

Protex Identify 방법

July - 2016

Mobile Open Source
(opensource.m@samsung.com)



Contents

I. Identify 개요

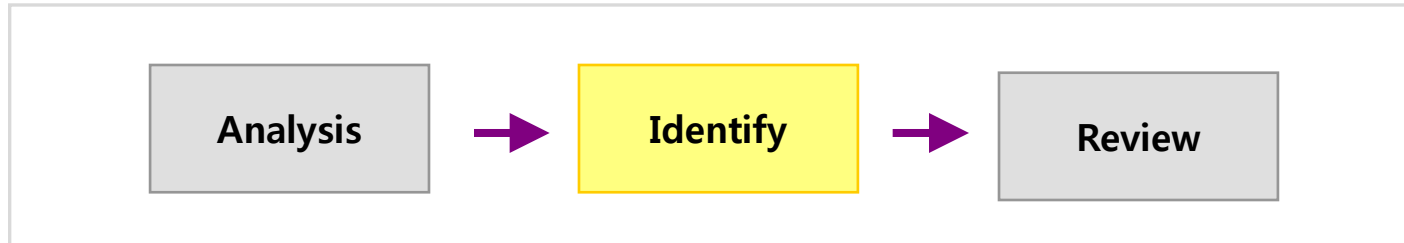
II. Tool 사용법

III. How To Identify

- 1) Comment 작성
- 2) Identify Step FAQ
- 3) 오픈 소스 검증 전반 FAQ

Identify 개요

개요 Identify 란?



■ Analysis 단계

- Identify 이전 단계
- 분석대상인 개발 소스와 Protex DB내의 오픈 소스를 비교하여 code match 여부를 분석
- Analysis 수행이 완료되면, 오픈 소스가 포함된 것으로 추측되는 파일(개발소스)이 검출됨

■ Identify 단계

- Analysis 단계에서 검출된 **파일의 오픈 소스 출처를 확인하는 단계**
- 출처 확인이란? Code가 일치하는 부분이 **어느 오픈 소스에서 가져와서 사용한 것인지** 확인하는 것
- 해당 개발 소스에 대하여 가장 잘 아는 사람(개발자)이 판별하는 것이 정확하다.
- License 에 대한 지식은 요구되지 않음

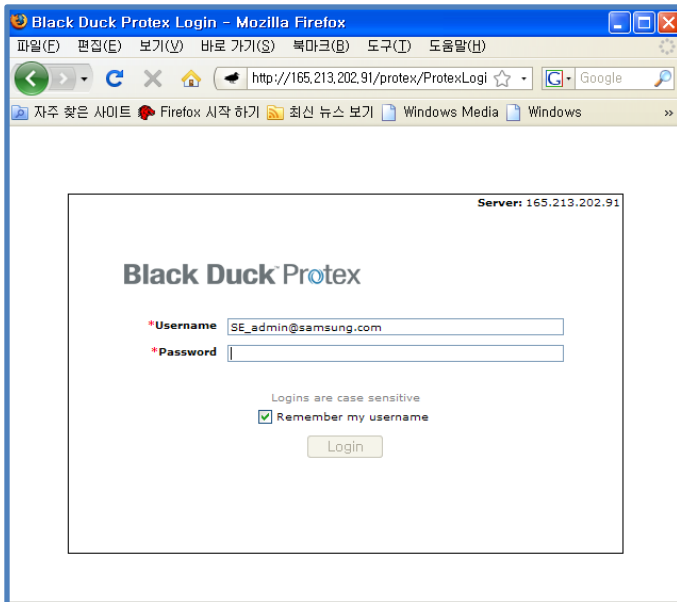
Tool 사용법

Tool 사용법 Server 접속

- Web Browser를 이용하여 Protex Server URL을 입력합니다.

IP Address	
Server 1	http://165.213.149.98
Server 2	http://165.213.202.91
Server 3	http://165.213.202.93
Server 4	http://165.213.149.172
Server 5 (For Tizen)	http://165.213.149.171
Server 6 (For Package)	http://165.213.149.173

- 부서별로 제공해드린 ID(xxx@samsung.com)와 PW를 입력합니다.



< Tip >

1. 브라우저는 **Mozilla Firefox** 또는 **Chrome** 을 권장합니다.
: Internet Explorer 보다 Mozilla Firefox 또는 Chrome 에서 수행속도가 훨씬 빠름
2. ID / PW 입력 아래에 위치한 **V Remember my username** 에 **체크**해 놓으면 다음 번부터는 다소 편하게 접속 하실 수 있습니다.

Tool 사용법 Start step

- Identify 대상 Project 를 선택합니다.

Black Duck Protex SE_admin@samsung.com

My Protex | Manage | Identify | Review | Report [Tools](#) | [Help](#) | [Logout](#)


Server: 165.213.202.91
Server Version: 5.2.3

Select a Project

Search: aries

81 Projects (1 - 30 [30](#) per page) [Jump...](#) | [First](#) | [< Prev](#) | [Next >](#) | [Last](#)

Project Name	Code Matches	String Searches	Dependencies	Pattern Matches	Pending Review	Approved	Disapproved	License Conflicts	Declared	No Discoveries	Last Analyzed
<input type="checkbox"/> 0000V6ZW9A3GAU-MR-1-11-Android-Aries EUR OPEN-Android(Gingerbread)	0	0	0	0	826	3,774	70	4,034	4	950	January 21, 2011 07:23:25 PM
<input type="checkbox"/> 0000V6ZW9A3GAU-MR-1-11-Android-2 OPEN-BootLoader(Gingerbread)	0	0	0	0	429	3	2	309	188	239	January 20, 2011 07:49:26 PM
<input type="checkbox"/> 0000V6ZW9A3GAU-MR-1-11-Android-3 OldScript(Gingerbread)	0	0	0	2	5	0	3	499	0	145	January 20, 2011 07:33:42 PM

click 

Tool 사용법 Manage step

- Manage 단계는 건너뛰고 바로 Identify 단계로 넘어갑니다.

Black Duck Protex

SE_admin@samsung.com

My Protex **Manage** Identify Review Report

Current Project: 0000V6ZW9A3GAU-MR-1-11-Android-Aries EUR OPEN-Android(Gingerbread)

Tools Help Logout

Server: 165.213.202.91

Server Version: 5.2.3

Project Status: 0 826 3,774 70 4,034

click

Project Settings Analysis Identification Patterns Users

Information

*Project Name: 0000V6ZW9A3GAU-MR-1-11-Android-Aries EUR OPE

Description: <Clone history>
GalaxyS EUR 2nd
date: 2011/01/20

License

License: Samsung Proprietary License (view)

No License

Code Label Information

Furnished By: Samsung Electronics.

Source Code URL:

내용 수정 ❌



Manage 단계 는

Admin 사용자가 Project를 생성하고 Policy Setting을 하기 위하여 설정 하는 부분으로, 기본사용자는 바로 Identify 단계로 넘어갑니다.

Tool 사용법 Identify step - 1

- 좌측의 Source Tree를 열어 Folder/File 명이 굵은 글씨로 표시된 파일을 선택합니다.

Black Duck Protex SE_admin@samsung.com

My Protex Manage **Identify** Review Report

Current Project: [test]-2-11-MMP-Trump RU SERC [C3332]-ThirdPartySource
Project Status: 75% Folders -

Tools Help Logout
Server: 165.213.202.91
Server Version: 5.2.3

Show: Pending Identification - ALL
New Filter...
Expand Collapse
 Only Show Matches

Folder: [test]-2-11-MMP-Trump RU SERC [C3332]-ThirdPartySource/3p_lwip_ops

11 Components (30 per page) | First | < Prev | Next > | Last

Approved	License Conflict	Component Type	Component	Version	License	Usage	Ship Status	# Code Match	# Depends	# Search	Used By
N/A	N/A		[test]-2-11-MMP-Trump RU SERC [C3332]-ThirdPartySource	Unspecified	Samsung Proprietary License	Original Code	Ship	2	0	0	
✓		LP	c-ares Library	Unspecified	MIT License V2 [Reviewed and Approved]	Component	Ship	1	0	0	
		FD	Capros-Omapl138	Unspecified	Unspecified	Component	Ship	1	0	0	
		LP	Copyright Massachusetts Institute of Technology	Unspecified	[template] Basic Proprietary Commercial License	Component	Ship	22	0	0	
		LP	Copyright Swedish Institute of Computer Science	Unspecified	BSD 2.0 [Reviewed and Approved]	Snippet (+ File, Component)	Ship	19	0	0	
		LP	Copyright_Global Election Systems Inc	Unspecified	[template] Basic Proprietary Commercial License	Snippet (+ Component)	Ship	8	0	0	
		FD	Cryptic Allusion DCDev - kallistios	1.3.x	BSD 2.0 [Reviewed and Approved]	Component	Ship	1	0	0	
		CP	Declared_Ignore Matching	Unspecified	Unspecified	Component	Ship	1	0	0	
		FD	DragonFlyBSD-tcp-md5	Unspecified	[template] Basic Proprietary Commercial License	Component	Ship	1	0	0	
		FD	lwIP - A Lightweight TCP/IP stack	Unspecified	Regents of University of California License	Component	Ship	17	0	0	

Tool 사용법 Identify step - 2

- Code Matches 탭을 열어 검출된 오픈 소스 Component 와 License를 확인합니다.

The screenshot displays the Black Duck Protex interface. At the top, there are navigation tabs: My Protex, Manage, Identify (selected), Review, and Report. The current project is identified as [test]-2-11-MMP-Trump RU SERC [C3332]-ThirdPartySource. The project status is shown with a bar chart: 39 (yellow), 2,551 (blue), 189 (green), 0 (red), 252 (orange), 695 (light blue), and 7,778 (white).

The main area shows the 'Code Matches' tab selected, with a red circle and the number '2' around it. A mouse cursor is clicking on this tab, with the word 'click' written next to it. Below the tabs, the file path is shown: File: /[test]-2-11-MMP-Trump RU SERC [C3332]-ThirdPartySource/3p_lwip_ops/lwIP/dist/src/include/netif/etharp.h. A search bar is present with a 'Clear' button. The 'Show' dropdown is set to 'Precision'.

A table lists 18 components (30 per page). The table has columns: ID, Approved, Type, Component Name, Release Dates, Usage, Status, %, Matched File, Line, and Lines. The first row shows 'Amiga Research OS' with a status of 'Precision Match' and a percentage of 22. The second row shows 'Cryptic Allusion DCDev - kallistios' with a green checkmark in the 'Approved' column, a status of 'Precision Match', and a percentage of 22. The third row shows 'devkitPro - devkitPPC' with a red 'X' in the 'Approved' column, a status of 'Precision Match', and a percentage of 22. The fourth row shows 'devkitPro - libogc' with a red 'X' in the 'Approved' column, a status of 'Precision Match', and a percentage of 22.

On the left side, there is a file tree view. The 'netif' folder is expanded, and 'etharp.h' is selected, with a red circle and the number '1' around it. A mouse cursor is clicking on 'etharp.h', with the word 'click' written next to it.

At the bottom, a code viewer shows the content of 'etharp.h', including a copyright notice: 'Copyright (c) 2001-2003 Swedish Institute of Computer Science. All rights reserved.' and a license statement: 'This software is provided as is and use in source and binary forms, with or without modification, is permitted provided that the following conditions are met:'.

Tool 사용법 Identify step - 3

- **“Compare All Matches in New Window”**를 클릭하시면 새 창에서 이 소스파일(Samsung)과 오픈 소스가 어느 부분이 일치하는 지 확인 할 수 있습니다.

Black Duck Protex

SE_admin@samsung.com

My Protex Manage Identify Review Report

Current Project: [test]-2-11-MMP-Trump RU SERC [C3332]-ThirdPartySource

Project Status: 39 2,551 189 0 252 695 7,778

Tools Help Logout

Server: 165.213.202.91

Server Version: 5.2.3

Show: Pending Identification - ALL

Expand Collapse

Only Show Matches

18 Components (30 per page)

ID	Approved	Component Type	Component Name	Version	License	Release Dates	Usage	Status	%	Matched File	Line	Li
		id	Amiga Research OS	Unspecified	[template] Basic Proprietary Commercial License		Snippet	Precision Match	22	AROS-20060617-contrib-source.tar.bz2/contrib/lwip/lwip/src/include/netif/etharp.h	29	41
	✓	id	Cryptic Allusion DCDev - kallistios	1.3.x	BSD 2.0 [Reviewed and Approved]	2005-05-07	Snippet	Precision Match	22	kos-osmode-1.3.x.zip/1.3.x/loadable/modules/net/net_lwip/lwip/src/include/netif/etharp.h	29	41
	X	id	devkitPro - devkitPPC	Unspecified	GPL 2.0 [Reviewed and Rejected]	2005-01-08 - 2005-02-18	Snippet	Precision Match	22	devkitPPC_r11b2-Mac-OSX.tar.bz2/devkitPPC_r11b2/powerpc-gekko/include/netif/etharp.h	29	41
	X	id	devkitPro - libogc	Unspecified	GPL 2.0 [Reviewed and	2004-08-14 - 2005-05-21	Snippet	Precision	22	libogc-src-20040813.tar.bz2/libogc-	29	41

Your File: etharp.h

```
1. /*
2. * Copyright (c) 2001-2003 Swedish Institute of Computer Science.
3. * All rights reserved.
4. *
5. * Redistribution and use in source and binary forms, with or without modification,
6. * are permitted provided that the following conditions are met:
7. *
```

Tool 사용법 Identify step - 4

Code Match 확인

좌측

개발 소스(Samsung)

우측

오픈 소스

```
Matches: Amiga Research OS: AROS-20060617-contrib-source.tar.bz2/contrib/lwip/lwip/src/include/netif/etharp.h
1. /*
2.  * Copyright (c) 2001-2003 Swedish Institute of Computer Science.
3.  * All rights reserved.
4.  *
5.  * Redistribution and use in source and binary forms, with or without modif
6.  * are permitted provided that the following conditions are met:
7.  *
8.  * 1. Redistributions of source code must retain the above copyright notice
9.  * this list of conditions and the following disclaimer.
10. * 2. Redistributions in binary form must reproduce the above copyright not
11. * this list of conditions and the following disclaimer in the documenta
12. * and/or other materials provided with the distribution.
13. * 3. The name of the author may not be used to endorse or promote products
14. * derived from this software without specific prior written permission.
15. *
16. * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMP
17. * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
18. * MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN
19. * SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL
20. * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROC
21. * OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSIN
22. * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
23. * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
24. * IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSI
25. * OF SUCH DAMAGE.
26. *
27. * This file is part of the lwIP TCP/IP stack.
28. *
29. * Author: Adam Dunkels <adam@sics.se>
30. *
31. */
32.
33. #ifndef __NETIF_ETHERARP_H__
34. #define __NETIF_ETHERARP_H__
```



노란색? 흰색?

- 노란색 : Code Match 가 된 부분
- 흰색 : Code Match 가 안된 부분

Tool 사용법 Status Bar

Status Bar

Black Duck Protex
Mobile_admin@samsung.com

My Protex Manage Identify Review Report

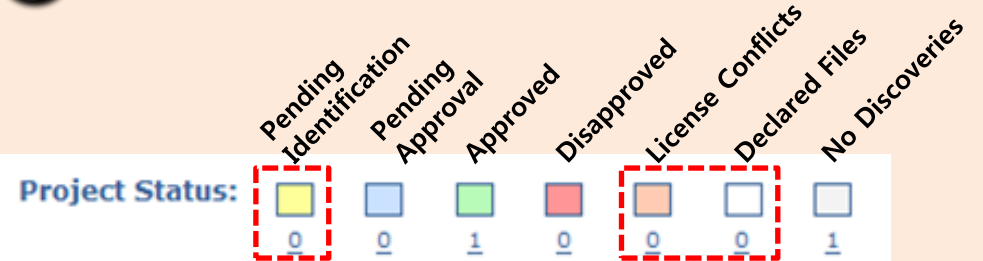
Current Project: 00016ZQXKA3GAU-1-10-WindowsMobile-Cetus US AT&T-System
Project Status: 0 0 1 0 0 0 1

Project: 00016ZQXKA3GAU-1-10-WindowsMobile-Cetus US AT&T-System

Approved	License Conflict	Component Type	Component	Version	License	Us
N/A	N/A		00016ZQXKA3GAU-1-10-WindowsMobile-Cetus US AT&T-System	Unspecified	Samsung Proprietary License	Origin Code
✓		KB	Animate.NET	Unspecified	MIT License V2 [Reviewed and Approved]	Comp
✗	▲	KB	BallCalc	Unspecified	GPL 2.0 [Reviewed and Rejected]	Comp
					MIT License	

This Project is not currently declared.

Project Status



Pending Identification: Protex 에서 오픈소스가 포함되어 있다고 판단한 Identify 대상파일 (Identify 할 때마다 숫자가 줄어듦)

License Conflicts : Identified files 중 License 충돌 이 있는 파일
Declared Files : Bill of Materials 탭에서 declare 로 처리한 파일

※ **주의 :** Identify 처리는 Code Matches 탭 에서 진행해야 합니다.
Code Matches 탭에서 "This file pattern is set to "Requires ID". You must declare this file on the Bill of Materials tab to resolve it." 문구가 나오는 경우에만 Bill of Materials 탭에서 Declare 로 처리해야 합니다.

How To Identify

How To Identify 설명

Identify 절차 (general case)

Black Duck Protex

My Protex Manage Identify Review Report

Current Project: Protex_Edu_98_3-Android

Project Status: 1,047 2 15 0 0 1 180

SE_admin@samsung.com
Tools Help Logout
Server: 165.213.149.98
Server Version: 5.2.3

Show: Pending Identification - ALL

Bill of Materials Code Matches Searches Dependencies

File: /Protex_Edu_98_3-Android/Android/external/bluetooth/bluez/audio/a2dp.c

Search: [] Go Clear Show: Precision

5 Components (30 per page) [First] < Prev | Next > | Last

ID	Approved	Type	Component Name	Version	License	Release Dates	Usage	Status	%	Matched File
x		Android - platform - external - bluetooth - bluez	2.3_r1	GPL 2.0 [Reviewed and Rejected]	2010-12-17	Snippet	Precision Match	82		android-android-2.3_r1-platform-external-bluetooth-bluez.tar.gz/audio/a2dp.c

Your File: a2dp.c Matched File: a2dp.c

*Component: Android - platform - external - bluetc (homepage) File/Folder Comment

Version: 2.3_r1

License: GPL 2.0 [Reviewed and Rejected] (view license)

Usage: Component

IDENTIFY Matches

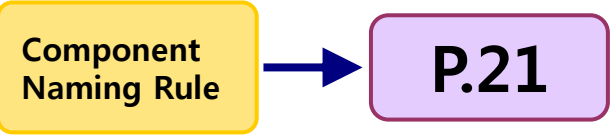
EXCLUDE Component (Recalculate Precision Matches)

ORIGINAL TO: Protex_Edu_98_3-Android

RESET Identification

Create Local Component... Save Cancel

- ① Source Tree 에서 Identify 하고 자 하는 File 을 선택합니다.
(굵게 표시된 폴더/파일명으로 Identify 대상임을 알 수 있습니다.)
- ② Code Matches 탭을 선택합니다.
- ③ Code가 일치하는 오픈 소스들이 List로 나열되는데, 이 오픈 소스들 중에서 가져다 사용한 출처를 선택합니다.
- ④ IDENTIFY Match 를 선택하고, Component명을 입력합니다.



- ⑤ Comment 를 입력한 후,
- ⑥ save comment 를 눌러 저장합니다.
- ⑦ SAVE 버튼을 눌러 Identify한 내용을 Protex 에 반영합니다.

How To Identify 설명

Identify 선택 종류

*Component: ([homepage](#))

Version:

License: ([view license](#))

Usage:

IDENTIFY Matches

EXCLUDE Component (Recalculate Precision Matches)

ORIGINAL TO: Protex_Edu_98_3-Android

RESET Identification

IDENTIFY Match
EXCLUDE Component
ORIGINAL TO *Project_name*

IDENTIFY Match	이 오픈 소스와 (전체/부분)일치한다는 것을 인정하는 경우 (즉, 개발 소스 코드의 출처는 이 오픈 소스임을 의미)
EXCLUDE Component	이 오픈 소스 Component를 Code Match 비교 대상에서 제외하고자 할 경우 이 항목을 선택하면, 이 Project 전체에 대하여 재분석을 하게 됩니다. Exclude 항목 선택 시에는 주의를 요하므로, 이 항목은 선택하지 않도록 합니다.
ORIGINAL TO <i>Project_name</i>	이 항목도 선택하지 않도록 합니다.

How To Identify 설명

■ 오픈 소스 Component 선택하기

The screenshot shows the Black Duck Protex interface. The main window displays a table of components identified in the file `/Protex_Edu_98_3-Android/Android/external/bluetooth/bluez/audio/a2dp.c`. The table has columns for ID, Approved, Type, Component name, Version, License, Release Dates, Usage, Status, %, and Matched File. Two components are listed:

ID	Approved	Type	Component name	Version	License	Release Dates	Usage	Status	%	Matched File
	X	⊕	Android - platform - external - bluetooth - bluez	3.3_r1	GPL 2.0 [Reviewed and Rejected]	2010-12-17	Snippet	Precision Match	82	android-android-2.3_r1-platform-external-bluetooth-bluez.tar.gz/audio/a2dp.c
	X	⊕	BlueZ	Unspecified	GPL 2.0 [Reviewed and Rejected]	2010-06-12 - 2010-09-08	Snippet	Precision Match	82	bluez-4.66.tar.gz/bluez-4.66/audio/a2dp.c

The 'Component name' column for the first row is highlighted with a red box. Below the table, there are sections for 'Your File: a2dp.c' and 'Matched File: a2dp.c' showing code snippets.

선택의 기준은?

어떤 오픈 소스를 가져다 사용했는가!
(목표 : 출처 확인)

나는 이 Code가 어떤 오픈 소스를 사용한 것인지 **알고있다.**
(ex: 신규 구현기능이라 개발자가 직접 구현한 경우)

P.21

나는 이 Code가 어떤 오픈 소스를 사용한 것인지 **전혀 모르겠다.**
(ex: Legacy Code인 경우)

P.18

How To Identify 출처 확인

■ 소스 코드 출처 확인 방법

- 직접 구현한 소스가 아니라서 오픈 소스와 일치하는 부분의 출처를 모른다면, 다음의 방법으로 출처를 확인할 수 있습니다.

1. **최초 개발자에게 문의** (이 방법이 가장 정확하지만, 개발자가 누군지 불분명하거나 퇴사한 경우는 2,3번 방법으로)
2. **소스 코드 상단의 주석 확인** (올바르게 오픈 소스를 가져다 사용한 경우라면, 소스 코드 상단에서 Copyright 문구를 확인 하실 수 있습니다.)
3. **검색 엔진을 통한 확인** (1,2번 둘 다 해당사항이 없는 경우에는 검색 엔진을 통해 확인하는 방법이 있습니다.)

The screenshot shows a search tool interface. At the top, there's a search bar and a 'Go' button. Below it, a table lists search results. The table has columns for ID, Approved, Component Type, Name, Version, License, Release Dates, Usage, Status, %, Matched File, Line, and Lines. One result is shown with a red 'X' in the 'Approved' column and a blue 'B' in the 'Component Type' column. The license is 'GPL 2.0 [Reviewed and Rejected]'. Below the table, a code editor window shows the source code for 'drv_callbacks.c'. Lines 5 through 8 are highlighted in orange, showing a copyright notice for Jouni Malinen and a reference to the GNU General Public License version 2.

ID	Approved	Component Type	Name	Version	License	Release Dates	Usage	Status	%	Matched File	Line	Lines
	X	B	hostap-sae	Unspecified	GPL 2.0 [Reviewed and Rejected]		Snippet	Precision Match	91	hostap-sae-master-20110415.tgz/cozybit-hostap-sae-77dc829/src/ap/drv_callbacks.c	1	234

```
1. /*
2. + hostapd / Callback functions for driver wrappers
3. + Copyright (c) 2002-2009, Jouni Malinen <j@w1.fi>
4. +
5. + This program is free software: you can redistribute it and/or modify
6. + it under the terms of the GNU General Public License version 2 as
7. + published by the Free Software Foundation.
8. +
9. + Alternatively, this software may be distributed under the terms of BSD
```

주석을 확인해보니, “GNU General Public License version 2” 라는 문구가 포함되어 있습니다.

따라서, 이 소스는 GPL 2.0 으로 배포된 오픈 소스라고 쉽게 판단할 수 있습니다.

주석이 없는 경우? 주석으로도 확인이 불가능한 경우?

→ 검색 엔진으로 확인하기! (다음 장에서 설명합니다.)

How To Identify 출처 확인

- 코드 검색 엔진 종류 (아래 2개 site 추천)

1. Chromium Code Search <https://cs.chromium.org/>

2. searchcode <https://searchcode.com/>

3. Krugle <http://opensearch.krugle.org/>

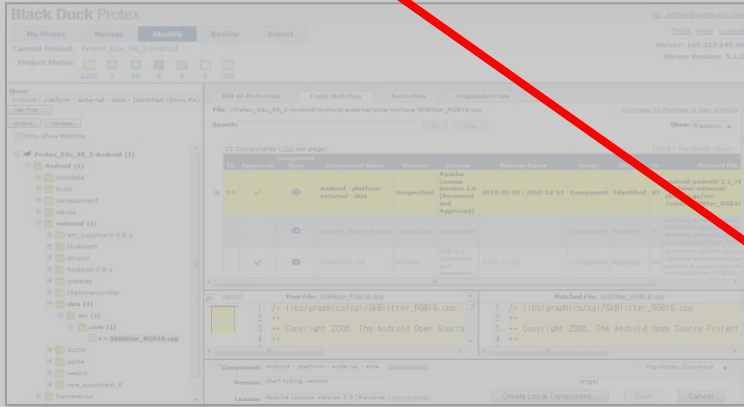


```
14: (132, 175, 207, 207) // 666h,
15: 207, 206, 207, 463,
16: 477, 46, 0xffffffff, 0xffffffff,
17: 0xffffffff, 0xffffffff, 0, 0,
18: },
19: static uint8_t F10A0000(UNITS ch)
20: {
21: }
22: static const uint8_t z_f10a0000[16] = {
23: 0x0, 0x0, 0x0, 0x0,
24: 0x0, 0x0, 0x0, 0x0,
25: 0x0, 0x0, 0x0, 0x0,
26: };
27: return z_f10a0000[ch];
28: }
29:
30: static int32_t ReadBits(Mozilla::Endian endian, uint32_t read)
31: {
32: return read;
33: }
34:
35: static void Convert2IP2(const uint8_t* src, uint8_t* dst)
36: {
37: }
38:
39: // RFC 2027 Section 5.3 Switch Frame
40: // 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0
41: // P | P | Q | Q | P | P | R | R |
42: // P | P | Q | Q | P | P | R | R |
43: //
44: // P = padding
45: // Q = frame type
46: // R = quality indicator
47: //
48: // RFC 2027 Section 5.3.2.1
49: // 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0
50: // (R2) - (R1) | (P1) - (P0) |
51: //
52: // Note: The conversion needs to change the endianness of the byte
53: // bits and shift the data by 4 bits
54: int frameType = (src[0] && 0x0f) << 3;
55: int frameBytes = (src[1] && 0x0f) << 4;
56: // Note: The conversion needs to change the endianness
57: // bits and shift the data by 4 bits
58: int frameType = (src[0] && 0x0f) << 3;
59: int frameBytes = (src[1] && 0x0f) << 4;
60: int i = 0;
61: for (int i = 0; i < frameBytes; i++)
```

오픈 소스와 일치하는 부분에서
특징적인 String (주로 함수명)으로
검색하여 License 확인!

How To Identify 오픈 소스 List 확인

■ 검출된 오픈 소스 List 확인



검출된
오픈 소스 List

검출된 오픈 소스 List 안에 이 Code의
오픈 소스 출처가 있다.



P.15

검출된 오픈 소스 List 안에 이 Code의
오픈 소스 출처가 없다.



P.21

Q : “왜 Component List 안에 없죠? 그럴 수 있나요?”

A : 없을 수 있습니다. Protex 가 비록 방대한 Database 를 가지고 있다고는 하나, Open Source 는 워낙 많기 때문에 모든 Open Source 를 다 가지고 있을 수 없기 때문입니다. 또한 최근에 등록된 Open Source 의 경우, Protex 의 Database 에 Update 되기까지는 일 정 시간이 소요될 수 있습니다.

How To Identify Component 변경



Component 명명 규칙

- 라이브러리 파일 (*.so, *.jar, ...) : 파일명을 컴포넌트명으로 사용

예) File: /S-201607146514-1-16-CharmBySamsung_1.0.3/app/src/main/libs/android-support-v4.jar
→ **android-support-v4.jar**

File:

Search:

1 Component [Columns...](#) [\(clear sort\)](#)

Approved	License Conflict	Component Type	Component	Version	
✓		LP	android-support-v4.jar	Unspecified	Apache License Ve Approved]

- 소스코드 : File Path에서 버전명을 제외한 해당모듈의 경로를 '-'로 연결, 라이선스 정보 괄호로 추가 ('-' 사이에 띄어쓰기가 필요합니다)

예) Folder: /S-201606248928-1-16-PCGallery_1.0.0.0/Gallery/CommonLibraries/Pensive/MediaInfo/MediaInfoLib
→ **PCGallery - Gallery - CommonLibraries - Pensive - MediaInfo - MediaInfoLib (BSD 2-clause)**

Folder:

Search:

2 Components (500 per page) [Columns...](#) [\(clear sort\)](#)

Approved	License Conflict	Component Type	Component
✓		LP	PCGallery - Gallery - CommonLibraries - Pensive - MediaInfo - MediaInfoLib (BSD 2-clause)

*라이선스 예시

- (BSD 2-clause)
- (BSD 2.0)
- (Apache 2.0)
- (MIT)
- (GPL 2.0)
- (LGPL 2.1)

How To Identify Component 변경

1 Component 이름을 Typing 하면 리스트가 나타납니다

2 IDENTIFY Matches

3 Save 버튼이 활성화 됩니다.

"0 Suggestions Found" 라고 나오며 선택이 불가능하다.

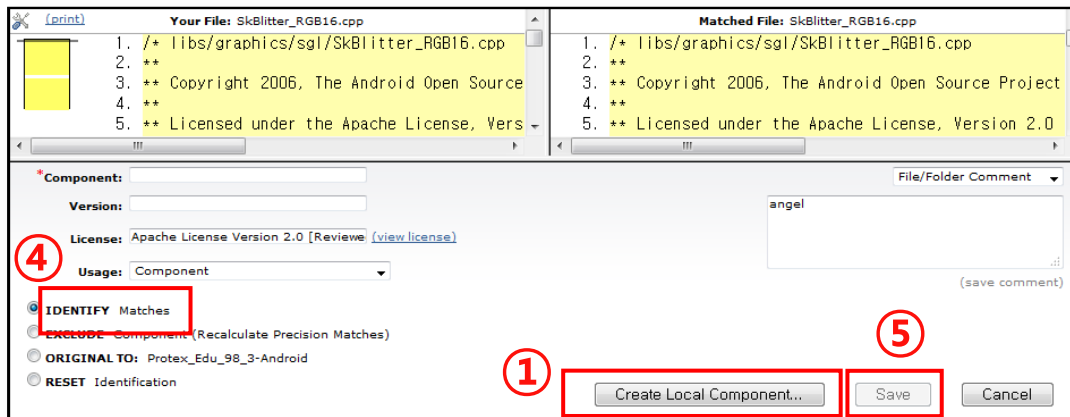
P.23

How To Identify 신규 Component 등록

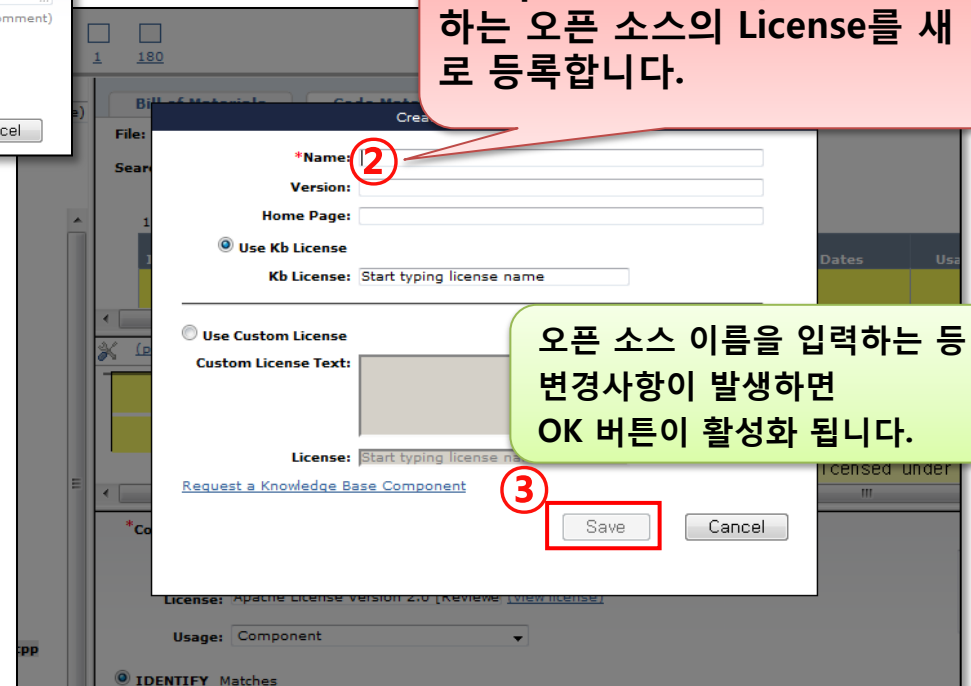
■ 신규 Component 등록하여 Identify 하기

Component명이 존재하지 않을 경우, 최초 1회 신규 Component 등록이 필요합니다.

등록된 Component는 삭제할 수 없으므로, 중복 등록하지 않도록 합니다.



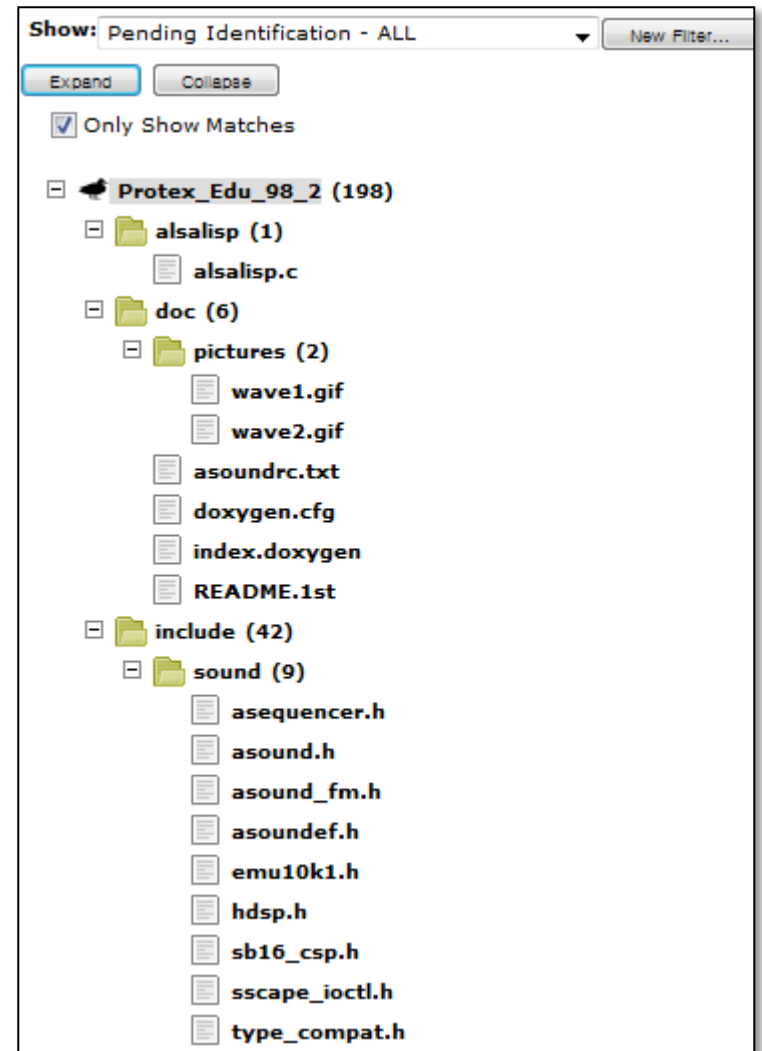
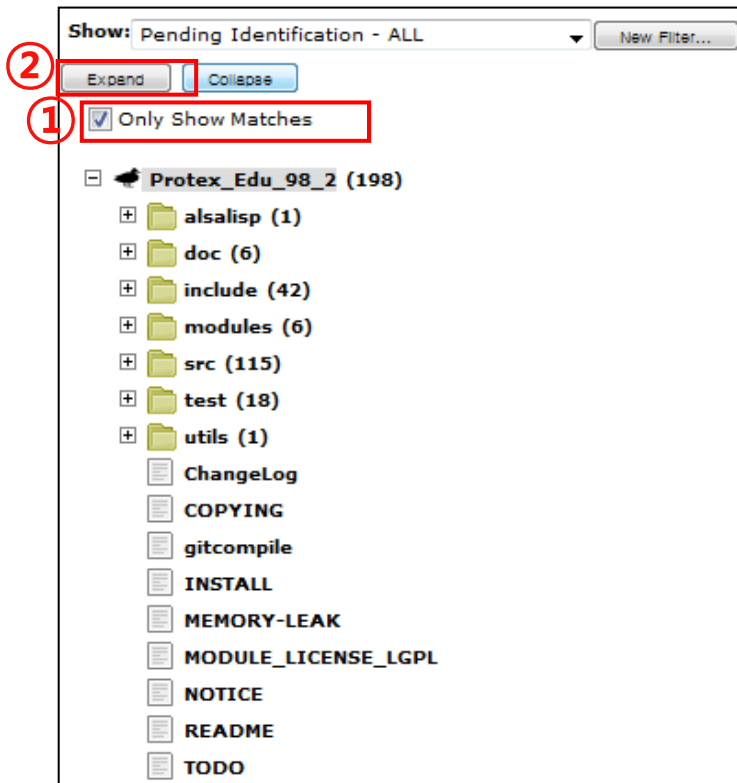
Component 이름과 출처에 해당하는 오픈 소스의 License를 새로 등록합니다.



How To Identify Tip

■ Identify 대상 파일만 보기

- Project 내에 폴더가 많아서 Identify 대상 파일을 찾기 힘든 경우, 좌측 Source Tree 에 위치한 Option을 선택하면 Identify 대상 파일을 쉽게 찾을 수 있습니다.



How To Identify 설명(Search tab)

Identify 절차 (Search tab)

Bill of Materials		Code Matches		Searches		Dependencies			
Standard	BSD License					Pending Identification	9	27	0
Standard	BSD License 1.0					Pending Identification	12	36	0
Standard	General Public License 2.0					Pending Identification	1	1	0
Standard	General Public License keywords					Pending Identification	1	1	0
Standard	Public Domain					Pending Identification	1	1	0
>> Custom	GPL 2.0 except LGPL 2.0/2.1	KB	General Public License 2.0	Unspecified	GPL 2.0 [Reviewed and Rejected]	Component Identified	1	0	2

2 Viewing results (1,000 per page)

[First] < Prev | Next > | La

File	Match
<input checked="" type="checkbox"/> DEV_Froyo/Android/extension/brcm/adaptation/dtun/include/dtun.h	PARTICULAR PURPOSE. See the GNU General Public License for more details. A copy
<input checked="" type="checkbox"/> DEV_Froyo/Android/extension/brcm/adaptation/dtun/include/dtun.h	the terms of the GNU General Public License, version 2, as published by

*Component: (homepage)

Version:

License: (view license)

Usage:

(save comment)

- ① Search 탭을 선택합니다.
- ② 키워드가 일치하는 파일들이 List로 나열되는데, 이 중 공개 의무가 있는 라이선스를 선택합니다.

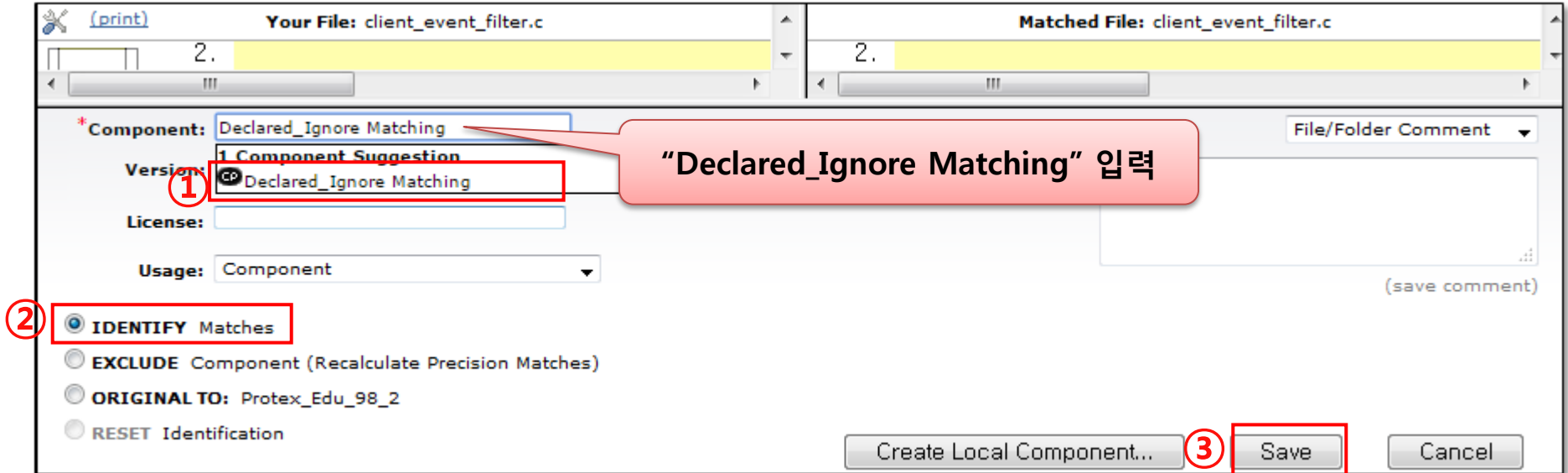
<공개의무가 있는 라이선스>

GPL 2.0 except LGPL 2.0 / 2.1
 LGPL 2.1 except GPL
 Mozilla Public License 1.1

- ③ 해당 파일이 false가 아닌지 확인 후 파일을 선택합니다.
- ④ Component 명을 입력 합니다
- ⑤ 해당 라이선스를 입력 합니다.
- ⑥ comment 를 작성 후, save comment 버튼을 눌러 저장합니다. SAVE 버튼을 눌러 Identify한 내용을 Protex 에 반영합니다.

How To Identify Ignorable Code

- 저작권의 영향을 받지 않는 코드에 대해서는 "Declared_Ignore Matching"으로 Identify 처리



저작권의 영향을 받지 않는 코드

- 1) 단순 상수
- 2) #include, import, etc...
- 3) Standard reference code (standard spec code)
- 4) Auto-generated code

How To Identify Ignorable Code Example

1) 단순 상수

```
15149. 0xd8ffd6ef, 0xda3fd91f, 0xdc8fd55f, 0xbfdd009f, 0xfdfdfcfe, 0x3fe2
15150. 0xe8efe1cf, 0xeb1feaf, 0xe75fec3f, 0x009fee8f, 0xaff0bfe, 0x2fec
15151. 0xf3c1f201, 0xfb1ff4e1, 0xfd5ffc3f, 0xcc9ffe7f, 0xfadff90d, 0x013f
15152. 0xf9e381f0, 0x074f06e1, 0x098f086f, 0x24cf0aaf, 0xf0f06e38d, 0x4fd0
15153. 0x15ef130d, 0x172f16df, 0x00000048, 0x0000000d, 0x00000000, 0x000
15154. 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x0001
15155. 0x02580001, 0x00020001, 0x02580000, 0x00030001, 0x04b00000, 0x0003
15156. 0x00000003, 0x00000004, 0x00000005, 0x00000006, 0x00000007, 0x0001
15157. 0x0000000b, 0x0000000c, 0x0000000d, 0x0000000e, 0x00000012, 0x0001
15158. 0x00000016, 0x00000017, 0x00000018, 0x00000019, 0x0000001a, 0x0001
15159. 0x000000e7, 0x0000004b, 0x0000004c, 0x0000004e, 0x00000078, 0x0001
15160. 0x000000ca, 0x000000cd, 0x000000e7, 0x000000aa, 0x000000a8, 0x0001
15161. 0x000000ba, 0x000000bb, 0x000000c6, 0x000000c7, 0x000000be, 0x0001
15162. 0x0000001e, 0x0000001f, 0x00000020, 0x00000021, 0x00000022, 0x0001
15163. 0x00000026, 0x00000027, 0x00000028, 0x00000029, 0x0000002a, 0x0001
15164. 0x0000002f, 0x00000030, 0x00000031, 0x00000033, 0x000000ab, 0x0001
15165. 0x00000037, 0x00000038, 0x00000049, 0x0000004a, 0x0000003e, 0x0001
15166. 0x00000076, 0x0000000f, 0x00000010, 0x00000083, 0x00000084, 0x0001
15167. 0x00000050, 0x00000051, 0x00000052, 0x00000053, 0x00000054, 0x0001
15168. 0x00000058, 0x00000059, 0x0000005a, 0x0000005b, 0x0000005c, 0x0001
15169. 0x00000060, 0x00000061, 0x00000062, 0x00000063, 0x00000064, 0x0001
15170. 0x00000068, 0x00000069, 0x0000006a, 0x0000006b, 0x0000006c, 0x0001
15171. 0x00000070, 0x00000071, 0x00000072, 0x00000073, 0x00000074, 0x0001
15172. 0x00000081, 0x00000080, 0x00000087, 0x00000088, 0x00000089, 0x0001
15173. 0x0000008f, 0x00000090, 0x00000091, 0x00000092, 0x00000093, 0x0001
15174. 0x00000097, 0x00000098, 0x00000099, 0x0000009a, 0x0000009b, 0x0001
15175. 0x0000009e, 0x0000009f, 0x000000a0, 0x000000a1, 0x000000a2, 0x0001
15176. 0x000000b0, 0x000000b1, 0x000000b2, 0x000000b3, 0x00000046, 0x0001
15177. 0x000000ce, 0x000000cf, 0x000000d0, 0x000000d1, 0x000000d3, 0x0001
15178. 0x000000d7, 0x000000d8, 0x000000d9, 0x000000da, 0x000000db, 0x0001
15179. 0x000000df, 0x000000e0, 0x000000e1, 0x000000e2, 0x000000e3, 0x0001
15180. 0x000000cc, 0x000000cb, 0x000000c3, 0x000000c2, 0x000000c1, 0x0001
15181. 0x000000ea, 0x000000eb, 0x000000ec, 0x000000ed, 0x000000ee, 0x0001
15182. 0x000000f2, 0x000000f3, 0x000000f4, 0x000000f6, 0x000000f7, 0x0001
15183. 0x000000f5, 0x000000f8, 0x000000f9, 0x000000fa, 0x000000fb, 0x0001
```

```
466. unsigned long L5[] = { 0xAD6EA6B0, 0x49A70F7D, 0x4B7A70E9, 0x
467. CSigs[i++] = NewCryptSig("blowfish", L5 , 8, NULL, 0, -4, 0,
468.
469. unsigned long L6[] = { 0x3F258C7A, 0x1E213F2F, 0x30FB40D4, 0x
470. CSigs[i++] = NewCryptSig("cast", L6 , 7, NULL, 0, -4, 0, CRYP
471.
472. unsigned long L7[] = { 0x7789F8B7, 0xE6C1121B, 0xE216300D, 0x
473. CSigs[i++] = NewCryptSig("cast", L7 , 7, NULL, 0, -4, 0, CRYP
474.
475. unsigned long L8[] = { 0x00000002, 0x00000003, 0x00000004, 0x
476. 0x0000000A, 0x0000000B, 0x0000000C, 0x0000000D, 0x0000000E, 0x
477. 0x000000E2, 0x000000E3, 0x000000E4, 0x000000E5, 0x000000E6, 0x
478. 0x000000EA, 0x000000EB, 0x000000EC, 0x000000ED, 0x000000EE, 0x
479. 0x000000F2, 0x000000F3, 0x000000F4, 0x000000F5, 0x000000F6, 0x
480. 0x000000FA, 0x000000FB, 0x000000FC, 0x000000FD, 0x000000FE, 0x
481. 0x00000002, 0x00000003, 0x00000004, 0x00000005, 0x00000006, 0x
482. 0x0000000A, 0x0000000B, 0x0000000C, 0x0000000D, 0x0000000E, 0x
483. 0x00000008, 0x0000000C, 0x0000000D, 0x0000000E, 0x0000000F, 0x
484. 0x00000013, 0x00000014, 0x00000015, 0x00000016, 0x00000017, 0x
485. 0x0000001B, 0x0000001C, 0x0000001D, 0x0000001E, 0x0000001F, 0x
486. 0x00000023, 0x00000024, 0x00000025, 0x00000020, 0x00000021, 0x
487. 0x00000025, 0x00000026, 0x00000027, 0x00000028, 0x00000029, 0x
488. 0x0000002D, 0x0000002E, 0x0000002F, 0x00000030, 0x00000031, 0x
489. 0x00000035, 0x00000036, 0x00000037, 0x00000038, 0x00000039, 0x
490. 0x0000003D, 0x0000003E, 0x0000003F, 0x00000040, 0x00000041, 0x
491. CSigs[i++] = NewCryptSig("des", L8 , 128, NULL, 0, -4, 0, CRYP
492.
493. unsigned long L9[] = { 0x0000002E, 0x0000002F, 0x00000030, 0x
494. 0x00000036, 0x00000037, 0x00000038, 0x00000039, 0x00000041, 0x
495. 0x00000045, 0x00000046, 0x00000047, 0x00000048, 0x00000049, 0x
496. 0x0000004D, 0x0000004E, 0x0000004F, 0x00000050, 0x00000051, 0x
497. 0x00000055, 0x00000056, 0x00000057, 0x00000058, 0x00000059, 0x
498. 0x00000063, 0x00000064, 0x00000065, 0x00000066, 0x00000067, 0x
499. 0x0000006B, 0x0000006C, 0x0000006D, 0x0000006E, 0x0000006F, 0x
500. 0x00000073, 0x00000074, 0x00000075, 0x00000076, 0x00000077, 0x
501. CSigs[i++] = NewCryptSig("des", L9 , 64, NULL, 0, -4, 0, CRYP
502.
503. unsigned long L10[] = { 0x02080800, 0x00080000, 0x02000002, 0x
```

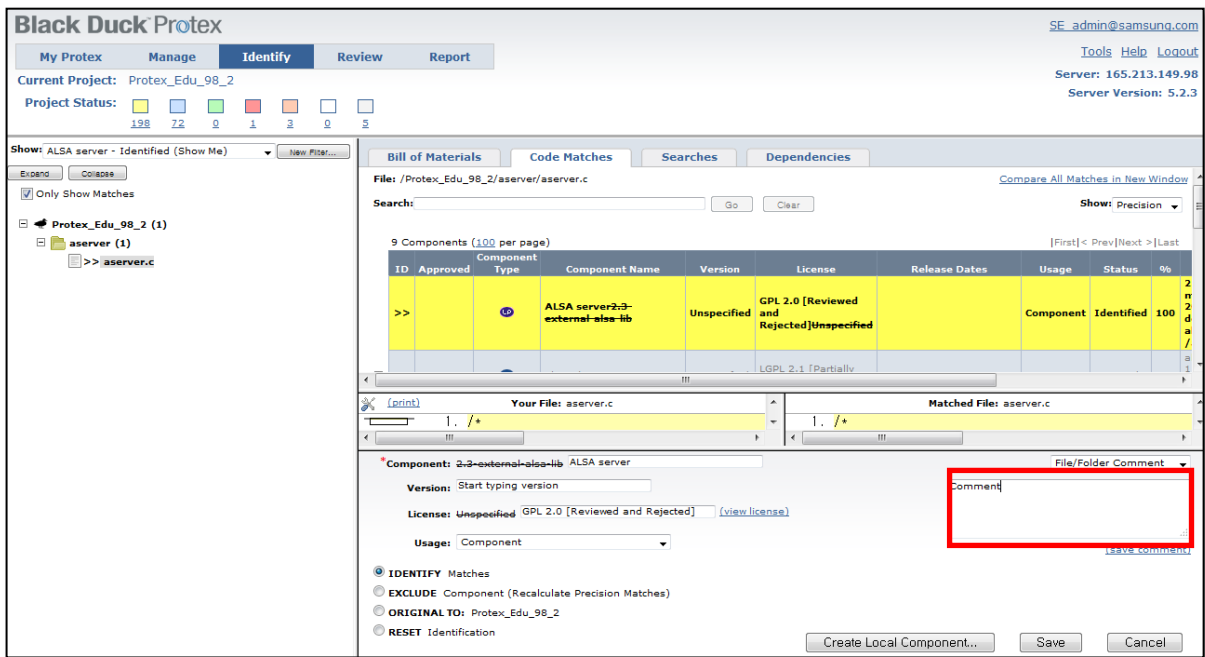
2) #include, import, etc...

```
2.
3. import android.app.Activity;
4. import android.app.AlertDialog;
5. import android.content.ComponentName;
6. import android.content.Context;
7. import android.content.DialogInterface;
8. import android.content.Intent;
9. import android.content.ServiceConnection;
10. import android.content.SharedPreferences.Editor;
11. import android.content.pm.PackageManager;
12. import android.net.wifi.p2p.WifiP2pDevice;
13. import android.os.Bundle;
14. import android.os.Handler;
15. import android.os.IBinder;
16. import android.preference.PreferenceManager;
17. import android.util.Log;
18. import android.view.MotionEvent;
19. import android.view.SurfaceHolder;
20. import android.view.View;
21. import android.view.WindowManager;
22. import android.widget.ImageView;
23. import android.widget.LinearLayout;
24. import android.widget.RelativeLayout;
25. import android.widget.TextView;
26. import android.media.MediaPlayer;
27.
```

```
19.
20. import android.app.Activity;
21. import android.app.AlertDialog;
22. import android.content.ComponentName;
23. import android.content.DialogInterface;
24. import android.content.Intent;
25. import android.content.ServiceConnection;
26. import android.content.SharedPreferences;
27. import android.content.SharedPreferences.Editor;
28. import android.content.pm.PackageInfo;
29. import android.content.pm.PackageManager.NameNotFoundException;
30. import android.os.Bundle;
31. import android.os.Handler;
32. import android.os.IBinder;
33. import android.os.Message;
34. import android.os.RemoteException;
35. import android.preference.PreferenceManager;
36. import android.text.Html;
37. import android.view.View;
38. import android.view.View.OnClickListener;
39. import android.widget.Button;
40. import android.widget.ProgressBar;
41. import android.widget.TextView;
42.
43. import com.csipsimple.R;
44. import com.csipsimple.android.DownloadProgress;
```

Comment 작성

Comment 필요성



Identify 수행하는 사람과 Reviewer 가 각각 다른 사람이라면?

: Reviewer 는 Identify 수행한 결과(Component / License)만을 보고 판단해야 합니다.

어떠한 이유로 해당 Component / License 로 분류한 것인지에 대해 간단한 설명이 있어야 Review 과정도 수월할 것입니다.

Identify 수행하는 사람과 Reviewer 가 같은 사람이라면?

: Identify 대상 파일수가 많아지면 파일 별 Identified 내역을 모두 기억하고 있기란 매우 어렵습니다.

또한 이 Project 을 backbone 으로 하는 과제에서 Clone Project 으로 활용하는 경우,

Reviewer 가 달라지게 되므로, Identify 수행내역에 대한 간단한 설명이 있는 것이 좋을 것입니다.

Comment 작성

The screenshot shows the 'Code Matches' tab in a dependency management tool. The left sidebar displays a project tree for '00014NRAKA3GAU-3-10-Android-P1 E'. The main area shows a table with one component match:

ID	Approved	Component Type	Component Name	Version	License	Release Dates	Usage
>>	✓	UP	Android Open Source Project(check)Android	Unspecified	Apache License Version 2.0	2009-10-05 - 2010-03-18	Component

Below the table, the 'Your File' and 'Matched File' are both 'Camera.java'. The 'Component' dropdown is set to 'Android Open Source Project', and the 'License' is 'Apache License Version 2.0'. A red box highlights the 'File/Folder Comment' field, which contains the text: 'No Copyright / No License notice Code Matching : Android Open Source Project - Apache License 2.0'. A blue arrow points from this comment box to the text below.

어떠한 이유로 해당 Component / License 로 분류한 것인지에 대한 간단한 설명을 Comment 란에 적어주시면 됩니다.

Comment 작성 Rule – OMS 연동 : commercial license

■ Commercial License 인 경우 OMS Review 대상에 포함되도록 주석 작성하기

: 3'rd-party 와 계약하고 사용하는 상용 소스를 Identify 수행할 때,

Protex License 는 [template] Basic Proprietary Commercial License 로 지정하고 있습니다.

OMS 에 과제등록시 위의 License 는 Review 대상으로 분류가 되지 않도록 되어 있는데요,
간혹 “이 파일이 상용 소스가 맞는거 같긴 한데... 확실하지 않아서 담당개발자의 Review 를
받아봐야겠다.”는 판단이 들면 Protex 의 comment 란에 (commercial) 이라는
말머리는 붙이시면 됩니다. (다음 page 그림 참고)

※ OMS 에서는 Protex 주석에서 (commercial) 문구를 발견하면
해당 파일을 Review 대상으로 분류하도록 처리합니다.

Comment 작성 Rule – OMS 연동 : commercial license

ID	Approved	Component Type	Component Name	Version	License	Release Dates	Usage	Status	%	Matched File	Line	# Lines
>>	X	LP	Copyright (c) Cypress Semiconductor Corporation Sapo-A5-Kernel	Unspecified	[template] Basic Proprietary Commercial License GPL 2.0 [Reviewed and Rejected]		Component	Identified	51	Sapo-A5-Kernel-master-20110108.tgz/r3pek-Sapo-A5-Kernel-1933b96/drivers/input/touchscreen/issp_routines.c	166	103

Your File: tspsr2drv_C3512T_TST200_Download.c

```

379.     }
380.     bCurrByte = bCurrByte << 1;
381. }
382. }
383.
384.
385. // =====
386. // SendVector()
387. // This routine sends the vector specified. All vec
388. // in ISSP_Vectors.h. The data line is returned t
        
```

Matched File: issp_routines.c

```

165.     }
166.     bCurrByte = bCurrByte << 1;
167. }
168. }
169.
170. // =====
171. // SendVector()
172. // This routine sends the vector specified. All vectors cons
173. // in ISSP_Vectors.h. The data line is returned to HiZ aft
174. // sent
        
```


***Component:** Sapo-A5-Kernel Copyright (c) Cypress Semiconduct

Version:

License: GPL 2.0 [Reviewed and Rejected] [template] Basic Proprietary Comm [\(view license\)](#)

Usage:

IDENTIFY Matches

EXCLUDE Component (Recalculate Precision Matches)

ORIGINAL TO: 0009AAGH6tPMWL1000-1-11-SCP-1exto RU SERC-GV_E49XX-delta

RESET Identification

(commercial) 상용 라이선스인지 개발자 리뷰 필요

(save comment)

개발자 Review 가 필요한 파일의 경우, 주석작성시 (commercial) 로 시작하시면 됩니다.
 단, License 가 [template] Basic Proprietary Commercial License 인 경우에만 해당 됨.

Comment 작성 Rule – OMS 연동 : 파일단위 작성

The screenshot shows the OMS interface with the following components and details:

Component List:

- 0000V6ZW9A3GAU-MR-3-11-Android-Aries EUR OPEN-Kernel(GB) (Expanded)
- arch (151) (Highlighted with red box)
- block (1)
- crypto (4)
- Documentation
- drivers (513)
- firmware (2)
- fs (36)
- include (39)
- init (2)
- ipc
- kernel (20)
- lib (1)
- mm (3)
- net (13)
- samples (1)
- scripts
- security (1)
- sound (36)
- tools

Component Details (Linux Kernel):

- Version: Unspecified
- License: GPL 2.0 [Reviewed and Rejected]
- License Conflict Status: Conflicts with another component and the project's declared license
- Code Matches ID'd in: 2,050 file(s) or folder(s)
- Usage: Snippet (+ File, Component)
- Ship Status: Ship
- Component ID: linuxkernel642
- Release ID:
- License ID: gpl20

Comment Field: GPL여부 확인 필요 (Highlighted with red box)

폴더 단위 identify 시, 입력 한 comment는 OMS 리뷰대상 파일 별 comment에 반영되지 않습니다.
개발자 Review 가 필요한 파일의 경우, comment는 파일 단위로 입력하여 주세요

Comment 작성 Rule – OMS 연동 : Declared_Unused_Matching

The screenshot shows the OMS interface with a component match for 'Declared_Ignore'. The 'Component' field is set to 'Declared_Unused Matching'. The 'Usage' is set to 'Component'. The 'IDENTIFY Matches' radio button is selected. The 'Matched File' is 'mfc_opr.c'. The 'Component Name' is 'Declared_Ignore'. The 'Version' is 'Unspecified'. The 'License' is 'Unspecified'. The 'Release Dates' is 'Component'. The 'Usage' is 'Component'. The 'Status' is 'Identified'. The 'Component' field is highlighted with a red box. A red arrow points from the text below to this box.

ID	Approved	Component	Component Name	Version	License	Release Dates	Usage	Status
		Declared_Ignore	Declared_Ignore	Unspecified	Unspecified		Component	Identified

***Component:** CommunitySG: Declared_Unused Matching
Version:
License:
Usage: Component
 IDENTIFY Matches
 EXCLUDE Component (Recalculate Precision Matches)
 ORIGINAL TO: 0004BVULRtPMWL0000-2-11-Android-Hawk US T MOBILE-Android(GB)
 RESET Identification
Create Local Component... Save Cancel

모든 과제에 미포함인 경우 사용!

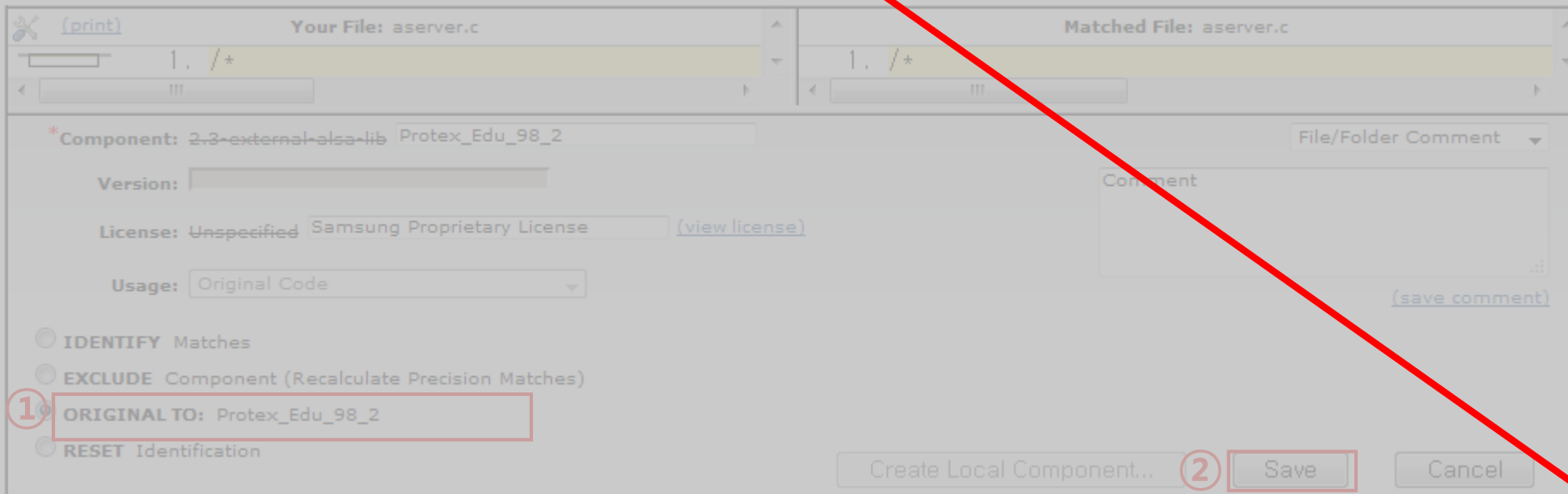
- ※ Declared_Unused_Matching으로 Component를 설정할 경우 Review가 필요없는 대상으로 보고 OMS로 결과를 가져오지 않기 때문에,
 1. 브랜치 검증의 경우 전과제에서 확실히 미포함인 경우만 Declared_Unused_Matching을 사용합니다.
 2. 과제의 경우 확실히 미포함인 경우에만 Declared_Unused_Matching을 사용합니다

Identify step FAQ

FAQ 오픈 소스가 아닌 경우의 처리

Q1 > 자체 구현한 소스입니다. 그런데도 오픈 소스와 일치하는 부분이 있다고 나오네요. 어떻게 해야 하나요?

A > 오픈 소스와 일치하는 부분이 있더라도 자체 구현한 부분이 확실하다면 다음과 같이 처리하시면 됩니다. 개발자 누구나 Common 하게 구현하는 형식이 라면 오픈 소스에서 가져다 사용했다고 볼 수 없기 때문입니다.

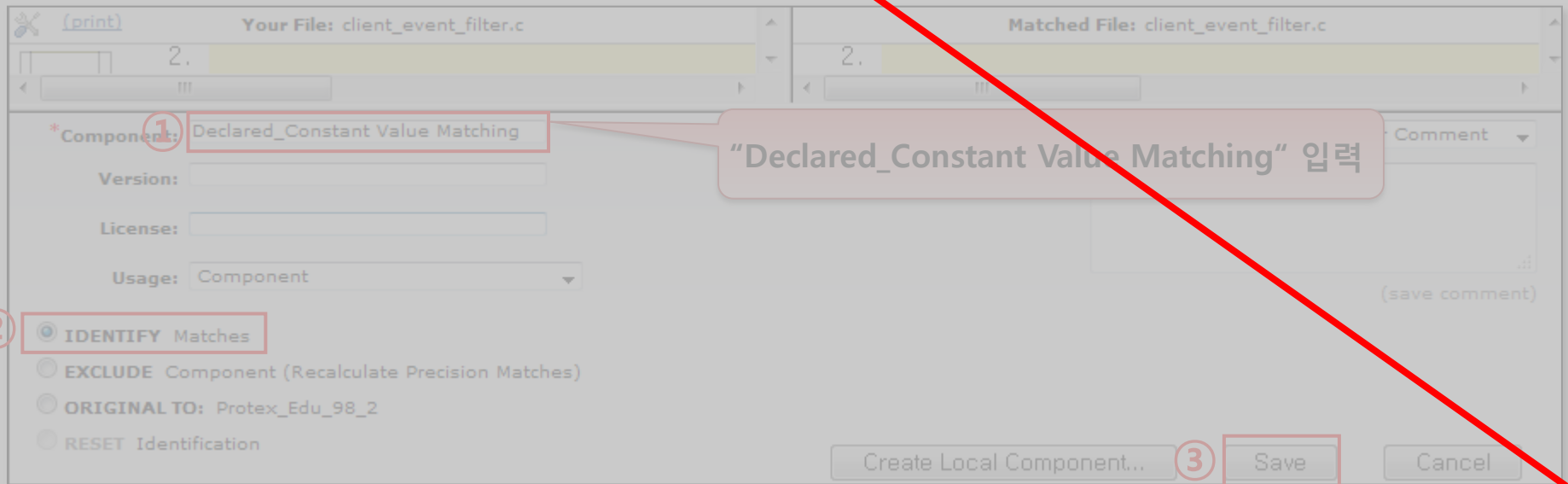


FAQ 상수 일치에 대한 처리

Q2 > 일부 상수만 일치 하는 것으로 보입니다. 이것도 오픈 소스와 일치한다고 볼 수 있나요?

A > 아닙니다. 상수만 일치하는 경우는 다음과 같이 처리하시면 됩니다.

```
15149. 0xd8ffd6ef, 0xda3fd911, 0xdc8fd55f, 0xbfdd009f, 0xf1fdfd6e, 0x3fa2
15150. 0xe8efefcf, 0xeb1feaf1, 0xe75fec3f, 0x009fee8f, 0xdfdf0bfef, 0x21ec
15151. 0xf3cf1201, 0xf1b1ff4e1, 0xfd5f1cf3f, 0xcc9fe7f, 0xfad1f900, 0x013f
15152. 0xffe38110, 0x074f06e1, 0x098f086f, 0x24cf0aaf, 0xf0f0e380, 0x4f0e
15153. 0x15ef1300, 0x172f1601, 0x00000048, 0x00000000, 0x00000000, 0x000f
15154. 0x00000000, 0x00000000, 0x02580000, 0x00000000, 0x04000000, 0x001f
15155. 0x02580001, 0x00220001, 0x02580000, 0x00030001, 0x04000000, 0x000f
15156. 0x00000003, 0x00000004, 0x00000005, 0x00000006, 0x00000007, 0x000f
15157. 0x0000000b, 0x0000000c, 0x0000000d, 0x0000000e, 0x00000012, 0x000f
15158. 0x00000016, 0x00000017, 0x00000018, 0x00000019, 0x0000001a, 0x000f
15159. 0x000000e7, 0x0000004b, 0x0000004c, 0x0000004e, 0x00000078, 0x000f
15160. 0x000000ca, 0x000000cd, 0x000000a7, 0x000000aa, 0x000000a8, 0x000f
15161. 0x000000ba, 0x000000bb, 0x000000c6, 0x000000c7, 0x000000be, 0x000f
15162. 0x0000001e, 0x0000001f, 0x00000020, 0x00000021, 0x00000022, 0x000f
15163. 0x00000026, 0x00000027, 0x00000028, 0x00000029, 0x0000002a, 0x000f
15164. 0x0000002f, 0x00000030, 0x00000031, 0x00000033, 0x0000004b, 0x000f
15165. 0x00000037, 0x00000038, 0x00000049, 0x0000004a, 0x0000003a, 0x000f
15166. 0x00000076, 0x00000001, 0x00000010, 0x00000083, 0x00000084, 0x000f
15167. 0x00000050, 0x00000051, 0x00000052, 0x00000053, 0x00000054, 0x000f
15168. 0x00000058, 0x00000059, 0x0000005a, 0x0000005b, 0x0000005c, 0x000f
15169. 0x00000060, 0x00000061, 0x00000062, 0x00000063, 0x00000064, 0x000f
15170. 0x00000068, 0x00000069, 0x0000006a, 0x0000006b, 0x0000006c, 0x000f
15171. 0x00000070, 0x00000071, 0x00000072, 0x00000073, 0x00000074, 0x000f
15172. 0x00000081, 0x00000080, 0x00000087, 0x00000088, 0x00000089, 0x000f
15173. 0x0000008f, 0x00000090, 0x00000091, 0x00000092, 0x00000093, 0x000f
15174. 0x00000097, 0x00000098, 0x00000099, 0x0000009a, 0x0000009b, 0x000f
15175. 0x0000009e, 0x0000009f, 0x000000a0, 0x000000a1, 0x000000a2, 0x000f
15176. 0x000000b0, 0x000000b1, 0x000000b2, 0x000000b3, 0x0000004e, 0x000f
15177. 0x000000ce, 0x000000c1, 0x000000d0, 0x000000d1, 0x000000d3, 0x000f
15178. 0x000000d7, 0x000000d8, 0x000000d9, 0x000000da, 0x000000db, 0x000f
15179. 0x000000d4, 0x000000e0, 0x000000e1, 0x000000e2, 0x000000e3, 0x000f
15180. 0x000000cc, 0x000000cb, 0x000000c3, 0x000000c2, 0x000000c1, 0x000f
15181. 0x000000ea, 0x000000eb, 0x000000ec, 0x000000ed, 0x000000ee, 0x000f
15182. 0x000000f2, 0x000000f3, 0x000000f4, 0x000000f6, 0x000000f7, 0x000f
15183. 0x000000f8, 0x000000f9, 0x000000fa, 0x000000fb, 0x000000fc, 0x000f
466. unsigned long L5[] = { 0xA0E6A6B0, 0x49A70F70, 0x4B7A70E9, 0x
467. CSigs[i++] = NewCryptSig("blowfish", L5, 8, NULL, 0, -4, 0,
468.
469. unsigned long L6[] = { 0x3F258C7A, 0xE213F2F, 0x30FB40D4, 0x
470. CSigs[i++] = NewCryptSig("cast", L6, 7, NULL, 0, -4, 0, CRYP
471.
472.
473. unsigned long L7[] = { 0x7789F87, 0xE6C11218, 0xE216300D, 0x
474. CSigs[i++] = NewCryptSig("cast", L7, 7, NULL, 0, -4, 0, CRYP
475.
476. unsigned long L8[] = { 0x0000002D, 0x00000003, 0x00000004, 0
477. 0x0000000A, 0x0000000B, 0x0000000C, 0x0000000D, 0x0000000E, 0
478. 0x00000002E, 0x00000003, 0x00000004, 0x00000005, 0x00000006, 0
479. 0x0000000A, 0x0000000B, 0x0000000C, 0x0000000D, 0x0000000E, 0
480. 0x000000FA, 0x000000FB, 0x000000FC, 0x000000FD, 0x000000FE, 0
481. 0x00000002, 0x00000003, 0x00000004, 0x00000005, 0x00000006, 0
482. 0x0000000A, 0x0000000B, 0x0000000C, 0x0000000D, 0x0000000E, 0
483. 0x00000013, 0x0000000C, 0x0000000F, 0x0000000E, 0x0000000F, 0
484. 0x00000013, 0x00000014, 0x00000015, 0x00000016, 0x00000017, 0
485. 0x0000001B, 0x0000001C, 0x0000001D, 0x0000001E, 0x0000001F, 0
486. 0x00000023, 0x00000024, 0x00000025, 0x00000026, 0x00000027, 0
487. 0x0000002B, 0x0000002C, 0x0000002E, 0x00000028, 0x00000029, 0
488. 0x0000002D, 0x0000002E, 0x0000002F, 0x00000030, 0x00000031, 0
489. 0x00000035, 0x00000036, 0x00000037, 0x00000038, 0x00000039, 0
490. 0x0000003D, 0x0000003E, 0x0000003F, 0x00000040, 0x00000041, 0
491. CSigs[i++] = NewCryptSig("des", L8, 128, NULL, 0, -4, 0, CRY
492.
493. unsigned long L9[] = { 0x0000002E, 0x0000002F, 0x00000030, 0x
494. 0x00000036, 0x00000037, 0x00000038, 0x00000039, 0x00000041, 0
495. 0x00000045, 0x00000046, 0x00000047, 0x00000048, 0x00000049, 0
496. 0x0000004D, 0x0000004E, 0x0000004F, 0x00000050, 0x00000051, 0
497. 0x00000055, 0x00000056, 0x00000057, 0x00000058, 0x00000059, 0
498. 0x00000063, 0x00000064, 0x00000065, 0x00000066, 0x00000067, 0
499. 0x0000006B, 0x0000006C, 0x0000006D, 0x0000006E, 0x0000006F, 0
500. 0x00000073, 0x00000074, 0x00000075, 0x00000076, 0x00000077, 0
501. CSigs[i++] = NewCryptSig("des", L9, 64, NULL, 0, -4, 0, CRYPT
502.
503. unsigned long L10[] = { 0x20808000, 0x00000000, 0x02000002, 0
```



FAQ License 정보가 있는 파일에 대한 처리

Q1 > Identify 대상 파일 내용이 별거 없는 듯 하지만, 코드 상단에 License 에 대한 내용이 있는 경우 어떻게 처리해야 하나요?

A > 실제 소스코드에는 저작권에 영향을 받지 않는 코드만 들어있더라도, 소스코드 상단의 명시된 라이선스를 따라야 합니다.

```
* This program is free software; you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation; either version 2 of the License, or
* (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program; if not, write to the Free Software
* Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
*
*/
```

GPL license text

```
typedef enum {
    AVDTP_ERROR_ERRNO,
    AVDTP_ERROR_ERROR_CODE
} avdtp_error_type_t;

#ifdef __SAMSUNG_PATCH__
typedef enum {
    AVDTP_SESSION_STATE_DISCONNECTED,
    AVDTP_SESSION_STATE_CONNECTING,
    AVDTP_SESSION_STATE_CONNECTED
} avdtp_session_state_t;
#endif
```

Ignorable code

FAQ Pattern match에 의한 검출건 처리 - (1)

Q2 > 오픈 소스 List 는 없고 다음과 같은 문구만 보입니다. 어떻게 해야 하죠?

This file pattern is set to "Requires ID". You must declare this file on the Bill of Materials tab to resolve it.

A > 파일명에 특정 확장자 혹은 특정 string을 가지고 있을 경우, Code Match 와 상관없이 무조건 검출되도록 한 도구의 Policy 때문입니다.

이렇게 Code Match 여부와 상관없이 검출되도록 하는 이유는 특정 string을 가지고 있는 파일이 오픈소스를 확인하는데 있어 도움이 되기 때문입니다.

예를 들어 파일명에 "*licen*" 이 포함된 파일인 경우, 관련 소스/Library의 License를 명시 해 놓은 파일 경우가 많습니다.

즉, 관련 소스를 Identify 할 때 도움이 될 수 있습니다. 다음 page 에 이어집니다.

FAQ Pattern match에 의한 검출건 처리 - (2)

A > 이전 page 이어서

그러나, 우연히 그러한 string 이 포함되었고 오픈소스 Code Match와 관련이 없다면, 아래와 같이 처리하시면 됩니다

1 Bill of Materials Code Matches Searches Dependencies

File: /Protex_Edu_98_3-Android/external/hostapd-0.8.x/src/ap/ieee802_1x.c

1 Component

Approved	License Conflict	Component
✓		

3 Component: Declared_Ignore Matching

Version: _____

License: _____

Usage: Component

DECLARE file

ORIGINAL TO: Protex_Edu_98_3-Android

4 Save

2 Declare File... Save

“Declared_Ignore Matching” 입력

※ 주의사항

폴더단위 처리 시,
해당 폴더 내 match된 모든 내용이
변경됨 (code match/String search의
identify 된 파일/아직 identify되지 않
은 파일 모두 변경)

따라서 파일단위 진행 필수

FAQ 폴더 단위 처리 - (1)

Q3 > 파일 단위로 하나하나 확인해야 하나요? 한꺼번에 처리할 수는 없을까요?

A > Protex 에서는 파일 외에 폴더 단위로도 Identify가 가능합니다.

폴더 단위로 Identify 하는 경우는 그 폴더 자체를 Open Source Project에서 그대로 가져온 경우에 하실 수 있습니다.

폴더 단위로 Identify 하면 Pending Identification 파일 수가 확 줄어들게 되므로 편리합니다. 단, 출처가 확실한 경우에만 수행하시기 바랍니다.

방법은 다음 Page를 참고하세요.

FAQ 폴더 단위 처리 - (2)

1 external (509)

Match	KB	Component	Version	License	Matched	Unmatched
✓	KB	libxml2	Unspecified	libxml2 License [Reviewed and Approved](and others)	202	0
✓	KB	libxml - The XML C parser and toolkit of Gnome	Unspecified	LGPL 2.1 [Partially Approved](and others)	192	0
	KB	CMISS Bioengineering Modelling Software - cmgui	2.3.3	Mozilla Public License 1.1	164	0
	KB	SafariMobile	Unspecified	Mozilla Public License 1.1	164	0
✓	KB	Libxml2 License	Unspecified	libxml2 License [Reviewed and Approved]	157	0
✗	KB	dia Win32 Installer - libxml2	Unspecified	GPL 2.0 [Reviewed and Rejected]	156	0
	KB	E/OS LX	Unspecified	Public Domain [Reviewed and Approved](and others)	154	0
✓	KB	Chromium Source	Unspecified	BSD 2.0 [Reviewed and Approved]	151	0
✓	KB	Chromium OS	Unspecified	BSD 2.0 [Reviewed and Approved]	149	0
				Apache License Version 2.0		

4 IDENTIFY Snippet and File Matches (202 items pending) (show me pending)

5 Save

Snippet Match는 소스의 일부분만 일치함을 의미하고, **File Match**는 소스전체가 일치하는 것을 의미합니다.

① Identify 하고자 하는 세부 폴더를 선택한 후, ② Code Matches 탭을 엽니다.

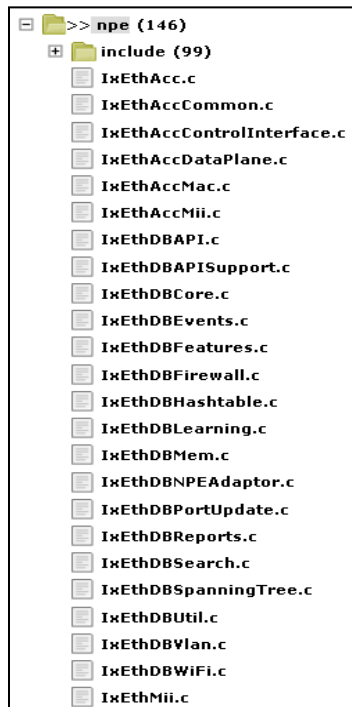
③ 선택된 오픈 소스 Project에서 가져온 게 맞나요? 그렇다면, 해당 Component를 선택하고

④ Identify 하시면 됩니다.

FAQ 특정 Component로 Identify된 파일 확인

Ex. Loke/android_uboot_smdkc110/cpu/ixp/npe/ 이하(146) : component – Copyright 2001-2005, Intel Corporation, license – BSD 2.0 이렇게 표시되어 BSD 2.0 license에 해당하는 것이 어떤 파일인지 확인하고 싶을 때

1. 좌측의 폴더 트리에서 원하는 폴더 선택(클릭)
2. 우측의 Bill of Materials 탭에서 보여지는 Component 리스트 중 원하는 Component 클릭
3. 하단의 Show me 클릭
4. Component에 해당하는 파일들이 좌측 폴더트리에 볼드체로 표시됨.
5. 이후 개별 파일별 내용 확인은 앞에서 설명한 방법 참조



Bill of Materials Interface (Right):

Folder: /10_Android_S1_EUR_i9000_Kernel(eclair)/Source/Loke/android_uboot_smdkc110/cpu/ixp/npe

3 Components (100 per page)

Approved	License Conflict	Component Type	Component	Version	License
✓		LP	Copyright 2001-2005, Intel Corporation	Unspecified	BSD 2.0 [Reviewed and Approved]
✗	⚠	LP	Free Software Foundation	Unspecified	GPL 2.0 [Reviewed and Rejected]

Component: Copyright 2001-2005, Intel Corporation

Version: Unspecified

License: BSD 2.0 [Reviewed and Approved] (view)

License Conflict Status: No License Conflicts

Code Matches ID'd in: 146 file(s) or folder(s) (show me)

Dependencies ID'd in: 0 file(s) or folder(s)

String Searches ID'd in: 0 file(s) or folder(s)

Usage: File (+ Component)

Ship Status: Ship

Component ID: c_1e36c8a9-db48-427d-9fbc-e7ee896a164d

Release ID:

License ID: bsd20

오픈 소스 검증 전반 FAQ

FAQ Dual License

Q : Dual License가 뭐죠?

A : 다음과 같은 사례를 Dual License라고 합니다.

- Matched된다는 원본 소스 파일을 다운 받음 (cairo-1.8.8.zip)
- 소스를 살펴보니 COPYING이라는 파일 존재



Cairo is free software.

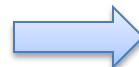
Every source file in the implementation[*] of cairo is available to be redistributed and/or modified under the terms of either the GNU Lesser General Public License (LGPL) version 2.1 or the Mozilla Public License (MPL) version 1.1. Some files are available under more liberal terms, but we believe that in all cases, each file may be used under either the LGPL or the MPL.

See the following files in this directory for the precise terms and conditions of either license:

COPYING-LGPL-2.1
COPYING-MPL-1.1

Please see each file in the implementation for copyright and licensing information, (in the opening comment of each file).|

[*] The implementation of cairo is contained entirely within the "src" and "pixman" directories of the cairo source distribution. There are other components of the cairo source distribution (such as the "test"



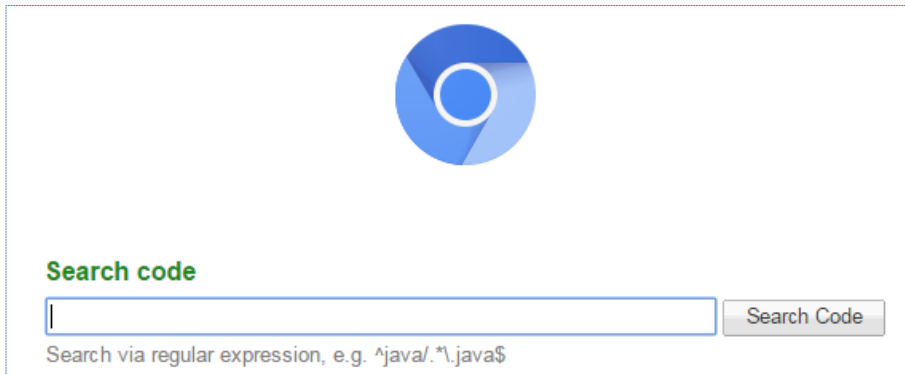
LGPL 2.1 혹은 MPL 1.1에 적용 받으므로 이와 같은 경우에는 MPL 1.1로 선택할 수 있다.
(LGPL 2.1이 공개 범위가 더 크므로)

FAQ 도움되는 사이트

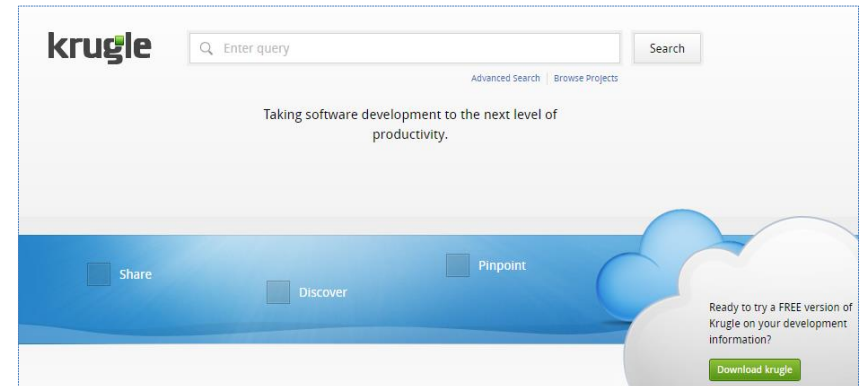
Q : Matched 된 오픈 소스 원본을 찾고 싶는데 어디서 찾아보죠?

A : 각 종 검색사이트에서 적절한 검색어 (파일명, directory명, 파일 내 특정 단어 등)을 입력하여 확인합니다. 그리고 하기와 같이 오픈 소스를 전문적으로 검색 할 수 있는 사이트도 있으니 참고하시기 바랍니다.

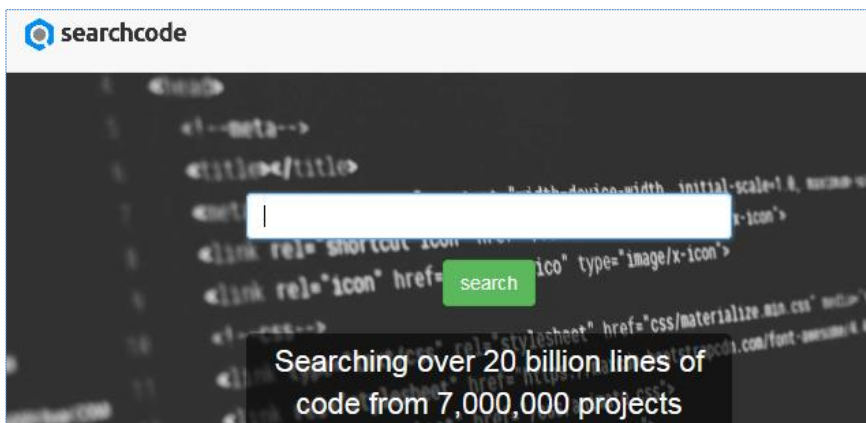
Chromium Code Search (<https://cs.chromium.org>)



Krugle Select (<http://www.krugle.org>)



searchcode (<https://searchcode.com>)



FAQ 사내 오픈 소스 전문가 커뮤니티

Q : 오픈 소스 라이선스에 대해서 좀 더 알고 싶어요, 혹시 사내 커뮤니티가 존재하나요?

A : MOSAIC 내 "[오픈소스 전문가 커뮤니티](http://mosaic.sec.samsung.net/club/102.club)(<http://mosaic.sec.samsung.net/club/102.club>)" 가 존재합니다. 유용한 정보가 많이 있으니 업무 참고하시기 바랍니다.

오픈소스 전문가 Open Source Expert ☆ 즐겨찾기 → 바로가기
<http://mosaic.sec.samsung.net/club/102.club>

전체 게시물 | 검색 | Master 정운환 | 2234 명 | 초대하기 | 설문 보기 | 내 커뮤니티 가기

서명덕님 [정회원]

© (GMT +9:00)

나의 활동정보

총 136회 방문
게시물 6 댓글 5
가입일 2014/04/22

개인정보변경 | 임시저장함

메뉴 | 소모임

전체 글 보기 (1244)
전체 동영상 보기 (0)

공지사항

- 오픈소스 정책/지침
- 오픈소스 검증 도구
- 오픈소스 FAQ
- 질의 및 응답

오픈소스 정책/지침

오픈소스 라이선스 교육자료	2012/08/09	493
오픈소스 SW정책	2012/02/24	403
오픈소스 SW공개지침	2012/02/24	290
오픈소스 라이선스 준수지침	2012/02/24	326
오픈소스 SW라이선스별 의무사항	2012/02/24	337
오픈소스 차별화(분리설계) 지침	2012/02/24	233
오픈소스 라이선스 위반방지 체크리스트	2012/02/24	210

질의 및 응답

[공지] [OSI] DMDUCK (165.213.149.122) 사용...	이재용	2015/10/15	62
[공지] [OSI] 코드 매치 화면이 보이지 않을 경...	김수현	2012/11/12	158
Default License 수정시 처리방법 문의	이낙수	2016/07/08	8

오픈소스 검증 도구

[공지] OSI_3.0.2_Release_101_20160108 (Protex SDK 7.1)	[3]	2015/10/05	675
[공지] AIRS v1.13 20141029 (Protex SDK 6.3.2)		2014/10/29	96
[공지] BSI v2.0 접속 정보 및 사용법 (파일럿 서버 운영 중단)		2014/06/24	143
[공지] Protex Admin 2.0 (Protex SDK 6.3.0)		2013/10/25	347
[공지] OSI_2.4.6_Release_97_20150211 (Protex SDK 6.3)	[8]	2012/04/28	4401
[공지] 질문은 "질의 및 응답" 게시판을 이용해주세요.		2012/02/22	46
AIRS v1.12 20140627 (Protex SDK 6.3.2)		2014/06/27	67

Toybox Release

[공지] Toybox 테스트 결과서		2014/07/15	58
toybox-0.6.0_SR24_20150930		2015/10/02	19
toybox-0.6.0_SR23_20150831		2015/08/31	2

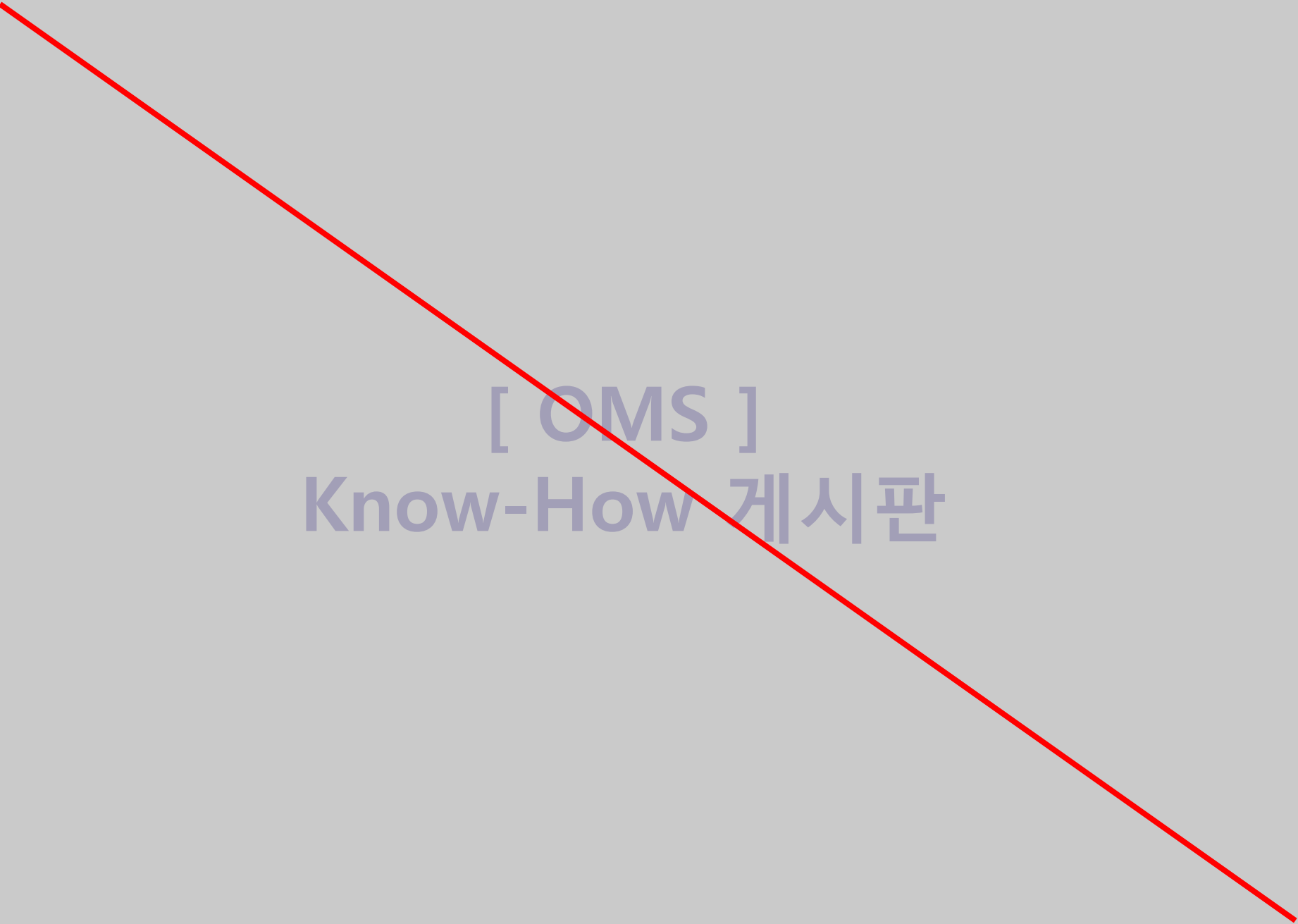
47

FAQ 강력 추천 사이트 – OLIS (for Korean)

한국 저작권 위원회에서 제작한 “오픈 소스SW 라이선스 종합정보 시스템”

: www.olis.or.kr

그 동안 어렵게만 느껴졌던 License 관련 정보를 **한글**로 확인할 수 있습니다.



[OMS]
Know-How 게시판

OMS 오픈 소스 Know-How 게시판

<http://165.213.149.180/oms/?mid=KnowHow>

다양한 case 별 identification Know-How 를 찾아볼 수 있습니다.

OMS Open Source Management System

HOME 검증신청 과제등록 리뷰결과이력판 Review 과제관리 **Process & Guide** Q & A 수평전개 지표 OSS Only

HOME > HOME

Process & Guide

- Process & Guide
- Know-How
- 오픈소스 참고 자료
- 라이선스 가이드

Open Source Verification Process

Android(Base) Android(Derived) Non-Android(Base) Non-Android

PIA DVR PV입고 PVR PRA SRA

Plan Development Production



Thank you !