



UNIVERSIDAD DE CHILE  
FACULTAD DE DERECHO  
DEPARTAMENTO DE DERECHO PÚBLICO

**RECONOCIMIENTO LEGAL EN CHILE DE LA  
MODIFICACIÓN DE SOFTWARE PARA PROTEGER LOS  
DERECHOS DE LOS USUARIOS**

TESIS PARA OPTAR AL GRADO DE LICENCIADO EN CIENCIAS  
JURÍDICAS Y SOCIALES

**AUTOR:** SAMUEL EMILIO ANTONIO MORALES PIMENTEL

**PROFESOR GUÍA:** SALVADOR MILLALEO HERNÁNDEZ

SANTIAGO

2018

## Agradecimientos

A mis padres, por el apoyo y el aliento a lo largo de la vida.

A mis amigos, por su compañía y sus consejos.

A los profesores que tuve en la Facultad, y especialmente al profesor guía de este trabajo, Salvador Millaleo, por sus correcciones y comentarios.

# Tabla de Contenido

Introducción .....	5
Planteamiento del problema y metodología .....	5
Capítulo 1: El Software y su modificación. ....	7
¿Qué es un software? .....	7
¿Qué es una modificación de software?.....	13
Capítulo 2: Las licencias de software.....	16
Definición de las licencias de software. ....	16
Características de las licencias de software.....	17
Tipos de licencias de software. ....	18
Análisis de los derechos involucrados en las licencias de software .....	23
Licencias de software propietario .....	24
Licencias de software libre.....	29
Conclusiones del capítulo.....	36
Capítulo 3: El ordenamiento jurídico chileno y la modificación de software .....	37
Constitución Política de la República .....	37
Ley de Propiedad Intelectual (17.336) .....	39
Ley de Protección al Consumidor (19.496).....	41
Ley que Tipifica Figuras Penales Relativas a la Informática (19.223) .....	47
Ley sobre Protección de la Vida Privada (19.628).....	49
Conclusión del capítulo.....	53
Conclusiones generales .....	53
Bibliografía .....	56

## Resumen

Este trabajo realiza un análisis del concepto de software y de modificación de software, y revisa el marco regulatorio vigente a efectos de determinar si se permite hoy en Chile la modificación de un software por parte de un usuario cuando aquel vulnera los derechos como persona que tiene el usuario del software, como la propiedad y la privacidad.

## Abstract

This work makes an analysis of the concept of software and software modification and makes a review of the current regulatory framework in Chile to determine if in Chile a software user can legally modify it to protect her personal rights, like property or privacy.

## Introducción

En el mundo actual, los programas computacionales o softwares tienen una relevancia cada vez mayor. Cada día que pasa usamos con más frecuencia algún software, ya sea directamente, por ejemplo, al encender y utilizar nuestro computador para navegar por Internet, al mandar mensajes desde nuestro teléfono móvil, o también indirectamente, al pasar una tarjeta BIP por los validadores en el transporte público, al pasar en frente de cámaras equipadas con reconocimiento facial e incluso al dar arranque a nuestro vehículo. En todas estas situaciones hay un software detrás que hace posible que las cosas funcionen. Viendo entonces, el sinnúmero de situaciones cotidianas donde nos vemos involucrados con un software, vale la pena preguntarse cuáles son los límites del uso que le damos al software.

### Planteamiento del problema y metodología

El tema de este trabajo son los usos que las personas pueden realizar con el software instalado en un artefacto.

El objeto es investigar los casos en que la limitación a la modificación de un software importa una limitación a ciertos derechos que tenemos las personas respecto del artefacto donde se instala ese software, como el derecho de propiedad, que a su vez importa la libertad de hacer lo que el propietario estime conveniente con ese objeto; y dilucidar cómo trata el ordenamiento jurídico chileno la licitud de tomar medidas, modificando el software, para restaurar el derecho de propiedad y la libertad sobre el artefacto.

Profundicemos en el problema de la investigación. La pregunta general de este trabajo es **¿Qué reconocimiento legal hay en Chile de la modificación o alteración de software para garantizar la libertad y la propiedad sobre el hardware que tienen los usuarios?**

Las preguntas específicas, que vamos a responder a lo largo del trabajo, son las siguientes:

- ¿De qué hablamos cuando nos referimos a alterar o modificar software?
- ¿Qué derechos, tanto constitucionales como legales, se pueden ver afectados por la imposibilidad de modificación?
- ¿Qué derechos se ven protegidos por la imposibilidad de modificación?
- ¿Qué derechos se adquieren con una licencia de software?

Todas estas preguntas se desprenden de la principal, y responderlas sirve a su vez para dilucidar la pregunta general.

Respecto a los objetivos del trabajo, el objetivo general es: **Aclarar si existe reconocimiento en el ordenamiento jurídico chileno de (1) la posibilidad de afectar derechos de las personas a través de ciertas cláusulas de las licencias de software y de (2) la posibilidad de incumplir o no aplicar esas cláusulas a través de la modificación de ese software.** Los objetivos específicos que necesitamos cumplir para lograr el objetivo general son:

- Aclarar qué podemos entender por modificación de software
- Establecer los derechos, tanto constitucionales como legales, que se pueden ver afectados cuando se toman medidas para imposibilitar la modificación del software
- Establecer los derechos, o bienes jurídicos, que protegen las cláusulas de no modificación de un software. (y, a contrario sensu, establecer que derechos se vulnerarían en caso de modificarlo)
- Establecer qué derechos se adquieren mediante una licencia de software.

La hipótesis del trabajo es **“Existe un reconocimiento parcial de la posibilidad de afectar derechos de las personas, que se expresa, por ejemplo, en las excepciones de la Ley de Propiedad Intelectual. Sin embargo, es parcial porque no se explicita como una afectación de derechos. En cuanto a la posibilidad de incumplir las cláusulas limitativas de derechos, tal posibilidad no se contempla en el ordenamiento jurídico actual”**

Respecto a la metodología, el trabajo se dividirá en dos partes: la primera se orientará a esclarecer los conceptos de software, modificación de software, licencias de software y derechos de los usuarios (primeros dos capítulos) Aquí la metodología a usar será fundamentalmente descriptiva.

La segunda parte se orientará al análisis del marco legal vigente en Chile y se utilizará una metodología documental. El análisis a realizar será valorativo, en la medida en que se interpretarán las normas para ver si coinciden o no con los objetivos de la investigación.

Cabe destacar que este tema parece ser muy poco tratado, tanto a nivel nacional como internacional. Esto, si bien significó un escollo a la hora de investigar, por la poca bibliografía

disponible, también es una oportunidad para desarrollar el tema y aportar con un nuevo trabajo a la discusión sobre el mismo.

Dicha ya la estructura del trabajo y dejando claro el problema, los objetivos y las preguntas a responder, vamos al desarrollo.

## Capítulo 1: El Software y su modificación.

Para responder las preguntas planteadas en la introducción resulta esencial responder dos preguntas orientadas a delimitar dos conceptos clave en esta investigación: el software y la modificación de software. En este primer capítulo vamos a abocarnos a responder, entonces, dos preguntas: ¿qué es un software? y ¿qué es una modificación de software?

### ¿Qué es un software?

En el ordenamiento jurídico chileno encontramos una definición legal de software en la Ley 17.336 de Propiedad Intelectual, en el artículo 5 letra t, como “conjunto de instrucciones para ser usadas directa o indirectamente en un computador a fin de efectuar u obtener un determinado proceso o resultado, contenidas en un cassette, diskette, cinta magnética u otro soporte material”. La misma letra define también la copia de un programa computacional, entendiendo como tal el “soporte material que contiene instrucciones tomadas directa o indirectamente de un programa computacional y que incorpora la totalidad o parte sustancial de las instrucciones fijadas en él”.<sup>1</sup>

Como podemos apreciar, la definición de software (o programa computacional, en español) que usa la Ley de Propiedad Intelectual se centra no tanto en el efecto que produce en la máquina donde se utiliza (refiere de forma genérica a un “determinado proceso o resultado”), como en el soporte material donde se almacena ese conjunto de instrucciones.

Para efectos de este trabajo, esa definición legal no parece suficiente, en la medida en que no explica técnicamente los efectos que tiene un software sobre una máquina. Por eso, vamos a ver algunas definiciones aportadas por profesionales de la ingeniería de software y las ciencias de la computación.

---

<sup>1</sup> Ley 17.336, art. 5 letra t

Ian Sommerville en su texto Ingeniería de Software, define a los softwares como “los programas de cómputo y la documentación asociada”.<sup>2</sup> La documentación asociada consiste en documentos del sistema que describen la estructura de este y documentación de usuario, como manuales de uso.<sup>3</sup> Este autor se refiere a los programas desarrollados de forma profesional, y normalmente, con un carácter comercial, según veremos más adelante.

Según Gregorio Martín, Francisco Toledo y Vicente Cerverón, el software es el soporte lógico de un computador, esto es, “el conjunto de programas que dirigen el funcionamiento de un computador: sistema operativo, programa de usuario, etc.”.<sup>4</sup> Los autores lo contraponen al concepto de soporte físico, es decir, al hardware de un computador.

Al decir “software” no nos estamos refiriendo a un concepto único, el software, como concepto, se puede dividir en varios tipos. La distinción más básica es entre software del sistema, que es el necesario para administrar y mantener los recursos del computador de una forma eficiente; y el software de aplicación, que son aplicaciones específicas que usan los recursos del computador. Martín, Toledo y Cerverón dividen a su vez el software del sistema en: programa de arranque; sistema operativo; programas de diagnóstico y mantenimiento; y utilidades generales y herramientas de programación.<sup>5</sup> Un ejemplo de software del sistema es un sistema operativo, como Microsoft Windows en un computador doméstico o Android en un teléfono inteligente. Un ejemplo de software de aplicación puede ser un navegador de internet, como Mozilla Firefox, o una aplicación para un teléfono inteligente, como WhatsApp.

Otra distinción es entre aplicaciones independientes y aplicaciones interactivas basadas en transacción. Las primeras son programas que se ejecutan localmente, en la misma máquina donde se utilizan, y no requieren conexión a una red. Por ejemplo, el bloc de notas de Windows. Las segundas son aplicaciones que se ejecutan remotamente y los que los usuarios acceden desde su PC o terminales. Según Sommerville, en este tipo de aplicaciones se incluyen aplicaciones web y sistemas empresariales, donde una organización brinda acceso a sus sistemas a través de un navegador web o un programa de cliente de propósito específico y servicios basados en la

---

<sup>2</sup> (Sommerville, 2011) p. 6

<sup>3</sup> Idem p. 5

<sup>4</sup> (Martín Quetglás, Toledo Lobo & Cerverón Lleó, 1995) p. 24

<sup>5</sup> Idem p. 46



nube.<sup>6</sup> Un ejemplo de este tipo de aplicaciones podría ser Gmail, el servicio de correo electrónico de Google, al que accedemos a través de un navegador, o también la red social Facebook, que también se accede a través de un navegador.

Respecto de esta última categoría de software se ha acuñado la expresión “software como servicio” (en inglés *software as a service* o SaaS), consistente en que los usuarios arriendan o “toman prestado” un software que se ejecuta en línea, en lugar de comprarlo e instalarlo en su propio computador. Todo el trabajo de procesamiento de los datos se realiza remotamente, y al usuario le llegan a su terminal los datos ya procesados. El método más frecuente de acceso es vía internet. El SaaS va más allá de ser un simple tipo de software, se ha transformado en un modelo de negocio para las grandes compañías productoras de programas computacionales, porque presenta ventajas como la centralización del software en sus propios servidores, por lo que es más fácil el diagnóstico de fallas y el soporte técnico a los usuarios. Asimismo, de cara a los usuarios, el software se presenta listo para usar y actualizado. También se da la posibilidad de que el uso de ese software sea cedido de forma gratuita, como ocurre con los servicios de Google, o estableciendo una tarifa de uso mensual o anual, que puede ser más atractiva que adquirir una copia del software por una sola vez, como ocurre con Office 365, de Microsoft, que es la conocida suite ofimática en un formato de software como servicio. Sin embargo, también se puede adquirir una copia “única”, ya sea físicamente (en un CD u otro medio de almacenamiento físico, como una memoria USB) o de forma virtual (descargando una copia desde los servidores de Microsoft).

Sin embargo, este modelo de software también presenta desventajas, especialmente de cara al usuario. Así, si los datos son procesados remotamente, esto implica que deben ser almacenados, aunque sea temporalmente, en los servidores de la empresa proveedora del software. Entonces, el usuario debe confiar bastante en la empresa para entregarle los datos, y a su vez, esta debe tener estándares elevados de seguridad y privacidad.<sup>7</sup>

Otra distinción muy conocida en el mundo del software es la distinción entre software libre y software propietario.

---

<sup>6</sup> Op cit. nota 2, p. 11

<sup>7</sup> (Gil, 2017)

El software libre (en inglés *free software*) no tiene una definición unívoca, puesto que en inglés *free* significa tanto libre como gratis, pero el software libre no necesariamente tiene que ser gratuito.

El software libre se enmarca dentro de un movimiento que tiene distintas ramificaciones, pero que en general concede cuatro libertades a los usuarios, como apuntan Bain y otros:<sup>8</sup>

- Libertad 0: ejecutar y usar el software para cualquier propósito
- Libertad 1: estudiar el programa y adaptarlo a las necesidades
- Libertad 2: distribuir copias
- Libertad 3: modificar el programa y liberar esas modificaciones al público

Estos autores distinguen, a su vez, entre el software libre en sentido estricto y el software de código abierto. El software libre en sentido estricto es el que contiene esas cuatro libertades fundamentales y propugna su mantenimiento a través de las modificaciones o trabajos derivados. Se relaciona con el origen del movimiento del software libre, con *la Free Software Foundation* (FSF) y Richard Stallman, uno de los principales activistas del software libre a nivel mundial. La forma que usa la FSF para mantener libre el software es a través de la licencia GNU GPL (*General Public License*), cuyos términos no permiten a los redistribuidores añadir a su licencia cualquier restricción adicional a las de la licencia original. Esto se conoce como *copyleft*, y obliga a los usuarios a continuar con la cadena de distribución abierta. La FSF defiende la idea de que el único software libre es el que se distribuye utilizando una licencia GPL, y por tanto el software de código abierto no sería un verdadero software libre.

El software de código abierto nació en 1998 como una escisión del software libre propugnado por la FSF, para superar la actitud conflictiva que se había asociado con el software libre en el pasado y acercar más el concepto al mundo empresarial. Tal como su nombre lo indica, en este tipo de software el código fuente es abierto a cualquier persona que lo quiera ver, estudiar, modificar o distribuir, pero las licencias mediante las que se distribuye no obligan al desarrollador a mantener esas mismas libertades.<sup>9</sup> Así, podría por ejemplo cerrar el código de

---

<sup>8</sup> (Bain, Gallego Rodríguez, Martínez Ribas & Sanjuán, s. f.)

<sup>9</sup> Ídem p. 26-28

modificaciones futuras y entregar solo una licencia de uso, y que esta sea pagada. Vamos a examinar esto con más detalle luego, cuando revisemos las licencias de software.

Por otro lado, el software propietario también tiene múltiples acepciones. En primer lugar, se puede entender como aquel software que es propiedad legal de una parte, donde el código fuente no está disponible y su uso está condicionado bajo términos contractuales (es decir, en una licencia de uso de software). Otra acepción dice relación con la disponibilidad del código fuente, estableciendo que es software propietario aquel cuyo código está cerrado, y se licencia bajo exclusivo derecho del titular del *copyright*. Una tercera acepción remarca el carácter comercial (y oneroso) del software propietario, estableciendo que es un software comercial donde el núcleo es mantenido en secreto por el desarrollador, los usuarios tienen que pagar un precio por el uso, y no tienen el derecho de distribuir el software ni de modificarlo.<sup>10</sup>

La FSF entiende el software propietario (o como lo denominan ellos, privativo) como “el software que no es libre”, definiendo este último como el software donde “Está prohibido su uso, redistribución o modificación, o requiere que se solicite permiso, o tiene tantas restricciones que de hecho no se puede hacer libremente.” Distinguen entre varios tipos de software privativo, como el *freeware*, que es aquel software en donde se permite la redistribución pero no la modificación, y donde el código fuente no está disponible; el *shareware*, donde se permite la redistribución pero quien continúa utilizando una copia debe pagar por la licencia (y establecen que el *shareware* tampoco es software libre, porque su código fuente no está disponible y no se puede modificar, y porque el software “no viene con permiso para hacer una copia e instarlo sin pagar la licencia, ni siquiera para personas que participan en actividades sin fines de lucro”); software privado, que es aquel que ha sido desarrollado para un usuario y este lo mantiene y lo utiliza, sin publicarlo; y software comercial, aquel que es desarrollado por una empresa como parte de su actividad comercial.<sup>11</sup>

Entonces, podemos apreciar que la visión tradicional contraponen software libre y software propietario, entendiendo que el primero entrega más libertades al usuario, mientras que el segundo las retiene en el desarrollador del software.

---

<sup>10</sup> ("What Is Proprietary Software", s. f.)

<sup>11</sup> ("Categorías de software libre y software que no es libre - Proyecto GNU -Free Software Foundation", s. f.)

Otra distinción que podemos hacer, ya no estrictamente técnica pero que sí puede tener relevancia para este trabajo, es entre el software hecho por encargo y el software adquirido de un tercero sin encargo o desarrollado para un uso propio.

El software desarrollado para uso propio o adquirido sin un encargo lo podríamos equiparar a la definición de software privado, o de software comercial según sea el caso, que hace la FSF. Es decir, sería un software que realiza un desarrollador para su uso particular, o alguien que adquiere un software “estándar” de parte de un tercero para darle un uso no aplicado a una tarea específica, sino que es un software de propósito general. Un ejemplo sería adquirir una copia y licencia de uso del sistema operativo Microsoft Windows: el desarrollador (en este caso, Microsoft) no realizó el programa por encargo de quien lo adquiere, sino que es un software comercial que la empresa ofrece a quien quiera comprar una licencia de uso.

El software hecho por encargo es aquel software que una persona (en nuestra opinión, podría ser una persona natural o jurídica) encomienda a otra su realización. Normalmente, esta relación jurídica va a estar mediada por un contrato de software por encargo.<sup>12</sup>

Esta distinción es importante en la medida en que es recogida por la Ley 17.336 de Propiedad Intelectual, porque establece en el artículo 8 inciso segundo que el derecho de autor de los programas computacionales será cedido a quien lo encargó.

Quedémonos entonces con la idea de que un software es el conjunto de instrucciones, expresado tanto en un código fuente como en código máquina, que sirve de soporte lógico al hardware donde está instalado, esto es, dirige su funcionamiento e incorpora las funciones que se va a poder realizar con ese hardware. Ya sea un computador doméstico, un teléfono inteligente, maquinaria industrial o el computador central de un vehículo autónomo, el hardware no puede funcionar correctamente sin uno o más softwares.

Ahora que hemos visto algunas definiciones y tipos de software, veamos qué podemos entender por “modificación de software”.

---

<sup>12</sup> (Fuentes Pinzón, 2003)

## ¿Qué es una modificación de software?

Definir una modificación de software es un tanto complejo, en la medida en que no existe un único método para modificar un software. Además, no toda modificación es legítima. Ya vimos que existe software libre y software propietario, y que en el primero el código fuente está disponible para que cualquier persona con los medios adecuados pueda verlo, estudiarlo, adaptarlo y realizar cualquier modificación (dependiendo de la licencia, como veremos más adelante). Así, en principio, si tenemos un software libre cualquier modificación sería legítima, puesto que el diseño del software y su licencia nos lo permiten. En el caso de un software propietario no tenemos acceso directo al código fuente, y tenemos que buscar otros medios que nos permitan modificarlo. En el caso de un software propietario las modificaciones entran en un terreno enmarañado, porque normalmente la licencia va a prohibir la modificación del software en cualquier forma. Sin embargo, pueden existir casos especiales, y eso es lo que este trabajo pretende dilucidar.

Ahora, para poder responder a la pregunta de qué es una modificación de software debemos ver primero cómo este se construye.

De acuerdo con Fuentes Pinzón, para crear un programa de computación se requiere una serie de pasos previos: Una fase de análisis para evaluar los fines, tareas y resultados perseguidos, necesidades del usuario y tipo de hardware adecuado; la determinación de la estructura o arquitectura del software; la descripción del programa en forma de diagramas y la de los módulos o partes; la escritura del software en un lenguaje de programación; la comprobación de resultados y corrección de los errores detectados; y la elaboración de los manuales de uso y documentación técnica.<sup>13</sup>

Si separamos cada uno de estos pasos, podemos apreciar que, estrictamente, el paso que corresponde a la construcción del software en sí es la escritura del mismo utilizando un lenguaje de programación.

Existen distintos lenguajes de programación, cada uno con características distintas que hacen que un programa pueda tener más o menos complejidad, y más o menos funciones. También, para poder crear un software, necesitamos a su vez otros softwares, o un conjunto de ellos,

---

<sup>13</sup> Idem p. 96

denominados entornos de desarrollo integrado (en inglés, *integrated development environment* o IDE)

Para poder modificar un software, lo más usual es conocer su código fuente, y a través de un IDE u otro programa similar, reescribirlo, cambiando el código fuente y consiguientemente cambiando el funcionamiento del software.

Sin embargo, como ya vimos, el software propietario no pone a disposición del público su código fuente, y en estos casos es necesario ocupar otras técnicas, como la ingeniería inversa.

La ingeniería inversa se puede definir como “aquel proceso analítico-sintético que busca determinar las características y/o funciones de un sistema, una máquina o un producto o una parte de un componente o subsistema”<sup>14</sup>

Aplicada al software, Teodoro Cipresso la define como “la práctica de analizar un sistema de software, como un todo o una parte, para extraer información del diseño e implementación”<sup>15</sup>

El mismo autor indica que la práctica de la ingeniería inversa -en general, no solo aplicada al software- nace de la curiosidad por investigar cómo y por qué las cosas funcionan. Explica que una vez identificados los componentes y dependencias de un software, se vuelve posible repararlos o incluso mejorarlos.

Delfín y Jiménez señalan que la ingeniería inversa a menudo se confunde con la piratería<sup>16</sup>, pero en principio la ingeniería inversa no tiene esa connotación “maliciosa” o “ilegal”.

Según Eldad Eliam, las aplicaciones de la ingeniería inversa se pueden dividir en dos categorías: ingeniería inversa relativa a cuestiones de seguridad e ingeniería inversa relativa a desarrollo de software.

En la primera categoría menciona como ejemplos que la ingeniería inversa es utilizada en investigación de mecanismos de encriptación para evaluar el nivel de seguridad que proporcionan; en conexión con software malicioso, tanto para quienes los desarrollan como

---

<sup>14</sup> (Delfín Vásquez & Jiménez López, 2017) p. 3

<sup>15</sup> (Cipresso, 2009) p. 2

<sup>16</sup> Op. Cit. nota 14 p. 3

quienes buscan combatirlos; y por *crackers*, con el objetivo de analizar y eventualmente romper esquemas de protección contra copias.

En la segunda categoría menciona como ejemplos de aplicación de la ingeniería inversa el alcanzar compatibilidad o interoperabilidad con software no documentado o parcialmente documentado, como un software propietario; también para determinar la calidad de un código fuente de terceros, como una librería o incluso un sistema operativo; asimismo, señala que es posible algunas veces usar técnicas de ingeniería inversa para extraer información valiosa de un producto de la competencia con el propósito de mejorar las tecnologías propias.<sup>17</sup>

Como se puede apreciar de estos ejemplos, en principio la ingeniería inversa tiene fines útiles e incluso loables, pero también puede ser usado para fines “maliciosos” o “ilegales”.

Dijimos que para modificar un software lo más usual es conocer su código fuente. Aquí es necesario precisar la distinción entre código fuente y código objeto. El primero se puede definir como el código escrito en un lenguaje de programación, que puede ser leído por el programador y por una persona humana. Este código no puede ser entendido por las máquinas, y entonces es necesario traducirlo a un lenguaje que la máquina pueda comprender. Esta traducción se realiza por medio de otros programas, denominados traductores. Así, se pasa del código fuente al código máquina o código objeto, que es el lenguaje que las máquinas pueden entender para llevar a cabo las instrucciones. De forma contraria a lo que ocurre con el código fuente, este es muy difícil de comprender para las personas humanas.<sup>18</sup>

Esta distinción entre código fuente y código objeto o código máquina es relevante en la medida en que también es reconocida por la ley chilena. La Ley de Propiedad Intelectual establece que tanto el código fuente como el código objeto de un programa computacional estarán protegidos.

En suma, podríamos decir que la modificación de software sería cualquier cambio en el funcionamiento previsto originalmente por los desarrolladores, ya sea a través de modificar el código fuente directamente o mediante otras técnicas, como la ingeniería inversa.

---

<sup>17</sup> (Eilam, 2005) p. 5-9

<sup>18</sup> (Joyanes Aguilar, 2008) p. 35

Clarificados entonces los conceptos de software y de modificación de software, vamos a revisar los distintos tipos de licencias de software que existen y qué derechos conceden o quitan, según sea el caso.

## Capítulo 2: Las licencias de software

En este segundo capítulo, abordaremos las licencias de software, su definición, características y tipos de licencias, para luego revisar qué derechos conceden (y *a contrario sensu*, qué derechos quitan) en relación con la posibilidad de modificar un software.

### Definición de las licencias de software.

Para revisar qué es una licencia de software, hay que señalar que esta se manifiesta a través de un contrato, por lo que tenemos que hablar de un contrato de licencia de software.

Para hablar de una licencia de software es necesario revisar qué es un contrato de licencia, esto es, el género de los contratos dentro de los que está el contrato de licencia de software.

Juan Carlos Monroy explica que un contrato de licencia es aquel cuyo objeto es autorizar uno o varios usos de explotación de una obra protegida. Quién otorga la autorización es el licenciante, y quien se beneficia de ella es el licenciatario. El mismo autor señala que “El contrato de licencia de software tiene por objeto autorizar al usuario ciertos usos o actos de utilización de un programa de computador, pudiendo comprender tanto la instalación en la memoria de un computador personal o servidor de red para el acceso de un determinado número de equipos cliente, como el acceso y uso de una aplicación en línea. Así mismo, el alcance de tales autorizaciones podrá variar según el tipo de licencia, pudiendo llegar a comprender inclusive la modificación y redistribución del programa de computador, así como el acceso al código fuente, como sería el caso por ejemplo del software libre de código abierto.”<sup>19</sup>

Según Humberto Carrasco, el contrato de licencia de software se podría definir como “aquel contrato en virtud del cual el licenciante autoriza al licenciatario para utilizar un software sobre el que es titular del derecho de autor”<sup>20</sup>

---

<sup>19</sup> (Monroy, 2012)

<sup>20</sup> (Carrasco Blanc)



Según Ramón Gómez Labrador, la licencia es un “contrato entre el desarrollador del software sometido a propiedad intelectual y a derechos de autor y el usuario, en el cual se definen con precisión los derechos y deberes de ambas partes. Es el desarrollador, o aquel a quien éste haya cedido los derechos de explotación, quien elige la licencia según la cual se distribuye el software”<sup>21</sup>

Kanan Madatli explica que una licencia de software se puede definir como “un contrato entre el licenciante del software y el comprador, definiendo los derechos del comprador para usar y para adquirir el software”<sup>22</sup>

### Características de las licencias de software.

Humberto Carrasco señala como características de este tipo de contratos las siguientes:

1. Es un contrato solemne, cuya solemnidad es que la autorización, y los derechos que en ella se contienen, consten por escrito.
2. Es un contrato generalmente oneroso, que excepcionalmente puede ser gratuito. El autor menciona como ejemplos de licencias gratuitas las de los programas *shareware* o *freeware*. Señala también que la mayoría de los casos donde la licencia es gratuita se dan en las licencias otorgadas electrónicamente, como en programas descargados de Internet.
3. Por regla general, son contratos de adhesión, salvo cuando se trata de software por encargo o software a la medida.
4. Generalmente contienen una serie de especificaciones para el correcto funcionamiento del software, como las características que debe tener el hardware donde se instalará el programa, o información del rendimiento en cuanto a su capacidad de ejecución. Carrasco señala que esa información viene contenida en el envase del software. Aquí, es necesario precisar que, como vimos en el capítulo anterior, hoy la mayoría del software se obtiene -y se adquiere, en el caso del software pagado- a través de Internet, y por tanto, esas especificaciones están contenidas en otros soportes, como los manuales de usuario, que son muchas veces electrónicos, en los sitios web de los programas.

---

<sup>21</sup> (Gómez Labrador)

<sup>22</sup> (Madatli)

5. Hay diversas cláusulas que se incluyen en este tipo de contratos. El autor menciona como ejemplos cláusulas de garantía de funcionamiento, violación de derechos de terceros o exclusividad.

A su vez, Juan Carlos Monroy señala que los contratos de licencia de software pueden caracterizarse como bilaterales, onerosos o gratuitos, conmutativos, atípicos, principales o accesorios, e innominados.

También señala que una característica esencial de los contratos de licencia (en general, no solo de software “consiste en que el titular del derecho de autor (generalmente el licenciante) no transfiere o cede la titularidad de tales derechos, sino que los mantiene en su poder, y se limita a autorizar, de manera exclusiva o no, los distintos usos o actos de explotación de que la obra puede ser objeto. Así mismo, la amplitud de sus derechos exclusivos de autor le brinda al licenciante la posibilidad jurídica de definir con detalle el alcance y las restricciones de las autorizaciones que otorga.”<sup>23</sup>

Kanan Madatli hace notar que “la licencia de software típica establece que la compañía de software entregará un producto final de software al usuario. En muchos casos, la compañía de software también se obliga a corregir cualquier defecto o bug que se descubra, y a proveer periódicamente al usuario versiones mejoradas y actualizadas del software. El usuario del software, a su vez, paga una tarifa por el software”<sup>24</sup>

En relación con las cláusulas del contrato, Madatli señala que “el usuario del software se compromete, a menudo, a que no (1) divulgará cualquier información confidencial que la compañía proporcione al usuario, (2) reproducirá o copiará el software, y (3) asignará o sublicenciará el software sin el consentimiento de la compañía de software.”<sup>25</sup>

## Tipos de licencias de software.

<b>Criterio de clasificación</b>	<b>Tipos</b>	<b>Ejemplos de licencias</b>
----------------------------------	--------------	------------------------------

---

<sup>23</sup> Op. Cit. Nota 1

<sup>24</sup> Op. Cit. Nota 4

<sup>25</sup> Ídem

<b>Según el grado de libertad que entregan</b>	Licencias de software propietario	<ul style="list-style-type: none"> <li>- Contrato de Licencia de Usuario Final de Microsoft Windows</li> <li>- Contrato de Licencia de Usuario Final de iOS (sistema operativo del iPhone)</li> </ul>
	Licencias de Software libre	<ul style="list-style-type: none"> <li>- GNU General Public License</li> <li>- Licencia Apache</li> <li>- Mozilla Public License</li> </ul>
<b>Según el grado de estandarización del software</b>	Licencias de software genérico	<ul style="list-style-type: none"> <li>- Condiciones de uso de Google Chrome</li> </ul>
	Licencias de software personalizado	<ul style="list-style-type: none"> <li>- Licencia de un software ERP diseñado para una empresa en particular.</li> </ul>
<b>Según la forma de celebración del contrato</b>	Licencias celebradas por escrito	<ul style="list-style-type: none"> <li>- Licencia de software complejo que requiera muchas especificaciones</li> </ul>
	Licencias celebradas por medios electrónicos	<ul style="list-style-type: none"> <li>- Licencia de uso de WinRAR, que aparece al instalar el programa.</li> </ul>
	Licencias celebradas por otros medios válidos de expresión del consentimiento	<ul style="list-style-type: none"> <li>- Licencia incorporada en la caja de un software (Monroy)</li> </ul>

Tabla 1. Tipos de licencias de software<sup>26</sup>

<sup>26</sup> Tabla de elaboración propia usando como referencia los criterios propuestos por Juan Carlos Monroy. Véase (Monroy, 2012)

Existen varios criterios de clasificación para dividir los tipos de licencias de software. Hemos hecho un resumen en la tabla que aparece arriba para facilitar la comprensión, que incorpora ejemplos para cada tipo de licencia.

Al igual que en el primer capítulo, podemos dividir las licencias de software según este sea propietario o libre.

En el caso del software propietario, la licencia va a depender del titular de los derechos de autor del software en cuestión, que normalmente va a ser quien lo crea o quien lo ofrece.

Por ejemplo, tenemos el Contrato de Licencia de Usuario Final (CLUF, en inglés *End User License Agreement, EULA*) del sistema operativo Windows, cuyo licenciante es la empresa que lo desarrolla y ofrece, Microsoft.

Otro ejemplo es el Contrato de Licencia de Usuario Final de Apple, diferenciado por el tipo de producto. Así, hay un contrato distinto para el sistema iOS, que usan los dispositivos móviles (iPhone, iPad y iPod Touch); para el sistema macOS (llamado anteriormente OS X), que usan los computadores de escritorio y portátiles; y para otros productos de software ofrecidos por la compañía, como la suite ofimática iWork o el programa de música iTunes.

Dentro de las licencias de software propietario podemos distinguir dos sub tipos: la licencia *shareware* y la licencia *freeware*. Según Monroy las licencias del primer tipo “corresponden a un tipo de distribución de aplicaciones que consiste en liberar gratuitamente una versión con funcionamiento limitado. Esa limitación puede ser temporal (después de determinada cantidad de días deja de operar), por funciones (desde el comienzo, o a partir de determinado momento, hay funciones que el programa deja de realizar) o una combinación de las mencionadas (el programa empieza con todas sus funciones y deja de realizar algunas al cabo de cierto tiempo). En cualquier caso, al finalizar el período de prueba el usuario debe pagar la licencia o desinstalar el programa”.<sup>27</sup> Un ejemplo de este tipo de software es WinRAR, un programa ampliamente usado en Windows para comprimir y descomprimir archivos. Las licencias del segundo tipo "son licencias que permiten obtener, instalar y/o usar gratuitamente un programa de computador. No es necesario que el usuario se registre, se suscriba o compre una licencia. La gran diferencia entre el freeware y el código abierto (“open source”) es que el primero es cerrado, es decir, el

---

<sup>27</sup> Op. Cit. Nota 1

usuario solo tiene acceso al código objeto del programa, no al código fuente”.<sup>28</sup> Un ejemplo de este tipo de licencia (y de programa) es el antivirus para Windows Avast, que se puede descargar en forma gratuita desde Internet.

En el caso del software libre, si bien cada desarrollador puede utilizar la licencia que desee, como ocurre con el software propietario, está más extendida la práctica de licenciar un software libre bajo determinadas licencias creadas principalmente por organizaciones, como la licencia General Public License (GNU GPL), creada y promovida por la Free Software Foundation, la licencia Apache, la licencia Mozilla Public License, creada y promovida por la Mozilla Foundation y usada en su producto más conocido, el navegador web Mozilla Firefox, entre otras.

Suele ocurrir con el software libre que, al ser un conjunto de aportes de distintos desarrolladores, cada “parte” de un software tenga licencias distintas. Así ocurre, por ejemplo, con el sistema operativo móvil Android y la iniciativa de código abierto que lo construye y mantiene, el *Android Open Source Project* (AOSP), Proyecto de Código Abierto de Android. En su sitio web, el AOSP recomienda el uso de la licencia Apache para liberar el software relacionado con Android, sin embargo, hay partes de Android que se liberan bajo otras licencias, como la licencia GPL para el núcleo (*kernel*) del sistema, puesto que es el núcleo del sistema operativo Linux, utilizado no solamente como base de Android sino también como base para sistemas operativos de escritorio, de servidores, de sistemas embebidos (por ejemplo, cajeros automáticos o navegadores GPS), que está liberado bajo la licencia GPL y, como veremos más adelante, esta licencia obliga a liberar las obras derivadas bajo los mismos términos, esto es, usando la misma licencia GPL, para así mantener las libertades fundamentales del software libre que expusimos en el primer capítulo.

Además de esta distinción entre licencias de software libre y licencias de software propietario, existen otras clasificaciones de las licencias.

Juan Carlos Monroy propone una clasificación según el grado de libertad de uso que se le entrega al licenciario (aquí están las licencias de software libre y de software propietario), según el grado de estandarización de los términos de la licencia, y según la forma de celebración del contrato.

---

<sup>28</sup> Ídem

Según el grado de estandarización, Monroy distingue entre licencias de software genérico (empaquetado) con contratos de adhesión y licencias de software personalizado. Las licencias otorgadas mediante contratos de adhesión se llaman también licencias *shrink-wrap*

Según la forma de celebración del contrato, distingue entre licencias celebradas por escrito; licencias celebradas por otros medios válidos de expresión del consentimiento y licencias celebradas por medios electrónicos). El autor señala que las licencias celebradas por escrito sería un contrato consensual, que se perfecciona por el solo consentimiento de las partes. En el caso de las licencias celebradas por otros medios válidos de expresión del consentimiento, Monroy destaca la aceptación tácita de los términos de la licencia señalando dos casos: cuando existen actos positivos por parte de quien adquiere el software y cuando existe silencio o inactividad por parte de quien adquiere el software. El autor expone la regulación de la oferta y la aceptación en Colombia, que establece que la aceptación tácita debe ser manifestada por hechos inequívocos de ejecución del contrato propuesto. En la doctrina chilena también aparece la concepción de que la aceptación tácita involucra la ejecución de determinados actos que signifiquen conformidad con la oferta.<sup>29</sup> Los actos positivos que señala Monroy son el desempaquetado del software y su instalación. Establece, eso sí, dos requisitos: que el usuario “tenga previo y pleno conocimiento de la licencia, con todas las restricciones y condiciones de uso en su integridad”; y “tenga conocimiento de las consecuencias jurídicas de su conducta (desempacar el software, instalarlo, usarlo, etc.) en cuanto a la conciencia de que por ese acto quedará vinculado jurídicamente tanto a los derechos como a las obligaciones contenidas en ese documento”. Por eso, señala que si, el usuario desempaca el software y luego encuentra un documento de licencia que señala que por solo haber desempacado el software se entiende haber aceptado los términos y condiciones de uso, no podría entenderse que allí hubo un acto de aceptación tácita, porque no se le dio al usuario la oportunidad del conocimiento previo de la licencia.

En el caso de la aceptación tácita mediante el silencio o la inactividad del usuario, Monroy expone que en la regulación internacional, como el artículo 18.1 de la Convención de Viena sobre Compraventa Internacional de Mercaderías o el artículo 2:204 de los Principios Europeos del Derecho de Contratos, el silencio o la inactividad no constituyen aceptación. Por eso, el

---

<sup>29</sup> (Segura)

autor sostiene que para que se consideren tales como una aceptación tácita es necesario que concurren tres requisitos: (1) que exista una negociación en donde “el usuario previamente expresó mediante un acto positivo e inequívoco su interés en recibir la oferta y celebrar el contrato”; (2) que “el usuario tenga pleno y previo conocimiento de la licencia, con todas las condiciones y restricciones de uso en su integridad” ; y (3) que “tenga conocimiento de las consecuencias jurídicas de su conducta (abstenerse de devolver la copia adquirida al vendedor o al productor) en cuanto a la conciencia de que por ese acto quedará vinculado jurídicamente tanto a los derechos como a las obligaciones contenidas en ese documento”.<sup>30</sup> Monroy señala que el primer requisito es determinante y fundamental. Dado el enfoque de este trabajo, pensamos que en muy pocos casos se va a dar la negociación, dado que la mayoría del software orientado a usuarios finales hoy en día se licencia mediante contratos de adhesión, y siendo así, si seguimos a Monroy no cabría la aceptación tácita.

Las licencias celebradas por medios electrónicos también se conocen como licencias *click-wrap*. Según Monroy, “el *click-wrap agreement* es el acuerdo de licencia presentado a los usuarios de un sitio web antes de iniciar la descarga de un programa informático. El acuerdo o contrato es similar a los de los programas informáticos convencionales. Por lo general estos acuerdos obligan al usuario a llevar a cabo algún tipo de acción como hacer clic sobre algún botón regularmente denominado Aceptar”.<sup>31</sup> Aquí, el consentimiento del usuario se expresa en el clic que debe hacer, de ahí su nombre en inglés.

### Análisis de los derechos involucrados en las licencias de software

El método que utilizaremos para analizar los derechos que conceden (o quitan) las licencias de software, será revisar las cláusulas de las licencias más representativas, dividiéndolas entre licencias de software propietario y software libre. En cada caso, vamos a analizar la última versión disponible de la licencia (que en el caso de las licencias de software propietario, se corresponde con la última versión del software objeto de la licencia)

---

<sup>30</sup> Op. Cit. Nota 1

<sup>31</sup> Ídem

## Licencias de software propietario

La primera licencia de software que vamos a analizar va a ser el Contrato de Licencia de Usuario Final del sistema operativo Microsoft Windows, por ser el sistema operativo de escritorio más utilizado.

Este contrato, en su parte 2 dispone:

### “2. Instalación y Derechos de uso.

**a. Licencia.** El software se cede bajo licencia y no es objeto de venta. En virtud del presente contrato y siempre y cuando cumpla con todos los términos del mismo, le otorgamos el derecho de instalar y ejecutar una instancia del software en su dispositivo (el dispositivo con licencia), para que lo utilice una persona a la vez. Si utiliza software de Microsoft o de orígenes autorizados para actualizar software que no es original, no convertirá su versión inicial ni la versión actualizada en original y, en ese caso, no dispondrá de una licencia para utilizar el software.

**b. Dispositivo.** En este contrato, “dispositivo” se refiere a un sistema hardware (ya sea físico o virtual) con un dispositivo de almacenamiento interno capaz de ejecutar el software. Una partición o división (blade) de hardware se considera un dispositivo.

**c. Restricciones.** El fabricante o el instalador y Microsoft se reservan todos los derechos (como los posibles derechos en virtud de determinadas leyes de propiedad intelectual e industrial) no concedidos expresamente en este contrato. Por ejemplo, esta licencia no le concede ningún derecho con respecto a las siguientes operaciones; específicamente, usted no puede:

- (i) utilizar o virtualizar las características del software por separado;
- (ii) publicar, copiar (excepto en el caso de la copia de seguridad permitida), alquilar, arrendar o dar en préstamo el software;
- (iii) transmitir el software (excepto en los casos permitidos por este contrato);
- (iv) eludir las restricciones o limitaciones técnicas del software;
- (v) utilizar el software como software de servidor (para hospedaje comercial), poner el software a disposición de varios usuarios mediante una red para que lo utilicen de forma simultánea,



instalar el software en un servidor y permitir que los usuarios accedan a él de forma remota ni instalar el software en un dispositivo para uso exclusivo de usuarios remotos;

(vi) utilizar técnicas de ingeniería inversa, descompilar ni desensamblar el software o intentar hacerlo, excepto y solo en la medida en que la restricción anterior (a) esté permitida por la legislación aplicable; (b) este permitida por los términos de licencia que rigen el uso de componentes de código abierto que pueden incluirse con el software; o (c) sea necesaria para depurar cambios en cualquier biblioteca licenciada en virtud de la licencia GNU Lesser General Public License que se incluye y vincula mediante el software; y

(vii) al utilizar características basadas en Internet, no podrá utilizarlas de forma tal que pueda interferir con el uso que otras personas hagan de ellas o tratar de utilizar o acceder a algún servicio, dato, cuenta o red sin autorización.”<sup>32</sup>

Como podemos apreciar, se deja patente que el software se concede bajo licencia, para instalar y ejecutar una instancia del software y ser utilizada por una sola persona. Además, se hace presente que, si se tiene un software no concedido bajo licencia, actualizarlo por medios autorizados no hace que se obtenga una licencia de uso.

Asimismo, se establecen ciertas restricciones en el uso del software. Entre las que interesan para los efectos de este trabajo, tenemos la prohibición de eludir las restricciones técnicas del software, y de utilizar técnicas de ingeniería inversa. Sin embargo, se establecen excepciones, como la legislación aplicable y términos de otras licencias aplicables que sí la permitan.

Otra licencia de software propietario que resulta conveniente analizar, es la del sistema operativo móvil de Apple, iOS, debido a que es una plataforma muy utilizada en la actualidad (está presente en dispositivos iPhone y iPad) y que además tiene antecedentes de modificaciones, por ejemplo, el llamado *jailbreak*.

Los apartados 2 y 3 del Contrato de Licencia de Software iOS disponen:

“2. Usos permitidos y restricciones de la Licencia.

---

<sup>32</sup> (Términos de Licencia del Software de Microsoft – Windows 10)

(a) De acuerdo con las condiciones de esta Licencia, se te concede una licencia limitada no exclusiva

para utilizar el Software iOS en un único Dispositivo iOS de marca Apple. Salvo lo expresamente permitido en la sección 2(b) siguiente, y salvo que medie un acuerdo separado entre tú y Apple, esta Licencia no permite que el Software iOS esté instalado en más de un Dispositivo iOS de marca Apple al mismo tiempo, y no podrás distribuir ni colocar el Software iOS en una red donde pueda ser utilizado por varios dispositivos al mismo tiempo. La presente Licencia no te concede ningún derecho para el uso de las interfaces propiedad de Apple, así como ningún derecho de propiedad intelectual con respecto al diseño, desarrollo, fabricación, concesión de licencias o distribución de dispositivos y accesorios de otros fabricantes o de aplicaciones de software de terceros que se usen con el Dispositivo iOS. Algunos de estos derechos se recogen en otras licencias de Apple. Para obtener más información acerca del desarrollo de dispositivos y accesorios de otros fabricantes para Dispositivos iOS, visita el sitio <https://developer.apple.com/programs/mfi/>. Para más información sobre el desarrollo de aplicaciones de software de otros fabricantes para Dispositivos iOS, visita el sitio <https://developer.apple.com>

(b) De acuerdo con las condiciones de esta Licencia, se te concede una licencia limitada no exclusiva para descargar las Actualizaciones del Software iOS que Apple lance para tu modelo de Dispositivo iOS con el fin de actualizar o restaurar el software de cualquier Dispositivo iOS que poseas. Esta Licencia no permite actualizar o restaurar ningún Dispositivo iOS que no sea de tu propiedad, ni distribuir o colocar las Actualizaciones del Software iOS en una red donde puedan ser utilizadas por varios periféricos u ordenadores al mismo tiempo. Si descargas una Actualización del Software iOS en tu ordenador, podrás realizar una copia de las Actualizaciones del Software iOS almacenadas en tu ordenador en un formato legible únicamente a efectos de copia de seguridad, siempre y cuando esta contenga toda la información relativa a derechos de autor y copyright, u otros datos relativos a la propiedad, contenida en el original.

(c) Si Apple ha preinstalado aplicaciones de marca Apple desde App Store en tu Dispositivo iOS en el momento de la compra (“Aplicaciones preinstaladas”), tendrás que iniciar sesión en App Store y asociar a tu cuenta de App Store estas Aplicaciones preinstaladas para poder

utilizarlas en el Dispositivo iOS. Cuando asocies una Aplicación preinstalada a tu cuenta de App Store, se asociarán automáticamente todas las demás Aplicaciones preinstaladas de tu Dispositivo iOS. Al asociar las Aplicaciones preinstaladas a tu cuenta de App Store, aceptas que Apple pueda transmitir, recopilar, mantener, procesar y usar tanto el ID de Apple que utiliza en tu cuenta de App Store como el identificador de hardware único de tu Dispositivo iOS como identificadores únicos de la cuenta con el fin de verificar si tu solicitud es apta y proporcionarte acceso a las Aplicaciones preinstaladas a través de App Store. Si no deseas utilizar una Aplicación preinstalada, puedes eliminarla del Dispositivo iOS en cualquier momento.

(d) No podrás —y aceptas no hacerlo ni permitir a otras personas que lo hagan— copiar (excepto en los casos expresamente permitidos en la presente Licencia), descompilar, desensamblar o intentar obtener el código fuente, como tampoco descifrar, modificar o crear trabajos derivados del Software iOS o de cualquier servicio proporcionado por el Software iOS ni de ninguna de sus partes (excepto en la medida en que la legislación aplicable o las condiciones de licencia que rigen el uso de componentes de código abierto o de código de muestra que puedan estar incluidos en el Software iOS prohíban cualquiera de las restricciones anteriores).

(e) El Software iOS puede utilizarse para reproducir materiales siempre que dicho uso se limite a la reproducción de materiales sin copyright, materiales de los que poseas el copyright o materiales que puedas reproducir porque cuentas con la autorización o el permiso legal correspondientes. Los derechos de titularidad y de propiedad intelectual de cualquiera de los contenidos que se muestran o almacenan en el Dispositivo iOS, o a los que se accedes a través de este, pertenecen a los propietarios del contenido correspondiente. Dicho contenido puede estar protegido por las leyes y los tratados vigentes en materia de propiedad intelectual y de derechos de autor y quedan sujetos a las condiciones de uso de la tercera parte suministradora del contenido. A menos que se establezca lo contrario en el presente documento, esta Licencia no te concede ningún derecho para el uso de dicho contenido ni garantiza que el contenido siga estando disponible para tu utilización.

(f) Te comprometes a utilizar el Software iOS y los Servicios (tal como se definen en el apartado 5 de este documento) de conformidad con todas las leyes aplicables, incluidas las leyes locales del país o la región donde residas o donde te descargues y utilices el Software iOS y los Servicios. Es posible que las funciones del Software iOS y los Servicios no estén disponibles en

todos los idiomas o regiones, que algunas funciones varíen según la región y que tu proveedor de servicios restrinja algunas funciones o no las ofrezca. Algunas de funciones del Software iOS y Servicios requieren una conexión de datos móviles o Wi-Fi.

(g) Necesitas una combinación única de nombre de usuario y contraseña —conocida como ID de Apple— para poder utilizar App Store. El ID de Apple también es necesario para acceder a actualizaciones de aplicaciones y determinadas funciones del Software iOS y los Servicios.

(h) Reconoces que muchas funciones, apps integradas y Servicios del Software iOS transmiten datos y podrían conllevar cargos para tu plan de datos, y que te haces cargo de dichos cargos. Puedes ver y controlar qué aplicaciones tienen autorización para utilizar datos móviles y ver una estimación de la cantidad de datos que consumen en la sección “Datos móviles” de Ajustes. Para obtener más información, consulta el Manual del usuario de tu Dispositivo iOS.

(i) Si seleccionas la opción para permitir la actualización automática de aplicaciones, el Dispositivo iOS buscará periódicamente actualizaciones para las aplicaciones de tu dispositivo y, si las hay, las descargará e instalará automáticamente. Las actualizaciones automáticas se pueden desactivar en cualquier momento en Ajustes, donde debes pulsar “iTunes Store y App Store” y, en “Descargas automáticas”, desactivar “Actualizaciones”.

(j) En determinadas circunstancias, el uso del Dispositivo iOS podría distraerte y dar lugar a una situación peligrosa (por ejemplo, no escribas mensajes de texto mientras conduces ni utilices los auriculares cuando vayas en bicicleta). Al usar el Dispositivo iOS asumes tu responsabilidad a la hora de respetar las normas que prohíben o restringen el uso de teléfonos móviles o auriculares (por ejemplo, el requisito de utilizar opciones manos libres para realizar llamadas durante la conducción).

### 3. Transmisión.

No puedes alquilar, arrendar, prestar, vender, redistribuir ni sublicenciar el Software iOS.

No obstante, puedes transferir tus derechos contenidos en esta Licencia de forma permanente y una sola vez siempre y cuando: (a) transmitas el Dispositivo iOS y la totalidad del Software iOS, incluidos todos los componentes y esta Licencia; (b) no conserves ninguna copia, total o

parcial, del Software iOS, incluidas las copias almacenadas en un ordenador o en cualquier otro dispositivo de almacenamiento; y (c) la otra parte lea y acepte las condiciones de esta Licencia”<sup>33</sup>

Al igual que ocurre con la licencia de Windows, se explicita que se trata de una licencia de uso del software, y que esta es no exclusiva. Además se vuelve a repetir la obligación del dispositivo único. En la letra d de las condiciones, prohíbe la copia, el descompilado, desensamblado o la obtención del código fuente. Asimismo, se prohíbe la modificación o creación de trabajos derivados. En el punto 3 tampoco se permite alquilar, arrendar, prestar, vender, redistribuir ni sublicenciar el software iOS.

Como se puede apreciar, en las licencias de software propietario solamente se autoriza el uso del programa, sin poder hacer otra operación que no esté expresamente autorizada por el licenciante, que además, se reserva los derechos de propiedad intelectual. La única excepción a esto parece ser la legislación aplicable, que vamos a revisar en detalle en el próximo capítulo.

### Licencias de software libre

Vamos a analizar dos licencias de software libre: la licencia General Public License (GPL) y la licencia Apache, por ser dos licencias muy usadas, por tanto, sirven para dar una idea suficiente de qué derechos se conceden o quitan en el software libre.<sup>34</sup>

La GPL establece, en su numeral 3, lo siguiente:

“3. Permisos básicos.

Todos los derechos concedidos bajo esta Licencia se conceden durante la duración de los derechos de autor (“copyright”) del Programa, y son irrevocables siempre que se cumplan las condiciones establecidas. Esta Licencia afirma explícitamente su ilimitado permiso para ejecutar el Programa sin modificar. La salida de la ejecución de una obra amparada está amparada por esta Licencia solo si la salida, dado su contenido, constituye una obra amparada. Esta Licencia

---

<sup>33</sup> (Contrato de Licencia de Software iOS de Apple)

<sup>34</sup> Si bien las licencias que son válidas legalmente son las versiones originales en inglés, en este trabajo analizaremos las respectivas traducciones al español, para facilitar la comprensión.

reconoce sus derechos de uso razonable u otro equivalente, según lo establecido por la ley de derechos de autor (“copyright”).

Usted podrá realizar, ejecutar y difundir obras amparadas que usted no transmita, sin condición alguna, siempre y cuando no tenga otra licencia vigente. Podrá distribuir obras amparadas a terceros con el único propósito de que ellos hagan modificaciones exclusivamente para usted, o proporcionarle ayuda para ejecutar estas obras, siempre y cuando cumpla con los términos de esta Licencia en la transmisión de todo el material del cual usted no controle los derechos de autor (“copyright”). Aquellos que realicen o ejecuten las obras amparadas por usted, deben hacerlo exclusivamente en su nombre, bajo su dirección y control, en los términos que le prohíban realizar ninguna copia de su trabajo con derechos de autor (“copyright”) fuera de su relación con usted.

La transmisión bajo otras circunstancias se permite únicamente bajo las condiciones expuestas a continuación. No está permitido sublicenciar, la sección 10 hace que sea innecesario.”<sup>35</sup>

Como podemos ver, este tipo de licencia es más permisiva que una licencia de software propietario, al conceder un permiso ilimitado de ejecución. También concede permisos de difusión y transmisión bajo ciertas condiciones.

Estas condiciones se establecen en los numerales 5 a 7 de la licencia:  
“5. Transmisión de copias literales.

Usted podrá distribuir copias literales del código fuente del Programa tal cual lo ha recibido, por cualquier medio, siempre que publique visible y apropiadamente en cada copia el correspondiente aviso de derechos de autor (“copyright”); mantenga intactos todos los avisos que establezcan que esta Licencia y cualquier cláusula no-permisiva añadida acorde con la cláusula 7 son aplicables al código; mantenga intactos todos los avisos de ausencia de garantía; y proporcione a todos los destinatarios una copia de esta Licencia junto con el Programa.

---

<sup>35</sup> (Traducción de la licencia GPLv3 al español)

Usted podrá cobrar cualquier importe o no cobrar nada por cada copia que distribuya, y podrá ofrecer soporte o protección de garantía mediante un pago.

6. Transmisión de Versiones Modificadas de la Fuente.

Usted puede transmitir una obra basada en el Programa, o las modificaciones para generarla a partir del Programa, en la forma de código fuente bajo los términos de la sección 4, suponiendo que además cumpla las siguientes condiciones:

- a. La obra debe incluir avisos destacados indicando que usted la ha modificado y dando una fecha pertinente.
- b. La obra debe incluir avisos destacados indicando que está liberada bajo esta Licencia y cualquier otra condición añadida bajo la sección 7. Este requerimiento modifica los requerimientos de la sección 4 de “mantener intactos todos los avisos”.
- c. Usted debe licenciar la obra entera, como una unidad, bajo esta Licencia para cualquier persona que esté en posesión de una copia. Esta Licencia se aplicará por consiguiente, junto con cualquier término aplicable adicional de la sección 7, a la totalidad de la obra, y a todos sus componentes, independientemente de como estén empaquetados. Esta Licencia no da permiso para licenciar la obra de otra forma, pero no invalida esos permisos si usted los ha recibido de forma separada.
- d. Si la obra tiene interfaces de usuario interactivas, cada una debe mostrar los Avisos Legales Apropriados; sin embargo, si el Programa tiene interfaces interactivas que no muestren los Avisos Legales Apropriados, tampoco es necesario que su obra lo haga.

Una recopilación de una obra amparada con otras obras separadas e independientes, que no son por su naturaleza extensiones de la obra amparada, y que no se combinan con ella con el fin de formar un programa más grande, en o sobre un volumen de un medio de almacenamiento o distribución, es llamado un “agregado” si la recopilación y su resultante derechos de autor (“copyright”) no son usados para limitar el acceso o los derechos legales de los usuarios de la recopilación más allá de lo que las obras individuales permitan. La inclusión de una obra amparada en un agregado no provoca que esta Licencia se aplique a los otros componentes del agregado.

## 7. Transmisión en Forma de No-Fuente.

Usted puede transmitir una obra amparada en forma de código objeto bajo los términos de las secciones 4 y 5, siempre que también transmita la Fuente Correspondiente legible por una máquina bajo los términos de esta Licencia, de una de las siguientes formas:

- a. Transmitir el código objeto en, o embebido en, un producto físico (incluyendo medios de distribución físicos), acompañado de la Fuente Correspondiente en un medio físico duradero habitual para el intercambio de software.
- b. Transmitir el código objeto en, o embebido en, un producto físico (incluyendo medios de distribución físicos), acompañado de un ofrecimiento escrito, válido durante al menos tres años y válido mientras usted ofrezca recambios o soporte para ese modelo de producto, de dar a cualquiera que posea el código objeto o (1) una copia de la Fuente Correspondiente de todo el software en el producto amparado por esta Licencia, en un medio físico duradero habitual para el intercambio de software, por un precio no más elevado que el coste razonable de la realización física de la transmisión de la fuente, o (2) acceso para copiar la Fuente Correspondiente de un servidor de red sin costo alguno.
- c. Transmitir copias individuales del código objeto con una copia del ofrecimiento escrito de proveer la Fuente Correspondiente. Esta alternativa está permitida solo ocasionalmente sin fines comerciales, y solo si usted ha recibido el código objeto con ese ofrecimiento, de acuerdo con la subsección 6b.
- d. Transmitir el código objeto ofreciendo acceso desde un lugar determinado (gratuitamente o mediante pago), y ofrecer acceso equivalente a la Fuente Correspondiente de la misma manera en el mismo lugar sin cargo adicional. No es necesario exigir a los destinatarios que copien la Fuente Correspondiente junto con el código objeto. Si el lugar para copiar el código objeto es un servidor de red, la Fuente Correspondiente puede estar en un servidor diferente (gestionado por usted o un tercero) que soporte facilidades de copia equivalentes, siempre que mantenga instrucciones claras junto al código objeto especificando dónde encontrar la Fuente Correspondiente. Independientemente de qué servidor albergue la Fuente Correspondiente, usted seguirá estando obligado a asegurar que está disponible durante el tiempo que sea necesario para satisfacer estos requisitos.



- e. Transmitir el código objeto usando una transmisión peer-to-peer, siempre que informe a los otros usuarios donde se ofrece el código objeto y la Fuente Correspondiente de la obra al público general de forma gratuita bajo la subsección 6d.

Una porción separable del código objeto, cuyo código fuente está excluido de la Fuente Correspondiente, como una Biblioteca del Sistema, no necesita ser incluida en la distribución del código objeto de la obra.

Un “Producto de Usuario” es o (1) un “producto de consumo”, lo que significa cualquier propiedad tangible personal que es usada habitualmente con fines personales, familiares o domésticos, o (2) cualquier cosa diseñada o vendida para ser incorporada en una vivienda. A la hora de determinar cuando un producto es un producto de consumo, los casos dudosos serán resueltos en favor de la cobertura. Para un producto concreto recibido por un usuario concreto, “uso habitual” se refiere a un uso típico y común de esa clase de producto, sin tener en cuenta el estado del usuario concreto o la forma en la que el usuario concreto realmente use, o espera o se espera que use, el producto. Un producto es un producto de consumo independientemente de si el producto tiene usos esencialmente comerciales, industriales o no comerciales, a menos que dicho uso constituya el único modo de uso significativo del producto.

La “Información de Instalación” de un Producto de Usuario quiere decir cualquier método, procedimiento, clave de autorización, u otra información requerida para instalar y ejecutar versiones modificadas de la obra amparada en ese Producto de Usuario a partir de una versión modificada de su Fuente Correspondiente. La información debe ser suficiente para garantizar que el funcionamiento continuado del código fuente modificado no es prevenido o interferido por el simple hecho de que ha sido modificado.

Si usted transmite una obra en código objeto bajo esta sección en, o con, o específicamente para usar en, un Producto de Usuario, y la transmisión tiene lugar como parte de una transacción en la cual el derecho de posesión y uso de un Producto de Usuario es transferido a un destinatario en perpetuidad o por un periodo establecido (independientemente de cómo se caracterice la operación), la Fuente Correspondiente transmitida bajo esta sección debe estar acompañada de la Información de Instalación. Pero este requisito no se aplica si ni usted ni ningún tercero tiene

la capacidad de instalar código objeto modificado en el Producto de Usuario (por ejemplo, la obra ha sido instalada en la ROM).

El requisito de proveer de la Información de Instalación no incluye el requisito de continuar proporcionando asistencia, garantía, o actualizaciones para una obra que ha sido modificada o instalada por el destinatario, o para un Producto de Usuario en el cual ha sido modificada o instalada. El acceso a una red puede ser denegado cuando la modificación en sí afecta materialmente y adversamente el funcionamiento de la red o viola las reglas y protocolos de comunicación de la red.

La Fuente Correspondiente transmitida, y la Información de Instalación proporcionada, de acuerdo con esta sección debe estar en un formato que sea documentado públicamente (y con una implementación disponible para el público en formato de código fuente), y no deben necesitar contraseñas o claves particulares para la extracción, lectura o copia.”<sup>36</sup>

Como vemos, permite la transmisión en varios casos, siempre que se respete la licencia GPL al hacer la transmisión, y se haga saber a los destinatarios. Además, y a diferencia de las licencias de software propietario, se permite la transmisión de modificaciones u obras derivadas.

En el caso de la licencia Apache, en el numeral 2 se especifican las condiciones en que se otorga la licencia:

“2. Concesión de licencia de derechos de autor.

De acuerdo con las condiciones de esta Licencia, por la presente, cada Contribuyente le otorga a Usted una licencia de derechos de autor irrevocable, perpetua, internacional, no exclusiva, sin cargas ni regalías, para reproducir, preparar Obras Derivativas, mostrar u operar públicamente, sublicenciar y distribuir la Obra y tales Obras Derivativas en forma de Código o de Objeto.”<sup>37</sup>

---

<sup>36</sup> Ídem

<sup>37</sup> (Licencia Apache 3.0 – Traducción al español)

Como se puede apreciar, al igual que ocurre con la licencia GPL, en la licencia Apache se deja reproducir, crear obras derivadas, sublicenciar y distribuir esas obras, ya sea en forma código fuente o código objeto.

En los numerales 4 y 5 de la licencia se establecen las condiciones para reproducir y distribuir:

#### “4. Redistribución.

Usted podrá reproducir y distribuir copias de la Obra o de las Obras Derivativas de la misma en cualquier medio, con o sin modificaciones, en forma de Código o de Objeto, siempre que cumpla las siguientes condiciones:

1. Deberá facilitar a todos los demás receptores de la Obra u Obras Derivativas una copia de esta Licencia y
2. Deberá indicar claramente las modificaciones que haya realizado en los archivos; y
3. Deberá conservar en la forma de Código de todas las Obras Derivativas que Usted distribuya todas las advertencias relativas a los derechos de autor, patentes, marcas registradas y atribuciones en la forma de Código de la Obra, con exclusión de las advertencias que no pertenezcan a ninguna parte de las Obras Derivativas, y
4. Si la Obra incluye un archivo de texto de "ADVERTENCIA/AVISO" como parte de su distribución, todas las Obras Derivativas que Usted distribuya deberán incluir una copia legible de las advertencias de atribución contenidas en dicho archivo de ADVERTENCIA, con exclusión de las advertencias que no pertenezcan a ninguna parte de las Obras Derivativas, en al menos uno de los lugares siguientes: En el contenido del archivo de texto de ADVERTENCIA o AVISO distribuido como parte de la Obra Derivativa; en el contenido de la forma de Código o en la documentación, si estos se facilitan junto con las Obras Derivativas; o mediante un dispositivo de muestra generado por las Obras Derivativas, en el lugar en que normalmente aparezcan tales advertencias externas. El contenido del archivo de ADVERTENCIA es meramente informativo y no modifica la Licencia. Usted podrá añadir Sus propias advertencias de atribución en las Obras Derivativas que Usted distribuya, junto con el texto de ADVERTENCIA de la Obra, o como adición al mismo, siempre que estas advertencias de atribución adicionales

no puedan interpretarse como una modificación de la Licencia. Usted podrá añadir Su propia declaración respecto a los derechos de autor en Sus modificaciones y podrá añadir distintas condiciones en la Licencia para el uso, la reproducción o la distribución de Sus modificaciones o de las Obras Derivativas en conjunto, siempre que Su uso, reproducción y distribución de la Obra cumpla por lo demás las condiciones establecidas en esta Licencia.

## 5. Envío de contribuciones.

A menos que Usted indique lo contrario, todas las Contribuciones que Usted envíe deliberadamente al Licenciador para su inclusión en la Obra, estarán sujetas a las condiciones de esta Licencia sin aplicación de condiciones adicionales. No obstante, ninguna disposición de este documento invalidará ni modificará las condiciones de cualquier otro acuerdo de Licencia que usted pueda haber suscrito con el Licenciador en relación con tales Contribuciones.”<sup>38</sup>

Como podemos observar, las condiciones son similares a las de la licencia GPL en la medida en que también se exige facilitar una copia de la licencia en las modificaciones u obras derivadas. Sin embargo, a diferencia de la licencia GPL, el numeral 5 permite que esas modificaciones u obras derivadas se puedan liberar bajo otra licencia. Por eso se dice que la licencia Apache es una licencia permisiva, ya que otras licencias de software libre como la licencia GPL obligan a liberar las obras derivadas bajo la misma licencia.

## Conclusiones del capítulo

Luego de lo que hemos revisado en este capítulo, podemos sacar las siguientes conclusiones:

1. Las licencias son contratos entre el desarrollador o quien tiene los derechos de propiedad del software y quien lo adquiere o usa, donde se establecen las condiciones de uso de aquel software
2. Existen variados tipos de licencias, algunas más permisivas que otras. En general, las licencias propietarias conceden solo el derecho de uso, reservando el resto de los derechos para el licenciante y prohibiendo expresamente las

---

<sup>38</sup> ídem

modificaciones no autorizadas. En cambio, las licencias libres permiten el uso y además otros derechos, como el de modificar y hacer obras derivadas, pero bajo condiciones diversas. Sin embargo, las licencias propietarias autorizan las modificaciones siempre que estén amparadas por la legislación vigente, y eso es a lo que nos vamos a dedicar en el siguiente capítulo.

## Capítulo 3: El ordenamiento jurídico chileno y la modificación de software

En este capítulo vamos a revisar las normas de la Constitución Política de la República, y las normas de rango legal que, de acuerdo a la materia que regulan, consideramos que tienen relación con permitir o prohibir la modificación de un software por parte del usuario final. El método de análisis a emplear va a ser revisar las disposiciones de la ley, interpretarlas y ver qué grado de libertad entregan o quitan.

### Constitución Política de la República

La Constitución aborda los derechos y deberes constitucionales en el artículo 19. Interesa especialmente a efectos de este trabajo el numeral 25, referido al derecho de propiedad, y más específicamente, al derecho de propiedad intelectual. Sin embargo, desde el punto de vista del usuario o consumidor del software, resulta interesante analizar también los numerales 4 y 5 del mencionado artículo, referidos al respeto y protección de la vida privada, y a la inviolabilidad de las comunicaciones, respectivamente.

El numeral 25 dispone que (La Constitución asegura a todas las personas) “La libertad de crear y difundir las artes, así como el derecho del autor sobre sus creaciones intelectuales y artísticas de cualquier especie, por el tiempo que señale la ley y que no será inferior al de la vida del titular. El derecho de autor comprende la propiedad de las obras y otros derechos, como la paternidad, la edición y la integridad de la obra, todo ello en conformidad a la ley. Se garantiza, también, la propiedad industrial sobre las patentes de invención, marcas comerciales, modelos, procesos tecnológicos u otras creaciones análogas, por el tiempo que establezca la ley.”

En base a esta disposición, podemos afirmar que el derecho de autor del software, como creación intelectual, pertenece a quien lo crea, y por tanto es esta persona quien decide que se puede y qué no se puede realizar con el software, a través de las licencias, según vimos en el segundo capítulo.

El numeral 4 del artículo 19 establece que (La Constitución asegura a todas las personas) “El respeto y protección a la vida privada y a la honra de la persona y su familia, y asimismo, la protección de sus datos personales. El tratamiento y protección de estos datos se efectuará en la forma y condiciones que determine la ley”

Resulta interesante analizar esta disposición, ya que ha sufrido una reciente modificación, establecida por la ley 21.096, publicada el 16 de junio del año 2018, que agregó la referencia la protección de los datos personales. Esto tiene bastante relevancia, a nuestro juicio, ya que esta norma operaría como límite a lo que un software puede realizar con nuestros datos personales, procurando siempre que el tratamiento se realice de forma que se protejan, en conformidad a lo establecido en la ley (que, específicamente, se trata de la ley 19.628, sobre protección de la vida privada, que analizaremos más adelante). También podría servir como argumento, en una hipotética acción de protección motivada por la vulneración a este numeral, para señalar que sí se podría modificar un software cuando el tratamiento de datos personales que haga no se conforme a lo establecido en la Constitución y las leyes.

El numeral 5 del artículo 19, señala que (la Constitución asegura a todas las personas) “La inviolabilidad del hogar y de toda forma de comunicación privada. El hogar sólo puede allanarse y las comunicaciones y documentos privados interceptarse, abrirse o registrarse en los casos y formas determinados por la ley”. Este numeral puede considerarse, a nuestro juicio, una extensión de lo establecido en el numeral anterior, ya que un software también podría estar contraviniendo la Constitución y las leyes cuando no puede asegurar que las comunicaciones privadas sean inviolables, o al menos, cuando no se tomen las medidas adecuadas a proteger estas comunicaciones, que cada vez más se realizan por medios electrónicos.

En suma, la Constitución Política de la República establece dos derechos básicos que podemos aplicar a la relación autor del software-software-usuario: primero, el derecho de propiedad del autor del software sobre su obra, y por tanto la libertad para decidir lo que los usuarios pueden hacer con él. Segundo, el derecho de las personas -principalmente los usuarios-, a que no se

vean afectados sus datos personales y sus comunicaciones privadas, que a su vez opera como límite al primer derecho, y como posible argumento para omitirlo en caso de que se produzca una vulneración.

## Ley de Propiedad Intelectual (17.336)

Por su especialidad, la primera norma de rango legal que vamos a revisar es la ley 17.336 de Propiedad Intelectual.

El objeto de protección está señalado en el artículo 3° número 16 de la ley, que señala “Quedan especialmente protegidos con arreglo a la presente ley: (...) 16) Los programas computacionales, cualquiera sea el modo o forma de expresión, como programa fuente o programa objeto, e incluso la documentación preparatoria, su descripción técnica y manuales de uso.”

De acuerdo a lo revisado en el primer capítulo, podemos decir que la ley de Propiedad Intelectual entrega una protección amplia al software, dado que no solo se protege el código fuente, también el código objeto y la documentación asociada. E incluso, lo amplía más al establecer que cualquiera sea el modo o forma de expresión, la protección es aplicable.

El título III de la ley trata acerca de las limitaciones y excepciones al derecho de autor y derechos conexos. En particular, se refiere al software en el artículo 71 Ñ:

“Artículo 71 Ñ. Las siguientes actividades relativas a programas computacionales están permitidas, sin que se requiera autorización del autor o titular ni pago de remuneración alguna:

a) La adaptación o copia de un programa computacional efectuada por su tenedor, siempre que la adaptación o copia sea esencial para su uso, o para fines de archivo o respaldo y no se utilice para otros fines.

Las adaptaciones obtenidas en la forma señalada no podrán ser transferidas bajo ningún título, sin que medie autorización previa del titular del derecho de autor respectivo; igualmente, las copias obtenidas en la forma indicada no podrán ser transferidas bajo ningún título, salvo que lo sean conjuntamente con el programa computacional que les sirvió de matriz.

b) Las actividades de ingeniería inversa sobre una copia obtenida legalmente de un programa computacional que se realicen con el único propósito de lograr la compatibilidad operativa entre programas computacionales o para fines de investigación y desarrollo. La información así obtenida no podrá utilizarse para producir o comercializar un programa computacional similar que atente contra la presente ley o para cualquier otro acto que infrinja los derechos de autor.

c) Las actividades que se realicen sobre una copia obtenida legalmente de un programa computacional, con el único propósito de probar, investigar o corregir su funcionamiento o la seguridad del mismo u otros programas, de la red o del computador sobre el que se aplica. La información derivada de estas actividades solo podrá ser utilizada para los fines antes señalados.”

Podemos observar que hay tres hipótesis de excepción, donde no se requiere autorización del autor o titular ni pago de remuneración. Estas son: (1) adaptación o copia del programa, (2) actividades de ingeniería inversa y (3) actividades realizadas sobre una copia de un programa para probar, investigar o corregir su funcionamiento o la seguridad. Cada excepción tiene determinados requisitos, así, la primera excepción requiere que la adaptación o copia sea esencial para el uso del programa, o tenga fines de archivo o respaldo exclusivamente (porque no debe utilizarse para otros fines). Además, señala que estas adaptaciones no pueden ser transferidas bajo ningún título, por lo que son de uso personal. La segunda excepción, a su vez, requiere que las actividades de ingeniería inversa se realicen con un fin específico, cual es el de lograr compatibilidad operativa entre programas o con fines de investigación y desarrollo. Al igual que ocurre con la primera excepción, aquí también se establece un requisito que podríamos llamar “de uso personal”, al establecer que la información obtenida no podrá utilizarse para producir o comercializar un programa computacional similar. La tercera excepción requiere que la copia del programa sobre la que se hacen las actividades haya sido obtenida legalmente. Además, restringe los propósitos, al señalar como únicos propósitos lo de probar, investigar, corregir el funcionamiento (del programa) u otros programas, como también de la red o del computador sobre el que se aplica (el programa).

Entonces, podemos señalar como una nota común en las excepciones, el llamado “requisito de uso personal”, es decir, que los trabajos derivados no se compartan con otras personas. Además,



otro requisito general en las excepciones a los derechos de autor y derechos conexos, es que no exista alguna “actividad relacionada con la venta y con el lucro”<sup>39</sup>, como apunta Álvaro Díaz.

En base a esta regulación, podemos afirmar que en Chile sí se permite la modificación de software, pero dados los requisitos que aparecen, estas modificaciones serían solo para uso personal y sin perseguir fines de lucro.

Según vimos en el capítulo anterior, estas excepciones no serían aplicables para buena parte del software libre, en la medida en que sus licencias son permisivas y van más allá de lo que establece la Ley de Propiedad Intelectual.

### Ley de Protección al Consumidor (19.496)

Resulta interesante incorporar en el análisis de la legislación a la Ley de Protección al Consumidor, en la medida en que normalmente, uno no tiende a relacionar en primera instancia al proveedor de un software como un sujeto al que le resultaría aplicable esta ley, ni a quien utiliza (y modifica) un software como un consumidor. Sin embargo, creemos que sí resultarían aplicables las disposiciones de esta ley si hablamos de un software estandarizado, y de venta al público general.

El artículo 1 de la ley establece las definiciones usadas en la misma. Este artículo establece en su número 1 lo que debe entenderse por consumidores o usuarios, estableciendo que son tales “las personas naturales o jurídicas que, en virtud de cualquier acto jurídico oneroso, adquieren, utilizan, o disfrutan, como destinatarios finales, bienes o servicios.” En base a esta definición, podríamos señalar que quien adquiere un software y paga un precio por él, se encuentra protegido por esta ley. En el número 2 del mismo artículo define a los proveedores, entendiendo como tales “las personas naturales o jurídicas, de carácter público o privado, que habitualmente desarrollen actividades de producción, fabricación, importación, construcción, distribución o comercialización de bienes o de prestación de servicios a consumidores, por las que se cobre precio o tarifa (...)”. Tal como una persona que compra un software puede ser entendida como un consumidor en el sentido de esta ley, pensamos que una empresa que vende software (ya sea en un medio físico o virtual, o como vimos en el primer capítulo, entregado como un servicio)

---

<sup>39</sup> Díaz p. 51

también puede ser entendida como un proveedor conforme a la Ley de Protección al Consumidor.

Si asumimos entonces que la adquisición de un software está sujeta a la Ley de Protección al Consumidor, tenemos que revisar qué derechos y qué deberes concede a cada parte la ley.

El artículo 3 expone los derechos y deberes básicos de los consumidores. Dispone lo siguiente:

“Artículo 3º.- Son derechos y deberes básicos del consumidor:

a) La libre elección del bien o servicio. El silencio no constituye aceptación en los actos de consumo;

b) El derecho a una información veraz y oportuna sobre los bienes y servicios ofrecidos, su precio, condiciones de contratación y otras características relevantes de los mismos, y el deber de informarse responsablemente de ellos;

c) El no ser discriminado arbitrariamente por parte de proveedores de bienes y servicios;

d) La seguridad en el consumo de bienes o servicios, la protección de la salud y el medio ambiente y el deber de evitar los riesgos que puedan afectarles;

e) El derecho a la reparación e indemnización adecuada y oportuna de todos los daños materiales y morales en caso de incumplimiento de cualquiera de las obligaciones contraídas por el proveedor, y el deber de accionar de acuerdo a los medios que la ley le franquea,

y

f) La educación para un consumo responsable, y el deber de celebrar operaciones de consumo con el comercio establecido.”

De los derechos y deberes que enumera este artículo, nos parecen importantes a efectos de este trabajo los de las letras B, C, E y F, esto es, el derecho a una información veraz y oportuna sobre los bienes y servicios ofrecidos; el derecho a la seguridad en el consumo de bienes o servicios; el derecho a una reparación e indemnización adecuada; y el deber de educación para un consumo responsable.

Esto, en la medida en que, si los consumidores están informados acerca de las características del software que están adquiriendo, probablemente en el futuro tengan menos problemas respecto

al uso que le dan al mismo. Por ejemplo, si un consumidor sabe que necesita extender las funciones de un software en el tiempo, pero no tiene los recursos económicos para contratar un software a medida, probablemente sea mejor optar por un software libre que le permita realizar esa extensión de funciones sin tener que recurrir a un permiso del titular de derechos de autor, que es lo que tendría que hacer si en un principio hubiese adquirido un software propietario (salvo que esas modificaciones puedan considerarse dentro de los casos de excepción contemplados en la Ley de Propiedad Intelectual que ya vimos, aunque en la mayoría de los casos donde se busca un software extensible en el tiempo, es porque se aplica a alguna actividad productiva, y en esos casos habría fines de lucro, lo que imposibilita la aplicación de las excepciones).

El otro derecho que consideramos importante es la seguridad en el consumo de bienes y servicios, y aquí surge una pregunta: ¿es posible hablar de seguridad en el consumo (o mejor dicho, en el uso) de los softwares? En nuestra opinión es un absoluto sí, dada la importancia creciente que tienen los programas computacionales en la vida actual, por ejemplo, sistemas automáticos que controlan el movimiento de los trenes por las vías; los sistemas informáticos de cientos de bancos alrededor del mundo; el sistema operativo de nuestro *smartphone* o computador; el software que está detrás de sistemas de reconocimiento facial que están empezando a implementarse cada vez con más frecuencia ya no solo en espacios privados, también en espacios abiertos y públicos; y de cara al futuro, el software que controlará los vehículos autónomos que en los próximos años circularán por las carreteras. Ya hemos conocido casos donde por el hackeo o el mal uso de un software se ha puesto en riesgo la seguridad de sistemas críticos o afectado en gran medida a una empresa, organización o poniendo en riesgos a muchas personas. Por ejemplo, el caso del *ransomware*<sup>40</sup> Wannacry, que afectó a muchas empresas e incluso gobiernos a nivel mundial, entre ellos la red interna de la empresa española Telefónica y los sistemas de hospitales públicos en el Reino Unido<sup>41</sup>, o el caso del *bug* llamado *Stagefright* en el sistema operativo Android, que abrió una puerta para el hackeo de millones de

---

<sup>40</sup> Un *ransomware* (del inglés *ransom*, rescate) es un software malicioso que encripta y oculta los archivos presentes en una máquina o computador para luego pedir un pago en dinero, servicios o especies para que el propietario de los archivos los pueda recuperar.

<sup>41</sup> ("El ciberataque con el virus WannaCry se extiende a nivel mundial", 2017).

dispositivos que usan dicho sistema.<sup>42</sup> (En cualquier caso, al momento de escribir este trabajo, muchas de esas vulnerabilidades han sido solucionadas)

Entonces, una vez que hemos dicho que la seguridad en el uso de un software es una materia muy importante, surge otra pregunta ¿qué ocurre si por un *bug* de un software nos vemos expuestos, o peor, resultamos atacados o vulnerados y se compromete, por ejemplo, nuestra seguridad o privacidad? Aquí es donde una posible respuesta la podemos encontrar en la letra E del artículo 3 de la Ley de Protección al Consumidor. Es decir, en el derecho a una reparación e indemnización adecuada y oportuna de todos los daños materiales y morales. No vamos a entrar en este trabajo a revisar las condiciones para que una posible indemnización se haga efectiva, tan solo nos vamos a limitar a afirmar que el derecho a una indemnización se puede ver atenuado por el hecho de que la ley requiere un incumplimiento de las obligaciones contraídas por el proveedor, y en la mayoría de los casos ocurre que en los contratos de licencia de software, el proveedor se exime de cierta responsabilidad, y además, se someten los posibles conflictos a arbitraje, lo que en nuestra opinión permite negociar de forma más amplia una salida al conflicto, que no necesariamente puede ser una indemnización.<sup>43</sup>

Por último, como colofón de los derechos y deberes de -en este caso- los usuarios de un software, es importantísimo a nuestro juicio destacar el deber señalado en la letra F del artículo 3, la educación para un consumo responsable. Si no nos informamos acerca de cómo funciona un software, qué posibles vulnerabilidades puede tener, cómo nos pueden afectar y cómo se pueden solucionar, vamos a estar en una mala posición para poder, primero, solucionar esas vulnerabilidades y controlar los daños que provocan; y segundo, poder accionar y obtener indemnizaciones según sea el caso.

Continuando con el análisis de la Ley de Protección al Consumidor, el párrafo 3º establece las obligaciones del proveedor. El artículo 12 entrega una suerte de regla general, al establecer que “Todo proveedor de bienes o servicios estará obligado a respetar los términos, condiciones y modalidades conforme a las cuales se hubiere ofrecido o convenido con el consumidor la entrega del bien o la prestación del servicio.”. El artículo 12 A se refiere a los contratos celebrados por medios electrónicos, y resulta interesante analizarlo en este trabajo toda vez que buena parte de

---

<sup>42</sup> ("The 'Stagefright' exploit: What you need to know", 2017)

<sup>43</sup> Véase por ejemplo la parte 10 del Contrato de Licencia de Usuario Final del sistema operativo Microsoft Windows usado en el capítulo anterior, donde se establece una cláusula de arbitraje.

los contratos de licencia de software se suscriben hoy electrónicamente. Este artículo reza “En los contratos celebrados por medios electrónicos, y en aquéllos en que se aceptare una oferta realizada a través de catálogos, avisos o cualquiera otra forma de comunicación a distancia, el consentimiento no se entenderá formado si el consumidor no ha tenido previamente un acceso claro, comprensible e inequívoco de las condiciones generales del mismo y la posibilidad de almacenarlos o imprimirlos. La sola visita del sitio de Internet en el cual se ofrece el acceso a determinados servicios, no impone al consumidor obligación alguna, a menos que haya aceptado en forma inequívoca las condiciones ofrecidas por el proveedor. Una vez perfeccionado el contrato, el proveedor estará obligado a enviar confirmación escrita del mismo. Ésta podrá ser enviada por vía electrónica o por cualquier medio de comunicación que garantice el debido y oportuno conocimiento del consumidor, el que se le indicará previamente. Dicha confirmación deberá contener una copia íntegra, clara y legible del contrato.”

Este artículo lo podemos relacionar con lo que vimos en el capítulo 2 respecto de la aceptación tácita de los contratos de licencia, viendo que si aplicamos esta regla, no se podría entender como perfeccionado el consentimiento si el usuario del software no es capaz de comprender las condiciones generales de un contrato de licencia.

El artículo 20 trata acerca de los casos en que un consumidor puede optar por la reparación, reposición o devolución de la cantidad pagada. Establece:

“Artículo 20.- En los casos que a continuación se señalan, sin perjuicio de la indemnización por los daños ocasionados, el consumidor podrá optar entre la reparación gratuita del bien o, previa restitución, su reposición o la devolución de la cantidad pagada:

- a) Cuando los productos sujetos a normas de seguridad o calidad de cumplimiento obligatorio no cumplan las especificaciones correspondientes;
- b) Cuando los materiales, partes, piezas, elementos, sustancias o ingredientes que constituyan o integren los productos no correspondan a las especificaciones que ostenten o a las menciones del rotulado; c) Cuando cualquier producto, por deficiencias de fabricación, elaboración, materiales, partes, piezas, elementos, sustancias, ingredientes, estructura, calidad o condiciones sanitarias, en su caso, no sea enteramente apto para el uso o consumo al que está destinado o al que el proveedor hubiese señalado en su publicidad;

d) Cuando el proveedor y consumidor hubieren convenido que los productos objeto del contrato deban reunir determinadas especificaciones y esto no ocurra;

e) Cuando después de la primera vez de haberse hecho efectiva la garantía y prestado el servicio técnico correspondiente, subsistieren las deficiencias que hagan al bien inapto para el uso o consumo a que se refiere la letra c). Este derecho subsistirá para el evento de presentarse una deficiencia distinta a la que fue objeto del servicio técnico, o volviere a presentarse la misma, dentro de los plazos a que se refiere el artículo siguiente;

f) Cuando la cosa objeto del contrato tenga defectos o vicios ocultos que imposibiliten el uso a que habitualmente se destine;

g) Cuando la ley de los metales en los artículos de orfebrería, joyería y otros sea inferior a la que en ellos se indique.

Para los efectos del presente artículo se considerará que es un solo bien aquel que se ha vendido como un todo, aunque esté conformado por distintas unidades, partes, piezas o módulos, no obstante que éstas puedan o no prestar una utilidad en forma independiente unas de otras. Sin perjuicio de ello, tratándose de su reposición, ésta se podrá efectuar respecto de una unidad, parte, pieza o módulo, siempre que sea por otra igual a la que se restituye.”

Según podemos ver, este artículo resulta más apto de aplicar en situaciones donde los bienes involucrados son bienes físicos, tangibles. Sin embargo, la ley no distingue, así que también se puede aplicar a un software. Esto implica, en nuestra opinión, que si un software no cumple con las especificaciones acordadas, se podría pedir la restitución de lo pagado, o la reparación del software. Al tratarse de un bien intangible y de fácil reproducción, creemos que en un caso así no procedería la devolución del mismo, principalmente porque no es necesario que el proveedor tenga acceso a esa copia defectuosa si el resto de las copias deberían ser idénticas, y por tanto, tener los mismos defectos.

Como conclusión del análisis de la Ley de Protección al Consumidor, podemos decir que en este cuerpo legal no se abordan hipótesis de prohibir o permitir la modificación de un software o de cualquier otro producto para subsanar sus defectos o proteger la seguridad, libertad o propiedad del usuario. Simplemente se dan reglas para ver cómo accionar contra un proveedor

en caso de fallas, pero la ley no se pronuncia acerca de posibles modificaciones realizadas por el mismo usuario en el producto.

### Ley que Tipifica Figuras Penales Relativas a la Informática (19.223)

Antes de entrar al análisis de esta ley, debemos tener en cuenta que data del año 1993, por lo que varias de sus disposiciones han quedado superadas por la realidad tecnológica actual. Tanto es así, que incluso hay iniciativas para reemplazarla o actualizarla, como dos proyectos de ley presentados durante el año 2015, los boletines 9998-07 y 10145-07. Esta ley se ha criticado por no ser capaz de tipificar delitos como el *phishing* o el *pharming*.<sup>44</sup> La doctrina existente en Chile respecto de esta ley, ha señalado que como la ley resulta insuficiente para caracterizar esos delitos, se ha intentado tipificar como estafa los delitos ya señalados, aunque no resultaría en todos los casos. Así, Romina Moscoso señala que cabría tipificar como estafa un delito de *phishing*, esto es, de falsificación de un sitio web, pero que no se podría calificar como estafa una interceptación de las claves.<sup>45</sup> Además, la misma autora señala que “La Ley 19.223 presenta varias deficiencias, no sólo en cuanto a formulación del bien jurídico, como se argumentó, sino que también en cuanto al manejo de conceptos claves, nociones jurídicas y entendimiento del fenómeno informático, las que condujeron a un resultado que peca por exceso, al abarcar su protección al soporte físico y por abarcar los sistemas no automatizados de tratamiento de información, y por ende la información contenida en soportes materiales; peca por defecto al no integrar otras formas de vulneración a la informática; y peca por ser repetitiva en sí misma”.<sup>46</sup>

Aún con todo lo ya dicho, vamos a revisar el texto legal para ver si entrega directrices que pueden ser útiles para nuestro trabajo.

El artículo 1 señala: “El que maliciosamente destruya o inutilice un sistema de tratamiento de información o sus partes o componentes, o impida, obstaculice o modifique su funcionamiento, sufrirá la pena de presidio menor en su grado medio a máximo.

---

<sup>44</sup> El *phishing* consiste en hacer pasar una página web falsa como si fuera la original (por ejemplo, de un banco) para así poder robar información sensible, como contraseñas. En el *phishing* se usan técnicas de ingeniería social para hacer que la víctima caiga. El *pharming* también consiste en la suplantación de un sitio web y robo de información confidencial, pero en lugar de usar técnicas de ingeniería social que induzcan al engaño, se reemplaza directamente la dirección legítima del sitio web, usando técnicas como la redirección a través del sistema DNS.

<sup>45</sup> (Moscoso Escobar, 2014) p. 64

<sup>46</sup> Ídem p. 73

Si como consecuencia de estas conductas se afectaren los datos contenidos en el sistema, se aplicará la pena señalada en el inciso anterior, en su grado máximo.”

Como podemos ver aquí, el lenguaje utilizado es bastante amplio, y tal como señala Romina Moscoso se incorpora tanto al soporte físico (hardware) como al soporte lógico (software). Además, al señalar “maliciosamente” implica que para llevar a cabo la conducta típica se requiere dolo directo y conciencia de la antijuridicidad. En lo que interesa para este trabajo la clave está en donde se señala “(...) modifique su funcionamiento (...)”, pero ¿qué se entiende por modificar el funcionamiento? La ley no lo señala, y creemos que ahí cae en ambigüedades que hacen difícil su aplicación. Además, en el supuesto que manejamos para este trabajo, que es el de un usuario final que necesita modificar un software para proteger sus derechos, como libertad, privacidad, seguridad o propiedad, no creemos que se dé lugar a una conciencia de antijuridicidad, sino más bien a una conciencia de vulneración de derechos que necesitan ser restaurados.

El artículo 2 reza “El que con el ánimo de apoderarse, usar o conocer indebidamente de la información contenida en un sistema de tratamiento de la misma, lo intercepte, interfiera o acceda a él, será castigado con presidio menor en su grado mínimo a medio”

Aquí vuelve a hacerse patente la amplitud de los conceptos, y no tenemos claro qué implicaría interceptar, interferir o acceder a un sistema de tratamiento de la información. Además, se señala que el ánimo con que se hagan tales acciones debe ser “indebido”, y tampoco queda claro qué se entiende por tal concepto. En nuestra opinión, se intenta establecer que también sería necesario dolo y una conciencia de antijuridicidad, y tal como señalamos en el análisis del artículo 1 de esta ley, una modificación de software en el sentido que quiere darle este trabajo no busca ser antijurídica.

El artículo 3 dispone “El que maliciosamente altere, dañe o destruya los datos contenidos en un sistema de tratamiento de información, será castigado con presidio menor en su grado medio”

Este artículo es quizás un poco más preciso, dado que en nuestra opinión establece resultados que son más fáciles de comprobar. Así, donde había información es relativamente sencillo verificar si está se alteró o se destruyó. Sin embargo, tampoco resultaría aplicable a nuestro caso,



dado que nuevamente requiere una acción “maliciosa”, y en nuestro concepto, una modificación de software para proteger derechos no tiene porqué ser tal.

Vamos a omitir el análisis del último artículo de la ley, dado que se refiere a la revelación y difusión maliciosa de datos de un sistema informático, situación que consideramos poco relevante a efectos de este trabajo, porque al hacer una modificación de un software no es necesario, generalmente, difundir los datos.

### Ley sobre Protección de la Vida Privada (19.628)

Resulta de interés analizar la ley sobre Protección de la Vida Privada, también llamada Ley de Protección de Datos Personales, por cuanto la forma en que un software podría vulnerar los derechos de los usuarios, en lo que concierne a la privacidad, es justamente haciendo un tratamiento incorrecto de sus datos personales, de forma que el carácter privado de ellos se pierda.

Esta ley tiene un enfoque amplio respecto al tratamiento de información, y abarca no solamente el tratamiento de datos por medios electrónicos, sino también por otros medios. El artículo 2 define ciertos conceptos usados en la ley. La letra f de este artículo define qué se entiende por datos personales. Dispone “Datos de carácter personal o datos personales, los relativos a cualquier información concerniente a personas naturales, identificadas o identificables.” La letra o define el concepto de tratamiento de datos. Establece “Tratamiento de datos, cualquier operación o complejo de operaciones o procedimientos técnicos, de carácter automatizado o no, que permitan recolectar, almacenar, grabar, organizar, elaborar, seleccionar, extraer, confrontar, interconectar, disociar, comunicar, ceder, transferir, transmitir o cancelar datos de carácter personal, o utilizarlos en cualquier otra forma.” La letra ñ define al titular de los datos, estableciendo que es “la persona natural a la que se refieren los datos de carácter personal”. Aquí cabe destacar la diferencia con el responsable del tratamiento de los datos, dado que la letra n del artículo 2 señala “Responsable del registro o banco de datos, la persona natural o jurídica privada, o el respectivo organismo público, a quien compete las decisiones relacionadas con el tratamiento de los datos de carácter personal.”. Así, podemos apreciar que el titular de los datos de carácter personal siempre va a ser una persona natural, mientras que el responsable del tratamiento de esos datos, puede ser tanto una persona natural como una persona jurídica.

En lo que toca a los mecanismos de protección que establece la ley, el artículo 4 establece que el tratamiento de datos personales solo se va a poder efectuar cuando la misma ley u otra disposición legal lo autoricen o cuando el titular consienta expresamente en ello. También establece que la persona que autoriza debe ser debidamente informada respecto del propósito del tratamiento de datos, y posible comunicación al público. Es decir, la ley establece un consentimiento anterior al tratamiento de datos.

El artículo 6 entrega la regla de eliminación y modificación de los datos, señalando “Los datos personales deberán ser eliminados o cancelados cuando su almacenamiento carezca de fundamento legal o cuando hayan caducado. Han de ser modificados cuando sean erróneos, inexactos, equívocos o incompletos. Se bloquearán los datos personales cuya exactitud no pueda ser establecida o cuya vigencia sea dudosa y respecto de los cuales no corresponda la cancelación. El responsable del banco de datos personales procederá a la eliminación, modificación o bloqueo de los datos, en su caso, sin necesidad de requerimiento del titular.”

El artículo 11 de esta ley consagra la responsabilidad de quien esté a cargo de la base de datos de cuidar de ellos “con la debida diligencia, haciéndose responsable de los daños”.

El título II de la ley establece los derechos de los titulares de los datos. Señala, en el artículo 12, que toda persona puede exigir al responsable de la base de datos información sobre sus datos personales, el uso que se les da e información sobre el destinatario. Además, cuando los datos sean erróneos, inexactos, equívocos o incompletos, el titular podrá solicitar su modificación, y cuando el almacenamiento carezca de fundamento legal o los datos sean caducos, podrá solicitar su eliminación. El artículo señala, asimismo, que estos procedimientos deben ser gratuitos, y que el titular puede pedir una copia del registro alterado, debiendo ejercerse este derecho personalmente.

El artículo 13 establece que “El derecho de las personas a la información, modificación, cancelación o bloqueo de sus datos personales no puede ser limitado por medio de ningún acto o convención”.

El artículo 15 señala una limitación respecto a la solicitud de modificación o eliminación de datos personales. Dispone “No obstante lo dispuesto en este Título, no podrá solicitarse información, modificación, cancelación o bloqueo de datos personales cuando ello impida o

entorpezca el debido cumplimiento de las funciones fiscalizadoras del organismo público requerido, o afecte la reserva o secreto establecidos en disposiciones legales o reglamentarias, la seguridad de la Nación o el interés nacional. Tampoco podrá pedirse la modificación, cancelación o bloqueo de datos personales almacenados por mandato legal, fuera de los casos contemplados en la ley respectiva.” Este artículo lo podemos concordar con el artículo 20 del mismo cuerpo legal, que establece que “El tratamiento de datos personales por parte de un organismo público sólo podrá efectuarse respecto de las materias de su competencia y con sujeción a las reglas precedentes. En esas condiciones, no necesitará el consentimiento del titular.” Así, podemos afirmar que el consentimiento y la integridad de los datos personales se ven disminuidas frente a las facultades de los organismos públicos, siempre que el tratamiento de esos datos se realice en el ámbito de su competencia.

El artículo 16 de la ley establece un procedimiento para el caso en que el responsable del banco de datos no se pronuncie sobre la solicitud que realice el titular, destinada a modificar o eliminar sus datos personales, o la deniegue por causas distintas a las establecidas en el artículo 15. Este procedimiento es denominado por la doctrina recurso de *habeas data*. Se tramita, en primera instancia, ante el juez civil del domicilio del responsable del banco de datos, y la sentencia definitiva es apelable, siendo este último fallo no susceptible del recurso de casación. Si la causal para denegar la modificación o eliminación de los datos fuere la seguridad nacional, el procedimiento debe tramitarse ante la Corte Suprema, la que, si lo estima conveniente podrá ordenar traer los autos en relación y oír a los abogados de las partes. Finaliza el artículo 16 señalando que “En caso de acogerse la reclamación, la misma sentencia fijará un plazo prudencial para dar cumplimiento a lo resuelto y podrá aplicar una multa de una a diez unidades tributarias mensuales, o de diez a cincuenta unidades tributarias mensuales si se tratare de una infracción a lo dispuesto en los artículos 17 y 18. La falta de entrega oportuna de la información o el retardo en efectuar la modificación, en la forma que decrete el Tribunal, serán castigados con multa de dos a cincuenta unidades tributarias mensuales y, si el responsable del banco de datos requerido fuere un organismo público, el tribunal podrá sancionar al jefe del Servicio con la suspensión de su cargo, por un lapso de cinco a quince días.”

Por último, el artículo 23 consagra la responsabilidad por las infracciones a la ley, disponiendo que “La persona natural o jurídica privada o el organismo público responsable del banco de

datos personales deberá indemnizar el daño patrimonial y moral que causare por el tratamiento indebido de los datos, sin perjuicio de proceder a eliminar, modificar o bloquear los datos de acuerdo a lo requerido por el titular o, en su caso, lo ordenado por el tribunal. La acción consiguiente podrá interponerse conjuntamente con la reclamación destinada a establecer la infracción, sin perjuicio de lo establecido en el artículo 173 del Código de Procedimiento Civil. En todo caso, las infracciones no contempladas en los artículos 16 y 19, incluida la indemnización de los perjuicios, se sujetarán al procedimiento sumario. El juez tomará todas las providencias que estime convenientes para hacer efectiva la protección de los derechos que esta ley establece. La prueba se apreciará en conciencia por el juez. El monto de la indemnización será establecido prudencialmente por el juez, considerando las circunstancias del caso y la gravedad de los hechos.”

Tenemos, entonces, que la Ley 19.628 consagra un procedimiento para asegurar la adecuada protección de los datos personales, atribuye responsabilidad y establece sanciones. Como se usan términos amplios y la ley no distingue, consideramos que sería perfectamente aplicable este procedimiento a un caso de incorrecto tratamiento de datos personales por parte de un software, en el sentido de ser un sistema de tratamiento automatizado de información. Con todo, hay quienes señalan que este procedimiento no es adecuado, y que falta avanzar todavía más en la materia. Romina Garrido sostiene “Como se aprecia, la ley contempla en su texto muchas garantías, pero sin herramientas procesales efectivas para hacer frente a posibles vulneraciones a la norma, y es este su mayor problema. El procedimiento que fija la ley es ante la justicia civil, y hasta hoy en día, quince años desde su vigencia no ha tenido una aplicación práctica, masiva y concreta. Las legislaciones actuales contemplan entes especializados para hacer frente a las transgresiones a este derecho, entes que además fiscalizan y controlan a quienes tratan datos. El procedimiento de la ley chilena, no se condice con la realidad tecnológica actual, y más aún carga el peso de la prueba en el afectado”<sup>47</sup>

Es interesante la discusión sobre la adecuada protección que entre la ley actual a los datos personales, pero excede los objetivos de este trabajo, y, por tanto, nos quedamos con que la Ley 19.628 establece un procedimiento y sanciones para cautelar el adecuado tratamiento de los datos personales.

---

<sup>47</sup> (Garrido, 2013)

## Conclusión del capítulo

En este capítulo hemos revisado las normas de la Constitución y de las leyes que más podrían tener relación con la modificación de un software. Sin embargo, nos hemos encontrado con que ninguna aborda esta cuestión de forma específica. La Constitución asegura el derecho de propiedad tanto a los autores del software sobre su obra, como en general a todas las personas sobre sus bienes. Por otro lado, también asegura la protección de la vida privada, la protección de los datos personales y la inviolabilidad de toda forma de comunicación privada. Además, entrega una acción para cautelar todos estos derechos, la acción de protección. En cuanto a las normas de rango legal, la que más se acerca, por su especialidad, es la Ley de Propiedad Intelectual, que aun así, solo trata ciertos aspectos, como los requisitos que esas modificaciones deben tener para no vulnerar la propiedad intelectual del software. La Ley de Protección al Consumidor no contempla la posibilidad de que sea el usuario final el que subsane los defectos del producto -el software- modificándolo por sí mismo, solo da reglas generales para que el consumidor o usuario le reclame al proveedor por los defectos. La Ley que Tipifica Figuras Penales Relativas a la Informática, tampoco resulta aplicable al caso de una modificación de software que busca proteger o restaurar derechos, por su amplitud y poca precisión conceptual, y por la exigencia de malicia o antijuridicidad que establece. Por último, la Ley de Protección de la Vida Privada contempla sanciones y un procedimiento, pero solo para el caso en que se deniegue injustificadamente la solicitud de modificar o eliminar datos personales de una base de datos, y con algunas limitaciones. Es decir, el ámbito de aplicación es más bien restringido.

## Conclusiones generales

Al finalizar el trabajo, estamos en condiciones de responder a las preguntas planteadas en la introducción. Comencemos entonces con las preguntas específicas. La primera pregunta a responder era ¿De qué hablamos cuando nos referimos a alterar o modificar software? De acuerdo a lo que revisamos en el primer capítulo, un software es un conjunto de instrucciones que puede ser expresado tanto en un código fuente, legible por una persona humana, como en código máquina, un lenguaje entendible solo por los dispositivos. El software sirve de soporte lógico al hardware donde está instalado, dirigiendo su funcionamiento y estableciendo las funciones que se van a poder hacer con los dispositivos. También, existen diversos tipos de

software, cada uno con sus características. Al parecer, existen programas que entregan más libertad al usuario, como los que podemos denominar software libre. En cambio, un software propietario no le entrega tanta libertad al usuario.

En cuanto a la modificación de software, quedó claro que no se trata de un concepto unívoco, sino que hay varias formas de modificar un software, como cambiar el código fuente o usar técnicas de ingeniería inversa en caso de no conocerlo. Vimos que, en general, resulta más sencillo modificar un software libre que un software propietario, por el mayor acceso al código fuente y a la documentación asociada. Al final, nos quedamos con la idea de que modificar un software implica cambiar el funcionamiento previsto originalmente por los desarrolladores del mismo.

Otra pregunta planteada en la introducción fue ¿Qué derechos se adquieren con una licencia de software? En el segundo capítulo vimos que la licencia de software es un contrato entre el desarrollador o quien posea los derechos de autor sobre el software y quien lo adquiere o usa, donde se establecen las condiciones de uso de aquel software. De acuerdo a las definiciones que revisamos, estas condiciones implican autorizaciones para el usuario, es decir, qué puede y qué no puede hacer con el software. Además, llegamos a la conclusión de que existen algunas licencias más permisivas que otras, como pudimos desprender del análisis de algunos contratos de licencia paradigmáticos, como la licencia de Microsoft Windows, la de Apple iOS o la licencia GPLv3. Así, las licencias propietarias conceden solo el derecho de uso, y reservan el resto de los derechos para el licenciante. Además, prohíben expresamente las modificaciones no autorizadas. Las licencias libres, por el contrario, permiten el uso y otros derechos, como modificar y hacer obras derivadas. Sin embargo, en las licencias propietarias hay excepciones a las prohibiciones que se establecen, como la legislación vigente en el territorio del licenciatario.

Respecto a la pregunta ¿Qué derechos, tanto constitucionales como legales, se pueden ver afectados por la imposibilidad de modificación?, conforme al análisis que hicimos, podemos decir que, principalmente los derechos afectados son el de propiedad, no sobre el software, cuya propiedad pertenece a quien tiene los derechos de autor, sino sobre el hardware, ya que las condiciones de uso del software pueden llegar a bloquear ciertas características que puede llegar a ofrecer el hardware. También, otro derecho que se puede ver afectado es el derecho a la privacidad, o más específicamente y usando el lenguaje de la Constitución, el derecho a la vida

privada, a la protección de datos personales, y a la inviolabilidad de las comunicaciones privadas, especialmente si consideramos que buena parte del software actual se conecta a Internet y transmite datos que revelan el uso que le dan los usuarios. En relación con la pregunta ¿Qué derechos se ven protegidos por la imposibilidad de modificación?, podemos decir que es una suerte de contrapartida de la pregunta anterior. Aquí, según vimos en el capítulo segundo el derecho que se busca proteger es el derecho de autor del propietario del software, que muchas veces va a ser el desarrollador, y a su vez, el licenciante en un contrato de licencia de software.

Respondidas las preguntas más específicas, vamos a la pregunta general y a la hipótesis: ¿Qué reconocimiento legal hay en Chile de la modificación o alteración de software para garantizar la libertad y la propiedad sobre el hardware que tienen los usuarios?

Según revisamos en el tercer capítulo, en Chile es posible afirmar que existe un reconocimiento legal de las modificaciones de un software, que se expresa en la existencia de excepciones al derecho de autor en la Ley de Propiedad Intelectual. Sin embargo, en ningún cuerpo normativo se establece explícitamente la posibilidad de modificar un software para proteger o cautelar posibles derechos de los usuarios. Solo existen algunos procedimientos, como el contemplado en la Ley de Protección al Consumidor para reclamar por defectos del software como producto, o el procedimiento contemplado en la Ley de Protección de Datos para reclamar por la negativa injustificada a modificar o eliminar datos personales. Así, con el marco legal actual, a lo más se podría intentar obtener una indemnización de parte del proveedor del software, pero no se contempla un camino para que el mismo usuario pueda subsanar posibles deficiencias del software, como realizar una modificación al funcionamiento.

Entonces, nuestra hipótesis “Existe un reconocimiento parcial de la posibilidad de afectar derechos de las personas, que se expresa, por ejemplo, en las excepciones de la Ley de Propiedad Intelectual. Sin embargo, es parcial porque no se explicita como una afectación de derechos. En cuanto a la posibilidad de incumplir las cláusulas limitativas de derechos, tal posibilidad no se contempla en el ordenamiento jurídico actual”, se cumple parcialmente, ya que existe ese reconocimiento parcial, que es un reconocimiento de la posibilidad de modificación y de excepción a los derechos de propiedad (o derechos de autor) del titular. E incluso, también se reconoce una afectación de los derechos de las personas, entendiendo este último concepto como persona natural, en la Ley de Protección de Datos Personales y se establece un procedimiento

para subsanarla, pero bajo causales y circunstancias muy específicas. En cuanto a la no explicitación de la afectación de derechos, efectivamente se cumple, porque como ya dijimos, ningún cuerpo legal contempla expresamente que se pueda modificar un software “porque se afectan los derechos de los usuarios”. Y finalmente, respecto a la posibilidad de incumplir cláusulas limitativas de derechos, aquí la hipótesis queda descartada, dado que el marco legal vigente sí entrega tal posibilidad. Eso sí, de forma excepcional y con requisitos. Así, las excepciones consagradas en la Ley de Propiedad Intelectual, permiten saltarse las cláusulas de las licencias en donde se prohíbe, por ejemplo, la ingeniería inversa, pero solo para ciertos fines y especialmente, que la modificación que se haga sea sin fines de lucro y para uso personal.

En suma, de esta investigación se puede desprender que, bajo la regulación actual no se contempla la posibilidad de que un usuario persona natural modifique por sí mismo un software cuando este afecta sus derechos, pero sí tiene distintas vías de defensa frente a esa vulneración, más o menos específicas, cuya nota común es que son todas *ex post*, es decir, se pueden ejercer una vez que la vulneración de derechos ya se haya producido, como la acción de protección, el recurso de *habeas data* o el procedimiento contemplado en la Ley del Consumidor por defectos de un producto. Así, al ordenamiento jurídico chileno todavía le falta avanzar más en la especificidad de conductas, requisitos, permisiones y prohibiciones, para poder entregar un marco más claro acerca de qué podemos y qué no podemos hacer los usuarios con un software, y especialmente, en la posibilidad de tomar medidas preventivas, con anterioridad a la producción de un daño, algo de suma importancia en medio de la revolución tecnológica en que estamos inmersos.

## Bibliografía

- Bain, M., Gallego Rodríguez, M., Martínez Ribas, M., & Sanjuán, J. *Aspectos Legales y de explotación del software libre. Parte I*. Universitat Oberta de Catalunya. Recuperado a partir de <http://libros.metabiblioteca.org/bitstream/001/311/8/84-9788-120-6.pdf>
- Carrasco Blanc, H. (2000). *Contratación electrónica y contratos informáticos* (1ra ed.). Santiago, Chile: La Ley.



- *Categorías de software libre y software que no es libre - Proyecto GNU -Free Software Foundation. Gnu.org.* Recuperado 11 de julio de 2017, a partir de <http://www.gnu.org/philosophy/categories.es.html>
- Cipresso, T. (2009). *Software reverse engineering education* (Master). San Jose State University.
- *Contrato de Licencia de Software iOS de Apple* (pp. 92-103). Recuperado a partir de <http://images.apple.com/legal/sla/docs/iOS11.pdf>
- Delfín Vásquez, J., & Jiménez López, E. *Ingeniería Inversa: Metodología y aplicaciones*. Instituto Superior Tecnológico de Cajeme. Recuperado a partir de <http://www.itesca.edu.mx/investigacion/foro/carp%20ponencias/28.pdf>
- Eilam, E. (2005). *Reversing*. Indianapolis: Wiley Publishing.
- El ciberataque con el virus WannaCry se extiende a nivel mundial. (2017). Recuperado de <http://www.elmundo.es/tecnologia/2017/05/12/5915e99646163fd8228b4578.html>
- Fuentes Pinzón, F. (2003). Contratación para la elaboración del software individualizado. Estudio comparado de la legislación Iberoamericana. *Revista de Ciencias Jurídicas Universidad de Costa Rica*, (102). Recuperado a partir de <https://revistas.ucr.ac.cr/index.php/juridicas/article/view/13373/12639>
- Garrido, R. (2013). El Habeas data y la ley de protección de datos en Chile. Serie Bibliotecología Y Gestión De Información, (83). Recuperado a partir de <https://sitios.vtte.utem.cl/seriebibliotecologia/wp-content/uploads/sites/11/2018/05/Serie-N%C2%B0-83-Junio-2013El-Habeas-data-y-la-ley-de-proteccio%CC%8In-de-datos-en-Chile.pdf>
- Gil, P. (2017). What Is 'SaaS' (Software as a Service)?. *LifeWire*. Recuperado a partir de <https://www.lifewire.com/what-is-saas-software-2483600>
- Gómez Labrador, R. (2005). *Tipos de licencias de software*. Recuperado a partir de <http://www.informatica.us.es/~ramon/articulos/LicenciasSoftware.pdf>
- Joyanes Aguilar, L. (2008). *Fundamentos de programación*. Madrid [etc.]: McGraw-Hill.
- *Licencia Apache 3.0 - Traducción al español. Moo.com*. Recuperado 4 October 2017, a partir de <https://www.moo.com/es/about/fonts/apache-licence.html>

- Madatli, K. (2015). Software License Agreements: Main Types, Legal Aspects, and Regulations. *Baku State University Law Review*, 2(1)
- Martín Quetglás, G., Toledo Lobo, F., & Cerverón Lleó, V. (1995). *Fundamentos de informática y programación*. Valencia: V.J.
- Monroy, J. (2012). Cuestiones jurídicas entorno a los contratos de desarrollo y licencia de software. *Revista La Propiedad Inmaterial*, (16). Recuperado a partir de <http://revistas.uexternado.edu.co/index.php/propin/article/view/3267>
- Moscoso Escobar, R. (2014). La Ley 19.223 en general y el delito de hacking en particular. *Revista Chilena De Derecho Y Tecnología*, 3(1). Recuperado a partir de <http://dx.doi.org/10.5354/0719-2584.2014.32220>
- Palazuelos, F. (2017). *Cronología de WannaCry: así se expandió el virus que paralizó a medio mundo*. *Genbeta.com*. Recuperado el 20 de noviembre de 2017, de <https://www.genbeta.com/a-fondo/cronologia-de-wannacry-asi-se-expandio-el-virus-que-paralizo-a-medio-mundo>
- Segura Riveiro, F. (2006). Algunas cuestiones actuales en torno a la oferta y la aceptación. *Cuadernos De Análisis Jurídicos. Universidad Diego Portales*, (3), 23-60. Recuperado a partir de [http://derecho.udp.cl/wp-content/uploads/2016/08/cuestiones\\_actuales\\_oferta\\_aceptacion\\_franciscoSegura.pdf](http://derecho.udp.cl/wp-content/uploads/2016/08/cuestiones_actuales_oferta_aceptacion_franciscoSegura.pdf)
- Sommerville, I., Campos Olguín, V., & Fuenlabrada Velázquez, S. (2011). *Ingeniería de software*. Madrid: Pearson Educación de México.
- *Términos de licencia del software de Microsoft - Windows 10*. (2016). *Microsoft.com*. Recuperado 4 October 2017, a partir de [https://www.microsoft.com/en-us/Useterms/Retail/Windows/10/UseTerms\\_Retail\\_Windows\\_10\\_Spanish.htm](https://www.microsoft.com/en-us/Useterms/Retail/Windows/10/UseTerms_Retail_Windows_10_Spanish.htm)
- *The 'Stagefright' exploit: What you need to know*. (2017). *Android Central*. Recuperado el 20 de noviembre de 2017, de <https://www.androidcentral.com/stagefright>
- *Traducción de la licencia GPLv3 al español*. *Lslspanish.github.io*. Recuperado el 4 de octubre 2017, a partir de [https://lslspanish.github.io/translation\\_GPLv3\\_to\\_spanish/](https://lslspanish.github.io/translation_GPLv3_to_spanish/)
- *What Is Proprietary Software*. *IGI Global*. Recuperado el 11 de julio de 2017, a partir de <https://www.igi-global.com/dictionary/proprietary-software/23877>