汉坤法律评述
2022 年 11 月 22 日

HANKUN
汉 坤 律 师 事 务 所
Han Kun Law Offices

北京 | 上海 | 深圳 | 海口 | 武汉 | 香港

“开源合规”系列之三 — SaaS 应用场景下的开源合规

作者：段志超 | 鲁学振 | 杨可依[1]

随着互联网科技的高速发展，软件服务不再局限于一次性地将软件部署于客户本地的传统交付模式，SaaS 软件服务在日常生活中随处可见。但基于传统软件交付服务产生的开源协议，如 GPL、LGPL 协议，无法也没有预料 SaaS 的应用场景，其具体条款也无法约束 SaaS 厂商对开源软件的使用。为解决此类问题，GNU Affero General Public License（AGPL）和 Server Side Public License（SSPL）等特殊协议便应运而生了。

本文将结合 AGPL 和 SSPL 协议的具体内容，讨论 SaaS 应用场景下的开源风险，并给出一定的合规建议。

一、传染性开源许可证的“演进”

最典型的带有开源风险的协议莫过于 GNU 计划的 GPL 和 LGPL 系列开源许可证，此类协议触发传染性的条件是分发[2]。而在 SaaS 服务场景中，因为用户没有直接获得软件本身，通常不被认定构成分发，因而 GPL 类开源软件的传染性对 SaaS 场景失效。

以 GPL v2 为例，GPL v2 的传染性通常被理解为包括“纵向传染”和“横向传染”。

具体地，根据该协议第 2（b）条的约定，“纵向传染”主要指 GPL 类程序会“传染”自身的修改版本（Modifications），也就是说，在 GPL 程序上进行修改所获得的修改版本也将用 GPL 进行开源：

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:（你可以修改你的程式副本的任意部分，以构成本程式的派生作品，并在满足上述条款及以下三点要求的前提下复制和分发该修改版：）

b）You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this

[1] 实习生梁杰对本文的写作亦有贡献。

[2] https://www.gnu.org/licenses/old-licenses/gpl-2.0.html。

License.（你必须使你分发或发布的作品，部分或全部包含本程式或其派生作品，允许第三方在本协议约束下使用，并不得就授权收费。）

—— *GPL v2 Article 2（b）*[3]

根据 GPL v2 许可证第 0 条和第 1 条的约定，“横向传染”主要指 GPL 类程序会“传染”包含其全部或部分源代码或修改版的作品，也就是说，只要一个作品为 GPL 类程序的“派生作品”，则该作品在复制和分发时，必须也采用相同的 GPL 类许可证进行开源：

0. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".)（下文中所说的“程式”指任何程序或作品，而“程式的派生作品”指该程式或者版权法认定的派生作品，即全部或部分包含了该程式的作品，无论是原样包含或做了修改，乃至翻译成了其他语言（后文中“修改”涵盖翻译）。

—— *GPL v2 Article 0*[4]

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.（你必须使你分发或发布的作品，部分或全部包含本程式或其派生作品，允许第三方在本协议约束下使用，并不得就授权收费。）

—— *GPL v2 Article 1*[5]

简而言之，任何人都可以以提供技术服务为目的，运行私有修改的 GPL 许可下的程序，只要不发布软件，则不需要公开源代码。但由于在 SaaS 服务场景中，用户通过云端访问软件，软件商不存在将软件源代码提供给用户的实质动作，也就没有触及 GPL 协议中“分发（Distribute）”的概念，也因此，GPL 协议被认为是拥有“SaaS Loophole”（SaaS 服务漏洞）[6]。在很长一段时间内，没有成规模的开源许可证可以针对性地对 SaaS 服务进行约束。

二、为 SaaS 而生的 AGPL 和 SSPL 协议

（一）为 SaaS 厂商特制的开源协议：AGPL

面对 SaaS 服务的崛起，出于保护其自研软件的目的，Affero Inc.发布了 AGPL v1，以限制 SaaS 服务提供商对其自研软件的自由使用[7]。自由软件基金会又在此基础上发布了 AGPL v3，后续被广泛应用到各个开源软件中。

AGPL v3 的诞生填补了 GPL 中只针对传统软件的分发模式赋予开源义务的“漏洞”，在 GPL 协议的基础上又增加了针对 SaaS 服务的限制。根据 AGPL v3 协议第 13 条的约定，如果对 AGPL 开源软

[3] https://www.gnu.org/licenses/old-licenses/gpl-2.0.html。

[4] https://www.gnu.org/licenses/old-licenses/gpl-2.0.html。

[5] https://www.gnu.org/licenses/old-licenses/gpl-2.0.html。

[6] https://www.synopsys.com/blogs/software-security/saas-companies-open-source-risk/。

[7] https://web.archive.org/web/20080315231323/http://www.softwarefreedom.org/technology/blog/2007/nov/21/stet-and-agplv3/。

件进行了修改，则需要向所有**通过计算机远程网络交互**的用户免费提供相应的源代码：

13. Notwithstanding any other provision of this License, if you modify the Program, your modified version must prominently offer all users interacting with it remotely through a computer network （if your version supports such interaction） an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge, through some standard or customary means of facilitating copying of software. This Corresponding Source shall include the Corresponding Source for any work covered by version 3 of the GNU General Public License that is incorporated pursuant to the following paragraph.

（尽管本许可有任何其他规定，如果您修改本程序，您的修改版本必须显著地向所有通过计算机网络远程与其交互的用户（如果您的版本支持这种交互）提供一个机会，通过一些便于复制软件的标准或惯用方式，使其可从网络服务器免费访问相应的源代码。此对应源应包括根据以下段落合并的 GNU 通用公共许可证第 3 版所涵盖的任何工作的对应源）。

—— AGPL v3 Article 13[8]

首先，需明确，AGPL 协议有针对地赋予了 SaaS 服务的开源义务：上述第 13 条中提到的“**通过计算机远程网络交互**”的场景包括网络和邮件服务器、基于互动的网络应用程序和在线播放的游戏服务器等，当然这也包括了 SaaS 服务。换句话说，AGPL 在继承了 GPL 开源义务的基础上，成功地将 SaaS 服务也纳入开源义务的约束中。根据 AGPL 协议第 13 条，在 SaaS 应用场景下，触发 AGPL 协议传染性的条件是对原 AGPL 程序进行了“修改（Modify）”。如果对 AGPL 程序进行了修改，那么需要开源包括 AGPL 修改版程序在内的帮助运行 SaaS 服务的所有程序的源代码，也就是进行整体开源。虽然在 SaaS 场景中只有“修改”才会触发 AGPL 的传染风险，在实践中，大多数 SaaS 服务商为了避免自研代码泄露的风险，对 AGPL 软件一律采取强硬的拒绝态度。例如，谷歌在其内部政策中明确禁止开发者使用 AGPL 软件[9]，对此，谷歌的技术大牛 Chris DiBona 称因为需要“非常非常小心地”来使用 AGPL 软件，不然就会面临共享代码的风险，公司的禁令“节省了开发时间”[10]。

常见的以 AGPL v3 开源的软件包括 Humhub、Grafana、MongoDB（后改用 SSPL）等。Flask 的开发者 Armin Ronacher 曾认为 AGPL v3 使得双重许可，即商业许可和开源许可证的双许可模式变得可行，对于许多初创公司而言，它可以使用 AGPL v3 协议以及商业许可开源自研产品，但使用者往往碍于 AGPL v3 的限制继而选择购买商业许可来获得源代码[11]。

（二）进阶的 AGPL：SSPL

对于一些软件商来说，仅靠 AGPL v3 保护商业利益是不够的，因此，SSPL 由此诞生。2018 年，MongoDB Inc.宣布其开发的 MongoDB 开源软件不再适用 AGPL v3，而是改用 SSPL 开源。MongoDB 在 SSPL 许可协议第 13 条中明确规定，如果将 SSPL 程序或程序的修改版本的功能作为服务向第三方提供时，则需要提供服务源代码。

13. If you make the functionality of the Program or a modified version available to third parties as a service,

[8] https://www.gnu.org/licenses/agpl-3.0.en.html。

[9] https://opensource.google/documentation/reference/using/agpl-policy。

[10] https://www.theregister.com/2011/03/31/google_on_open_source_licenses/。

[11] https://lucumr.pocoo.org/2013/7/23/licensing/。

you must make the Service Source Code available via network download to everyone at no charge, under the terms of this License. Making the functionality of the Program or modified version available to third parties as a service includes, without limitation, enabling third parties to interact with the functionality of the Program or modified version remotely through a computer network, offering a service the value of which entirely or primarily derives from the value of the Program or modified version, or offering a service that accomplishes for users the primary purpose of the Program or modified version.

"Service Source Code" means the Corresponding Source for the Program or the modified version, and the Corresponding Source for all programs that you use to make the Program or modified version available as a service, including, without limitation, management software, user interfaces, application program interfaces, automation software, monitoring software, backup software, storage software and hosting software, all such that a user could run an instance of the service using the Service Source Code you make available.

(13.如果您将本程序的功能或修改版本作为服务提供给第三方，您必须根据本许可的条款通过网络下载免费向所有人提供服务源代码。将本程序的功能或修改后的版本作为服务提供给第三方，包括但不限于使第三方能够通过计算机网络远程与本程序或修改后的版本的功能交互，提供一项服务，其价值完全或主要来源于本程序或修改版本的价值，或提供为用户实现本程序或修改版本的主要目的的服务。

"服务源代码"是指程序或修改版本的对应源代码，以及您用来使程序或修改版本作为服务提供的所有程序的对应源代码，包括但不限于管理软件、用户界面、应用程序接口、自动化软件、监控软件、备份软件、存储软件和托管软件，所有这些都是为了让用户可以使用您提供的服务源代码运行服务实例。)

—— ***SSPL Article 13***[12]

由此可以看出，区别于 GPL 和 AGPL 协议，SSPL 不再将"修改"作为触发开源传染性的条件，而是明确了，如果在为 SaaS 服务开发的软件中使用了 SSPL 软件，那么需要把所有使该软件能运行成 SaaS 服务的相应源码都进行开源。需要特别指出的是，SSPL 扩大了开源的范围，其第 13 条明确了需要开源的"服务源代码"包括但不限于"管理软件、用户界面、应用程序接口、自动化软件、监控软件、备份软件、存储软件和托管软件，所有这些都是为了让用户可以使用您提供的服务源代码运行服务实例"[13]，也就是说开源范围不仅仅包括了本程序的源码、还要包括和本程序配套使用的所有程序的源码。业内人员理解 SSPL 的本质要求是，如果提供服务的"本程序"中包含了 SSPL 组件，"你就要大方地把提供服务的整个配套程序（包括前后台）都贡献出来"[14]。

MongoDB 是一个面向文档的数据库，实现了分布式储存，解决了应用程序开发社区中的大量现实问题，一直以来都被广泛应用。MongoDB Inc.称这一改动主要是针对一些亚洲云服务提供商在自己的商业数据库版本中使用 MongoDB 源代码而不遵守开源许可证的现象[15]。MongoDB 公司曾向 OSI 申请将 SSPL 加入开源许可证认证列表，但被 OSI 拒绝[16]，后 MongoDB 撤回了申请。无独有偶，各大操作系统开发商，包括 Debian、Fedora 和 Red Hat Enterprise Linux 纷纷在各自发行版中删除了 MongoDB，

[12] https://www.mongodb.com/licensing/server-side-public-license。

[13] https://www.mongodb.com/licensing/server-side-public-license。

[14] https://my.oschina.net/vigor23/blog/5129221。

[15] https://www.zdnet.com/article/its-mongodbs-turn-to-change-its-open-source-license/。

[16] https://www.oschina.net/news/127950/osi-say-sspl-not-open?fr=vx。

Fedora 认为 SSPL 对商业用户具有严重歧视性，违背了自由软件许可证的初衷。一些开发者则认为，开源许可证的要求会越来越 Copyleft，“如果 MongoDB 换了一次，那么很可能会有第二次”[17]。

此外，除 MongoDB 外，还有一些软件选择将其许可证变换为 SSPL。譬如，Elastic NV 开发的 Elasticsearch 和 Kibana 从 7.11 版本开始不再使用 Apache 2.0 和他们自己的 Elastic License，而是变为使用 SSPL 和 Elastic License 进行“开源”[18]。面对这种变化，曾依赖 Elastic NV 产品的亚马逊 AWS 决定计划在之前以 Apache 2.0 开源的版本上开发分支（fork）项目以继续开发许可为 Apache 2.0 的版本[19]，以避免使用以 SSPL 开源的软件。

由此可知，开源软件许可证的变换确实能给开源软件开发者带来商业机会，同一开源软件的开源许可证可能会根据软件版本更新迭代进行变化，增减开源义务，或者增加商业版本许可。当然在这个过程中也许会造成现有客户的流失，不过开源软件开发者试图寻求通过修改开源许可证来达成平衡，而我们也需要思考如何应对这些日益“严苛”的开源义务，做到在 SaaS 服务中合规使用开源软件。

三、SaaS 场景下开源合规的常见误区

在这样的背景下，SaaS 服务商实际上处于使用开源软件的两难境地。一方面，基于应用服务开发的需求，使用各式各样的开源软件是必不可少的。但是在另一方面，使用开源软件可能面临项目被传染而被迫开源的风险，这对于 SaaS 服务商的产品差异化竞争是极为不利的[20]。

基于此，如何在使用开源软件的同时尽量避免开源义务的产生对于云服务商而言成为了亟待解决的问题。在分析可能的解决路径之前，我们有必要指出常见的面对这些开源软件时的合规性误区。

误区 1：对开源软件均采取动态链接库、管道等隔离地址空间的方法以消除开源风险，或是干脆不在商业版本中使用以 AGPL v3 开源的软件。

是否需要采取隔离地址空间的措施规避开源风险不能一概而论。如前所述，当 GPL 开源软件被分发，LGPL 开源软件被修改且分发时，会触发开源义务，但服务商可以通过对 GPL 或 LGPL 开源软件采取隔离地址空间的做法以避免开源。不过，正如前文所指出的，面对软件服务的日益进化，开源许可证本身也在随着时代发展而变化，更不用说作者对于开源许可证附加的保留、限制条款，开源许可证在如今呈现出多样化、动态化的趋势。对开源软件采取隔离地址空间的使用方式是应对 GPL 或 LGPL 的常见方法。对于适用 AGPL 协议的开源软件而言，若其被分发给用户，则可以参照 GPL 的规避方法采取隔离地址空间的方法以避免开源义务。在本文所关注的 SaaS 场景下，AGPL 开源组件处于服务器端用于为客户提供服务时，若该组件被修改了，则需要公开源代码以降低开源风险；但若不存在修改，则不触发开源义务。因此，是否需要采取隔离地址空间的方法以避免开源义务，是需要结合使用场景和协议类型进行具体分析的。

误区 2：开源软件仅应用于服务器端，权利人无法获得源码提起诉讼，不需进行风险合规。

对于仅应用于服务器端，不会分发给用户的开源软件，即使权利人获知开源软件被使用的概率较低，但在特定的商业应用场景中，如果不合规地使用开源软件，不能排除被诉侵权的风险。如在将自研 SaaS 服务设施提供给用户的场景中，即使 SaaS 服务商对开源软件采取了网络屏蔽、行为监测等措施，权利人仍然可

[17] https://opensource.stackexchange.com/questions/11554/mongodbs-sspl-license-what-does-offering-as-a-service-actually-mean。

[18] https://www.elastic.co/pricing/faq/licensing#summary-of-licensing-change。

[19] https://www.zdnet.com/article/aws-as-predicted-is-forking-elasticsearch/。

[20] 孙福洲，钱瑾，杨静&钱岭.（2020）.云服务时代下的开源发展.电信科学（11），156-164。

能通过招聘广告、企业宣传的内容获知开源代码被使用。如著名三维设计软件 CATIAV5 的权利人达索公司就曾在中国发起较大规模的软件著作权侵权诉讼[21]，达索公司通过企业网页招聘要求（如写明招聘者需精通三维设计软件 CATIAV5[22]）、商业沟通（如与企业电话咨询其产品服务详情[23]）等方式得知有企业使用了其开发的软件，并由此向法院请求对企业的计算机证据保全并进行进一步的代码比对，或向文化执法机关请求检查所使用的软件，由此获得证据提起侵权之诉。又如，在 To B 的 SaaS 服务模式中，SaaS 服务商将包含开源软件的 SaaS 服务出售给企业，企业利用 SaaS 服务与用户进行连接。在商业实践中，云服务商往往将同一类的 SaaS 服务出售给同一类型的企业客户，即使 SaaS 服务商与企业客户签订有保密协议，但是由于客户繁多，且客户内部的保密管理优劣不一，仍无法排除泄露 SaaS 服务使用开源软件的信息的可能性。

综上所述，开源软件的合规方式因开源许可证条款、权利人保留限制条款及开源软件使用场景的不同而不同，对使用的开源软件一律采取隔离地址空间的方法或者干脆不在商用版本中使用带有风险的开源软件既不是防止风险的“万金油”，也是不经济不合理的。

四、SaaS 与开源软件共存的合规建议

在避免了误区的前提下，结合 AGPL 和 SSPL 协议的要求，对于 SaaS 场景下的开源合规的风险筛查具体路径，我们归纳了如下流程图，供企业合规实务中参考。

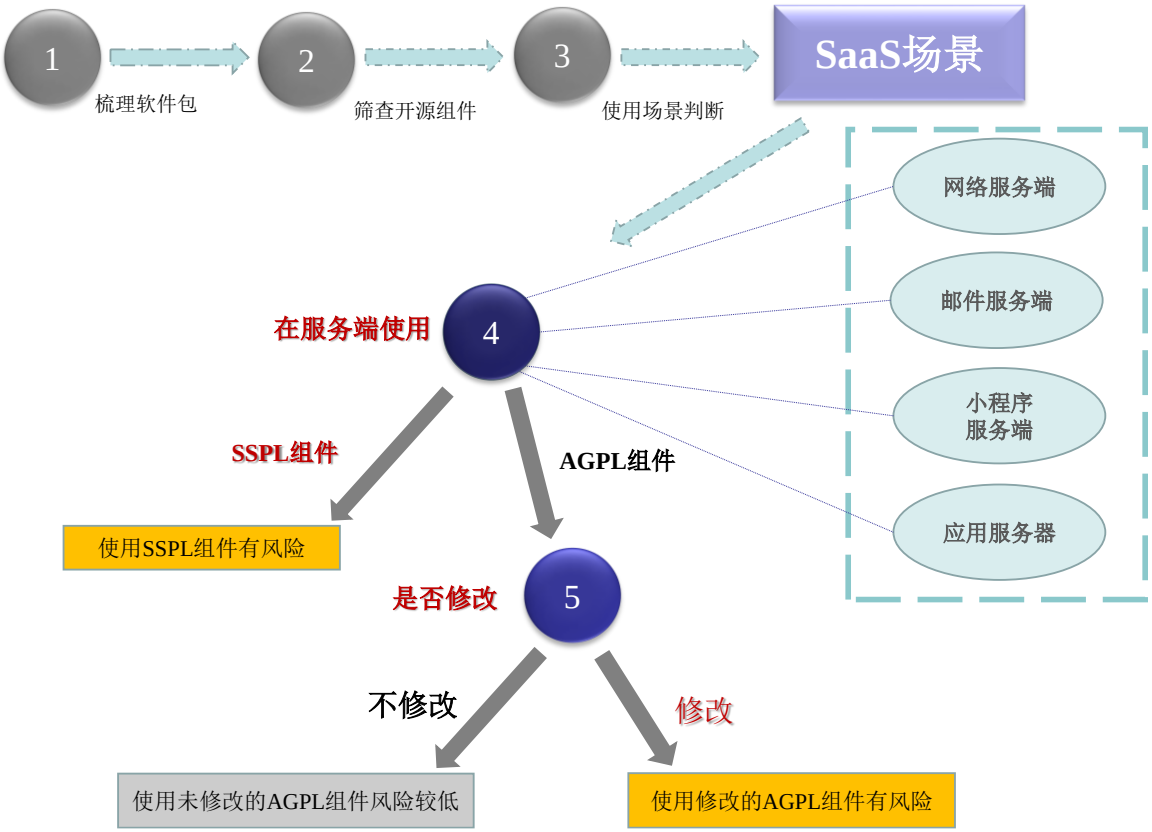

图 1 开源合规风险筛查的具体路径

对于已经开发好的或者正在开发中的产品，**首先**，是梳理筛查出该产品中使用的开源组件，此处的使用

[21] CATIAV5 软件的权利人为达索公司，在北大法宝以当事人关键词“达索”检索著作权侵权案，有约 100 起案件。

[22] （2017）皖 02 民初 138 号民事判决书。

[23] （2021）最高法知民终 1560 号民事判决书。

方式不仅指产品中包含该开源组件，也包括（1）在产品中以动态链接、静态链接，（2）通过产品的插件或服务器下载，或（3）通过网络连接或云服务提供等方式；开源组件不仅包括开源软件本身（全部或部分），还包括开源软件的任何代码与文件（包含且不限于：源代码、二进制代码、API 或修改过后的代码）。**其次**，应对该组件的具体应用场景作出判定，确定应用场景是否为云服务、是否涉及“分发”和“不分发”，是否是在服务端使用，是否涉及“修改”或“不修改”；其中“分发”系指指以任何方式对外（譬如用户）提供开源软件，包含但不限于用户自行下载该产品、私有部署等情况。

在本文所关注的 SaaS 服务的场景下，则尤为需要注意开源协议的风险。这里所说的 SaaS 服务包括但不限于网站服务、邮件服务、小程序云服务、应用云服务。若使用的组件适用 SSPL，则存在开源风险，需替换或购买商业版本；若使用的组件适用 AGPL，则需要通过避免修改或购买商业版本来避免风险。

五、结语

十多年前的软件行业以开源软件、独立商业版本软件、SaaS 服务为主呈现出“三足鼎立”的态势，软件开发者往往围绕着其中一种软件服务模式开发产品。随着软件技术的改革和发展，如今的软件开发更趋向多方协作，产品类型不再仅限于开源、独立商业版本或是 SaaS 服务其中一种，而是三者融合、相互扶助，也才有了我们看到的被包裹在 SaaS 服务中的开源软件的应用。

软件的应用场景瞬息万变，开源软件因为其自身的可变化性和适应性强而被广泛应用，也因此造就了开源软件许可证的持续变化。一方面是要保护开源社区、促进开源精神的长足发展，软件开发者想要在众多生存开发者中寻求稳定的生存之路，正如我们在 MongoDB 中看到的开源许可证从 AGPL v3 到 SSPL 的变换所带来的影响一样。另一方面，开源许可证的不断演化同时造就了挑战，企业因开源许可证的变化而对开源合规进行不断地更新与调整，在这个过程中抓住机会并合理地规避风险做到开源合规，乃是抢占市场先机、牢筑竞争优势的关键。

特别声明

汉坤律师事务所编写《汉坤法律评述》的目的仅为帮助客户及时了解中国或其他相关司法管辖区法律及实务的最新动态和发展，仅供参考，不应被视为任何意义上的法律意见或法律依据。

如您对本期《汉坤法律评述》内容有任何问题或建议，请与汉坤律师事务所以下人员联系：

段志超

电话： +86 10 8516 4123
Email： kevin.duan@hankunlaw.com