Bypassing the Integrity Checking of Rights Objects in OMA DRM: a Case Study with the MelOn Music Service

Jusop Choi¹, William Aiken², Jungwoo Ryoo², Hyoungshick Kim¹ ¹Computer Science and Engineering, Sungkyunkwan University, Republic of Korea ²Information Science and Technology, Pennsylvania State University, USA ¹{cjs1992, hyoung}@skku.edu ²{wva5029, jryoo}@psu.edu

ABSTRACT

Commercial digital music is typically distributed in the music source market via Digital Rights Management systems (DRM). DRM systems help remotely control the music contents. The Open Mobile Alliance (OMA) DRM became the de facto standard after major market adoption because of its support for a wide variety of different business and usage models. In OMA DRM, a popular business model is a (monthly) subscription enforced by controlling the period of playback time; once the given period of time expires, the music cannot be played. In this paper, we demonstrate how to bypass the integrity checking of the rights object in the OMA DRM system through a case study of MeIOn (a well-known music distribution service in South Korea) by reverse engineering its media player equipped with a DRM agent.

Categories and Subject Descriptors

K.6.5 [Management of Computing and Information]: Security and Protection—*Digital Rights Management*; D.2.7 [Software Engineering]: Distribution and debugging—*Restructuring, reverse engineering, and reengineering*

Keywords

OMA DRM, MelOn, bypassing, reverse-engineering

1. INTRODUCTION

Digital Rights Management (DRM) is a fundamental technology that makes today's music distribution market possible [14]. DRM technologies allow content providers and publishers to control the whole distribution chain and apply flexible usage rules [8]. Content providers distribute their digital contents online, but after its distribution they can still exercise control of those contents via DRM [13]. DRM systems have been designed to prevent unauthorized access to and use of protected digital content.

For example, DRM systems permit users to play digital music within a predetermined time interval and/or for a predetermined number of times. Such permission/constraint parameters are typically included in a DRM-protected file and governed by a DRM

IMCOM '16, January 04-06, 2016, Danang, Viet Nam © 2016 ACM. ISBN 978-1-4503-4142-4/16/01...\$15.00 DOI: http://dx.doi.org/10.1145/2857546.2857609 agent that is strongly coupled with a music player to access the protected file [9]. To achieve this, DRM-protected files are securely encrypted with a cryptographic key, which can only be decrypted by authorized DRM agents under specific usage rules and conditions.

However, the effectiveness of these DRM technologies in practice is still controversial. Many security experts agree that secure implementation of DRM technology is very tricky, and many commercial DRM systems have proven insecure (e.g., against memory analysis [16]). To make matters worse, even with a secure DRM implementation, there exists an inevitable problem called the *analog hole* [13] which is the duplication of DRM-protected contents by analog means (e.g., recording the analog sound which is produced by a music player).

OMA (Open Mobile Alliance¹) DRM is an open-standards based framework for DRM technologies that supports a wide variety of different usage models, and it has now been widely adopted as the de facto standard for many business applications in the mobile industry [8]. In OMA DRM, one of the most popular business models is a (monthly) subscription by controlling the period of playback time; once the given period of time expires, the music cannot be played. In this paper, we analyze the security of this usage model through a case study of MelOn (a well-known music distribution service similar to iTunes), which has controlled about 39.1% of the online music Korean market share since 2009. According to a 2006 survey [11], the Korean digital music market was worth 191 million USD at that time [11], and it has continuously grown since then. In MelOn, digital music content is securely distributed in the encoded DRM Content Format (DCF), a file format defined by OMA DRM.

This paper examines how to forcibly modify the permissions and constraints (e.g., the time interval constraint) in an OMA DCF file by disabling the integrity checking of the MelOn media player. The integrity check can be bypassed by removing a conditional jump instruction at runtime. By doing this, DRM-protected contents can be used without any usage constraint (e.g., the time interval constraint can be set to any value). Our implementation illustrates how difficult it is to securely implement a DRM agent in the real-world.

This paper is structured as follows. In Section 2, we provide additional information about DRM and OMA DRM used by MelOn. In Section 3, we show how to bypass the OMA DRM protection with a modified DCF file. We present a discussion on countermeasures to mitigate such attacks in Section 4. In Section 5, we discuss the ethical and legal implications of this work. And lastly, in Section 6, we provide our overall conclusion.

¹http://www.openmobilealliance.org

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

2. BACKGROUND

DRM technology aims to achieve the reliable and efficient distribution of digital content over the Internet by providing a trustworthy system to securely enforce the rights of the copyright holders, distribution dealers, and content creators [16].

In terms of DRM, content is defined as valuable intellectual property that should be protected from access by unauthorized users. A user is a subject who can access DRM contents under granted "rights" and "conditions". The "rights" are the permissions granted for DRM content, and the "conditions" include requirements for (and limitations of) how the permissions can actually be executed. By granting specific rights and conditions, content providers can control distributed digital contents as they want. Surely, these specified rights and conditions should not be able to be directly modified by unauthorized entities (e.g., the users). In general, protection from such modifications can be achieved with the implementation of cryptographic algorithms. The use of DRM contents can also be traced and monitored [14].

The main components of DRM typically include the following: the DRM agent, the Content Issuer (CI), the Rights Issuer (RI), and the user [3]. In particular, the roles of these components are very important since the overall security of a DRM system depends on the secure implementation of a DRM agent and its relationship to the other components. The DRM agent is responsible for enforcing permissions and constraints associated with DRM content as well as controlling access to that content; the CI manages the secure delivery of DRM content; the RI defines permissions and constraints associated with DRM contents and creates a rights object for expressing them; and the user is the consumer of the DRM content [3].

DRM solutions are generally classified into non-cryptographic and cryptographic techniques. Non-cryptographic DRM techniques usually operate based on identifying the user by physical aspects, usually relying on manuals, disks, or other hardware for authentication. On the other hand, cryptographic DRM techniques operate based on confirming that the user has certain access rights, using a cipher when a user tries to access any DRM content. This approach is especially useful when content distribution is done digitally, and as a result, most recent DRM services have been launched based on cryptographic techniques [16].

The digital music market has been one of the biggest adopters of cryptographic DRM techniques. Since digital files can be easily redistributed (e.g., through peer-to-peer services), many music marketplaces have no choice but to use DRM systems to enforce their end-user license agreements. However, even with DRM, some possibilities of unauthorized distribution still exist [13, 16]. Moreover, DRM is unpopular with end-users because DRM makes digital contents very uncomfortable to use and causes compatibility issues with various devices, media players, and media file formats [6, 10]. Nonetheless, music source companies still want to maintain their DRM systems because the use of DRM systems are currently regarded as the best efforts to limit unauthorized use of their digital content (e.g., unauthorized distribution to others).

2.1 OMA DRM

OMA DRM was invented by the Open Mobile Alliance with several stakeholders, such as content providers, licence providers, and device manufacturers to support a wide variety of different usage models.

The OMA DRM 1.0 specification was first released in 2002 as the world's first open DRM standard for mobile devices [2]. As a follow-up, OMA DRM 2.0 was developed in 2006, with a focus on a wide variety of different distribution mechanisms to enhance end-user experiences. The DRM architecture document particularly specified a file format named the DRM Content Format (DCF) to represent a file format for securely downloading and controlling DRM contents, which are based on the ISO base media file format (ISO14496-12) [2].



Figure 1: The OMA DRM 2.0 functional architecture (adapted from [2]).

The functional architecture of OMA DRM 2.0 is illustrated in Fig. 1. As shown in this figure, when a user downloads DRM-controlled content from the CI, the associated rights object, which includes a cryptographic key for accessing the DRM content, can be delivered from the RI in a separate manner. Also, the DRM content can be re-distributed in a local network through a distribution mechanism called *super-distribution*. Those delivery mechanisms were designed to enhance flexibility and convenience for sharing DRM-protected contents through peer-to-peer networks [2].



Figure 2: Structure of DCF (adapted from [1]).

Fig. 2 shows the structure of a DCF file. As shown in this figure, the DCF file contains one or more containers that comprise the encrypted digital content (i.e., the content object) with meta-data that represent information such as author, title, media type and the URL to obtain the associated rights object to unlock the content. The rights object is typically implemented as an XML file that describes the permissions and constraints granted to a DRM agent when accessing a specific DCF file. Various permissions and constraints can be defined by Rights Expression Language (REL) [12]. The permission information specifies how a digital content can be accessed such as play, display, execute, print, and export; the constraint information specifies the number of times the content can be accessed and/or the expiration time of the content. The rights object also contains the content encryption key needed to decrypt the encrypted digital content. The rights object is integrity-protected by a Message Authentication Code (MAC) such as HMAC-SHA1 and contains a list of content object identifiers and their respective usage permissions. The MAC key is securely protected using a PKI mechanism. Only the authorized DRM agent with its own private key can access the MAC key and the content encryption key [15].

2.2 MelOn Service

MelOn², a popular online music store in South Korea, uses OMA DCF files to securely distribute its music files and control them with predefined permissions and constraints (see Section 2.1). In this service, downloaded DCF files can only be played with a paying user's MelOn music player (see Fig. 3) — the MelOn player, a program which can be downloaded from the MelOn website, is necessary to use MelOn services.

| MelOn Player * 🖞 Login * 📃 B 🗆 X | | | | | | | | | |
|----------------------------------|---|-------------------------|-----|-------------------------------|---------|----------------|---------------------------|--------------|-----|
| PLAYER WEB | | MelÓn 44 M | | | | | | | |
| R +4 | | (R | | ✔ 검색어를 입력해주세요. | | 0 | | | |
| 🖻 마이뮤직 | | | | | | | | | 0 |
| 18 현재재명목록 | 1 | 열론차드 > 일론 TOP 100 + 실시간 | | | | | | | |
| 0、関展為単 | | □ 윤위 | 변동 | 곡명 | A122 | 아티스트 | 앨명 | 좋아요 | 뮤 |
| | | | (B) | B 위험해 (SO, DANGEROUS) | 03:29 | 입텐션 (UP10T) | - 紙秘密(留音問留) | @1.274 | |
| · 실시간 | H | 0.1 | - 0 | 6 건용 | 03:21 | 이유 관지(GOD | 무한도전 영동고속도로 가 | @116,692 | |
| · 금상승 | | 2 | - 0 | lla 컵 (Feat, 태양) | 04:09 | 승민호 | 쇼미더머니 4 Episode 5 | @ 60,494 | |
| · 월간 | | 3 | - 0 | 🚯 Dumb Dumb | 03:22 | Red Velvet (2) | The Red - The 1st Album | @ 26,078 | |
| · 주2: | | □ 4 | - 0 | 타 맙소사 | 03:11 | 황태지 (황광희, | 무한도전 영동고속도로 가 | @ 81,680 | |
| · 82 | v | 5 | - 0 | 🗈 Lion Heart | 03:44 | 소녀시대 | Lion Heart - The 9h Album | @48.344 | |
| | Ľ | 6 | +1 | 屆 오늘부터 우리는 (Me Gustas | 03:40 | 여자친구 | 여자전구 2nd Mini Album | @34,104 | |
| | | 07 | +1 | 屆 스뽄서 (\$ponsor) | 04:05 | 으뜨거따시 (하 | 무한도전 영동고속도로 가 | @ 50,186 | |
| | | 8 | - 0 | 局 오빠차 | 03:13 | 인크레더블,타블 | 쇼미더머니 4 Episode 4 | (@ 32,905 | |
| | | 9 | +1 | E Okey Dokey | 84:08 | 송민호,지코 (Zl | 쇼미더미니 4 Episode 6 | @ 20,304 | |
| | | 10 | +1 | 品 거북선 (Feat 팔로알토) | 04:05 | 지베즈 (Ja Mez | .ΩOICHOILI 4 Episode 2 | @ 48.897 | |
| | | 0.11 | +1 | 토 위일위일 | 03:13 | 핵요 (hyukoh) | 20 | @ 106.832 | |
| | | 4 | | | | | | | .P. |
| | | ✓ 전체관 | 控 | 문기 🍐 디운 🔸 담기 🎂 선물 🕨 T | 0P100 전 | | | 101 〒 06:12: | 8 |
| · (*) 연결시작 · (1) 컨버팅 | | | | | | | h. | | |

Figure 3: MelOn player interface.

MelOn launched a service called "*Free-Club*" to offer unlimited downloads and unlimited music streaming on a monthly basis. If the user's subscription is expired and is not renewed, then the user's MelOn music player will no longer play the downloaded DCF files when the expiration date of the user's subscription, explicitly given in those files, has passed.

3. BREAKING THE DRM PROTECTION IN MELON

Before playing a DCF file, the DRM agent, embedded in the MelOn media player, checks whether the DCF file should be played with its current permissions and constraints. For example, once the downloaded DCF file is past its expiration date, that DCF file cannot be played anymore. In this paper, however, we show how difficult it is to deploy an access policy-based DRM implementation in this way by breaking the DRM protection in the MelOn player. In our experiments, we used a PC (with running 32-bit Windows 7).

Overall, there are two ways of breaking the OMA DRM protection; we need to bypass either (1) the checks for permissions and constraints on a DCF file or (2) the check for the integrity of the rights object in a DCF file. In general, the check codes for (1) can be located in several different regions (e.g., the check code for subscription expiration time is different from the check code for count limitation). Therefore, for a more generalized breaking method, we focus on how we bypass the check code for (2).

This process can be subdivided into two phases: (A) identifying the locations of permissions and constraints in a rights object, and (B) disabling the integrity check in the MelOn media player application. Each of these phases will be discussed in the following subsections.

The first step is to identify the locations of permissions and constraints, including (but not limited to) the subscription expiration date, to be modified in a DCF file. To achieve this goal, we can compare the headers of DCF files with different permissions and constraints.

| 100001D0 | 52 46 43 3 | 2 36 3 | 13 30 3E | 65 56 | 61 6C 7 | 75 65 3D | 22 | RFC2630;eValue=" |
|--|---|--|--|--|--|---|--|--|
| 00001E0 | 41 39 39 3 | 4 44 4 | 1 31 38 | 41 37 | 32 32 3 | 34 41 36 | 41 | A994DA18A7224A6A |
| 00001F0 | 35 41 45 3 | 7 46 4 | 16 45 38 | 42 39 | 43 35 3 | 34 33 31 | 43 | 5AE7FFE8B9C5431C |
| 0000200 | 44 31 41 3 | 5 45 3 | 0 39 31 | 44 41 | 36 38 4 | 16 34 30 | 35 | D1A5E094DA68F405 |
| 0000210 | 32 36 30 3 | 5 43 3 | 15 35 36 | 38 31 | 42 35 4 | 1 35 41 | 36 | 2605055681858586 |
| 0000220 | 22 0D 0A 5 | 5 73 6 | 1 67 65 | 2D 43 | 6F 6E 7 | 4 72 6F | 6C | " Usage-Control |
| 0000230 | 3A 54 69 6 | D 65 2 | 2D 4E 6F | 74 2D | 41 66 7 | 4 65 72 | 2F | :Time-Not-After/ |
| 10000240 | 32 30 31 3 | 4 30 3 | 7 33 36 | 32 33 | 35 39 3 | 15 39 5A | 2B | 281487382359597+ |
| 10000250 | 38 39 26 4 | 1 66 6 | IC 6F 77 | 61 62 | 60 65 2 | PD 53 65 | 72 | 09.Allowable-Ser |
| 0000260 | 76 69 63 6 | 5 73 2 | F 30 30 | 30 30 | 30 30 3 | 80 30 3B | 63 | vices/00000000:c |
| 0000270 | 69 64 54 7 | 9 70 6 | i5 3D 4C | 49 4E | 3B 65 4 | 1 75 74 | 68 | idTune=MIN:eAuth |
| 10000280 | 43 6F 64 6 | 5 3D 2 | 2 53 53 | 65 41 | 75 74 f | 8 2D 31 | 2F | Code="SSeAuth-1/ |
| 0000290 | 31 38 64 6 | 5 37 6 | 5 30 61 | 61 30 | 65 39 3 | 39 64 61 | 31 | 10de7e0aa0e99da1 |
| 0000240 | 32 37 61 3 | 8 22 6 | ID ØA 40 | 65 74 | 61 2D 4 | 13 6F 6E | 74 | 27a8" Meta-Cont |
| 0000000 | 61 69 6E 6 | 5 72 3 | A 48 53 | 5F 43 | 5F 35 3 | 6 30 31 | 2D | ainer:KS_C_5601- |
| ппппл2Вп | | 2 38 4 | 1D 45 4F | 55 3D | 22 6D 7 | 0 33 22 | 38 | 1992:MENII="mn3": |
| 10000280 100002C0 | 31 39 39 3 (a) Ti | me-1 | Not- | Afte | er = 2 | 20140 | 73(|) |
| 10000280 100002C0 | 31 39 39 3 (a) Ti | me-1 | Not- | Afte | er = 2 | 20140 | 73(|) |
| 19999229 19999229 199999109 | 31 39 39 3 (a) Ti: | me-1 | Not- | Afte | er = 2 | 20140 [°] '5 65 3D | 73(|) RFC2630;eValue="] |
| 19999229 19999229 199999209 199999129 | 31 39 39 3 (a) Ti 52 46 43 3 35 36 45 3 | me-1 | Not- 83 30 3E | Afte | er = 2 $61 \ 6C \ 7$ $34 \ 31 \ 4$ | 20140 <u>'5 65 3D</u> 13 34 34 | 73(22 42 |) RFC2630;eValue=" 56E26731AE41C448 |
| 00000280 00000200 00000100 000001E0 000001E0 | 31 39 39 3 (a) Ti: 52 46 43 3 35 36 45 3 43 39 37 3 | me−1 2 36 3 2 36 3 5 45 3 | Not- | Afte | $e_{r} = 2$ $\frac{61 \ 6C}{34 \ 31} \frac{7}{4}$ $\frac{36 \ 41}{4} \frac{41}{4}$ | 20140 <u>5 65 3D</u> 13 34 34 12 42 42 | 73(22 42 45 |) RFC2630;eValue=" 56E26731AE41C44B C975E529EC6ABBBE |
| 000002C0 000002C0 000001D0 000001E0 000001F0 000001F0 | 31 39 39 3 (a) Ti: 52 46 43 3 35 36 45 3 43 39 37 3 42 36 41 3 | me−1 2 36 3 2 36 3 5 45 3 7 38 3 | Not- 33 30 3E 37 33 31 35 32 39 36 34 36 | Afte | $e_{r} = 2$ $\frac{61 \ 6C}{34 \ 31} \frac{7}{4}$ $\frac{36 \ 41}{36 \ 38} \frac{4}{38}$ | 20140' ^{75 65 3D} ^{13 34 34 34} ^{12 42 42 42} ^{15 31 36} | 73(22 42 45 41 |) RFC2638;eUalue=" 56E26731AE41C448 C975E529EC6ABBBB B6A7864486E68516A |
| 00000200 00000200 00000100 000001E0 000001E0 000001E0 000001E0 00000210 | 31 39 39 3 (a) Ti 52 46 43 3 35 36 45 3 43 39 37 3 42 36 41 3 41 43 31 3 | me−1 2 36 3 2 36 3 5 45 3 7 38 3 7 33 3 | Not- 33 30 3E 37 33 31 35 32 39 36 34 30 38 38 37 | Afte | $e_{r} = 2$ $\frac{61 \ 6C}{34 \ 31 \ 4}$ $\frac{36 \ 41}{36 \ 38}$ $\frac{43}{33}$ $\frac{33}{33}$ | 20140' ^{75 65 3D} ^{13 34 34} ^{12 42 42} ^{13 31 36} ^{13 41 44} | 73(22 42 45 41 33 |) RFC2638;eUalue=" 56E26731AE41C448 C975E529EC6ABBBE B6A786406E68516A Ac1738872EC39AD3 |
| 00000200 00000200 00000100 00000100 00000100 00000200 00000210 00000210 | 31 39 39 3 (a) Ti: 52 46 43 3 35 36 45 3 43 39 37 3 42 36 41 3 41 43 31 3 22 00 06 5 | me−1 2 36 3 2 36 3 5 45 3 7 38 3 7 38 3 7 33 3 5 73 6 | Not- 33 30 38 37 33 31 35 32 39 36 34 30 38 38 37 31 67 65 | Afte 65 56 41 45 45 43 36 45 32 45 20 43 | er = 2 $\frac{61}{34} \frac{6C}{31} \frac{7}{4}$ $\frac{36}{36} \frac{31}{41} \frac{4}{4}$ $\frac{36}{38} \frac{38}{33} \frac{3}{3}$ $\frac{6F}{6E} \frac{7}{7}$ | 20140' ^{75 65 3D} ^{13 34 34} ^{14 42 42 42} ^{15 31 36} ^{19 41 44} ^{14 72 6F} | 73(42 45 41 33 60 |) RFC2630;eValue=" 56E26731AE416448 C975E522E66AB8BE 86A786406E68516A AC1738872EC39AD3 " Usaqe-Control |
| 300000200 300000200 300000100 3000001E0 3000001F0 300000210 300000210 300000220 300000220 | 31 39 39 3 (a) Ti 52 46 43 3 35 36 45 3 43 39 37 3 42 36 41 3 41 43 31 3 22 60 64 5 36 54 5 41 5 41 5 41 5 41 5 41 5 41 5 41 5 | me-1 2 36 3 2 36 3 5 45 3 7 38 3 7 33 3 5 73 6 0 65 2 | Not- 33 30 38 37 33 31 35 32 39 36 34 30 38 38 37 31 67 65 20 46 6F | Afte 65 56 41 45 45 43 36 45 32 45 20 43 74 20 | $e^{r} = 2$ $61 \ 6C \ 7$ $34 \ 31 \ 4$ $36 \ 41 \ 4$ $36 \ 38 \ 3$ $43 \ 33 \ 33$ $6F \ 6E \ 7$ $41 \ 66 \ 7$ | 20140 5 65 3D 13 34 34 12 42 42 12 31 36 19 41 44 74 72 6F 74 65 72 | 73(42 45 41 33 60 2F |) RFC2630;eUalue=" 56226731AE41C44B C975E529EC6ABBBB 56A7864406668516A AC1738872EC39AD3 "Usage-Control :Time-Not-After/ |
| 00000200 00000200 00000100 00000100 00000100 00000200 00000200 00000220 00000220 00000220 00000220 | 31 39 39 3 (a) Ti: 52 46 43 3 35 36 45 3 43 39 37 3 42 36 41 3 42 36 41 3 41 43 31 3 22 00 04 5 34 54 69 6 32 30 31 3 | me-1 2 36 3 2 36 3 5 45 3 7 38 3 7 4 8 3 7 4 8 3 7 4 8 4 8 4 8 4 8 4 8 4 9 | Not- 33 30 3E 37 33 31 35 32 39 36 34 30 38 38 37 51 67 65 20 4E 6F 34 33 30 | Afte 65 56 41 45 45 43 32 45 20 43 74 20 32 33 | $e^{r} = 2$ $61 \ 6C \ 7$ $34 \ 31 \ 4$ $36 \ 41 \ 4$ $43 \ 33 \ 3$ $6F \ 6E \ 7$ $41 \ 66 \ 7$ $35 \ 39 \ 3$ | 20140 5 65 3D 13 34 34 14 242 14 25 31 14 44 74 72 6F 74 65 72 15 39 5A | 73 22 42 45 41 33 60 2F 2B |) RFC2638;eUalue="" 6626731AE41C44B C975E529E06ABBBE B6A7864406C68516A c1738872EC39AD3 "Usaqe-Control :Time-Not-After/ 20154042025952+ |
| 00000280 00000200 00000100 00000100 00000200 00000210 00000220 00000220 00000220 00000220 00000220 | 31 39 39 3 (a) Ti | me-1 2 36 3 2 36 3 2 36 3 5 45 3 7 38 3 7 3 6 3 7 4 5 7 6 5 7 6 5 7 6 7 6 7 6 7 6 7 6 7 6 7 | Not- 33 30 3E 37 33 31 35 32 39 36 34 37 36 38 38 37 51 67 65 20 4E 6F 34 33 36 56 6F 77 | Afte 45 56 41 45 45 43 36 45 32 45 32 45 32 43 74 20 32 33 61 62 | $e_{r} = 2$ $\frac{61}{34} \frac{6C}{31} \frac{7}{41}$ $\frac{43}{36} \frac{38}{33} \frac{3}{33}$ $\frac{43}{36} \frac{38}{33} \frac{3}{33}$ $\frac{6F}{6E} \frac{7}{7}$ $\frac{41}{35} \frac{69}{39} \frac{3}{35}$ $\frac{6C}{65} \frac{65}{2}$ | 20140 5 65 3D 3 34 34 42 42 42 5 31 36 39 41 44 74 72 6F 74 572 5 39 5A 20 53 65 | 73 42 42 45 41 33 60 2F 2B 72 |) 56E26731RE41C44B C975E529EE668516A B6478640668516A AC1728872EC39003 :'Usaqe-Control :'Lime-Not-After/ 2015643022595924 09,Allowable-Ser |
| 00000280 00000200 00000120 00000200 00000200 00000200 00000220 00000220 00000220 00000220 00000220 00000220 00000240 00000250 | 31 39 39 3 (a) Ti: 52 46 43 3 35 36 45 3 43 39 37 42 36 41 3 43 39 37 3 42 36 41 3 42 36 90 9 36 32 30 31 33 36 35 36 45 34 | me - 1 2 36 3 2 36 3 2 36 3 5 45 3 7 33 3 5 73 6 0 65 2 5 30 3 1 6C 6 5 73 2 | Not- 33 30 3E 37 33 31 35 32 39 36 34 30 38 38 37 31 67 65 20 4E 6F 34 33 30 46 37 65 20 4E 77 27 30 36 | Afte 65 56 41 45 45 43 36 45 32 45 20 43 74 20 32 36 132 33 | er = 2 61 6C 7 34 31 4 36 41 4 36 38 3 43 33 3 6F 6E 7 41 66 7 35 39 3 36 38 3 36 38 3 37 4 43 38 3 43 38 3 43 38 3 43 38 3 43 38 3 43 38 3 43 38 3 44 4 45 4 46 4 47 4 48 4 4 | 20140 ⁷ 5 65 3D 13 34 34 14 2 42 5 31 36 5 41 44 74 72 6F 74 65 72 85 39 5A 15 365 30 38 30 38 | 73 42 45 41 33 60 2F 28 72 63 |) 562267318E41C448 C975E529EC6ABBBE 8667864406C685164 401738872C239AD3 "Usage-Control 11ne-Not-After/ 201504302359592+ 09,Allowable-Ser vices/80080080; |
| 19999269 19999269 19999269 199996169 199996169 19999269 19999239 19999229 19999229 19999229 19999229 199992629 199992629 199992629 199992629 199992629 | 31 39 39 (a) Ti 52 46 43 35 36 45 35 36 45 343 39 37 42 36 41 31 31 31 22 60 64 32 36 31 36 39 22 41 43 31 22 60 64 33 31 31 36 39 22 36 31 33 36 39 22 47 66 69 63 69 64 54 7 | me - 1 2 36 3 2 36 3 5 45 3 7 33 3 5 73 6 0 65 2 5 30 3 1 6C 6 5 73 2 9 70 6 | Not- 33 30 3E 37 33 31 35 32 39 36 34 30 38 38 37 31 67 65 20 42 6F 34 33 36 36 6F 77 2F 30 30 35 30 40 | Afte 4 65 56 41 45 45 43 36 45 20 43 74 20 32 33 61 62 38 38 49 4E | $e_{T} = 2$ $\frac{61}{34} \frac{60}{34} \frac{7}{34} \frac{7}{34} \frac{7}{36} \frac{7}{$ | 20140' 5 65 3D 13 34 34 12 42 42 15 31 36 15 35 15 36 15 36 16 | 73(22 42 45 41 33 60 2F 2B 72 63 68 |) BFC2630;eValue=" 56226731AE41C44B C975E529EC648B86 B6478640668516A AC1728872EC29003 "Usaqe-control "Isane-Not-After/ 20150430235952+ 0,Allouable-Ser vices/00000000;c IdType=Ntn:eauth |
| 10000260 10000200 100000100 10000200 10000200 10000220 10000220 10000220 10000250 10000250 10000250 10000250 | 31 39 39 3 (a) Ti: 35 36 43 39 37 35 36 45 3 39 37 3 42 36 41 3 39 37 3 42 36 41 3 41 43 34 43 36 41 3 42 30 90 5 36 54 59 63 54 59 63 54 59 63 6 32 30 31 33 39 22 4 36 36 36 36 36 36 36 36 36 36 36 69 63 6 64 54 7 6 9 36 6 64 64 64 54 36 6 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 <td< th=""><th>me] 2 36 3 2 36 3 5 45 3 7 38 3 5 73 6 5 73 6 5 73 6 7 8 6 5 73 6 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8</th><th>Not- 33 30 3E 37 33 31 35 32 35 36 34 36 38 38 37 31 67 65 30 4E 6F 34 33 36 55 30 4E 55 30 4E 22 53 53</th><th>Afte 4 65 56 41 45 45 43 36 45 32 45 32 43 74 20 32 33 61 62 38 30 38 36 39 48 49 48 49 48</th><th>$e_{T} = 2$ $\frac{61 \ 6C}{34 \ 31} \ \frac{3}{33} \ \frac{3}{3} \$</th><th>20140' 5 65 30 13 34 34 14 24 42 42 15 31 36 19 41 44 14 72 6F 14 65 72 15 39 5A 10 53 65 10 30 3B 11 75 74 18 2D 31</th><th>73(22 42 45 41 33 60 2F 2B 72 63 68 2F</th><th>) S62267318E41C448 C975E529EC688B86 B6678644066685166 Act738872EC39A03 "Usage-Control Iime-Not-After/ 2015043023595924 09,Allouable-Ser vices/0000000; IdType-HIN;eAuth- Code="XSeAuth-1/</th></td<> | me] 2 36 3 2 36 3 5 45 3 7 38 3 5 73 6 5 73 6 5 73 6 7 8 6 5 73 6 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 | Not- 33 30 3E 37 33 31 35 32 35 36 34 36 38 38 37 31 67 65 30 4E 6F 34 33 36 55 30 4E 55 30 4E 22 53 53 | Afte 4 65 56 41 45 45 43 36 45 32 45 32 43 74 20 32 33 61 62 38 30 38 36 39 48 49 48 49 48 | $e_{T} = 2$ $\frac{61 \ 6C}{34 \ 31} \ \frac{3}{33} \ \frac{3}{3} \ $ | 20140' 5 65 30 13 34 34 14 24 42 42 15 31 36 19 41 44 14 72 6F 14 65 72 15 39 5A 10 53 65 10 30 3B 11 75 74 18 2D 31 | 73(22 42 45 41 33 60 2F 2B 72 63 68 2F |) S62267318E41C448 C975E529EC688B86 B6678644066685166 Act738872EC39A03 "Usage-Control Iime-Not-After/ 2015043023595924 09,Allouable-Ser vices/0000000; IdType-HIN;eAuth- Code="XSeAuth-1/ |
| 10000220 10000220 10000100 10000110 10000110 10000210 10000220 1000020 1000020 1000020 1000020 1000020 1000020 1000020 1000020 1000020 1000020 1000020 1000020 10000000 10000000 10000000 10000000 10000000 10000000 10000000 10000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 1000000 10000000 100000000 | 31 39 39 3 (a) Ti 35 36 43 3 35 36 45 3 43 39 33 42 36 43 34 31 3 32 30 31 31 32 30 31 33 30 32 30 31 30 30 32 30 31 30 30 32 30 31 30 30 22 45 45 7 45 66 69 63 69 63 69 64 54 7 43 66 45 64 54 7 43 66 45 66 45 67 43 66 44 34 44 34 44 34 44 36 36 66 66 66 66 66 66 66 66 66 66 66 66 66 66 66 | me - 1 2 36 3 2 36 3 5 45 3 7 38 3 7 3 3 8 3 7 3 2 9 70 3 7 3 2 9 70 2 7 30 2 7 30 3 7 3 2 7 38 3 7 3 2 7 3 3 7 3 3 3 7 3 3 7 3 3 3 7 3 3 7 3 3 3 | Not- 33 30 3E 37 33 31 35 32 35 36 34 36 36 38 38 37 36 38 38 37 36 48 6F 34 33 36 57 48 42 37 43 42 37 43 42 37 43 42 38 55 37 43 42 39 55 30 55 | Afte 41 45 45 43 36 45 32 45 32 45 32 45 32 45 32 45 32 33 61 62 33 36 49 4E 65 41 37 37 | er = 2 34 31 4 36 41 4 36 38 3 43 33 3 67 66 7 41 66 7 36 39 3 67 65 2 38 30 3 38 65 4 75 74 6 44 35 3 | 20140' 5 65 30 3 34 34 4 2 42 42 5 31 36 9 41 44 4 72 6F 4 65 72 5 39 5A 5 39 5A 5 39 5A 10 53 65 10 30 3B 11 75 74 8 20 31 36 22 31 36 32 41 | 73(22 42 45 41 33 60 2F 2B 72 63 68 2F 42 |) RFC2630;eValue=" 56220731AE41CA4B C975E529E0648BB6 860786440668516A AC1728872EC29A03 "Usage-Control 'Line-Not-After/ 20150402835952+ 09,Allowable-Ser vices/00000005; CdType-HIN;eAuth Code-"SseAuth-1/ An4F97CB77D562AB |
| 10000220 10000220 10000220 10000210 10000210 10000210 10000220 1000020 1000020 1000020 1000020 10000020 10000020 10000020 100000000 10000000 10000000 100000000 | 31 39 39 3 (a) Ti | me - 1 2 36 3 2 36 3 2 36 3 5 45 3 7 38 3 5 73 6 5 73 2 5 30 3 1 6C 6 5 73 2 5 30 3 1 6C 6 5 73 2 9 30 2 6 39 3 9 22 6 | Not- 33 30 3E 37 33 31 35 32 35 36 34 30 38 38 37 51 67 65 20 4E 6F 43 33 36 46 6F 77 27 30 36 55 30 40 22 53 53 37 43 42 30 0 4 40 30 40 3 | Afte 45 56 41 45 36 45 32 45 32 45 32 45 32 45 32 45 32 43 36 43 32 45 32 43 36 43 32 45 32 45 32 45 32 45 32 45 32 45 34 43 36 45 32 45 36 45 36 45 36 45 37 4 20 38 30 49 4E 37 45 37 45 37 45 37 45 37 45 37 45 38 30 49 45 37 45 3 | $e_{T} = 2$ $\frac{61}{34} \frac{60}{34} \frac{7}{34} \frac{7}{34} \frac{7}{34} \frac{7}{34} \frac{7}{34} \frac{7}{34} \frac{7}{35} \frac{7}{39} \frac{7}{36} \frac{7}{36} \frac{7}{36} \frac{7}{37} \frac{7}{44} \frac{7}{36} \frac{7}{36} \frac{7}{37} \frac{7}{74} \frac{6}{44} \frac{44}{35} \frac{3}{36} \frac{7}{20} \frac{7}{44} \frac{7}{36} \frac{7}$ | 20140 ⁷ 5 65 30 13 34 34 12 42 42 15 31 36 19 41 44 14 72 65 15 30 57 15 30 58 10 30 38 11 75 74 18 20 31 10 63 22 41 13 6F 6E | 73(22 42 45 41 33 60 2F 2B 72 63 63 63 63 2F 42 72 72 72 72 72 72 72 72 72 7 |) S6E26731RE41C448 C975E529EC6A8B86 B6A786A96668516A AC1738872EC39AD3 IIme-Not-After/ 201504302355522 IIme-Not-After/ 20150430235552 IIme-Not-After/ 20150430235552 IIme-Not-After/ 20150430235552 IIme-Not-After/ 20150430235552 IIme-Not-After/ 20150430235552 IIme-Not-After/ 20150430235552 B0430416-201 E009 Heta-Cont |
| 100002260 100002200 10000200 10000200 10000200 10000210 10000220 10000220 10000220 10000220 10000220 10000220 10000220 10000220 10000220 | 31 39 39 3 (a) Ti: 52 46 43 3 35 36 45 3 42 36 45 3 42 36 44 3 14 33 97 3 36 54 64 6 31 41 33 41 36 39 20 4 36 39 20 4 36 45 4 76 69 56 4 37 6 69 56 4 43 6F 64 64 54 7 31 41 54 4 45 30 54 3 | me - 1 2 36 3 2 36 3 5 45 3 7 33 3 5 73 6 0 65 2 5 73 2 9 70 6 5 3D 2 6 3D 2 7 32 3 7 3 3 7 3 3 7 3 3 7 3 2 7 3 3 7 3 2 7 3 3 7 3 3 7 3 5 7 5 | Not- 33 30 3E 37 33 31 35 32 35 36 34 30 38 38 37 36 34 30 38 38 37 36 34 30 36 34 30 36 34 30 36 43 30 37 43 42 30 64 45 30 66 45 30 66 30 | Afte 4556 4145 4543 3645 3245 32245 323 7420 3233 6162 3836 494E 6541 3737 65743 | $e_{T} = 2$ $\frac{61}{34} \frac{60}{34} \frac{7}{34} \frac{7}{34} \frac{7}{34} \frac{7}{34} \frac{7}{35} \frac{7}{35} \frac{7}{35} \frac{7}{36} \frac{7}{$ | 20140 ⁷ ¹⁵ ⁶⁵ ^{3D} ¹³ ³⁴ ³⁴ ¹⁴ ¹⁴ ¹² ⁴² ⁴² ¹⁴ ¹⁴ ¹² ⁴² ⁴² ¹⁴ ¹⁴ ¹⁷ ²⁶ ⁵⁷ ¹⁵ ¹⁵ ⁵³ ⁶⁵ ¹⁶ ³⁰ ^{3B} ¹⁷ ¹⁷ ⁵⁷ ⁴⁴ ¹⁶ ³² ²⁴¹ ¹³ ¹⁶ ¹⁶ ¹⁶ ¹⁶ ¹⁶ ¹³ ¹⁶ ¹⁶ ¹⁶ ¹⁶ ¹³ ¹⁶ ¹⁵ ¹⁶ ¹⁶ ¹³ ¹⁶ ¹⁵ ¹⁶ ¹⁶ ¹³ ¹⁵ ¹⁶ ¹⁶ ¹³ ¹⁶ ¹⁵ ¹⁶ ¹⁶ ¹³ ¹⁶ ¹⁵ ¹⁶ ¹⁵ ¹⁶ | 73(22 42 45 41 33 60 2F 28 72 63 68 2F 42 74 20 |) 56220731AE41C44B C975E529E504E504BBBE 56786440668516A AC1728872EC29AD3 'UsageControl 'Line-Not-After/ 29158432959524 09,Allowable-Ser Uses/000000080; IdType-MIN:eAuth Code-"SSeAuth-1/ IAH570CB77D562AB E040' Heta-Cont aner:KS_C56MT- |
| 10000220 10000220 10000100 10000100 10000210 10000220 10000220 10000220 10000220 10000220 10000220 10000220 10000220 | 31 39 39 3 (a) Ti. 52 46 43 3 55 36 45 3 43 39 37 3 42 36 41 3 41 43 31 3 22 00 06 5 30 52 46 5 41 3 31 3 22 00 06 5 30 52 46 5 30 52 46 5 30 52 6 41 3 41 43 31 3 22 00 06 5 30 52 6 41 3 41 45 30 43 3 51 49 5 43 6 46 5 31 41 34 4 45 30 34 3 51 90 5 6 | me - 1 2 36 3 2 36 3 2 36 3 5 45 3 7 33 3 5 73 6 0 65 2 5 73 2 9 70 6 5 73 2 9 70 6 5 30 2 6 39 3 9 22 3 8 4 | Not- 33 30 3E 37 33 31 35 32 35 36 34 30 38 38 37 36 34 30 38 38 37 36 34 30 38 37 39 35 30 40 37 43 42 30 40 40 30 40 40 40 40 40 40 40 | Afte 4556 4145 4543 3645 3245 2043 7420 3293 6162 3038 494E 6541 3737 6574 55743 5530 | $e_{T} = 2$ $\frac{61}{34} \frac{61}{34} \frac{61}{34} \frac{7}{34} \frac{7}$ | 20140 ⁷ ¹⁵ ⁶⁵ ^{3D} ¹³ ³⁴ ³⁴ ¹⁴ ¹⁴ ¹² ⁴² ⁴² ⁴² ¹⁵ ¹³ ¹³ ¹⁶ ¹⁴ ¹⁴ ¹⁷ ¹⁷ ¹⁶ ¹⁷ ¹⁴ ¹⁷ ¹⁷ ¹⁷ ¹⁶ ¹⁵ ¹⁷ ¹⁷ ¹⁸ ¹⁵ ¹⁷ ¹⁷ ¹⁸ ¹⁶ ¹³ ¹⁷ ¹⁷ ¹⁸ ¹⁶ ¹³ ¹⁷ ¹⁶ ¹⁶ ¹⁶ ¹⁶ ¹³ ¹⁷ ¹⁷ ¹⁷ ¹⁶ ¹³ ¹⁷ ¹⁷ ¹⁷ ¹⁶ ¹³ ¹⁷ ¹⁷ ¹⁷ ¹⁶ ¹³ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁶ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁷ ¹⁶ ¹⁷ | 73(42 42 45 41 33 60 2F 2B 72 63 68 2F 22 72 72 20 38 |) BFC2630;eValue=" 56226731AE41C44B C975E529EC648BB6 B64786440E68516A AC1738872EC39A03 Usage-Control Time-Not-After/ 201504302355952- 0,Allouable-Ser vices/08090000;c. Idlype-HIN:eAuth Code-"SseAuth-1/ Af4F97CB775520B E049 Meta-Cont almer:KS_C_5601 E049 Meta-Cont almer:KS_C_5601 E049 Meta-Cont almer:KS_C_5601 E049 Neta-Cont almer:KS_C_5601 E049 B204 |

(b) Time-Not-After = $20150430\cdots$

Figure 4: Comparison of two DCF files that can be played until July 30, 2014 and until April 30, 2015, respectively. The red boxes represent the differences (eValue, Time-Not-After, and eAuthcode) between those DCF files.

Fig. 4 shows the comparison results between DCF files that can be played until July 30, 2014 and April 30, 2015, respectively. We attempted to identify the differences between these two DCF files using a hex editor. The differences between the two files are eValue, Time-Not-After, and eAuthcode (see the red boxes in Fig. 4). With those different parts, we examined each part one-by-one and found that Time-Not-After (see the middle box in each figure) represents the subscription expiration date consisting of the year, month, day, hour, minute, and second. Since the digest length for HMAC-SHA1 used in OMA DRM is 20 bytes (see Section 2.1), we surmise that eAuthcode is the MAC value used to ensure the integrity of the rights object in the DCF file.

When the locations of permissions and constraints are exactly identified, it becomes easy to play such DCF files without any restriction (e.g., expiration time constraint) since those permissions and constraints might be specified with any desired values. As we already described in Section 2.1, however, the integrity of the rights object in a DCF file is securely protected by its MAC. That is, the DRM agent, embedded in the MelOn music player, always checks if any of the permissions and constraints included in a DCF have been maliciously modified before playing the DCF file. Therefore, we need to disable the integrity check of the rights object in the OMA DCF files if we want to use it without any usage constraint.

3.1 Identifying the Locations of Permissions and Constraints

We performed a dynamic analysis of the MelOn media player with two popularly used tools: (1) we used $OllyDBG^3$ to obtain a list of all referenced strings by our target keywords such as eValue,

²http://www.melon.com/index.htm

³http://www.ollydbg.de/

Time-Not-After, and eAuthcode; and (2) we used IDA Pro⁴ to figure out the check code for the integrity of the rights object in a DCF file and analyze its behaviors. Through this dynamic analysis, we finally identified our target codes from the MelOn media player.

3.2 Disabling Integrity Check

For checking the integrity of a rights object in a DCF file, shown in Fig. 5, the DRM agent obtains some header information, such as cid type, Time-Not-After, allowable-services and the MAC key. Next, the DRM agent calculates the MAC value using HMAC-SHA1 with the obtained strings and values from the DCF file. Finally, the DRM agent compares the calculated MAC value with eAuthCode included in the DCF file. If they are identical, the rights object must not have been modified. The encrypted content in the DCF file can then be decrypted and accessed by the DRM agent. Otherwise, the rights object is identified as modified, and the DRM agent handles the DCF file as a corrupted file.



Figure 5: Procedure of Integrity Checking of Rights Objects

Fig. 6(a) shows the MAC comparison procedure at the assembly code level in the MelOn media player. We can see that the control flow can be changed at the "conditional jump" (jnz) instruction (see Line 1 in Fig. 6(a)): the difference between the calculated MAC value and eAuthCode is compared with 0; if they are not same, the control flow is changed to an error-handling routine by the jnz, short for Jump if Not Zero, instruction (see Line 2 in Fig. 6(a)); otherwise, the program executes the following instructions in sequence normally.



Figure 6: MAC verification procedure in the MelOn media player is disabled by replacing the jnz instruction with nop instructions.

Therefore, we can bypass this MAC verification procedure by replacing the jnz instruction with nop, short for No Operation, instruction in a simple manner (see Fig. 6(b)).

The jnz instruction is two bytes long. That is, we should replace this instruction with other instructions that are also two bytes

long in order to keep the other instructions at the same positions in the original code. In this paper, we use two nop instructions (0×90) to achieve this goal without changing any semantic of the original code⁵ With this technique, we cracked the MelOn media player in order to verify the feasibility of the proposed attack and found that the cracked MelOn media player runs normally with any DCF file even when its rights object is modified (e.g., the value of Time-Not-After is intentionally extended).

4. COUNTERMEASURES

By a reverse engineering analysis of the MelOn media player, we managed to reveal the structure of its machine codes and found a method to modify the assembly code to bypass security checks enforced by the DRM agent. Thus, a straightforward recommendation would be to use code obfuscation [4] and/or anti-debugging [5] techniques in order to hide critical codes and control flows. For example, a packing algorithm called "Alternate EXE Packer"⁶ provides a reasonable level of protection against debugging (see Fig. 7) — our disassembling program (i.e., IDA Pro) cannot produce decompiled codes with any meaningful string that provide clues to the functionality and various system calls made by codes. This would make the DRM agent program harder to analyze.

```
dd 086F60D08h, 30296064h, 1100805h, 21E408D8h, 55102020h
dd 28628685h, 9508E287h, 1FD48660h, 38142420h, 0C208182Ch
dd 4CA4278h, 76280C16h, 0CEB0EC98h, 14242818h, 3F3E16A3h
dd 60517508h, 1C307524h, 0EF51828h, 38871858h, 70F07520h
dd 1B161C03h, 0F8051786h, 22556430h, 0D49C30A4h, 17333878h
dd 0A48065h, 5D1C2C0Ch, 405BB894h, 384018A8h, 0E7911D16h
dd 688C3865h, 11242Ch, 6C05FB28h, 3F285C7Bh, 2C303407h
```

Figure 7: Decompiled MelOn program after packing the program with "Alternate EXE Packer".

However, software-based DRM solutions have inherent limitations since program codes themselves can always be reviewed and then compromised in practice, before or during runtime. Without strong protection for DRM agents, DRM techniques only offer a marginal security improvement. Therefore, we need to consider a mechanism for verifying the trustworthiness of DRM agents.

In general, the basic mechanism is as follows: Before running a media player, a service provider remotely verifies the media player integrity and grants access to that running only when it is not compromised. Such an integrity checking system can be used together with some hardware mechanism such as a Trusted Platform Module (TPM) [7] or ARM TrustZone [17]. Those mechanisms provide a means to reliably report the integrity of software and/or platform configurations with protected key storage to build a strong platform integrity verification mechanism by securely protecting the hash of the normal program binary image through a hardware chip embedded in the motherboard.

5. ETHICAL AND LEGAL ISSUES

The main motivation of our experiments is to analyze potential risks of OMA DRM services and to suggest reasonable countermeasures to mitigate such risks. Therefore, we evaluated whether DRM protection can be disabled through a case study of MelOn. The discovered vulnerability has been completely reported to Loen

⁴https://www.hex-rays.com/products/ida/support/download.shtml

⁵There are several different techniques to achieve the same goal. For example, instead of using two nop instructions with 0×90 and 0×90 , a sequence of " $0 \times 0F$ and $0 \times 0D$ " or " 0×50 and 0×58 " instructions can also be used.

⁶http://www.alternate-tools.com/pages/c_exepacker.php?lang=ENG

Entertainment Inc., the organization that operates the MelOn service. We also reported this vulnerability to the Korea Internet and Security Agency (KISA) running a bug bounty program for applications used in South Korea.

6. CONCLUSION

In this paper, we implemented a proof-of-concept cracking procedure to disable the protection of OMA DRM. We carefully examined DCF files with different constraints and identified their exact locations. With that information, we reverse-engineered the MelOn media player to bypass the integrity check of the rights object in a DCF file and successfully cracked the media player. Our study shows how difficult it is to securely protect digital items with software-based DRM solutions alone since DRM agents themselves can be compromised. Therefore, it is important to deploy appropriate defense mechanisms that would ensure the integrity of DRM agents with roots of trust at a low level (e.g., through hardware mechanisms). Although we currently limited our security analysis to the MelOn service alone, we believe this type of attack can also be applicable to other systems based on OMA DRM. For future work, we are planning to extend our security analysis to other DRM applications.

7. ACKNOWLEDGEMENTS

This work was supported by the NRF grant funded by the Korea government (No. 2014R1A1A1003707). This work was also funded in part by the ICT R&D program (2014-044-072-003, 'Development of Cyber Quarantine System using SDN Techniques') of MSIP/IITP. This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC support program (IITP-2015-H8501-15-1008) supervised by the IITP.

8. REFERENCES

- [1] Open Mobile Alliance. Candidate version 2.0–13. 2004.
- [2] Willms Buhse and Jan van der Meer. The open mobile alliance digital rights management [standards in a nutshell]. *Signal Processing Magazine, IEEE*, 24(1):140–143, 2007.
- [3] Feng-Cheng Chang, Chiao-Lin Wu, and Hsueh-Ming Hang. A Switchable DRM Structure for Embedded Device. In Proceedings of the Third International Conference on International Information Hiding and Multimedia Signal Processing, 2007.
- [4] Christian S. Collberg and Clark Thomborson. Watermarking, tamper-proffing, and obfuscation: Tools for software protection. *IEEE Transactions on Software Engineering*, 28(8):735–746, 2002.
- [5] Michael N Gagnon, Stephen Taylor, and Anup K Ghosh. Software protection through anti-debugging. *Security & Privacy, IEEE*, 5(3):82–84, 2007.
- [6] David Geer. Digital rights technology sparks interoperability concerns. *Computer*, 37(12):20–22, 2004.
- [7] Trusted Computing Group. TPM Main Specification Version 1.2 rev. 103. http://www.trustedcomputinggroup.org/ resources/tpm_main_specification, 2007.
- [8] F. Hartung and F. Ramme. Digital rights management and watermarking of multimedia content for m-commerce applications. *Communications Magazine, IEEE*, 38(11):78–84, 2000.
- [9] James Irwin. Digital rights management: The open mobile alliance DRM specifications. *Information Security Technical Report*, 9(4):22–31, 2004.

- [10] Ton Kalker. On Interoperability of DRM. In *Proceedings of* the ACM Workshop on Digital Rights Management, 2006.
- [11] Kyoung-Joo Lee. The coevolution of it innovation and copyright institutions: The development of the mobile music business in japan and korea. *The Journal of Strategic Information Systems*, 21(3):245–255, 2012.
- [12] Nicholas Paul Sheppard and Reihaneh Safavi-Naini. On the operational semantics of rights expression languages. In *Proceedings of the Nineth ACM Workshop on Digital Rights Management*, DRM '09, pages 17–28, New York, NY, USA, 2009. ACM.
- [13] Mark Stamp. *Information security: principles and practice*. John Wiley & Sons, 2011.
- [14] SR Subramanya and Byung K Yi. Digital rights management. *Potentials*, *IEEE*, 25(2):31–34, 2006.
- [15] Daniel Thull and Roberto Sannino. Performance Considerations for an Embedded Implementation of OMA DRM 2. In *Proceedings of the Conference on Design*, *Automation and Test in Europe*, pages 46–51, 2005.
- [16] Ruoyu Wang, Yan Shoshitaishvili, Christopher Kruegel, and Giovanni Vigna. Steal This Movie - Automatically Bypassing DRM Protection in Streaming Media Services. In Proceedings of the 22Nd USENIX Conference on Security, 2013.
- [17] J. Winter. Experimenting with ARM TrustZone Or: How I Met Friendly Piece of Trusted Hardware. In *Proceedings of Conference on Trust, Security and Privacy in Computing and Communications*, pages 1161–1166, 2012.

APPENDIX

A. APPENDIX OF ABBREVIATIONS

| ABBR. | Description |
|-------|------------------------------------|
| DRM | Digital Rights Management |
| OMA | Open Mobile Alliance |
| CI | Content Issuer |
| RI | Rights Issuer |
| RO | Rights Objects |
| REL | Rights Expression Language |
| DCF | DRM Content Format |
| MAC | Message Authentication Code |
| TPM | Trusted Platform Module |
| KISA | Korea Internet and Security Agency |