

¿A quién sirve realmente el servidor?

Richard Stallman

Lunes 7 de junio de 2010.

Con el software libre nosotros, los usuarios, recuperamos el control sobre nuestros datos. El software propietario sigue existiendo, pero podemos eliminarlo de nuestra vida, y muchos ya lo hemos hecho. Sin embargo, ahora se cierne sobre nuestros datos una nueva amenaza. El software como Servicio. Por el bien de nuestra libertad, también debemos que rechazarlo.

En Internet, el software propietario no es el único modo de perder la libertad. El software como Servicio es otra forma de entregar a alguien poder sobre nuestros datos.

Antecedentes: Cómo nos arrebató la libertad el software propietario

La tecnología digital puede darnos libertad; pero también puede arrebatárnosla. La primera amenaza para el control que tenemos sobre nuestros datos vino del software propietario: los programas que los usuarios no pueden controlar porque los controla su dueño (una empresa como Apple o Microsoft). El titular suele aprovecharse de este poder desigual insertando elementos maliciosos como programas espía, puertas traseras («back doors») o Gestión de Restricciones Digitales (DRM, Digital Restrictions Management) (a la que en su propaganda denominan «Digital Rights Management», Gestión de Derechos Digitales).

La solución para este problema es desarrollar software libre y rechazar el software propietario. El software libre significa que nosotros, como usuarios, gozamos de cuatro libertades esenciales: (1) manejar el programa como deseemos, (2) analizar y modificar el código fuente para que haga lo que nosotros fuéramos, (3) redistribuir copias fieles del programa original y (4) redistribuir copias de las versiones modificadas. (Véase la definición de software libre.)

Con el software libre nosotros, los usuarios, recuperamos el control sobre nuestros datos. El software propietario sigue existiendo, pero podemos eliminarlo de nuestra vida, y muchos ya lo hemos hecho. Sin embargo, ahora se cierne sobre nuestros datos una nueva amenaza. El software como Servicio. Por el bien de nuestra libertad, también debemos que rechazarlo.

Cómo nos arrebató la libertad el software como Servicio

El software como Servicio (SaaS, Software as a Service) significa que alguien configura un servidor de una red con programas que llevan a cabo determinadas tareas informáticas (gestionar hojas de cálculo, procesar textos, traducir un texto a otra lengua, etc.) y luego invita a los usuarios a realizar esas labores en ese servidor. Los usuarios envían la información al servidor, que ejecuta las operaciones con los datos suministrados y les devuelve directamente a ellos los resultados o el fruto de las tareas realizadas.

Estos servidores despojan del control a los usuarios de un modo aún más inevitable que el software propietario. Con el software propietario los usuarios suelen recibir un programa ejecutable, pero no el código fuente. Eso dificulta a quienes saben de programación estudiar el código con el que está hecho, de modo que es difícil determinar qué hace en realidad el programa, y complicado modificarlo.

Con el SaaS los usuarios ni siquiera reciben el programa ejecutable: está alojado en el servidor, donde los usuarios no pueden verlo ni acceder a él. Por tanto, les resulta imposible determinar qué hace en realidad, y muy difícil cambiarlo.

Además, el SaaS desemboca automáticamente en consecuencias perniciosas equivalentes a las prestaciones maliciosas de determinado software propietario. Por ejemplo, algunos programas propietarios son programas espía: envían información sobre la actividad informática de los usuarios. Microsoft Windows envía información a Microsoft sobre las actividades de los usuarios. Windows Media Player y RealPlayer informan de lo que los usuarios ven o escuchan.

A diferencia del software propietario, el SaaS no requiere un código oculto para obtener los datos del usuario. Por el contrario, son los usuarios quienes, para utilizarlo, deben remitir la información al servidor. El resultado es idéntico al de los programas espía: el gestor del servidor obtiene la información. Y la obtiene sin ningún esfuerzo adicional, por la propia naturaleza del SaaS.

Algunos programas propietarios maltratan a los usuarios mediante un sistema de control remoto. Por ejemplo, Windows tiene una puerta trasera con la que Microsoft puede modificar a la fuerza determinado software del equipo. El lector de libros digitales de Amazon, Kindle (cuyo nombre, que en inglés significa «prender» o «encender», hace pensar que está concebido para quemar los libros de las personas) tiene una puerta trasera orwelliana que Amazon utilizó en 2009 para borrar por control remoto las copias para Kindle de los libros de Orwell 1984 y Rebelión en la granja que los usuarios habían adquirido en Amazon.

El SaaS otorga intrínsecamente al gestor del servidor poder para cambiar el software que se utiliza o manipular los datos del usuario. Una vez más, no es necesario ningún código especial para hacerlo.

Por tanto, el SaaS equivale al programa espía por antonomasia, es una puerta trasera abierta de par en par y otorga injustamente al gestor del servidor poder sobre el usuario. No podemos aceptarlo.

Desenredar el problema del SaaS del problema del software propietario

El SaaS y el software propietario desembocan en situaciones perniciosas de similar naturaleza, pero los mecanismos causales son distintos. En el caso del software propietario, el origen reside en que disponemos de una copia del programa que es difícil o ilegal alterar, y la utilizamos. En el del SaaS, la causa es que utilizamos una copia de un programa que no tenemos.

Ambos asuntos suelen confundirse, y no sólo de forma involuntaria. Quienes desarrollan Internet utilizan el término vago de «aplicación web» para referirse tanto al software de un servidor como a los programas que se ejecutan en su nuestro equipo, en nuestro navegador. Algunas páginas web instalan temporalmente en nuestro navegador sin informarnos programas de JavaScript nada triviales o incluso muy voluminosos. Cuando estos programas JavaScript no son libres, son tan nocivos como cualquier otro software no gratuito. Sin embargo, aquí topamos con el problema del propio software del servidor.

Muchos defensores del software libre presuponen que el problema del SaaS se resolverá desarrollando software libre para los servidores. Por el bien del gestor del servidor, sería mejor que sus programas fueran libres; si son propietarios, sus titulares tendrán poder sobre el servidor. Es injusto para el gestor y no nos ayuda en absoluto. Pero si los programas del servidor son libres, no nos libramos de los efectos del SaaS como usuarios del servidor, pues otorgan libertad al gestor, pero no a nosotros.

Liberar el código fuente del software del servidor beneficia a la comunidad: los usuarios convenientemente entrenados pueden configurar servidores similares, quizá modificando el software. Pero ninguno de esos servidores nos dará el control sobre las operaciones que hacemos en él, a menos que sea nuestro. El resto sería todo SaaS. El SaaS siempre nos somete al poder del gestor del servidor, y la única solución es... ¡no utilizar el SaaS! No utilicemos el servidor de otro para hacer nuestras tareas informáticas suministrándole información.

Distinguir el SaaS de otro tipo de Servicios en red

¿Condenar el SaaS significa rechazar también todos los servidores de red? En absoluto. La mayoría de los servidores no plantean este problema, pues las tareas que realizamos con ellos no afectan a la información que suministramos salvo en un sentido muy trivial.

El propósito original de los servidores web no era realizar labores informáticas para nosotros, sino hacer pública información con el fin de que accediéramos a ella. Eso es lo que hacen incluso hoy día casi todos los sitios web sin plantear el problema del SaaS, ya que acceder a la información hecha pública por otro no consiste en realizar ningún tipo de operación informática. Tampoco lo es hacer públicos nuestros materiales a través de un sitio de blogs o un servicio de microblogs como Twitter. Eso mismo puede decirse para las comunicaciones no privadas, como las de los grupos. Las redes sociales pueden incurrir en el SaaS; sin embargo, su razón de ser es precisamente un mecanismo de comunicación y publicación, no de SaaS. Si utilizamos el servicio para retocar aspectos de edición menores de aquello que vamos a comunicar, el problema no es importante.

Los servicios como los motores de búsqueda recogen información de toda la red y nos permiten examinarla. Buscar entre la recopilación de información que nos ofrece no es realizar labores informáticas propiamente dichas en el sentido habitual (no hemos sido nosotros quienes hemos suministrado la información), de modo que utilizar un servicio de esta naturaleza para buscar en la web no es SaaS. (Sin embargo, utilizar el motor de búsqueda de alguien para implantar una prestación de búsqueda en nuestro sitio web sí es SaaS.)

El comercio electrónico no es SaaS, pues la tarea informática no es exclusivamente nuestra; más bien, la hacemos en colaboración con un tercero. De modo que no hay razón particular por la que esperemos controlar la operación en exclusiva. Lo peliagudo del comercio electrónico es si confiamos a ese tercero nuestro dinero y nuestros datos personales.

Utilizar servidores para un proyecto conjunto no es SaaS, ya que la tarea informática que realizamos con ello no es nuestra personalmente. Por ejemplo, si editamos páginas en Wikipedia no estamos haciendo una tarea informática propia ni ofreciendo información personal; más bien, estamos colaborando con la labor informática de Wikipedia.

Wikipedia controla sus servidores, pero los grupos pueden encontrarse con el problema del SaaS si realizan sus actividades en servidores ajenos. Por fortuna, desarrollar sitios para alojar a otros como Savannah o Sourceforge no plantea el problema del SaaS, ya que lo que los grupos hacen allí es en esencia exponer publicaciones y emitir comunicaciones públicas, no sus archivos informáticos privados.

Los juegos entre varios jugadores son una actividad de grupo que se desarrollan en el servidor de un tercero, lo que los convierte en SaaS. Pero cuando la información que se expone es únicamente el estado del juego y la puntuación, lo máximo que un gestor malintencionado podría hacer es favorecer a algún jugador. También podemos ignorar ese riesgo, ya que parece improbable y hay muy poco en juego. Por el contrario, cuando el juego se convierte en algo más que un juego, la cosa cambia.

¿Qué servicios online son SaaS? Google Docs es un ejemplo claro. Su actividad esencial es editar textos, y Google anima a las personas a que lo utilicen para elaborar sus documentos personales; eso es SaaS. Ofrece la ventaja añadida de se puede trabajar con otro en ello, pero añadir participantes no altera el hecho de que editar en el servidor es SaaS. (Además, Google Docs es inaceptable porque instala en los navegadores de los usuarios un voluminoso programa de JavaScript no libre.) Si utilizar un servicio para comunicarse o cooperar exige realizar también con él partes sustanciales de nuestra propia tarea informática, ese tipo de actividad informática es SaaS, aun cuando la mera comunicación no lo sea.

Algunos sitios web ofrecen servicios múltiples, y aunque alguno no sea SaaS, tal vez otro sí. Por ejemplo, el servicio principal que presta Facebook es el de redes sociales, y eso no es SaaS; sin embargo, sirve de soporte para aplicaciones de terceros, algunas de las cuales pueden ser SaaS. El principal servicio de Flickr es distribuir fotografías, que no es SaaS, pero también ofrece prestaciones para editar fotos, que sí lo es.

Algunos sitios web cuyo principal servicio es la publicación de documentos y la comunicación lo amplían con la «gestión de contactos»: almacenar y registrar las personas con quienes tenemos relación. Enviar correo electrónico a esas personas no es SaaS, pero guardar un registro de nuestras conversaciones con ellos, si interesan, sí es SaaS.

Si un servicio no es SaaS no quiere decir que esté bien. Hay otras cosas perniciosas que puede hacer un servicio. Por ejemplo, Facebook distribuye imágenes de video mediante Flash, lo que ejerce presión sobre los usuarios para que utilicen un software no libre, y les transmite una impresión falsa de privacidad. Estos también son asuntos importantes, pero el objeto de este artículo es la cuestión del SaaS.

La industria de las tecnologías de la información no favorece que los usuarios tengan en cuenta estas distinciones. Para eso sirve la «computación en nube», un concepto de moda. El término es tan nebuloso que podría referirse casi a cualquier utilización de Internet. El término incluye el SaaS y casi todo lo demás, de manera que no sirve más que para prestarse a generalizaciones inútiles.

El significado real de «computación en nube» es propiciar que nos despreocupemos de nuestros datos y tareas informáticas. Dice: «No hagas preguntas, simplemente confía en cualquier cosa sin vacilar. No te preocupes por quién controla tus tareas ni quien almacena tus datos. No compruebes si nuestro servicio esconde un anzuelo antes de haberlo mordido». En otras palabras: «piensa como un bobo». Por eso prefiero evitar ese concepto.

Abordar el problema del SaaS

Sólo una pequeña parte del conjunto de sitios web hace SaaS; la mayoría no plantean este problema. Pero, ¿qué debemos hacer con los que sí lo plantean?

Para el simple caso en el que estemos haciendo un tratamiento de información propia y por nosotros mismos, la solución es sencilla: utilicemos una copia de una aplicación de software libre. Editemos nuestros textos con nuestra propia copia de un programa de edición libre como GNU Emacs, o con un procesador de textos libre. Editemos nuestras fotografías con nuestra copia de un software libre como GIMP.

¿Pero qué pasa cuando trabajamos en colaboración con otras personas? Hoy día tal vez resulte difícil hacerlo sin utilizar un servidor. Si utilizamos uno, no confiemos en un servidor gestionado por una empresa privada. Un mero contrato de cliente no es protección suficiente, a menos que podamos detectar una infracción y denunciarla de verdad, para lo que la empresa seguramente redacta los contratos de forma que les permitan cometer un amplio abanico de abusos. La policía puede citar judicialmente a la empresa para que le suministre datos sobre nosotros con menos fundamento del que nos exige a nosotros para citarlos a ellos como acusados; eso suponiendo que la empresa no trabaje ya de forma voluntaria para ellos, como hicieron las compañías telefónicas estadounidenses que

pincharon los teléfonos de sus clientes cuando se lo pidió Bush. Si tenemos que utilizar un servidor, usemos uno cuyos gestores nos ofrezcan unos motivos para confiar que trasciendan la mera relación comercial.

Sin embargo, a más largo plazo podemos crear alternativas a la utilización de servidores. Por ejemplo, podemos crear un programa P2P mediante el cual los colaboradores puedan compartir datos encriptados. La comunidad del software libre debería desarrollar distribuciones sustitutivas P2P para «aplicaciones web» importantes. Quizá sea prudente lanzarlas bajo la Licencia Pública General de Affero, derivada de GNU, puesto que son candidatos adecuados para que alguien acabe convirtiéndolos en programas alojados en servidores. El proyecto GNU busca voluntarios para trabajar en este tipo de sustitutivos. También invitamos a otros proyectos de software libre a tener en cuenta este aspecto en su diseño.

Mientras tanto, si una empresa nos invita a utilizar su servidor para almacenar datos o realizar tareas informáticas, no cedamos; no utilicemos el SaaS. A menos que vayamos a utilizarlos con nuestro propio servidor, no compremos ni instalemos «clientes livianos», pues no son más que unos ordenadores tan endebletes que nos obligan a hacer el trabajo de verdad en un servidor. Utilicemos un ordenador de verdad y almacenemos en él la información. Por el bien de nuestra libertad, trabajemos con nuestra propia copia de un programa libre,.

(publicado originalmente en Boston Review)

Traducido para Rebelión por **Ricardo García Pérez**

Fuente: GNU.org y Rebelión