version 1.0

Edit Comment Agile Board More

**Details**

Type: ☒ Task • Status: CLOSED

Priority: ☒ P3 - Normal Queue Resolution: Fixed

Component/s: Black Duck HUB •

Labels: None ✎

Severity: 2-High

This Issue Relates to: Project

**Parent/Child Hierarchy**

Toggle Hierarchy Toggle Work

%	SP	R	O	W	T	P	S	B	Issue	Summary
100.00	0	0.00	0.00	0.00					<span>EDS-42286</span>	New Project Request: TSYS Version 1.0

**Description**

1. Project Name (Should exactly match with Forge Project) ...?: **TSYS**
2. Project Version?: **1.0**
3. Do you have an older project which can be cloned to create the new project?: **NO**

Please furnish the following information also required for project creation:

Project Owner: : **Martin Cespedes mcespedes**

Project Description: **TSYS 1.0**

What is your department / business unit? ..AGS/CC/DataSoln/CAE/DCA/EPT/Labs/GCS Ops/GO/PS/SME/UC/UCP/VID: **EPT**

What is the release date of expected version or release? ..YYYY/MM/DD: **2019/07/16**

What is the commit date of expected version or release? ..YYYY/MM/DD: **2019/07/26**

Do you want any component changes should be applicable across project? (Yes or No): **NO**

**Bamboo link:**

<https://bamboo.forge.avaya.com/browse/TSYS>

# Escanear Source Code Repository.



**Hay dos opciones para crear el plan de Bamboo:**

**Lo hacemos nosotros:** No lo recomiendo, todavía hay problemas con esto, no siempre funciona, y se pierde tiempo encontrando el problema o pidiendo ayuda.

**Lo hace la gente de Black Duck Hub:** Esto es lo ideal, porque inmediatamente después de crear el proyecto, corren el plan, y nos ahorran mucho trabajo. Para esto, debemos aclarar al crear el ticket cual es el plan Daily de bamboo, el cual usaran de base para crear el plan de Black Duck Hub y correr el script de escaneo.

**NOTA:** NO lo configuran para que se ejecute automáticamente, lo dejan ejecutado una vez exitosamente y listo. Nosotros debemos configurarlo para que se ejecute una vez al día.



# Review

Es muy importante que se preste atención a lo que se hace. Auditoria es cada vez mas exigente con lo que declaramos y cómo lo declaramos.

**NOTA:** Pedir al desarrollador la lista de componentes de la aplicación. Solo se debe hacer review de los componentes que nos pase, y no de TODOS los detectados, en la sección de Ignorar veremos mas en detalle.

Como hacer el Review de un componente:

A la derecha del componente, elegir "Edit"



Operational Risk  
of Components



24 Snippets

Need Confirmation

Match Status Confirmed X

Ignore Not ignored X

Filter components...

Add Filter

Usage	License	Security Risk	Operational Risk	
Dynamically Linked	Public Domain		High	 
Dev. Tool / Excluded	Apache-2.0		High	 
Dev. Tool / Excluded	Apache-2.0		High	 
Dev. Tool / Excluded	Apache-2.0		High	 
Dev. Tool / Excluded	Apache-2.0		High	 

Edit  
Ignore  
Comment

# Review



Dentro de la ventana de edición, se debe cambiar según corresponda, el nombre, Versión, el origen y uso del componente, también si se desea se puede dar una descripción del propósito de dicho componente en nuestra app.

**NOTA:** En el caso de los IVR todos los datos que vienen por defecto, son los correctos, por lo que no debemos cambiar nada aquí. Pero aun así, por si acaso, verificar que "Usage" sea siempre "Dynamically Linked" Salvo que sea un componente que solo se usara en desarrollo pero aun asi se desplegara en el cliente, de ser asi se debe elegir "Dev.Tool/Excluded".

Edit Component

Component \*

Apache Commons Codec

Version

1.9

Origin ID

maven commons-codec:commons-codec:1.9

Usage

Dynamically Linked

Purpose

Modification

☐



> Additional Fields

Cancel

Update

# Review

Luego, se debe verificar los datos de la licencia del componente.

Page	License	Security Risk	Operational Risk		
ynamically Linked	 Public Domain		High		
v. Tool / Excluded	Apache-2.0 		High		
v. Tool / Excluded	 Apache-2.0		High		
v. Tool / Excluded	 Apache-2.0		High		

Al hacer click en el nombre de la licencia detectada se abre la ventana de licencias.

# Review

Spring Framework 3.1.1 Component License

☒ Include in Notices File Report

Attribution Statement ▾  
Copyright (c) 2002-2011 SpringSource, a division of VMware, Inc.

License  
Apache License 2.0 ▾

Apache License 2.0  
(Apache-2.0)

Apache License 2.0  
Status: Unreviewed | Family: Permissive

Required	Forbidden	Permitted
> State Changes	> Hold Liable	> Distribute
> Include License	> Use Trademarks	> Private Use
> Include Notice		> Modify

**En esta ventana se debe verificar:**

- **Que se use una sola Licencia.**
- **Se debe completar el Attribution Statement con el Copyright del componente, esto es mandatorio, si no esta, será rechazado.**



## Que hacer en el caso de que haya mas de una Licencia?

Jetty :: Servlet Handling 9.2.26.v20180806 1 Match Transitive Dependency Dev. Tool / Excluded Apache-2.0 and 1 more...

En este caso al ingresar a Licencia, verán 3 opciones, dejar las 2 licencias, elegir una u otra. Jamás dejen las dos licencias porque será rechazado, deben elegir una. Siempre que se de opción entre Sun GPL y CDDL , elegir CDDL porque GPL puede traer conflictos.

Jetty :: Servlet Handling 9.2.26.v20180806 Component License

☐ Include in Notices File Report

Attribution Statement ▾

License

(Apache License 2.0 AND Eclipse Public License 1.0)

(Apache License 2.0 AND Eclipse Public License 1.0)

Apache License 2.0

Eclipse Public License 1.0


State Changes

Hold Liab

Distribute

# Review: ¿Como obtener la información necesaria?

**Primer método:** Si no es su primer Black Duck Hub y ya tienen otros proyectos hechos, pueden hacer click en el nombre del componente, y los llevara a la pagina del mismo, en ella verán una lista de las aplicaciones donde ustedes hicieron review de ese mismo componente, pueden obtener la Licencia y Copyright de alguna de ellas.


<http://www.eclipse.org/jetty>  
**Jetty :: Servlet Handling** ▸ 9.2.26.v20180806  
unknown Versions: 280

**Description**  
 Jetty Servlet Container

<b>Released</b>	<b>Newer Versions</b>	<b>Status</b>
Aug 6, 2018	29	Unreviewed

**Where Used**

Project	Version	Released	Phase
ADVCAEPHY-[AdvantageCare-Physicians]	Advantage-Care-Physicians-IVR-1.0	Feb 29, 2020	In Planning
DATAMARK-[Datamark]	Datamark-WCMBP-IVR-1.0	Dec 9, 2019	Pre-release
EPTLEXISNE-[EPT-LexisNexis]	DSS-EVA-Module-for-DB-and-Context-Store-1.0	Never	Archived
TSYS-[EPT-TSYS-IVR]	Client-1	Jul 15, 2019	Archived
TSYS-[EPT-TSYS-IVR]	Client-2	Feb 4, 2020	In Planning

Displaying 1-5 of 5

# Review: ¿Como obtener la información necesaria?

**Segundo método:** Si no tienen este componente en ningún otro proyecto porque nunca le hicieron el review, pueden buscarlo en Google, y deben bajar el source del mismo. Una vez bajado lo abren con winzip o winrar y buscan dentro de sus archivos la información necesaria.



# Review: ¿Como obtener la información necesaria?

Mvnrepository.com es una buena pagina para encontrar el componente y su source.



**MVNREPOSITORY** Search for groups, artifacts, categories

Home » org.eclipse.jetty » jetty-servlet » 9.2.26.v20180806

**Jetty :: Servlet Handling » 9.2.26.v20180806**

Jetty Servlet Container

License	Apache 2.0 EPL 1.0
HomePage	<a href="http://www.eclipse.org/jetty">http://www.eclipse.org/jetty</a>
Date	(Aug 06, 2018)
Files	<a href="#">jar (113 KB)</a> <a href="#">View All</a>
Repositories	Central
Used By	1,875 artifacts

Hacen click en "View All" y les abre la pagina con los archivos para bajar.



# Review: ¿Como obtener la información necesaria?

En la pagina verán todos los archivos, deben elegir el que sea sources y .jar

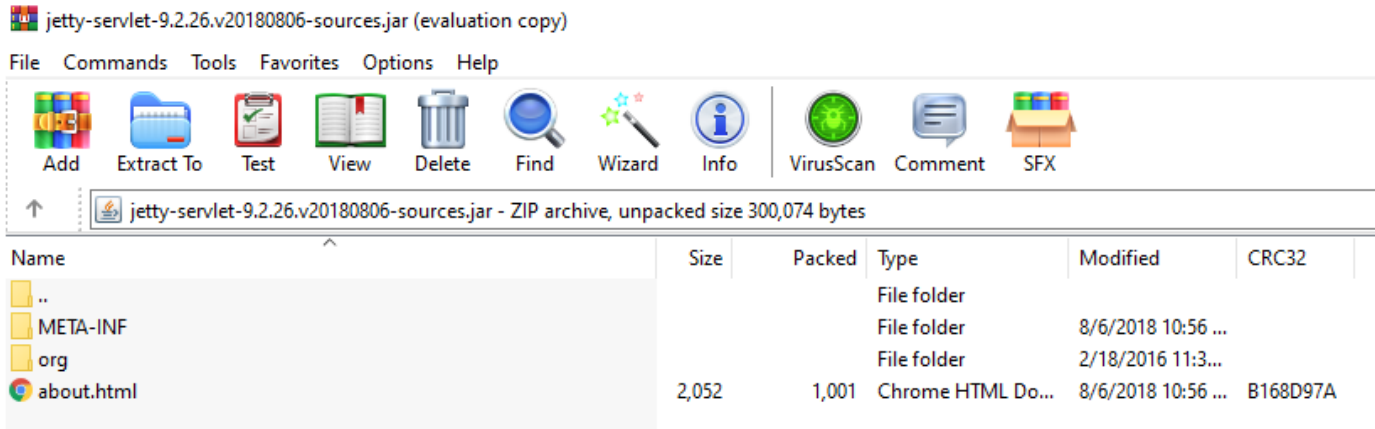
← → ↺ 🏠 🔒 repo1.maven.org/maven2/org/eclipse/jetty/jetty-servlet/9.2.26.v20180806/

## org/eclipse/jetty/jetty-servlet/9.2.26.v20180806

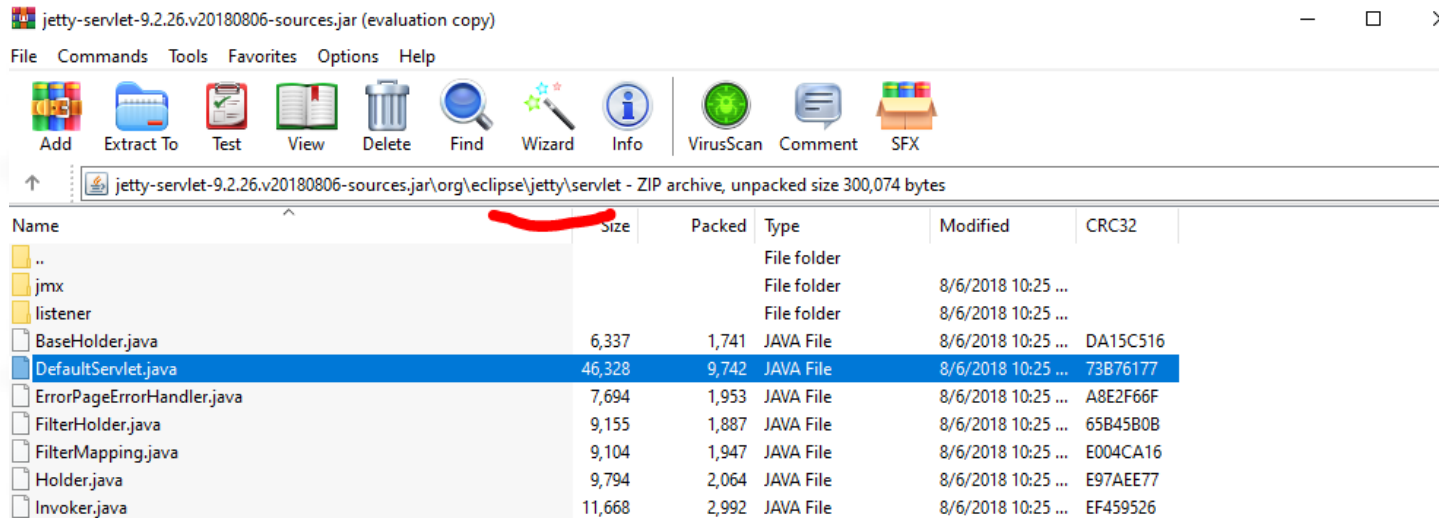
../		
<a href="#">jetty-servlet-9.2.26.v20180806-config.jar</a>	2018-08-06 15:56	606
<a href="#">jetty-servlet-9.2.26.v20180806-config.jar.asc</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-config.jar.asc...</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-config.jar.md5</a>	2018-08-06 15:56	32
<a href="#">jetty-servlet-9.2.26.v20180806-config.jar.sha...</a>	2018-08-06 15:56	40
<a href="#">jetty-servlet-9.2.26.v20180806-javadoc.jar</a>	2018-08-06 15:56	253599
<a href="#">jetty-servlet-9.2.26.v20180806-javadoc.jar.as...</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-javadoc.jar.as...</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-javadoc.jar.md...</a>	2018-08-06 15:56	32
<a href="#">jetty-servlet-9.2.26.v20180806-javadoc.jar.sh...</a>	2018-08-06 15:56	40
<a href="#">jetty-servlet-9.2.26.v20180806-sources.jar</a>	2018-08-06 15:56	66517
<a href="#">jetty-servlet-9.2.26.v20180806-sources.jar.as...</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-sources.jar.as...</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-sources.jar.md...</a>	2018-08-06 15:56	32
<a href="#">jetty-servlet-9.2.26.v20180806-sources.jar.sh...</a>	2018-08-06 15:56	40
<a href="#">jetty-servlet-9.2.26.v20180806-tests.jar</a>	2018-08-06 15:56	203173
<a href="#">jetty-servlet-9.2.26.v20180806-tests.jar.asc</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-tests.jar.asc...</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806-tests.jar.md5</a>	2018-08-06 15:56	32
<a href="#">jetty-servlet-9.2.26.v20180806-tests.jar.sha1</a>	2018-08-06 15:56	40
<a href="#">jetty-servlet-9.2.26.v20180806.jar</a>	2018-08-06 15:56	115713
<a href="#">jetty-servlet-9.2.26.v20180806.jar.asc</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806.jar.asc.asc</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806.jar.md5</a>	2018-08-06 15:56	32
<a href="#">jetty-servlet-9.2.26.v20180806.jar.sha1</a>	2018-08-06 15:56	40
<a href="#">jetty-servlet-9.2.26.v20180806.pom</a>	2018-08-06 15:56	3283
<a href="#">jetty-servlet-9.2.26.v20180806.pom.asc</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806.pom.asc.asc</a>	2018-08-06 15:56	801
<a href="#">jetty-servlet-9.2.26.v20180806.pom.md5</a>	2018-08-06 15:56	32
<a href="#">jetty-servlet-9.2.26.v20180806.pom.sha1</a>	2018-08-06 15:56	40

# Review: ¿Como obtener la información necesaria?

Luego de bajarlo, lo abren con Winzip o Winrar, o cualquier otro descompresor.



Exploran las carpetas del jar buscando archivos que contengan la licencia y el copyright que usa



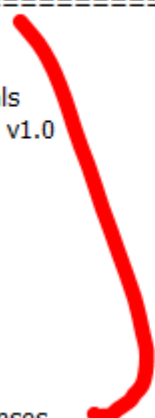
# Review: ¿Como obtener la información necesaria?

**Finalmente en los archivos.java encontré lo siguiente:**

View - BaseHolder.java

File Edit View Help

```
//  
// =====  
// Copyright (c) 1995-2018 Mort Bay Consulting Pty. Ltd.  
// =====  
// All rights reserved. This program and the accompanying materials  
// are made available under the terms of the Eclipse Public License v1.0  
// and Apache License v2.0 which accompanies this distribution.  
//  
// The Eclipse Public License is available at  
// http://www.eclipse.org/legal/epl-v10.html  
//  
// The Apache License v2.0 is available at  
// http://www.opensource.org/licenses/apache2.0.php  
//  
// You may elect to redistribute this code under either of these licenses.  
// =====  
//
```

A red bracket is drawn on the right side of the code block, spanning from the first line of the license section to the last line, highlighting the entire license information.

**Con esto se que puedo usar cualquiera de las dos licencias, cuando Apache es una opción siempre elijan Apache, porque es la que seguro no tendrán ningún conflicto. Y también ahí mismo esta el Copyright. Lo copian y listo.**

**Nota: A partir de ahora, una vez este aprobado este componente, ya les aparecerá en la lista de componentes que hicieron review y ya harán el review con el primer método.**

# Review: ¿Como obtener la información necesaria?

**Tercer método:** Este método es muy nuevo, todavía esta en Beta y no funciona la mayoría de las veces, pero ira mejorando por lo que en un futuro no muy lejano será el método preferido.

Se agregó (en los proyectos nuevos que pidan, no lo verán en los viejos, no es retroactivo) un martillo con un signo + junto a la licencia de algunos componentes.


Esto quiere decir que si hacen click abrirá una ventana con licencias atribuidas a ese componente:

Component ^	Source	Match Type	Usage	License
✓ Apache log4j 1.2.16	4 Matches	Direct Dependency, Transitive Dependency, Exact Directory	Dynamically Linked	M Apache-2.0 + ↕
✓ ... 1.5.7	4 Matches	Direct Dependency, Transitive Dependency, Exact Directory	Dynamically Linked	MIT + ↕



# Review: ¿Como obtener la información necesaria?

Hacen click en la flecha de la Licencia que corresponde a su componente, en este caso era Apache License 2.0



EPTHSBC-[EPT-HSBC] IVR-Seguros-1.0  
**Apache log4j ▸ 1.2.16**

License Declared License: Apache License 2.0 | 1 Policy Violation

Activate Deactivate

License ^	Active	License Family	Status	Last Updated
> Apache License 1.1	✓	Permissive	Unreviewed	May 14, 2020 by System User
> <b>Apache License 2.0</b>	✓	Permissive	Unreviewed	May 14, 2020 by System User
> Common Development and Distribution License 1.0	✓	Weak Reciprocal	Unreviewed	May 14, 2020 by System User
> Common Public License 1.0	✓	Weak Reciprocal	Unreviewed	May 14, 2020 by System User

## Y muestra archivos relacionados:


EPTHSBC-[EPT-HSBC] IVR-Seguros-1.0  
**Apache log4j ▸ 1.2.16**

License Declared License: Apache License 2.0 | 1 Policy Violation

Activate Deactivate

License ^	Active	License Family
> Apache License 1.1	✓	Permissive
▼ <b>Apache License 2.0</b>	✓	Permissive
<b>maven/log4j:log4j:1.2.16</b>	✓	
> Common Development and Distribution License 1.0	✓	Weak Reciprocal
> Common Public License 1.0	✓	Weak Reciprocal

## Review: ¿Como obtener la información necesaria?

Finalmente les muestra la siguiente pagina donde estarán los archivos relacionados y podrán ver el copyright, como dije es beta, así que no pude encontrar uno que me muestre algo, nos mostraron en una reunión como funcionaba, pero no tengo screenshots. Solo se queda loading en mis proyectos. En el futuro seguramente lo mejoraran y usaremos mas que nada este método.

### Reference Files

Apache License 2.0

Status: Unreviewed | Family: Permissive

Origin

 maven/log4j:log4j:1.2.16

### Files

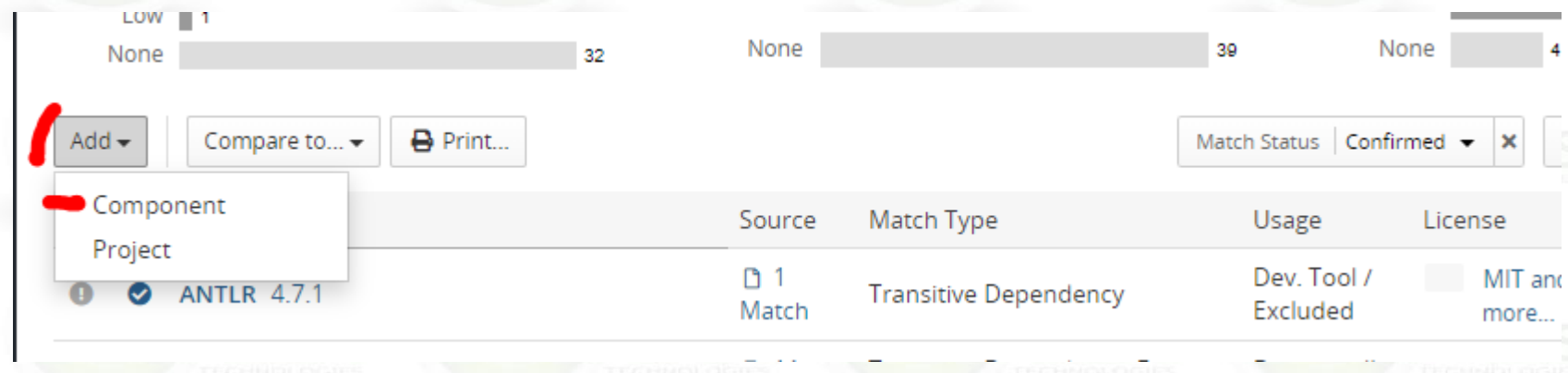


Loading...

## Review: Agregar componentes no detectados

Puede suceder que algún componente de la aplicación no sea detectado por el escaneo automático del plan de Bamboo.

En este caso se debe agregar manualmente de la siguiente manera:



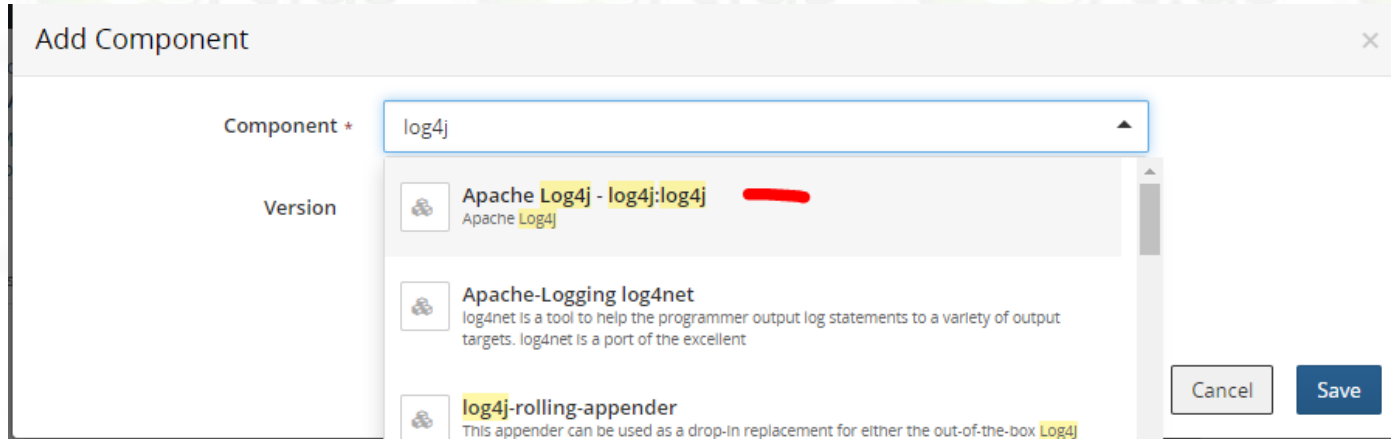
The screenshot shows the Certius interface with a table of components. The 'Add' button is highlighted with a red bracket. The table has columns for Component, Source, Match Type, Usage, and License. A dropdown menu is open for the 'Add' button, showing 'Component' and 'Project' options. The table lists one component: ANTLR 4.7.1, which is a Transitive Dependency, used as a Dev. Tool / Excluded, with a MIT license.

Component	Source	Match Type	Usage	License
ANTLR 4.7.1	1 Match	Transitive Dependency	Dev. Tool / Excluded	MIT and more...

Se elige agregar componente.

# Review: Agregar componentes no detectados

Luego, en la ventana que aparece, se escribe el nombre del componente que se desea agregar



Add Component

Component \*

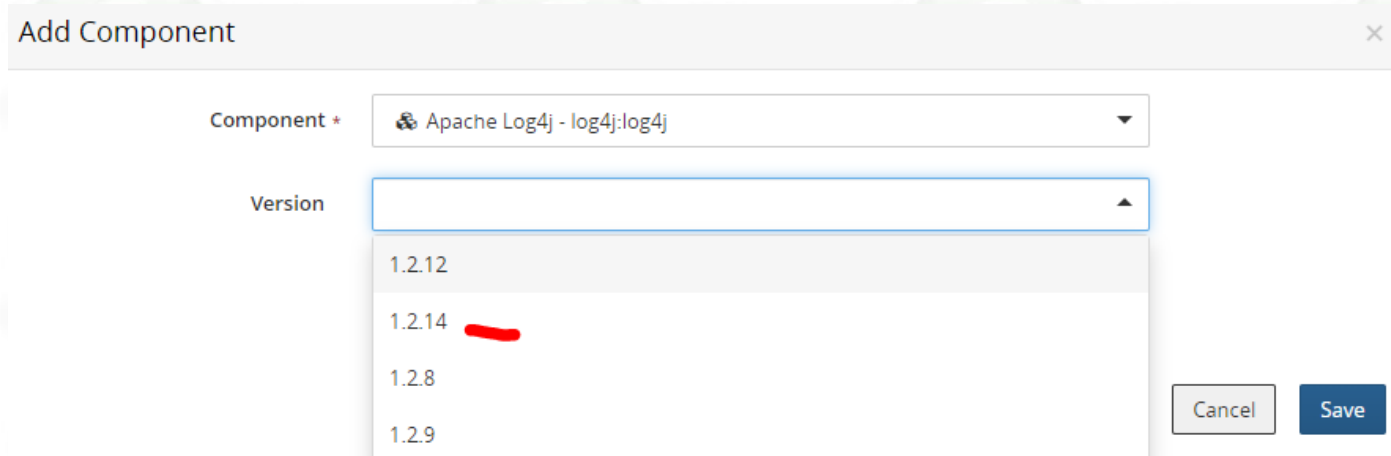
log4j

Version

- Apache Log4j - log4j:log4j
- Apache-Logging log4net
- log4j-rolling-appender

Cancel Save

Luego, se elige la versión, y se hace click en "Save".



Add Component

Component \*

Apache Log4j - log4j:log4j

Version

- 1.2.12
- 1.2.14
- 1.2.8
- 1.2.9

Cancel Save



## **Review: Agregar componentes no detectados**



**El componente ya deberá aparecer en la lista de componentes.**

**NOTA:** Puede pasar que el componente no aparezca para ser agregado, de ser así, es porque no está en la base de datos de Black Duck Hub, para resolverlo hay que crear un ticket de Jira a los administradores de Black Duck Hub para que lo agreguen.

# Review: Ignorar componentes.

**Primero que nada, debemos hacer una distinción muy importante. El script de Black Duck Hub detecta varios niveles de dependencias de los componentes, y nosotros solo hacemos review de los componentes que usamos, NO de sus dependencias.**

**Por lo que a la hora de ignorar hay dos tipos de componentes que ignoraremos. Primero: Los que ignoramos porque no usamos y son dependencias y Segundo: los que si usamos pero debemos ignorar por algún motivo. Primero: Como dije, serán estas dependencias que nosotros no usamos. Para ignorarlas simplemente vamos al margen derecho y hacemos click en la misma flecha que es para editar, pero en este caso elegimos "Ignore".**

	Usage	License	Security Risk	Operational Risk	
	Dev. Tool / Excluded	<input type="checkbox"/> MIT and 1 more...	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	Medium	 
act	Dynamically Linked	<input type="checkbox"/> Apache-2.0	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	H	 
	Dev. Tool / Excluded	<input type="checkbox"/> Apache-2.0	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	Medium	 

- Edit
- Ignore
- Comment

## **Review: Ignorar componentes.**

**Segundo: Puede pasar que haya componentes que no queremos se haga Review por algún motivo.**

**En este caso es mas complicado, porque se debe hacer un procedimiento especifico para ignorar dichos componentes.**

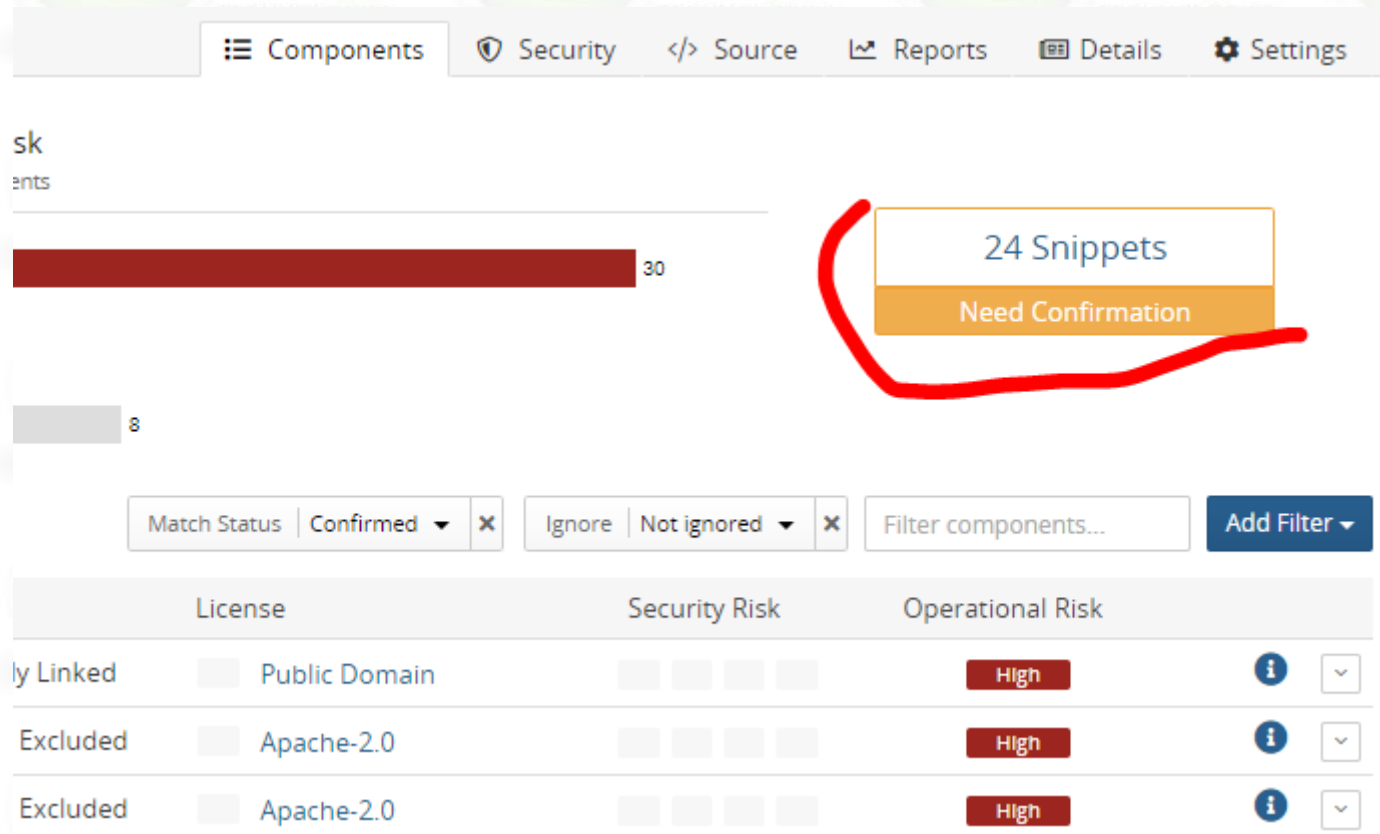
**El proceso dependerá del tipo de licencia que tenga dicho componente.**

**No es algo que nos pueda suceder por lo que no entrare en mas detalle al respecto. Si se desea hacer esto o al menos averiguar como se hace, pueden leer esta pagina:**

**[Cuando y como ignorar un componente](#)**

# Review: Snippets.

Los snippets son partes reusables de código. Osea el famoso copy paste de código que usamos.



The screenshot shows the Black Duck Hub interface with a navigation bar at the top containing 'Components', 'Security', 'Source', 'Reports', 'Details', and 'Settings'. Below the navigation bar, there are two horizontal bars: a red one labeled '30' and a gray one labeled '8'. A red circle highlights a box containing '24 Snippets' and 'Need Confirmation'. Below this, there are filter controls for 'Match Status' (Confirmed), 'Ignore', and 'Not ignored', along with a 'Filter components...' button and an 'Add Filter' button. The main table displays the following data:



	License	Security Risk	Operational Risk	
ly Linked	Public Domain	<div><div></div><div></div><div></div><div></div></div>	High	<div><div></div><div></div></div>
Excluded	Apache-2.0	<div><div></div><div></div><div></div><div></div></div>	High	<div><div></div><div></div></div>
Excluded	Apache-2.0	<div><div></div><div></div><div></div><div></div></div>	High	<div><div></div><div></div></div>

El escaneo de Black Duck Hub, también detecta snippets dentro de los componentes.



## Review: Snippets.

Estos snippets pueden tener derecho de autor, por lo que si encuentra alguno sospechoso en un componente lo marcara para que hagamos review del mismo. Para ello hacen click en "Need Confirmation"

Name	Component	Match Type
 AEntity.java	Apache Geronimo 1.2.0-beta	 1 Snippet
 ArrayOfgea_boleta.java	Apache Geronimo 1.2.0-beta	 1 Snippet
 ArrayOfgea_nosnet2.java	Apache Geronimo 1.2.0-beta	 1 Snippet

En la pagina verán el nombre del componente donde se detecto el snippet. Nosotros NUNCA modificamos componentes, por lo que jamás tendremos snippets en ellos. Siempre se deben elegir todos los snippets e ignorarlos. Pero, por las dudas, pueden preguntarle al desarrollador si modifico algún componente, de ser así, él mismo debe entrar aquí y ver si algún snippet corresponde a su modificación.

# Monitorear Componentes Con JIRA

**Esto esta deprecado asi que lo hare corto, cuando uno hacia review en Black Duck Hub se creaba un ticket de jira por cada componente, que luego legales aprobaba o pedía mas info y uno respondía hasta que finalmente quedaba aprobado.**

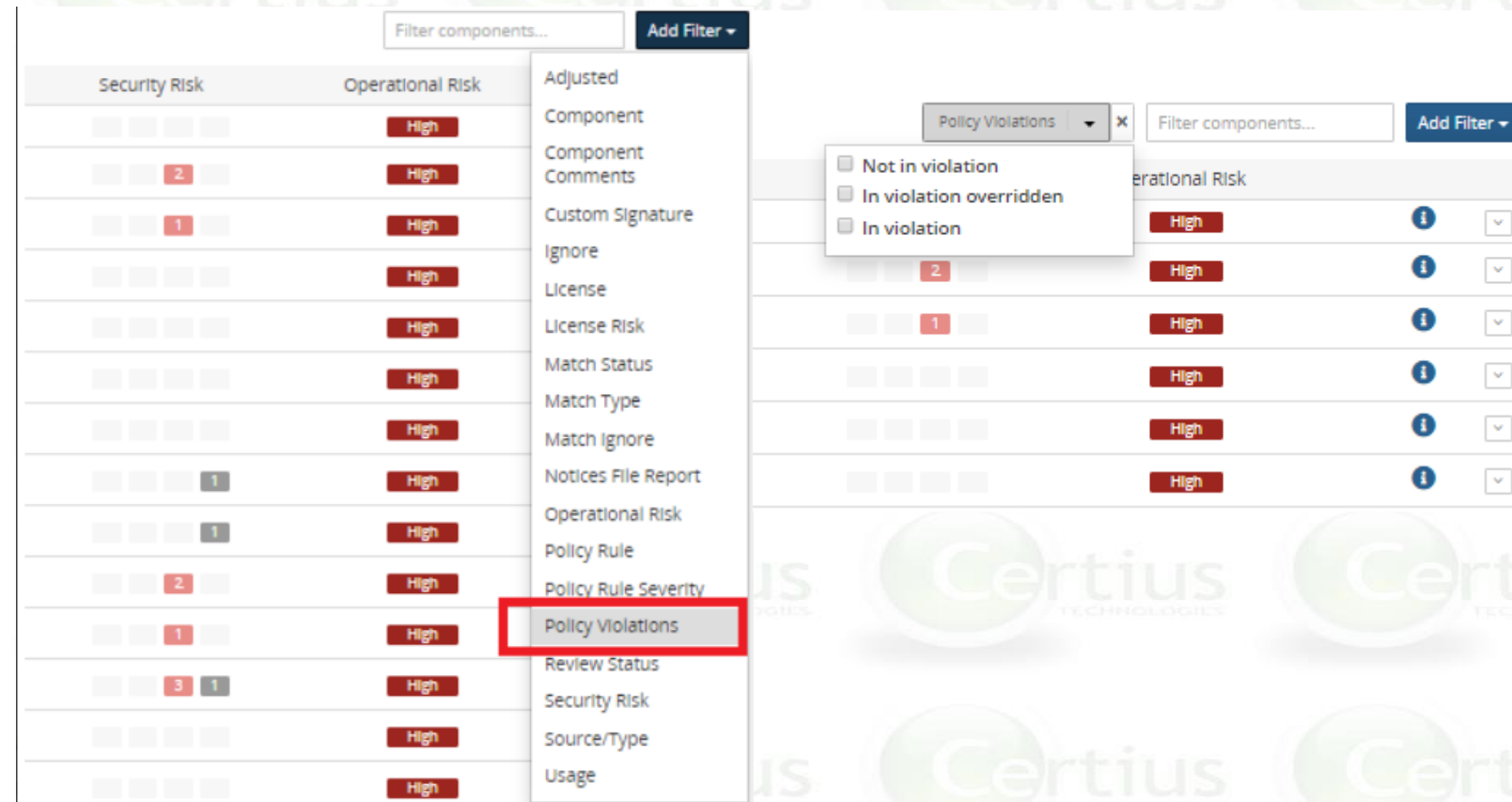
**Esto ya NO será así en adelante. Lo comento aquí porque los proyectos que ya funcionaban de esta manera lo seguirán haciendo, pero todos los proyectos nuevos no tendrán integración con Jira.**

**Por lo que no veo necesario explicar como funciona porque ninguno creara un proyecto en que se deba usar este método.**

**Así que pasemos directamente a Monitorear Componentes Sin integración con Jira**

# Monitorear Componentes Sin JIRA

Este procedimiento es nuevo, primero van a Filter y filtran por Policy violations y seleccionan "In violation"

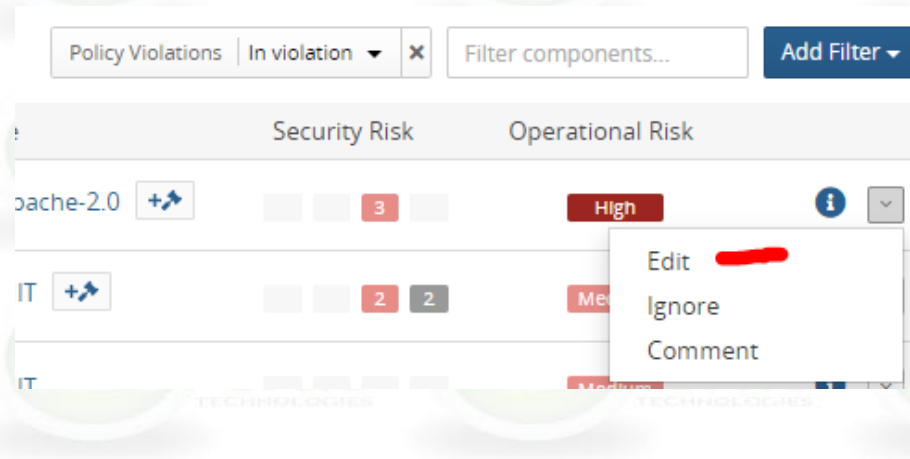


The screenshot displays the Certius interface for monitoring components. A dropdown menu is open, showing various filter options. The 'Policy Violations' option is highlighted with a red box. The background shows a table of components with 'Security Risk' and 'Operational Risk' columns. The 'Operational Risk' column shows 'High' for all components. The 'Security Risk' column shows various counts (e.g., 2, 1, 1, 2, 1, 3, 1).

Security Risk	Operational Risk
	High
2	High
1	High
	High
	High
	High
	High
1	High
1	High
2	High
1	High
3 1	High
	High
	High

# Monitorear Componentes Sin JIRA


Luego seleccionan Edit, en la esquina derecha de cada componente.



En la pagina de editar componente hacen click en Additional Fields



**Edit Component**

Component \*  Apache log4j

Version 1.2.16

Origin ID maven log4j:log4j:1.2.16

Usage Dynamically Linked

Purpose

Modification ☐

[Additional Fields](#)



# Monitorear Componentes Sin JIRA

Aquí es donde Legales se comunicara con nosotros.

**“Component type” lo completa legales, 99% de los casos es open source**



**“Notes from and to Legal and GSO”, e donde nos pedirán que hagamos cambios y donde les responderemos siempre con el handle + la fecha, y luego el comentario.**



**“Component Approval Status”, legales pone el status actual, nosotros solo lo modificamos de “Need Info” a “Open” en el caso de que agreguemos información que se nos solicitó en Notas.**



Edit Component

Component Type

Component Type (Open Source, Commercial, Freeware)  
☐ Commercial  
☐ Freeware  
☐ Open Source

Notes from and to Legal and GSO

Communication between teams. Notes instructions: <Global Handle> -- <Date> -- <Comments>  
Text Area

Component Approval Status

Status from Legal to Development Team  
☐ Need Info  
☐ Rejected  
☐ Approved  
☐ Open

GSO Conditions of Use

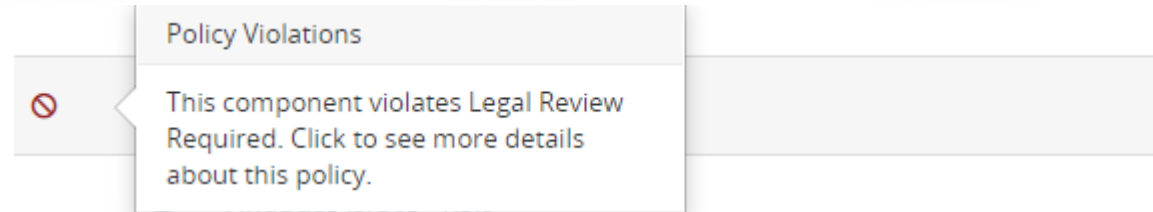
GSO Conditions of Use  
Text Area

Cancel

Update

# Monitorear Componentes Sin JIRA

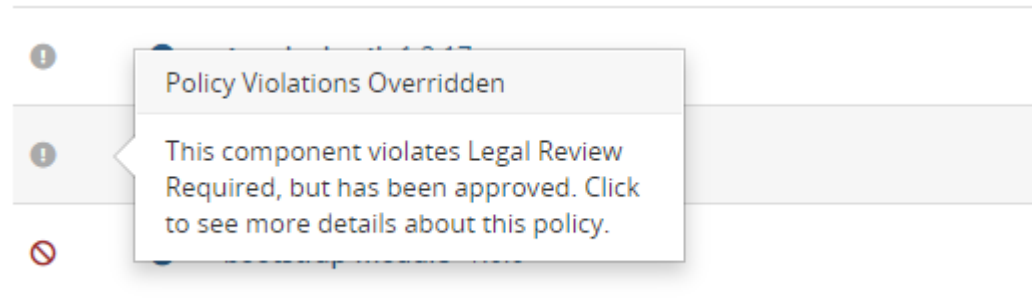
**Finalmente si el componente fue aprobado pasará de "In Violation" a "In Violation Overridden"**

A screenshot of a software component's status. On the left, there is a red circle with a white 'X' icon. To its right, a tooltip box is displayed with the title 'Policy Violations' and the text 'This component violates Legal Review Required. Click to see more details about this policy.' The component name 'Guava: Google Core Libraries for Java' is partially visible to the right of the tooltip.

Policy Violations

This component violates Legal Review Required. Click to see more details about this policy.

! ✓ Guava: Google Core Libraries for Java 11.0.2

A screenshot of a software component's status. On the left, there is a green circle with a white checkmark icon. To its right, a tooltip box is displayed with the title 'Policy Violations Overridden' and the text 'This component violates Legal Review Required, but has been approved. Click to see more details about this policy.' The component name 'Guava: Google Core Libraries for Java' is partially visible to the right of the tooltip.

Policy Violations Overridden

This component violates Legal Review Required, but has been approved. Click to see more details about this policy.

# Script de Monitoreo



Se puede monitorear componentes usando este script de Python. Todavía nunca lo use, porque es muy reciente y además no me parece que valga la pena si uno tiene menos de 100 componentes. Pero dejo el proceso aquí:

## Requerimientos:

Python 3.6 o superior.

Paquetes de Python *xlsxwriter* y *urllib3*.

Luego de bajar e instalar Python, van a consola y escriben “pip install xlsxwriter” y luego “pip install urllib3”

# Script de Monitoreo

Como ejecutarlo:

En la misma carpeta donde esta el script (archivo adjunto) ejecutar este comando:

***# python BlackDuck\_Hub\_Legal\_NeedInfo\_Rejected\_Status\_Components.py <hubName>***

Se generara un archivo Excel que obtiene toda la información de todas las versiones del proyecto. Pero solo los componentes que esten "in violation" y que están en estado "need info" o "rejected".

	A	B	C	D	E	F	G	H	I
1	Project Name	Project Version	Component Name	Component Version	Component Type	Legal and GSO Notes	Component Status	Policy Violation	Approval Violation
2	SOCDDSS-[State-of-Connecticut-DSS]	1.0	AOP Alliance (Java/J2EE AOP standard)	1.0	Open Source	The HUB indicates it is public domain licensed, but please go into the source code header files to confirm the license text. Does it dedicate the software to the public domain? And to confirm this please provide the license text in the license text box.	Need Info	IN_VIOLATION	IN_VIOLATION
4	SOCDDSS-[State-of-Connecticut-DSS]	1.0	Apache Neethi	2.0.4	Open Source	HUB is indicating that this file is been modified. Could you please confirm about the modification? - Kavya Bandaru	Need Info	IN_VIOLATION	IN_VIOLATION
5	SOCDDSS-[State-of-Connecticut-DSS]	1.0	axiom-impl	1.2.8	Open Source	License text is missing in the license text box. Please do provide license text. - Kavya Bandaru	Need Info	IN_VIOLATION	IN_VIOLATION
6	SOCDDSS-[State-of-Connecticut-DSS]	1.0	Backport JSR 166	3.1	Open Source	Please look in the source code header file for the exact license text. The license text is missing. YDT	Need Info	IN_VIOLATION	IN_VIOLATION
7	SOCDDSS-[State-of-Connecticut-DSS]	1.0	com.springsource.java.vax.mail	1.4.0	Open Source	The Match Type column says files were modified. Did Avaya modify these files? If yes, are the modifications separate from the Avaya proprietary code? Any modifications written by Avaya would have to be	Need Info	IN_VIOLATION	IN_VIOLATION
	SOCDDSS-[State-of-Connecticut-DSS]	1.0	Java Servlet API	4.0.0	Open Source	The component is shown as dual licensed under CDDL and GPL with class path exception 2.0. Could you please confirm about the license and copyright notice from the source	Need Info	IN_VIOLATION	IN_VIOLATION



# Script de Monitoreo



Script



BlackDuck\_Hub\_Legal\_NeedInfo\_Rejected\_Status\_Components-EPT.py

## Known Issues:

### 1. Despues de correr el script reciben el siguiente mensaje:

```
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after connection broken by 'SSLError(SSLCertVerificationError(1, '[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed: unable to get local issuer certificate (_ssl.c:1108)'))': /simple/xlsxwriter/
```

**Solución:** Desconectarse de la VPN y probar de nuevo.

### 2. Despues de correr el script reciben el siguiente mensaje:

```
Traceback (most recent call last): File "BlackDuck_Hub_Legal_NeedInfo_Rejected_Status_Components-EPT.py", line 1, in  
<module> import requestsModuleNotFoundError: No module named 'requests'
```

**Solución:** Instalar librería "requests". Ejecutar *"pip install requests"*

# Pedir PDF de licencias

Finalmente, una vez este todo aprobado (Verificar bien que todo este aprobado) pedimos el PDF de licencias y copyright.  
Se hace mediante un ticket de JIRA de la siguiente manera:

Project: **EES**


Issue Type: **Task**

Summary: **License File for Project ...**

Component/s: **Black Duck Hub**

Description: **Please provide license file for this Black Duck Hub Project, all components has been approved: proyecto y versión. <URL al proyecto en BDH>.**


## Create Issue

Project\*  EES Support (EDS) ▼

Issue Type\* ☒ Task ▼ ?



---

Summary\* License File for Project ...

Priority  P3 - Normal Queue ▼ ?






Severity None ▼

Level of impact issue had on community.

Component/s  Black Duck HUB × ▼ 

Start typing to get a list of possible matches or press down to select.

## Description

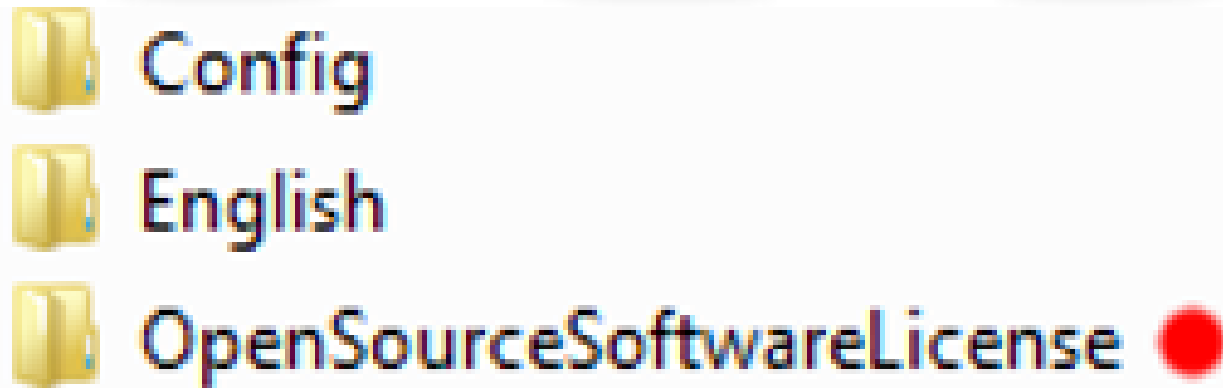
Style ▼ **B** *I* U A ▼  ▼  ▼    ▼ + ▼ ^

Visual Text

# Desplegar PDF de licencias



Una vez recibido el PDF debe ser copiado al Application Server donde están las aplicaciones , en un folder llamado "OpenSourceSoftwareLicense" dentro de la carpeta DATA. También debe subirse al repositorio.



# Agregar PDF de licencias en Confluence

**En confluence debe agregarse la siguiente información.**

**Black Duck Hub Project Name:**

**Black Duck Hub Version:**

**Link to Black Duck Hub Bamboo plan:**

**Licence and Copyright PDF file:**

**Ejemplo:**

## **Black Duck Hub**

Black Duck Hub Project Name: ARTARJNAR-[Tarjeta-Naranja-Argentina-Project]

Black Duck Hub Version: ARTARJNAR-IVR-Denuncias-1.0

Link to Black Duck Hub Bamboo plan: <https://bamboo.forge.avaya.com/browse/ARTARJNAR-BDH>

License and Copyright PDF file: ARTARJNAR-[Tarjeta-Naranja-Argentina-Project]-ARTARJNAR-IVR-Denuncias-1.0.pdf

# Cerrar ticket de Black Duck Hub



**Seguramente Guillermo les creo un ticket tipo Task pidiendo que hagan el Black Duck Hub del proyecto (sino, pregunten), una vez hayan agregado el PDF en confluence. Resuelven el ticket poniendo como comentario la URL a la página de confluence donde esta la información de Black Duck Hub y se lo asignan a la persona que se los asigno a ustedes.**



# The End?



**Felicidades!!! completaron Black Duck Hub...**

**Les prometo que después de hacer varios proyectos, les resultara igual de tedioso, pero si les sirve de consuelo, solía ser mucho peor...**

**Y de a poco, están mejorando el proceso.**

**Iré actualizando la información a medida que sea necesario.**