Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Chancellerie fédérale ChF
Transformation numérique et gouvernance de l'informatique TNI

14.09.2026

Em002-6 FAQ sur les OSS et l’art. 9 LMETA

Version 2.0-f544937

Table des matières

1. Contexte et objectif 3
1.1. Contexte et objectif de la FAQ 3
1.2. Structure et organisation 3
2. Notions et définitions 3
2.1. Logiciel open source 3
2.2. Logiciels et licences 3
2.3. Base légale dans la LMETA 4
3. Rôles et responsabilités 6
3.1. Rôle de la Chancellerie fédérale 6
3.2. Rôle des unités administratives 6
3.3. Collaboration avec des fournisseurs tiers 7
3.4. Communication de données personnelles 7
3.5. Acquisition 7
4. Processus 8
4.1. Instructions 8
4.2. Choix de la licence 9
4.3. Critères de qualité pour les logiciels open source 9
4.4. Documentation 10
4.5. Contributions de l’administration fédérale à des projets 11
4.6. Support 11
5. Aides 11
5.1. Instructions 11
5.2. Listes de contrôle 12
5.3. Questions de sécurité 12
6. Questions juridiques 13
6.1. Utilisation de l’anglais 13
6.2. Exclusion de responsabilité 14
6.3. Logiciels permissifs 15
6.4. Dispositions de licence 15

6.5. Rapport entre la LTrans et la LMETA 15

6.6. OSS et protection des données 16

7. Généralités 16

8. Annexe 18

8.1. Abréviations 18

8.2. Glossaire 19

> **NOTE** Ce document est un projet en cours d'élaboration et fait partie des aides destinées à soutenir l'administration fédérale dans la publication de code open source. Pour plus d'informations, voir le README.

1. Contexte et objectif

1.1. Contexte et objectif de la FAQ

Les aides à la publication de logiciels open source suscitent régulièrement des questions. La compilation ci-après répond aux questions les plus fréquentes et vise à améliorer la compréhension de certains thèmes et aspects.

1.2. Structure et organisation

La FAQ est organisée par thème:

2. Notions et définitions
3. Rôles et responsabilités
4. Processus
5. Aides
6. Questions juridiques
7. Généralités

2. Notions et définitions

2.1. Logiciel open source

Qu'entend-on par logiciel open source?

Cette notion est définie dans Em002-1 Guide pratique pour les logiciels open source dans l'administration fédérale.

2.2. Logiciels et licences

Qu'entend-on par logiciel?

Cette notion est définie dans Em002-1 Guide pratique pour les logiciels open source dans l'administration fédérale. Pour les licences, voir Em002-3 Guide des licences OSS.

Qu'entend-on par licences?

Il s'agit ici de licences de logiciels. De plus amples informations figurent dans Em002-3 Guide des licences OSS.

Les modèles d'IA développés en interne doivent-ils être considérés comme des logiciels et donc soumis aux exigences légales de l'art. 9 LMETA? Qu'en est-il des données d'entraînement?

L'open source désigne les logiciels dont le code source est librement accessible et modifiable à des conditions minimales. Cela comprend également les modèles d'IA entraînés en interne (y compris l'algorithme, la procédure d'entraînement et le code).

Les modèles d'IA sont entraînés à partir de données, le cas échéant des Open (Government) Data. Les Open Data désignent des jeux de données pouvant être librement utilisés, partagés et

retraités. Les modèles d'IA dont les données d'entraînement reposent sur des Open (Government) Data sont considérés comme des logiciels au sens de la LMETA et doivent donc également être publiés.

Toutefois, si les données d'entraînement contiennent des données personnelles (sensibles) ou des informations classifiées, il est possible de renoncer à la publication pour des motifs de sécurité. Lors de la publication d'un modèle d'IA en open source, les données d'entraînement ne doivent pas nécessairement être publiées; cela correspond cependant à l'esprit de l'open source et est vivement recommandé par la communauté open source. Pour de plus amples informations sur l'IA, nous renvoyons au Réseau de compétences en intelligence artificielle (CNAI).[1]

Les outils dotés de capacités d'IA générative (p. ex. GitHub Copilot) doivent-ils être contrôlés avant leur utilisation afin qu'aucun droit de propriété intellectuelle de tiers ou de la Confédération ne soit violé?

Les contenus générés par un modèle d'IA lui-même ne peuvent pas être protégés par le droit d'auteur, faute d'activité créatrice. Les contenus générés par l'IA qui s'appuient sur des œuvres protégées de tiers peuvent toutefois porter atteinte aux droits de ces tiers. De plus, les sorties des IA étant notoirement peu fiables, elles ne devraient être utilisées qu'après un examen attentif au cas par cas.

Par ailleurs, les exploitants de systèmes d'IA peuvent se réserver, dans leurs conditions d'utilisation, le droit d'employer les entrées pour entraîner le système. Cela peut à son tour conduire à ce que des entrées ou des parties d'entrées soient affichées à des tiers. Lorsque cela ne peut être exclu, il faut s'assurer que les entrées ne violent pas les droits de la Confédération ou de tiers.

Il est recommandé d'examiner et d'autoriser individuellement l'utilisation des outils d'IA. Les outils non autorisés ne peuvent pas être utilisés, mais les collaborateurs peuvent demander l'autorisation d'un outil. Il est souvent judicieux de régler ces questions contractuellement avec le fournisseur; certains fournisseurs proposent aussi des abonnements particuliers par lesquels ils respectent les droits de tiers, p. ex. DeepL.

2.3. Base légale dans la LMETA

Quelle est la base légale de la publication des logiciels open source?

Selon l'art. 9 LMETA[2], les autorités fédérales de l'administration fédérale centrale doivent divulguer le code source des logiciels qu'elles développent ou font développer. Toute personne peut utiliser, développer ou modifier le logiciel sans devoir acquitter d'émoluments.

Il ne peut être renoncé à la publication du code source qu'en raison de droits de tiers ou pour des motifs de sécurité.

De plus amples informations figurent dans Em002-2 Instructions pour la publication de logiciels open source.

Qu'entend-on par motifs de sécurité?

Les motifs de sécurité admis par l'art. 9 LMETA et la manière de les traiter sont définis au chapitre 3.2 de Em002-2 Instructions pour la publication de logiciels open source.

Que sont les droits de tiers?

Les droits de tiers au sens de l'art. 9 LMETA et la manière de les traiter sont définis au chapitre 3.3 de Em002-2 Instructions pour la publication de logiciels open source.

1. CNAI ainsi que l'art. 10 LMETA
2. RS 172.019

Les exceptions peuvent-elles être limitées dans le temps?

Les motifs justifiant une exception au sens de l'art. 9 LMETA (droits de tiers et motifs de sécurité) peuvent disparaître avec le temps.

Si tel est le cas, le code source du logiciel doit être publié.

Quelles conséquences juridiques faut-il envisager en cas de violation de l'art. 9 LMETA?

Il existe deux types de violations:

1. Des droits de tiers sont violés.
2. Une autorité fédérale refuse de publier un logiciel open source (p. ex. en invoquant des motifs de sécurité non fondés).

Réponse au point 1:

Il s'agira régulièrement d'une violation contractuelle et des prétentions en dommages-intérêts peuvent être invoquées conformément au code des obligations.

Réponse au point 2:

Cela n'a pas de conséquence directe. Si un tiers souhaite pouvoir utiliser le code source d'une autorité fédérale, il doit déposer une demande fondée sur la loi sur la transparence (LTrans; RS 152.3) auprès de l'autorité compétente.

En cas d'intérêt marqué, il peut être judicieux d'envisager une publication dans l'intérêt des deux parties. Il n'existe aucune obligation de publier rétroactivement, mais cela reste possible à titre volontaire.

Des prétentions en responsabilité peuvent-elles être invoquées en cas de dommage causé par un logiciel open source?

En principe, la responsabilité devrait être exclue dans les contrats. Il n'est toutefois pas possible d'exclure la responsabilité pour négligence grave ou faute. Dans de tels cas, une autorité fédérale pourrait être tenue pour responsable en vertu de la loi sur la responsabilité (LRCF; RS 170.32).

À partir de quelle date les logiciels doivent-ils être publiés?

Selon l'art. 9, al. 1, LMETA, les logiciels développés après le 1er janvier 2024 doivent être publiés.

Pour les logiciels développés avant le 1er janvier 2024, une publication devrait être examinée lorsque des modifications importantes sont prévues.

En cas d'intérêt considérable de tiers, les logiciels existants peuvent également être publiés à titre volontaire (le cas échéant avec une participation aux coûts). Dans ce cas, une communauté (voir Em002-4) devrait aussi être envisagée.

Pour les petits scripts, la publication n'est généralement pas la meilleure option. Il peut exister des dépôts d'outils qui passent alors ensemble par le processus de publication.

Pour les unités administratives de l'administration fédérale décentralisée soumises à la LMETA, l'obligation de publier s'applique depuis le 1er mai 2025. Des exceptions à l'obligation de publier existent pour les logiciels développés sans fonds fédéraux et pour les logiciels de recherche (art. 3, al. 1, OMETA).

Peut-on exiger la publication rétroactive de logiciels développés avant le 1er janvier 2024?

La loi ne prévoit pas de publication rétroactive.

En outre, les contrats conclus avant le 1er janvier 2024 peuvent contenir des droits de tiers qui s'opposent à une publication.

Si une nouvelle version majeure est prévue (major selon le Semantic Versioning[3]), une publication de l'ensemble du logiciel devrait être envisagée. Sinon, il faudrait au moins publier le code source des nouvelles fonctionnalités.

Sur quelle base un tiers devrait-il demander la publication du code source?

Un tiers devrait déposer une demande fondée sur la LTrans auprès de l'autorité compétente.

3. Rôles et responsabilités

3.1. Rôle de la Chancellerie fédérale

Qui met à disposition les aides de base pour la mise en œuvre?

Le secteur TNI de la Chancellerie fédérale met à la disposition des autorités fédérales des aides sous la forme d'un ensemble de documents accompagnant Em002. Cela comprend les directives stratégiques et d'autres guides pratiques, y compris la FAQ, ainsi que des listes de contrôle destinées à garantir une mise en œuvre conforme au droit.

Les aides OSS sont mises à jour par le secteur TNI.

La Chancellerie fédérale est-elle compétente pour accorder des exceptions (p. ex. pour des motifs de sécurité)?

Il n'existe aucun organe central qui statue sur les exceptions à l'obligation de publier prévue à l'art. 9 LMETA. Chaque autorité fédérale est responsable de sa propre mise en œuvre conforme au droit.

Les départements peuvent édicter des réglementations.

La liste de contrôle relative à l'exception générale selon l'art. 9 LMETA [BBL-CL] de l'OFCL devrait être remplie.

3.2. Rôle des unités administratives

Qui est responsable de la mise en œuvre opérationnelle des publications OSS?

Chaque autorité fédérale (p. ex. office, unité administrative) qui développe ou fait développer des logiciels est responsable de l'ensemble du processus de publication.

Les départements peuvent fixer des réglementations contraignantes et des mécanismes de contrôle dans leur domaine de compétence, p. ex. pour éviter des atteintes à la réputation.

Il est recommandé de désigner, par office fédéral, une personne responsable de l'OSS, p. ex. le responsable informatique (IM), l'architecte d'entreprise, le CSI ou le CPD, afin de garantir des processus uniformes et efficaces. Si rien n'est prévu, la direction de l'office est en principe responsable d'une mise en œuvre conforme.

Cette personne pourrait ensuite également être consultée pour l'octroi d'exceptions à l'obligation de publier.

3. Gestion de versions — Wikipédia

3.3. Collaboration avec des fournisseurs tiers

Comment les développements internes d'une autorité fédérale doivent-ils être publiés?

Le droit d'auteur naît auprès des personnes physiques qui ont développé le code source, mais il est généralement transféré à l'employeur par le contrat de travail. Dans les contrats avec des prestataires externes, il est important de délimiter clairement les nouveaux développements et de convenir de la cession ou du transfert contractuel du droit d'auteur. Enfin, pour les développements internes communs, les parties concernées doivent s'accorder sur la licence open source du code source sous laquelle le logiciel est publié en tant que projet open source autonome. Publier sous une licence open source ne signifie pas renoncer au droit d'auteur: celui-ci reste auprès des titulaires respectifs. Ce n'est que sur la base de leur droit d'auteur qu'ils peuvent se défendre juridiquement contre des violations de licence ou accorder, parallèlement aux licences OSS, des licences plus étendues.

3.4. Communication de données personnelles

À quelles conditions des données personnelles, par exemple celles des développeurs, peuvent-elles être publiées dans le cadre d'un logiciel open source?

Il n'existe pas de règle générale à ce sujet. La décision appartient au projet.

Dans tous les cas, les dispositions applicables en matière de protection des données doivent être respectées. Selon l'art. 36, al. 1, LPD, les organes fédéraux ne peuvent communiquer des données personnelles que s'ils disposent d'une base légale pour ce faire.

L'art. 36, al. 2, LPD prévoit notamment les exceptions suivantes:

(b) La personne concernée a consenti à la communication.

(e) La personne concernée a rendu ses données accessibles à tout un chacun et ne s'est pas opposée expressément au traitement.

Dans la plupart des cas, il est conseillé d'obtenir préalablement le consentement des personnes concernées, celles-ci ayant généralement elles-mêmes intérêt à ce que leur travail soit visible publiquement. Si le consentement n'est pas donné ou si d'autres motifs s'opposent à la publication, le code source devrait être publié sans données personnelles, en ne mentionnant que l'autorité fédérale concernée.

Ce consentement devrait être obtenu auprès des fournisseurs et des collaborateurs dans le cadre des processus d'acquisition et d'engagement.

3.5. Acquisition

Où trouver la documentation relative à l'acquisition de logiciels open source?

Avec la version 2.0, l'aide Em002-7 Aspects stratégiques de l'acquisition et logiciels open source a été mise à disposition.

Le Centre de compétence des marchés publics de la Confédération (CCMP) a publié un nouvel aide-mémoire intitulé Acquisition de logiciels et art. 9 LMETA [KBB-MB] ainsi que deux documents «Critères types – acquisition LMETA et open source.docx» et «Textes contractuels types – développement de logiciels.docx».

D'autres ressources sont disponibles sur la plateforme d'apprentissage et de modèles des marchés publics (www.perimap.admin.ch).

Une aide supplémentaire est disponible sur l'intranet de l'OFCL (accessible uniquement sur le réseau de la Confédération) avec la directive Open source dans les acquisitions [BBL-WL] et la

liste de contrôle relative à l'exception générale selon l'art. 9 LMETA [BBL-CL].

Les informations correspondantes se trouvent sur les pages intranet de l'OFCL (disponibles uniquement au sein de la Confédération).

Liens: boîte à outils de l'OFCL pour l'acquisition informatique

4. Processus

4.1. Instructions

Existe-t-il des instructions centrales sur la manière de traiter les logiciels open source dans l'administration fédérale?

Oui, le secteur TNI de la Chancellerie fédérale met à disposition une documentation complète à cet effet.

Il convient de noter que ces aides constituent une recommandation TIC et non une prescription contraignante.

Lien vers les aides OSS

Comment publier les développements internes d'une autorité fédérale réalisés dans le cadre de collaborations? Comment traiter les droits existants?

Lors du développement de logiciels, le droit d'auteur naît auprès des personnes physiques qui ont développé le code source.

Si la personne qui développe est liée par un contrat de travail, les développements appartiennent à l'employeur (art. 332 CO).

Dans les contrats avec des prestataires externes, il est donc important de délimiter clairement les nouveaux développements et de convenir de la cession ou du transfert contractuel du droit d'auteur. Enfin, pour les développements internes communs, les parties concernées doivent s'accorder sur la licence open source du code source sous laquelle le logiciel est publié en tant que projet open source autonome.

Publier sous une licence open source ne signifie pas renoncer au droit d'auteur: celui-ci reste auprès des titulaires respectifs.

Ce n'est que sur la base de leur droit d'auteur qu'ils peuvent se défendre juridiquement contre des violations de licence ou accorder, parallèlement aux licences OSS, des licences plus étendues.

Les droits originaux devraient donc, si possible, toujours revenir à la Confédération pour les nouveaux développements ou les développements ultérieurs. Pour les autres participants, il convient d'utiliser des Contributor Licence Agreements (CLA) ou un Developer Certificate of Origin (DCO). Des instructions à ce sujet figurent dans Em002-3 Guide des licences OSS, Em002-2 Instructions pour la publication de logiciels open source et Em002-4 Guide des communautés OSS.

Les modifications doivent-elles être publiées?

Selon l'art. 9 LMETA: oui, si elles ont été développées après le 1er janvier 2024.

Cela devrait toutefois se faire de manière à en tirer également un bénéfice. Le document Em002-2 Instructions pour la publication de logiciels open source informe sur le processus.

Quand la publication est-elle «suffisante» au sens de la LMETA?

L'art. 9 LMETA n'est pas précis à cet égard et laisse une grande marge de manœuvre. Le code source doit être accessible au public.

Les exigences de la loi sont par exemple déjà satisfaites par la simple publication du code source sous forme de fichier zip sur un site web.

Pour en tirer un bénéfice, notamment par la publication de la documentation, etc., cela devrait se faire dans un dépôt de code selon les bonnes pratiques. Des informations à ce sujet figurent dans Em002-2 Instructions pour la publication de logiciels open source.

À l'avenir, les bénéficiaires de prestations prescriront-ils aux fournisseurs de prestations la manière de publier le logiciel et d'en assurer la qualité?

Les règles de chaque autorité fédérale s'appliquent aussi bien aux fournisseurs qu'aux bénéficiaires de prestations.

Il est toutefois judicieux d'intégrer les aides correspondantes aux endroits appropriés dans les processus et les quality gates existants.

Sur la base de son expérience de développement, le fournisseur de prestations peut faire des propositions à ses bénéficiaires de prestations et traiter cela de manière uniforme (comme le processus de développement est traité de manière uniforme).

4.2. Choix de la licence

Sous quelle licence open source un logiciel doit-il être développé?

Il n'existe pas de prescriptions strictes quant au choix de la licence, pour autant que les conditions de licence des composants logiciels utilisés dans le logiciel à développer soient respectées.

Pour les développements entièrement nouveaux, il convient de choisir un type de licence permettant une base large et durable pour les développements ultérieurs.

Il est important à cet égard que la licence concernée soit largement acceptée dans la communauté de développeurs pertinente.

L'utilisation de l'AGPL (avec copyleft) et de la MIT est donc recommandée.

Ces deux licences représentent en quelque sorte les deux extrêmes du spectre des licences.

L'AGPL, avec un copyleft très fort, impose durablement le principe «public money – public code». En cas de développements ultérieurs, notamment de solutions propriétaires, cela s'atténue en direction de la LGPL.

La MIT est en revanche une licence très libérale, où presque tout est possible.

Ces deux licences couvrent l'ensemble du spectre. Si, pour les licences permissives, on attache une importance particulière à ne pas mentionner la Confédération, la BSD-3 est à préférer à la MIT.

Pour un choix plus détaillé des licences, nous renvoyons à Em002-3 Guide des licences OSS.

4.3. Critères de qualité pour les logiciels open source

Les logiciels open source doivent-ils satisfaire à certains critères de qualité?

En règle générale, tous les critères de qualité applicables à n'importe quel autre logiciel s'appliquent également. L'open source rend avant tout tout plus transparent. Pour respecter la licence, les critères d'une «bonne» publication et les éventuelles règles de la communauté,

quelques points supplémentaires sont nécessaires. Ils sont présentés dans Em002-2.2 Liste de contrôle analyse et préparation OSS et Em002-4.1 Liste de contrôle communauté OSS.

Quelles exigences sont prévues pour les quality gates?

Aucune prescription centrale n’est prévue. Les règles de l’autorité fédérale (unité administrative) qui développe le logiciel sont déterminantes.

Hormis les exigences générales, il n’existe pas d’exigences particulières concernant la qualité des logiciels open source. HERMES sera toutefois mis à jour pour tenir compte de l’art. 9 LMETA.

La publication est la manière dont l’administration fédérale se présente au public. Si cela n’est pas fait avec soin et professionnalisme, l’image publique d’une autorité fédérale peut rapidement en pâtir.

Comment les directives de développement sont-elles réglées?

Hormis les exigences générales figurant dans les aides, il n’existe pas de prescriptions particulières concernant le développement de logiciels open source. Des indications figurent dans Em002-2.2 Liste de contrôle analyse et préparation OSS et Em002-4 Guide des communautés OSS.

Les logiciels open source doivent-ils être publiés à un endroit précis?

La Confédération n’exploite actuellement pas de dépôt central propre. Les autorités fédérales sont donc en principe libres quant au lieu de publication. Comme de nombreuses autorités fédérales utilisent déjà GitHub, cette plateforme est actuellement le choix privilégié. Il existe en outre un compte GitHub central (swiss) qui peut être utilisé.

GitHub – swiss/index: un aperçu des organisations de dépôts actuelles.

C’est le cas en particulier lorsqu’une autorité fédérale ne dispose pas encore d’un compte GitHub («organisation») ou n’utilise pas d’autre plateforme telle que GitLab ou BitBucket.

Veuillez vous adresser à l’unité compétente du secteur TNI (opensource@bk.admin.ch) pour vérifier si une publication via le compte GitHub central (swiss) est judicieuse et possible.

Les autorités fédérales devraient dans tous les cas conserver une copie locale du code. Il est également conseillé que chaque autorité fédérale dispose d’une stratégie correspondante et de règles internes correspondantes.

D’autres recommandations figurent dans Em002-2 Instructions pour la publication de logiciels open source.

4.4. Documentation

Quelles exigences de documentation s’appliquent à la publication de logiciels open source?

Les aides mises à disposition par le secteur TNI de la Chancellerie fédérale informent sur les exigences légales; voir les aides OSS.

Il est recommandé de remplir les listes de contrôle OSS et de les conserver de manière centralisée dans chaque autorité fédérale (unité administrative).

4.5. Contributions de l'administration fédérale à des projets

Une unité administrative peut-elle et doit-elle contribuer du code à un projet «upstream», et satisfait-elle ainsi aux exigences de l'art. 9 LMETA?

- Si les projets sont open source, l'art. 9 LMETA est satisfait. Les projets doivent toutefois aussi rester open source. À défaut, le code doit être publié à nouveau.
- Les contributions upstream ont l'avantage que la maintenance a lieu dans le projet upstream et que le bénéfice pour la communauté est maximisé.

  Les inconvénients suivants existent: moins de contrôle et d'influence; il faut éventuellement signer un CLA ou un DCO (voir Em002-4 chapitre 3.3). Les contributions n'apparaissent en outre pas dans le publiccode.yml ni dans les répertoires de la Confédération. Un tout autre avantage est qu'avec de telles contributions, les développeurs continuent d'évoluer au sein de l'écosystème.
- Tant que les risques et la charge restent maîtrisables (contenu du CLA, collecte du code), les contributions upstream sont à saluer.
- La licence du projet upstream s'applique alors à la contribution. Cela correspond à la procédure décrite dans Em002-3.
- **Les forks de projets upstream sont à éviter:** toute la charge de maintenance et de mise à jour incombe alors à l'autorité fédérale. Un fork insuffisamment entretenu représente en outre un risque de sécurité. Le bénéfice pour la communauté est quasi nul.
- Les contributions à des projets upstream devraient en principe être appréciées positivement et activement discutées par les responsables de projet des autorités fédérales.
- Si le projet upstream appartient à l'administration fédérale et qu'il s'agit de contributions de tiers, Em002-7 chapitre 9 apporte une aide.

4.6. Support

L'autorité fédérale est-elle tenue d'assurer le support des logiciels publiés?

Non, mais elle peut le faire si elle le souhaite.

Selon l'art. 9, al. 5 et 6, LMETA, elle peut également percevoir des émoluments pour le support.

Comment une autorité doit-elle procéder si elle souhaite facturer le support?

Pour des raisons de ressources, cela ne fait pas partie du projet actuel de mise à disposition d'aides. Si nécessaire, cela pourrait être intégré aux aides dans une étape ultérieure. Pour l'instant, chaque autorité fédérale règle cela elle-même. (Voir aussi Em002-4 Guide des communautés OSS)

5. Aides

5.1. Instructions

Existe-t-il des instructions centrales sur le travail avec des logiciels open source dans l'administration fédérale?

Oui, le secteur TNI de la Chancellerie fédérale met à disposition une documentation complète à cet effet dans le cadre de Em002 Directives stratégiques pour les logiciels open source dans l'administration fédérale.

Les logiciels open source sont traités de manière générale dans Em002-1 Guide pratique pour les logiciels open source dans l'administration fédérale. La publication au sens de l'art. 9 LMETA est traitée dans Em002-2 Instructions pour la publication de logiciels open source.

5.2. Listes de contrôle

Les aides comprennent-elles également des listes de contrôle pour la publication de logiciels open source?

Oui, le secteur TNI de la Chancellerie fédérale met les listes de contrôle suivantes à libre disposition:

- Em002-2.1 Liste de contrôle évaluation préalable OSS
- Em002-2.2 Liste de contrôle analyse et préparation OSS
- Em002-2.3 Liste de contrôle validation et publication OSS
- Em002-4.1 Liste de contrôle communauté OSS

Voir également la **liste de contrôle relative à l'exception générale selon l'art. 9 LMETA** [BBL-CL] de l'OFCL.

5.3. Questions de sécurité

Existe-t-il des risques de sécurité propres aux logiciels open source?

Dans ce contexte, on mentionne par exemple les attaques sur la chaîne d'approvisionnement, les vulnérabilités zero-day et les attaques par typosquatting.

Ces questions de sécurité doivent être prises en compte pour les logiciels open source et leur acquisition, mais aussi de manière générale pour le développement de logiciels et l'acquisition de logiciels commerciaux. Elles peuvent tout au plus être quelque peu amplifiées si le projet OSS utilisé a manqué de rigueur lors de l'intégration de bibliothèques et de conteneurs.

À moyen terme, une stratégie générale est nécessaire pour atténuer ces risques (via le NCSC/OFCS). Si nécessaire, des sources sûres pour les conteneurs et les bibliothèques doivent être mises à disposition.

De manière générale, toutes les personnes responsables dans l'environnement informatique doivent connaître ces vecteurs d'attaque possibles. Les tentatives possibles comprennent:

- Attaques sur la chaîne d'approvisionnement: `XZ_Utils_backdoor` (https://en.wikipedia.org/wiki/XZ_Utils_backdoor), Codecov (https://blog.gitguardian.com/codecov-supply-chain-breach/)
- Vulnérabilités zero-day: Log4Shell (https://www.ibm.com/de-de/think/topics/log4j)
- Typosquatting: https://en.wikipedia.org/wiki/Typosquatting

La publication du code source augmente-t-elle le risque d'une copie malveillante d'une application? (p. ex. une application d'hameçonnage ou une fausse application)

Non. Retenir le code source n'offre aucune protection efficace contre les fausses applications et est de surcroît incompatible avec l'art. 9 LMETA, qui établit le principe de l'open source pour l'administration fédérale.

Le risque qu'un tiers crée une application visuellement identique pour collecter des données existe indépendamment de la publication du code source. Comme les systèmes de design de la

Confédération sont publiquement accessibles, l’apparence d’une application peut être reproduite par un attaquant sans accès au code source. Les applications peuvent en outre être décompilées afin d’extraire des éléments tels que des logos de la version publiée.

La sécurité et la confiance sont donc assurées non par l’obscurité, mais par la vérification et la sécurité cryptographique:

- **Signatures numériques:** les applications officielles doivent être signées cryptographiquement, ce qui garantit techniquement qu’une application est un original non modifié de la Confédération. Le système d’exploitation du smartphone vérifie cette signature.
- **Vérification par les magasins d’applications:** les magasins d’applications d’Apple et de Google proposent des procédures permettant de vérifier que les éditeurs sont des organes étatiques officiels. Google, par exemple, désigne explicitement les applications de la Chancellerie fédérale (ChF) comme des applications gouvernementales, ce qui donne aux utilisateurs confiance dans leur authenticité.
- **Builds reproductibles:** grâce à des builds indépendants et reproductibles, il est possible de démontrer techniquement que le code source publié correspond exactement à l’application disponible dans le magasin.

Une exception générale pour des motifs de sécurité au sens de l’art. 9 LMETA n’est donc pas applicable aux applications.

Recommandation:
Plutôt que de retenir le code source, l’accent devrait être mis sur les processus des magasins d’applications et sur la surveillance. Les canaux officiels doivent être tenus à jour et les fausses applications signalées sans délai afin d’être supprimées par les exploitants des plateformes (Apple et Google).

Est-il judicieux que les services demandeurs au sein de l’administration fédérale se regroupent pour acquérir des logiciels open source?

Oui, absolument. Cela peut se faire directement et de manière informelle entre unités administratives.

Le problème est plutôt que les unités administratives connaissent mal les besoins des autres.

Une solution pourrait consister à ce que les unités administratives tiennent une liste «most wanted» d’outils (ou de bibliothèques) open source à développer ou à mettre à disposition et qu’elles partagent cette liste.

Une telle liste gérée de manière centralisée pourrait aussi aider à mieux analyser le besoin en composants open source et à mieux évaluer le degré de maturité de l’unité administrative. En fin de compte, un dépôt d’artefacts serait judicieux.

La tenue d’une telle liste serait alors la tâche d’un OSPO, si un tel bureau était introduit. Au mieux, cela pourrait aussi se faire via Administration numérique suisse pour l’ensemble de la structure fédérale.

6. Questions juridiques

6.1. Utilisation de l’anglais

La Confédération peut-elle utiliser des licences dans leur langue originale, l’anglais?

Selon l’art. 9, al. 4, LMETA, il convient d’utiliser dans la mesure du possible et lorsque cela est judicieux des textes de licence établis au niveau international.

Insister sur des licences OSS en langue allemande réduirait largement à néant le but de l'art. 9, al. 4, LMETA, car il existe très peu de textes de licence établis au niveau international pour lesquels une version dans l'une des langues officielles de la Suisse (allemand, français et italien selon l'art. 70, al. 1, Cst.) est reconnue comme déterminante pour l'interprétation.

Cette situation est comparable à celle des filières en anglais dans les hautes écoles telles que l'EPFZ, où l'anglais est admis comme langue d'enseignement – sous réserve des trois conditions de l'art. 36 de la Constitution fédérale (commentaire saint-gallois de la Constitution fédérale, 4e éd. 2023, art. 70 N 29; il faut donc une base légale, un intérêt public et la proportionnalité pour que l'anglais soit admis).

Sur cette base, l'art. 9, al. 4, LMETA doit être interprété comme constituant la base légale de l'utilisation de licences en anglais, les licences internationales étant majoritairement rédigées en anglais. La disposition doit donc se référer à celles-ci.

L'intérêt public de l'art. 9 LMETA réside notamment dans l'échange avec les communautés de développeurs concernées.[4] Dans le secteur informatique, ces communautés sont typiquement internationales et utilisent l'anglais comme langue de travail. Limiter les licences à celles en langue allemande rendrait à tout le moins difficile, voire impossible, la coopération avec les communautés internationales, car un logiciel ainsi licencié ne serait guère utilisé au niveau international. Cela irait également à l'encontre du but visé par l'art. 9 LMETA.

De plus, les licences OSS établies au niveau international sont par définition valables dans le monde entier. Le législateur entendait manifestement permettre l'utilisation internationale des logiciels publiés. Limiter les publications OSS aux licences en langue allemande restreindrait fortement l'utilisabilité internationale des logiciels publiés.

Par ailleurs, l'anglais est solidement établi comme langue standard dans la sphère commerciale du secteur informatique, dont font également partie les publications OSS. L'utilisation de licences en anglais constitue donc une restriction raisonnable et proportionnée des droits des partenaires contractuels. De plus, les licences en anglais sont à la fois appropriées et nécessaires pour atteindre les objectifs visés.

6.2. Exclusion de responsabilité

La plupart des licences OSS contiennent des clauses d'exclusion de responsabilité. Est-ce suffisant ou des mesures supplémentaires sont-elles nécessaires? Qu'en est-il des logiciels défectueux? (en tenant compte de la loi sur la responsabilité)

Les licences applicables excluent la responsabilité pour négligence légère. **La responsabilité ne peut être exclue pour négligence grave ou acte intentionnel.** Les risques de responsabilité pratiques sont toutefois minimes.

Diffuser un logiciel modifié sous une licence incompatible avec celle du logiciel d'origine (par exemple sous une licence MIT sans copyleft alors que le code initialement utilisé était sous licence GPL avec copyleft) viole la licence d'origine. En l'absence de protection de la bonne foi en droit de la propriété intellectuelle, cela ne permet ni aux autres utilisateurs d'utiliser soudainement le logiciel d'origine sous la nouvelle licence, ni ne crée d'obligation exécutoire de proposer la nouvelle licence aux conditions de la licence d'origine. La seule conséquence est la violation de la licence d'origine, pouvant entraîner des prétentions en dommages-intérêts (analogie de licence) contre une autorité fédérale.

Ce risque peut toutefois être maîtrisé par un contrôle systématique conformément aux guides, comme cela est aujourd'hui la norme dans l'industrie du logiciel. Concrètement, la mise sous

4. FF 2022 804, p. 66

licence exige l'établissement d'une nomenclature de tous les composants logiciels préexistants intégrés dans le nouveau logiciel, qui doit être strictement respectée lors de la mise sous licence.

Lorsque des droits de tiers existent, l'art. 9, al. 1, LMETA n'exige de toute façon pas la publication du logiciel (ou éventuellement uniquement de ses parties nouvellement écrites; cela devrait être clarifié dans les guides).

Selon l'art. 11 de la loi sur la responsabilité, la responsabilité de la Confédération relève du droit civil lorsqu'elle agit comme sujet de droit civil. C'est le cas selon l'art. 9, al. 2, LMETA; une responsabilité supplémentaire fondée sur la loi sur la responsabilité est par conséquent exclue.

6.3. Logiciels permissifs

Si un logiciel repose sur une bibliothèque ou un composant de programme publié sous une licence permissive, le logiciel principal (sans la bibliothèque) peut-il être publié sous une licence non permissive?

Oui, cela ne pose pas de problème. Voir les indications données dans le guide Em002-3.

6.4. Dispositions de licence

Un logiciel peut-il être utilisé par plusieurs offices? Quelles dispositions de licence s'appliquent au sein de l'administration fédérale? L'administration fédérale est-elle considérée comme un groupe de sociétés? (Cela concerne les logiciels open source de tiers non modifiés.)

La diffusion de logiciels sous licences permissives ne pose généralement pas de problème.

Avec les licences copyleft, la question se pose toutefois de savoir si la diffusion au sein de l'administration fédérale centrale ou décentralisée déclenche des obligations de copyleft et risque ainsi de compromettre la confidentialité des développements propres à la Confédération. Lors de la remise à des institutions de l'administration fédérale décentralisée dotées de leur propre personnalité juridique, c'est effectivement le cas, alors que cela ne s'applique pas au sein de l'administration fédérale centrale, tous les départements agissant sous la même personnalité juridique.

Les sociétés d'un groupe sont juridiquement indépendantes mais restent sous le contrôle économique de la société mère. La doctrine juridique considère que la remise de code sous licence copyleft à une société du groupe déclenche des obligations de copyleft. Comme les sociétés d'un groupe reçoivent souvent des logiciels open source lors du développement de logiciels, elles ont besoin de leur propre droit d'utilisation; la remise à une société du groupe déclenche donc des obligations de copyleft.

En revanche, la remise au sein de l'administration fédérale centrale ne déclenche pas d'obligations de copyleft. Pour l'administration fédérale décentralisée, où existent des personnalités juridiques distinctes, ce qui précède s'applique: la remise déclenche des obligations de copyleft, et le logiciel doit être proposé sous forme de code source aux mêmes conditions de licence que le logiciel d'origine. Une instruction interdisant de remettre le logiciel à des tiers violerait la licence et pourrait également contrevenir à l'art. 9 LMETA.

6.5. Rapport entre la LTrans et la LMETA

Quel est le rapport entre la loi sur la transparence (LTrans) et la LMETA?

Selon l'art. 6 LTrans, les documents officiels doivent être remis. Est un document officiel au sens de l'art. 5, al. 1, LTrans toute information enregistrée sur un quelconque support, qui se trouve en possession d'une autorité et qui concerne l'accomplissement d'une tâche publique. Les logiciels open source relèvent en principe de la LTrans. La LTrans s'applique lorsqu'une personne demande à un département ou à un office le code source pour son propre usage. L'office doit divulguer le code source à la personne concernée, à moins que des exceptions au sens de l'art. 7 LTrans ne

s'appliquent. Si l'office refuse de divulguer le code source, la procédure LTrans s'applique. Le cas est différent lorsque quelqu'un exige que l'office publie le code source sur son site web. La procédure ne suit alors pas la LTrans mais la LMETA. Comme la LMETA ne prévoit elle-même aucune procédure, le droit administratif général doit suffire. Celui-ci offre soit la possibilité d'une décision, soit celle d'un acte matériel, soit celle d'une décision en constatation de l'office concerné. Cette décision peut ensuite être contestée devant le Tribunal administratif fédéral.

Si des obligations de divulgation pour des logiciels développés par une autorité fédérale sont concevables au sens de la LTrans (lorsque les conditions correspondantes sont remplies), il convient de noter que la divulgation d'un logiciel selon la LTrans n'accorde pas automatiquement une licence OSS. L'utilisation d'un logiciel divulgué selon la LTrans se limite donc aux buts fixés par la LTrans, tels que la consultation du code ou son exécution à des fins d'observation. La divulgation selon la LTrans ne comprend toutefois pas de licence pour une utilisation régulière du logiciel ni comme base pour le développement d'autres logiciels (détails dans T. Poledna/S. Schlauri/S. Schweizer, Rechtliche Voraussetzungen der Nutzung von Open-Source-Software in der öffentlichen Verwaltung, insbesondere des Kantons Bern [trad.: Conditions juridiques de l'utilisation de logiciels open source dans l'administration publique, en particulier dans le canton de Berne], Zurich 2017, https://carlgrossmann.com/?ddownload=11748, N 393 ss).

La décision de mettre un logiciel sous licence OSS, même s'il relève de la LTrans, continue de suivre la LMETA; il n'existe toujours pas de droit à l'octroi d'une licence.

Il est important de noter que les logiciels développés avant le 1er janvier 2024, bien que non soumis à la LMETA, doivent néanmoins être remis conformément à la LTrans.

6.6. OSS et protection des données

Les données personnelles des développeurs peuvent-elles être publiées sans autre? Quels aspects de la protection des données faut-il prendre en compte?

Du point de vue de la protection des données, la publication de données personnelles (p. ex. noms ou adresses électroniques) est problématique si les collaborateurs n'y ont pas consenti au préalable. En effet, la communication de données personnelles peut en principe être évitée par la pseudonymisation ou l'anonymisation des entrées du dépôt, ce qui serait requis selon le principe de la minimisation des données et de la proportionnalité (art. 6, al. 2, LPD).

Un tel consentement devrait toutefois être généralement facile à obtenir, les collaborateurs ayant souvent intérêt à montrer publiquement leurs compétences (ainsi Poledna/Schlauri/Schweizer, N 68, avec renvois). Le consentement peut aussi être implicite par l'utilisation des noms dans le dépôt, du moins en l'absence d'instruction contraire de l'employeur.

Il est néanmoins conseillé de régler dans les contrats la responsabilité du fournisseur pour les données personnelles de ses collaborateurs.

7. Généralités

Le fait qu'un logiciel open source soit utilisé à des fins commerciales ou non commerciales joue-t-il un rôle?

Les licences open source ne distinguent généralement pas entre utilisation commerciale et non commerciale. Un logiciel open source peut donc être utilisé à toutes fins, y compris pour des applications commerciales. Les fournisseurs commerciaux tentent souvent d'intégrer des composants open source dans des produits propriétaires. Cela n'est admissible que si les composants open source ne sont pas placés sous une licence à effet copyleft (voir aussi Em002-3 Guide des licences OSS).

Existe-t-il des restrictions quant aux organisations admises pour les communautés?

En fin de compte, chaque cas doit toujours être examiné individuellement: qui se trouve derrière, et quels engagements sont pris?

Tant qu’aucun argent ne circule et qu’aucun engagement formel n’est pris, la situation est relativement simple et sans problème.

Des affiliations à des associations sont possibles, tout comme une collaboration informelle.

Il est possible de déléguer des affiliations à eOperations ou à des organisations similaires.

Voir aussi Em002-4 Guide des communautés OSS, chapitre 3.1

Quel est le rapport entre la norme ISO 5230 et les guides?

La norme ISO 5230:2020 a été examinée et prise en compte lors de l’élaboration des guides. La liste ci-après présente les exigences de la norme et la manière dont elles ont été traitées dans les guides:

3.1 Programme foundation

- Les «Em002 Directives stratégiques pour les logiciels open source dans l’administration fédérale» et les stratégies référencées constituent le cadre. La mise en œuvre incombe aux départements et offices concernés.

3.2 Relevant tasks defined and supported

- Le guide «Em002-2 Instructions pour la publication de logiciels open source» et les listes de contrôle définissent des tâches concrètes pour la publication. Il n’existe actuellement pas d’organe central au sein de l’administration fédérale.

3.3 Open source content review and approval

- Le processus de publication est décrit dans l’aide «Em002-2 Instructions pour la publication de logiciels open source». Celle-ci renvoie également à l’existence d’une Software Bill of Materials (SBOM).
- L’aide «Em002-3 Guide des licences OSS» informe de manière complète sur la mise sous licence.

3.4 Compliance artifact creation and delivery

- Le processus de publication est décrit dans l’aide «Em002-2 Instructions pour la publication de logiciels open source». Celle-ci renvoie également à l’existence d’une Software Bill of Materials (SBOM).
- La mise en œuvre n’est pas décrite plus en détail et a lieu au sein des départements ou des offices.

3.5 Understanding open source community engagements

- L’aide «Em002-4 Guide des communautés OSS» décrit comment mettre en place et entretenir une communauté. Un concept doit être défini et mis en œuvre pour chaque projet.

3.6 Adherence to the specification requirements

- Le chapitre final énonce des exigences pour les organisations ou projets souhaitant confirmer explicitement le respect des exigences OpenChain.

La norme ISO esquisse d’autres domaines de responsabilité d’un Open Source Programme Office (OSPO) qui ne sont pas couverts par les guides.

8. Annexe

8.1. Abréviations

AO
Application owner

CCMP
Centre de compétence des marchés publics de la Confédération

ChF
Chancellerie fédérale

CLA
Contributor Licence Agreement

DCO
Developer Certificate of Origin

ANS
Administration numérique suisse (www.digital-public-services-switzerland.ch/strategy)

BP
Bénéficiaire de prestations

FP
Fournisseur de prestations

FSF
Free Software Foundation

HERMES
Handbuch der Elektronischen Rechenzentren des Bundes, une méthode de développement de systèmes (www.hermes.admin.ch)

IA
Intelligence artificielle

LMETA
Loi fédérale sur l’utilisation des moyens électroniques pour l’exécution des tâches des autorités

LTrans
Loi sur la transparence

OFCL
Office fédéral des constructions et de la logistique

OSI
: Open Source Initiative

OSPO
: Open Source Programme Office

OSS
: Logiciel open source

OSSD
: Développement de logiciels open source

UO
: Unité d’organisation (en règle générale un office fédéral)

USIC
: Unité de pilotage informatique de la Confédération (jusqu’en 2021)

SPDX
: Software Package Data Exchange

8.2. Glossaire

D’autres termes figurent dans TERMDAT, la banque de données terminologique de l’administration fédérale.

Domaine d’application
: Il s’agit des catégories de logiciels de bureautique (BA) telles que définies dans [A029] et d’autres classifications.

Branche
: Une branche de développement d’un logiciel open source.

Collaboration
: Le terme «collaboration» dérive du mot latin «collaborare», qui signifie «travailler ensemble».
Il décrit un mode de travail dans lequel plusieurs personnes ou équipes œuvrent ensemble à un objectif commun en y apportant leurs compétences et leurs ressources. L’échange mutuel, la transparence et le partage des connaissances sont au centre.
Selon le Gabler Wirtschaftslexikon, la collaboration désigne aussi la «coopération d’une entreprise avec ses clients et ses fournisseurs, en recourant aux technologies modernes de l’information pour intégrer les processus d’affaires internes et interentreprises».

Contribution
: Une contribution OSS est un apport au développement et à l’amélioration d’un logiciel open source (OSS). Cela peut prendre la forme de code, de documentation, de tests, de retours ou d’autres types de contributions. (Google AI)

Contributor Licence Agreement (CLA)
: Un Contributor Licence Agreement (CLA), également appelé Contributor Agreement, est un document qui fixe les conditions auxquelles de la propriété intellectuelle peut être apportée à un projet ou à une initiative. Il s’agit généralement d’un projet logiciel sous licence open source (Wikipédia).

Core developer
: Développeur d’un projet open source disposant de droits d’accès de committer au dépôt.

Analyse de marché

Une analyse de marché spécifique sert à identifier et à traiter les informations relatives au marché d’acquisition actuel et potentiel. L’analyse de marché a pour but d’identifier et de comprendre les structures du marché et, en particulier, la structure des fournisseurs au regard de toutes les caractéristiques déterminantes : situation des prix (élevés, bas, fluctuations) ; taille du marché ; répartition géographique des fournisseurs potentiels et des acteurs principaux, etc. (voir https://perimap.admin.ch/goto_perimap_file_38221_download.html).
Les sources possibles de logiciels OSS figurent dans Em002-1 Guide pratique OSS, chapitre 7.

OSS

Logiciel open source

Produit

Utilisé ici comme synonyme d'«application».

Le terme provient d’ITIL. Dans ITIL (Information Technology Infrastructure Library), un produit est une configuration de ressources conçue pour apporter de la valeur à un client ou à une organisation. Contrairement à un service, qui est typiquement un processus interactif et continu, un produit est une entité statique.

Les produits peuvent être des logiciels, du matériel, des données ou d’autres ressources fournies par l’organisation. (Google AI)

Standardisation des produits

Dans le présent document, la standardisation se réfère principalement à SD120, le service standard TIC pour la bureautique, ainsi qu’à d’autres standardisations du secteur TNI.

Subscription

Lors de l’acquisition d’un logiciel par abonnement, une subscription est conclue. Cela signifie que le logiciel n’est pas acheté.
Dans la plupart des cas, cela comprend non seulement l’autorisation d’utiliser le logiciel, mais aussi l’ensemble des prestations et du support pertinents.

Projet upstream

Upstream est un terme issu du développement logiciel distribué (souvent open source) et désigne la direction d’un correctif vers son origine (upstream), c’est-à-dire vers les développeurs ou mainteneurs d’origine du logiciel, ou vers le projet d’origine. Il peut aussi s’agir de bibliothèques logicielles.