Mode opératoire du mécanisme de confiance MAN

Version 1.13

7 février 2024

Récapitulatif des éditions

N° Version	Date de version	Nature de la modification	Auteur
0.1	22 mars 2022	Premier draft	Axel Tessier
1.0	25 avril 2022	Version validée	Axel Tessier
1.1	5 juillet 2022	- Utilisation de la notion d'opérateur signataire au lieu d'opérateur d'origine - Introduction des composants BPCO et GCO et suppression de la notion d'application MAN au profit de la plateforme MAN - Introduction de l'entête If-Modified-Since pour optimiser la récupération de la liste des certificats et CRL - §4, §5 : Inclusion des codes erreur SIP - §2.3 : remplacement d'appelant et destinataire par numéro appelant / numéro destinataire - §2.3 : Suppression de la liste des niveaux d'attestation pour un renvoi vers le document « MAN_Cas_Usages_Voix » - §2.5 : Modification du signataire de la CRL des certificats de l'autorité + introduction d'un backup de la CRL - §2.6.4 : Ajout contrainte sur date de début de validité d'un certificat non-test - §2.7.2.3 : Un certificat de test révoqué supprimé n'entraîne plus sa suppression de la CRL - §3.5.1.3 : Correction URL d'accès du certificat - §5.5.2.3 : Clarification du processus de validation des certificats CA - §8.7, §8.8 : modification des schémas afin d'avancer la réponse de confirmation à l'opérateur signataire de la création du certificat - §8.7.5 : Correction valeur de date de fin d'expiration du certificat - §8.7.6 : Remplacement des détails de la réponse de création de certificats directs par un renvoi aux guides de référence des APIs. - §9.7.3 : suppression dans le diagramme de la mention CRL - §13, §14 : ajout code erreur 503 + cas pour les codes erreur 403 et 404 - §13 : ajout contrainte sur la date de début de validité du certificat - §14.7 : ajout section accès certificat CA	Axel Tessier
1.2	28 septembre 2022	- §13 Mise à jour suite à la publication officielle de l'API Swagger GCO - Ajout propriété certificat *renewed_by*	Axel Tessier

		- Renommage propriétés du certificat *start_date / end_date* par *valid_from / valid_to* - Prise en compte de la liste des opérateurs signataires pour lesquels l'OPTS a un contrat - Ajout paramètres pour la procédure de renouvellement - Introduction du contexte de débrayage opérateur - §2.6.6: Limitation de l'algorithme à utiliser à ECDSA-256 - Réduction du paramètre since à 15 jours - Ajout document de référence Profil SIP 3.1 - Clarification de la valorisation des claims orig et dest du token PASSPORT	
1.2.1	30 septembre 2022	- Correction du format de la propriété info dans le champ Identity	Axel Tessier
1.3	18 janvier 2023	- Ajout dans la propriété KeyUsage du certificat racine CA de la valeur CrlSign - Ajout de précisions quant au contenu des certificats de l'autorité de certification - Modification des contrôles et codes erreurs retournés par le STI-VS pour se conformer à l'ATIS-1000082 - Remplacement des valeurs effectives de durée d'expiration des certificats, délais d'archivage et de suppression par une référence au code de procédures MAN fournissant ces valeurs - Simplification des sections §6 et §7 afin de faire référence au code de procédures MAN - Suppression du contenu des sections §8.5 et §15 au profit du code de procédures MAN - Suppression du principe de débrayage général en faveur de débrayages opérateur - §3 : Introduction de la procédure de débrayage STI-AS lors de l'émission des appels - §4, §5 : Introduction de la procédure de débrayage de vérification des appels en transit et en réception - §4, §5 : Prise en compte du débrayage STI-AS lors de la vérification des appels en transit et en réception - §14.5, §14.6 : renommage de la propriété « header » en « protected » - §2.5.1, §14.8.2 : le certificat responsable de la signature de la CRL de l'autorité de certification est maintenant le certificat	Axel Tessier
1.4	7 mars 2023	- §3.6.2 : Modification du format du token de débrayage - §4.5 : L'erreur associée au contrôle lié à la propriété alg doit être 437 et non 438 - §4.5, §5.5 : mention de la procédure de vérification de débrayage STI-AS	Axel Tessier

		- §6.2 : Correction de syntaxe - §8.7.2 : Fourniture de scripts d'exemple de génération de fichier CSR valide - §8.7.5, §8.8.8 : Mention de l'IHM pour la spécification des propriétés du certificat - §11.4.1 : Mention des informations à renseigner lors de la révocation d'un certificat - §11.4.4 : Correction de la réponse API retournée en cas de révocation. - §13.3.1, §13.4.1 : Modification de la propriété Requise à Oui pour valid_from - §13.3.1 : Suppression de la valeur par défaut des propriétés valid_from et valid_to	
1.5	30 mars 2023	- §3.5, §5.5 : Précisions du format Base 64 URL au lieu de Base 64 pour la génération du token PASSport - §6.3 : Suppression de la possibilité de modifier de manière autonome la liste des opérateurs liés à un OPTS - §8.7.2 : Simplification de la procédure de génération du fichier CSR - §10.3.2 : Précisions sur la date de début de validité d'un certificat créé dans le cas de renouvellement automatique - §10.4.3 : Précisions sur la date de début de validité d'un certificat créé dans le cas de renouvellement manuel	Axel Tessier
1.6	4 mai 2023	- §2.5.2, §2.5.3, §2.5.4 : Ajout de la propriété OU - §2.5.3 : Suppression de pathlen :0 pour l'extension x509 Basic Constraints des certificats intermédiaires - §2.6.1, §2.7.1.3, §11.3 : Nouveau statut « Invalidé » - §2.6.3 : Ajout contraintes pour les certificats de test - §2.6.3.1 : Ajout de diagramme dans le cas de suppression de certificat de tests avant expiration - §2.6.7 : Ajout de la propriété OU et d'un 2e CRL Issuer pour le certificat BPCO PA2 - §2.7.1.5, §2.7.2.3 : Suppression de cas code de procédures MAN + ajout cas invalidation - §3.5.1.3, §5.5.3.4 : Amélioration des procédures de génération/vérification de la signature du PASSporT - §6.3 : Retour arrière modifications version 1.5 + ajout liste de notification « Dépôt » - §8.7, §8.8, §13.3, 13.4, 13.11 : Suppression de la possibilité de renouvellement automatique pour les certificats de test - §9.2, §9.6.1, §9.7.1, §13.7 : Les certificats de test ne sont pas inclus dans la copie locale des certificats - §10.3, §10.4 : Ajout prérequis de renouvellement - §13.1.3 : Ajout codes retour 201, 202, 204 - §14.1.3 : Ajout code retour 400	Axel Tessier

1.7	16 juin 2023	- Correction du format des numéros de téléphone inclus dans les exemples pour se conformer aux règles techniques - §2.6.6: Clarification que seul l'algorithme autorisé pour la signature d'appels est ES256 - §3.5.1.2 : Ajout d'un exemple avec format de numéro court - §3.5.1.2, §5.5.2.4 : Précision du fuseau horaire pour le champ iat du token PASSporT - §5.5.2.3 : Ajout mention de récupération des informations de l'entête pour la génération de trace - $8.8.3 : Ajout d'une contrainte sur la période autorisée pour finaliser un certificat indirect - §10.4.1, §10.5.2, §13.3.1, §13.4.1, §13.11.1 Précision de l'utilisation de la propriété renewal_after - §14.5.2, §14.6.2 : Correction du format de la propriété version	Axel Tessier
1.8	27 juin 2023	- §2.6.3, §9.2 Modification de la logique liée à la copie locale pour ne plus embarquer les certificats de test - §2.5.2, §2.5.3, §2.5.4, §14.5.2, §14.6.2 Informations pour obtenir les certificats de l'autorité de certification - §2.5.5 Contenu de la CRL des certificats de l'autorité de certification - §2.7.2.2 Contenu de la CRL des certificats opérateurs - §3.5.1.3 Ajout d'exemples de commandes shell pour générer l'empreinte STI-AS du token PASSPorT - §3.6.2 Informations sur le mode d'obtention du token de débrayage - §9.6.3 Suppression d'une confusion potentielle quant au contenu des réponses des APIS BPCO /ca et /ca/certs - §13.3.1, §13.5.1 Ajout d'une référence à la section de création du CSR	Axel Tessier
1.9	2 août 2023	- §2.6.3, §10.5.2, §13.3.1, §13.4.1, §13.11.1 Suppression du renouvellement automatique pour les certificats de test - §2.8.6 Ajout précisions sur la valorisation du champ FROM et l'utilisation possible du PAI - §8.8 Ajout de contraintes à la finalisation de certificats indirects	Axel Tessier
1.10	27 septembre 2023	- §5.5.1 Clarification des contrôles pour les codes erreur 403 et 438 - §5.5.2.3, §5.5.2.4 Clarification de l'écart attendu pour le code erreur 403	Axel Tessier
1.11	6 novembre 2023	- §2.3.3, §5.5.1 Précisions sur les libellés associés aux codes réponse SIP en distinguant les libellés par défaut et les spécifiques employés par les opérateurs	Axel Tessier

		- §2.8.9 Nouvelle section indiquant l'utilisation du code réponse SIP 400 pour le modèle français - §4.5.1.1, §5.5.2.1 Précisions sur les contrôles attendus pour l'entête P-Identity-Bypass - §5.5.2.3 Préconisation de valeurs spécifiques à utiliser dans le cas du code réponse SIP 400 retourné	
1.12	14 décembre 2023	- §2.7.2.2, §2.7.2.4 Information sur le renouvellement automatique de la CRL - §9.6.1, §9.7.1, §13.7.2 Précisions sur le format attendu pour le serial number dans la copie locale - §14.3.1, §14.7.1 Précisions sur le format attendu pour le serial number dans l'URL du certificat de la BPCO	Axel Tessier
1.13	7 février 2024	- §8.7.2, §8.7.4, §8.8.5, §8.8.7 Précision sur l'interdiction du champ Serial Number dans le CSR - §8.8.3 Information sur la suppression automatique des demandes de certificats indirects non finalisés au bout de 3 mois - §10.2 Nouvelle contrainte pour la fonctionnalité de renouvellement automatique des certificats indirects concernant la validité du contrat OPTS - §13.3.1, §13.4.1 Suppression de la valeur par défaut de l'option *renewal_after*	Axel Tessier

Documents de référence

Titre	Version
Profil SIP 3.2 : IP interconnection Interface specification based on SIP/SDP	Version 3.2 d'août 2023
Plan Programme MAN	Version 1.3 du 5 juillet 2022
Glossaire MAN	Version 1.3 du 5 juillet 2022
Guides de référence des APIs de la plateforme MAN	Version 1.6.0 du 14 décembre 2023
Règles techniques MAN	Version 1.4 du 15 novembre 2023
MAN_Cas_Usages_Voix	Version 1.1.1 du 26 octobre 2022
MAN_Cas_Usages_Messages	Version 1.0 du 5 juillet 2022
Mode opératoire des incidents, signalements et métriques du MAN	Version 1.12 du 7 février 2024
Code de procédures MAN	Version 1.7 du 7 février 2024

Table des matières

1 Introduction

1.1 Contexte - Le plan programme MAN

Dans le cadre des dispositions introduites par la loi n° 2020-901 du 24 juillet 2020 visant à encadrer le démarchage téléphonique et à lutter contre les appels frauduleux, les opérateurs sont tenus de s'assurer que, lorsque leurs clients utilisateurs finals utilisent un numéro issu du plan de numérotation établi par l'ARCEP comme identifiant d'appelant pour les appels et messages qu'ils émettent, ces utilisateurs finals sont bien affectataires dudit numéro ou que l'affectataire dudit numéro a préalablement donné son accord pour cette utilisation. Les opérateurs sont tenus de veiller à l'authenticité des numéros issus du plan de numérotation établi par l'ARCEP lorsqu'ils sont utilisés comme identifiant d'appelant pour les appels et messages reçus par leurs clients utilisateurs finals.

En vue de cette obligation, les opérateurs ont établi et validé la mise en œuvre d'un dispositif sectoriel nommé **programme MAN** (Mécanismes d'Authentification des Numéros), composé :

- d'un mécanisme d'authentification des numéros lui-même constitué de :
  - D'un **mécanisme de confiance** : authentifier l'opérateur d'origine et garantir l'intégrité de l'information,
  - De traitements des cas d'usage d'appels : traitement et attestation des appels
- de compléments pour lutter contre les appels et messages avec modifications frauduleuses ou erronées des numéros appelants ou d'émetteur

1.2 Objectif du document

Le but de ce document est de fournir à l'ensemble des acteurs concernés la compréhension du mécanisme de confiance mis en place dans le cadre du programme MAN (Mécanismes d'Authentification des Numéros) ; il décrit les processus attendus pour l'implémentation de ce mécanisme et sa gestion dans le temps par les opérateurs.

Ce document couvre les aspects suivants :

- Le principe du mécanisme de confiance
- Les acteurs interagissant dans ce mécanisme et leurs responsabilités
- Les composants et l'architecture mis en place
- Les communications entre les composants
- Les processus à implémenter par les opérateurs dans le cadre du mécanisme de confiance
- Les fonctions disponibles pour la mise en place de ces processus
- Les cas d'usage et les cas particuliers à prendre en compte

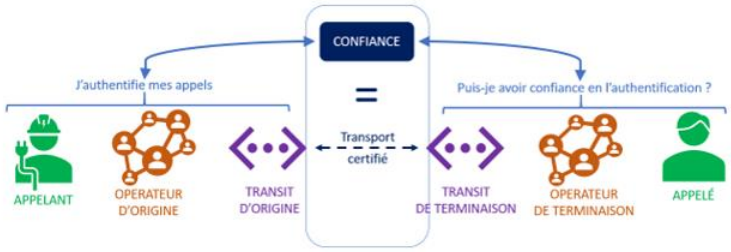
2 Mécanisme de confiance

2.1 Introduction

La problématique autour du démarchage téléphonique et des appels frauduleux touchant l'ensemble des pays, différentes normes et spécifications ont émergé en Amérique du Nord depuis 2017 afin de définir une solution STIR SHAKEN répondant à ce problème, solution mise en place par l'ensemble des opérateurs américains et canadiens depuis fin 2021.

Dans un but d'interopérabilité avec les opérateurs internationaux, le programme MAN s'appuie sur ces standards tout en les adaptant afin de créer un modèle français, appelé **mécanisme de confiance**.

Le mécanisme de confiance définit les règles de fonctionnement des certificats, le processus d'inscription à la communauté MAN, et les principes de vérification et rejet des appels.



Le but de ce document n'est pas de réexpliquer en détail les normes existantes mais de détailler le fonctionnement du modèle français, les acteurs prenant partie et les procédures à mettre en place par ces derniers.

2.2 Standards utilisés

Le tableau ci-dessous fournit la liste des standards RFC et des spécifications ATIS sur lesquels s'appuient les mécanismes de confiance américain et français.

Norme / Spécification	Organisme	Intitulé
RFC 8224	IETF	Authenticated Identity Management in the Session Initiation Protocol (SIP)
RFC 8225	IETF	PASSporT: Personal Assertion Token
RFC 8226	IETF	Secure Telephone Identity Credentials: Certificates
RFC 8588	IETF	Personal Assertion Token (PaSSporT) Extension for Signature-based Handling of Asserted information using toKENs (SHAKEN)
ATIS-1000074.v002	ATIS	Signature-based Handling of Asserted information using toKENs (SHAKEN)
ATIS-1000080.v004	ATIS	SHAKEN: Governance Model and Certificate Management
ATIS-1000082	ATIS	Technical Report on SHAKEN APIs for a Centralized Signing and Signature Validation Server
ATIS-1000084	ATIS	Technical Report on Operational and Management Considerations for SHAKEN STI Certification Authorities and Policy Administrators

Le mécanisme de confiance est conforme en tout point avec les standards RFC, mais prend plus de liberté quant aux spécifications ATIS afin de fournir une solution plus adaptée aux besoins du modèle français. La liste des différences d'implémentation de la solution STIR/SHAKEN entre les modèles français et américain est ainsi explicitée en section 2.8.

2.3 Solution STIR/SHAKEN

Le **mécanisme de confiance** retenu pour la France embarque les éléments de la solution STIR/SHAKEN, couvrant les besoins suivants :

- Authentification de l'appelant et de son numéro
- Signature des appels par l'opérateur d'origine/signataire (§2.4.3.1)
- Vérification de la signature des appels par l'opérateur de terminaison
- Coupure des appels dans le cas d'échec de leur vérification

Cette solution se base sur l'utilisation de certificats – appelés certificats opérateur (§2.6). C'est le socle du mécanisme d'authentification qui permet de signer et vérifier les appels SIP, en faisant confiance aux certificats opérateurs délivrés par l'autorité de confiance, ainsi que de faire circuler l'attestation SHAKEN dans les échanges SIP entre opérateurs interconnectés.

Les échanges utilisant des protocoles non-SIP ne sont pas concernés par ce document.

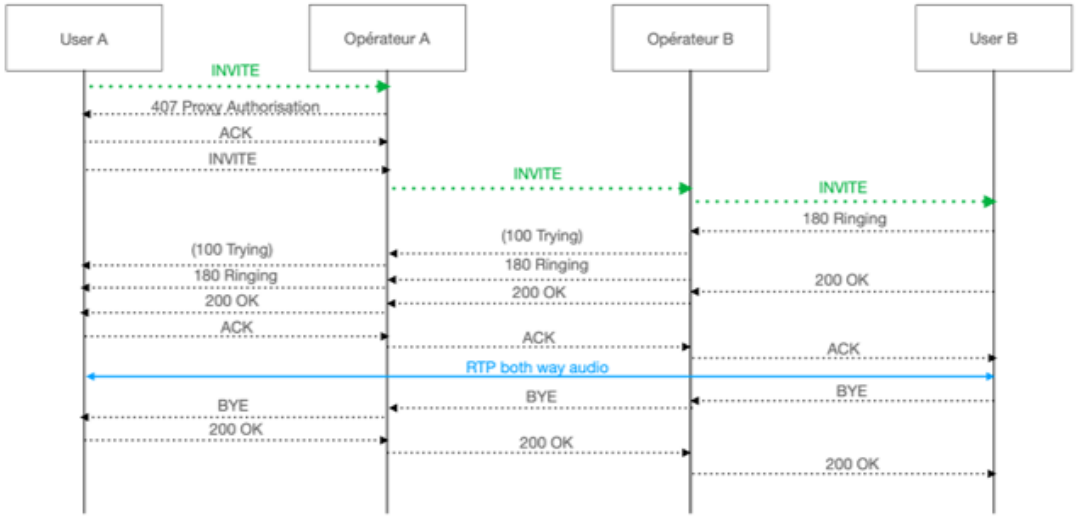
2.3.1 Protocole STIR pour la signature des appels

Les standards RFC **8224** (*Authenticated Identity Management in the Session Initiation Protocol (SIP)*), **8225** (*PASSporT: Personal Assertion Token*) et **8226** (*Secure Telephone Identity Credentials: Certificates*) décrivent le principe de signature d'un appel téléphonique SIP par l'intermédiaire de certificats, permettant l'authentification de l'opérateur d'origine/signataire et la vérification à la réception de l'appel de cette signature. Ce système est appelé protocole **STIR** (Secure Telephone Identity Revisited).

Comme indiqué par la RFC **8224**, la signature s'effectue par l'introduction d'un entête **Identity** au sein du message SIP **INVITE** envoyé lors de l'établissement de l'appel.

Le format de cet entête est décrit par la RFC **8225**, et prend la forme d'un JSON Web Token (RFC 7519) appelé **PASSport** (Personal Assertion Token), dont le header et le payload sont des objets JSON normalisés permettant de fournir **un numéro appelant et le numéro du destinataire utilisés pour l'établissement de l'appel** ainsi que la date d'émission. Ce token est signé avec un certificat X509 dont les détails sont fournis par la RFC **8226**.

Note : par défaut, le numéro appelant utilisé pour coder le JWT est celui qui est présenté à l'utilisateur appelé.



Le format final de l'entête **Identity** au sein du message INVITE est le suivant, une fois signé et haché :

```
Identity: eyJhbGciOiJFUzI1NiIsInBwdCI6InNoYWtlbiIsInR5cCI6InBhc3Nwb3J0IiwieDV1Ijoia
HR0cHM6Ly9kb21haW4tYnBjby9jZXJ0cy9jb2RlLWFwbmYvc24tY2VydGlmaWNhdGUuY2VyIn0.eyJhdHRl
c3QiOiJBIiwiZGVzdCI6eyJ0biI6WyIzMzgwMTAyMDMwNCJdfSwiaWF0IjoxNjg2ODMwNjg5LCJvcmlnIjp
7InRuIjoiMzMxMjM0NTY3ODkifSwib3JpZ2lkIjoiODIxZTg0ZmMtMmZhZi00YzJmLWFkMGItZjg3NzVlNz
M5YTdhIn0.1RO6mLLPsyET9X6UCGABTeBacw9OkuEi97EYEcjHZS_4OeFu4hxFcj9hbclsfnCwXYlNF0ZAl
GpUGv82U9cKMA;info=<https://domain-bpco/certs/code-apnf/sn-
certificate.cer>;ppt=shaken;alg=ES256
```

2.3.2 Extension SHAKEN du token PASSport pour attestation

La section 5.2.3 de la spécification **ATIS-100074** introduit une extension de type **SHAKEN** au token **PASSport** pour véhiculer le niveau d'attestation du numéro appelant tel que défini par l'opérateur d'origine/signataire, par l'intermédiaire d'une nouvelle propriété *attest* pouvant prendre les valeurs A, B ou C telles que définies dans le document « *MAN_Regles techniques* ».

Le format attendu pour cette extension SHAKEN a été ratifié par le standard RFC **8226**.

PASSport header avec extension ppt : shaken

```
{
  "alg": "ES256",
  "ppt": "shaken",
  "typ": "passport",
  "x5u": "https://domain-bpco/certs/code-apnf/sn-certificate.cer"
}
```

PASSport payload avec propriétés attest et origid

```
{
  "attest": "A",
  "dest": {"tn":["33801020304"]},
  "iat": 1686830689,
  "orig": {"tn":"33123456789"},
```

```
  "origid": "821e84fc-2faf-4c2f-ad0b-f8775e739a7a"
}
```

Pour des raisons de lisibilité, des retours à la ligne et des espaces ont été ajoutés aux exemples.

2.3.3 Vérification et coupure des appels

Dans le cas où la vérification d'un appel échoue et que celui-ci doit être coupé, et conformément à la spécification **ATIS-1000074**, section 5.3.2, l'opérateur en charge de la vérification doit répondre à la requête SIP INVITE avec un code réponse SIP correspondant au contrôle effectué. Ce code réponse doit être accompagné d'un libellé dont la section §5.5.1 fournit les valeurs par défaut, mais les opérateurs sont libres d'utiliser toute autre valeur.

Appels cassables

Le modèle français prévoit aussi la notion d'appel cassable, où un appel devant être cassé est néanmoins transmis. Les cas particuliers pour lesquels un appel doit être considéré comme cassable et non cassé sont les suivants :

- L'appel correspond à un appel d'urgence
- Le mécanisme MAN est en phase de rodage
- Le mécanisme de débrayage de l'opérateur émetteur est déclenché (cf. §3.6)

Dans ce cas, l'opérateur en charge de la vérification répond à la requête INVITE avec un code 200, mais a la possibilité d'inclure au sein d'un entête *Reason* de la réponse le code et le libellé de l'erreur correspondant au contrôle tel que fourni par l'opérateur :

```
Reason: SIP; cause=436; text="Bad Identity Info"
```

2.4 Architecture du modèle français

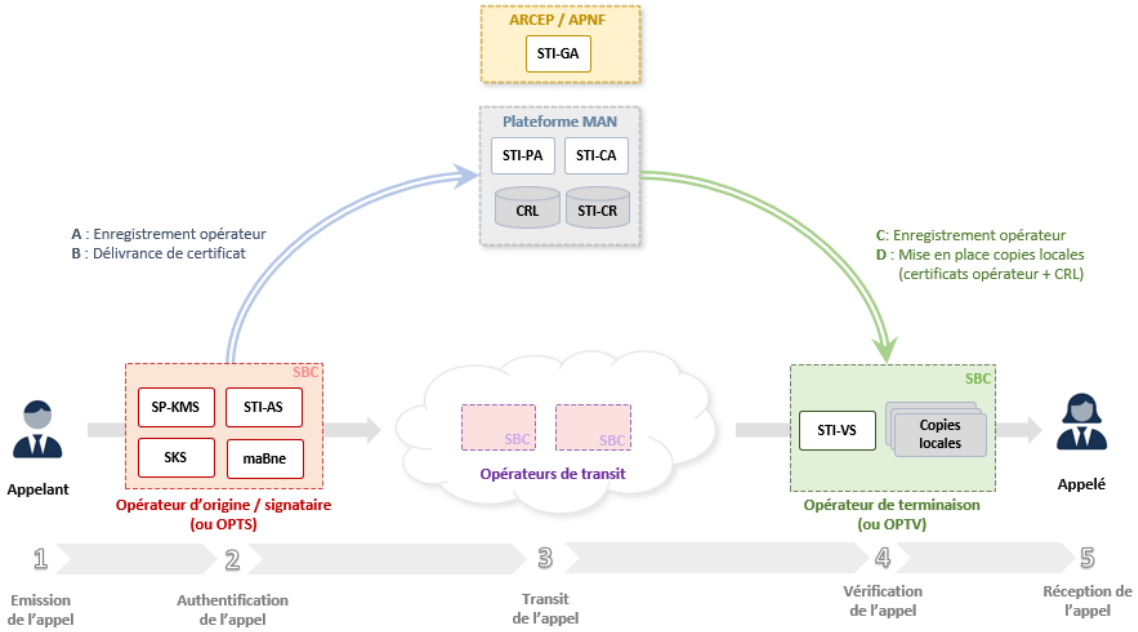
Le modèle français reprend l'architecture logique définie par la spécification ATIS **100080**, mais en simplifie la gestion en fonctionnant avec une autorité de certification **STI-CA** unique, permettant la centralisation des certificats délivrés aux opérateurs français au sein d'un seul **STI-CR** géré par cette entité.

Le **STI-PA** n'ayant plus à gérer de multiples **STI-CA**, sa mission se réduit à l'application des règles énoncées par le **STI-GA** pour l'admission des opérateurs à la communauté STIR. Pour des soucis de simplification, il est par conséquent décidé de regrouper les responsabilités de STI-PA et STI-CA au sein d'une seule et même entité, en charge d'une plateforme logicielle regroupant l'ensemble des fonctionnalités et appelée **plateforme MAN (**§2.4.2**)**.

D'autres acteurs s'adjoignent enfin à cette plateforme afin de couvrir les autres aspects du mécanisme de confiance :

- Une autorité de gouvernance (§2.4.1), prenant la responsabilité du **STI-GA**

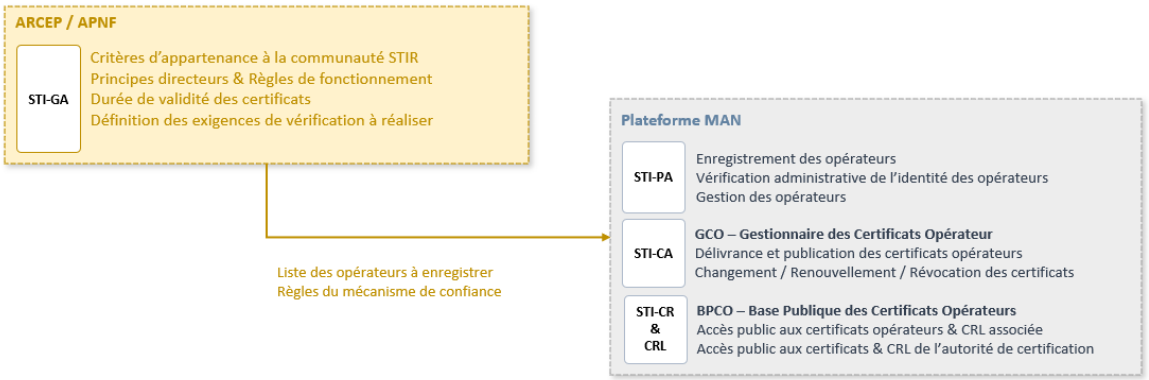
- Les opérateurs en charge de l'acheminement des appels (§2.4.3)



2.4.1 Autorité de Gouvernance

L'autorité de gouvernance (**STI-GA)** du mécanisme de confiance est représentée par l'APNF et l'ARCEP. Le rôle de ces entités est de définir les politiques et règles de gestion du mécanisme de confiance en France, incluant :

- la rédaction des règles de fonctionnement de l'entité STI-CA / STI-PA
- les règles de gestion des certificats (durée de validité, cycle de vie)
- la définition des critères d'appartenance d'un opérateur à la communauté STIR
- la fourniture à la plateforme MAN de la liste des opérateurs approuvés
- la fourniture à la plateforme MAN de la liste d'opérateurs à révoquer



2.4.2 Plateforme MAN

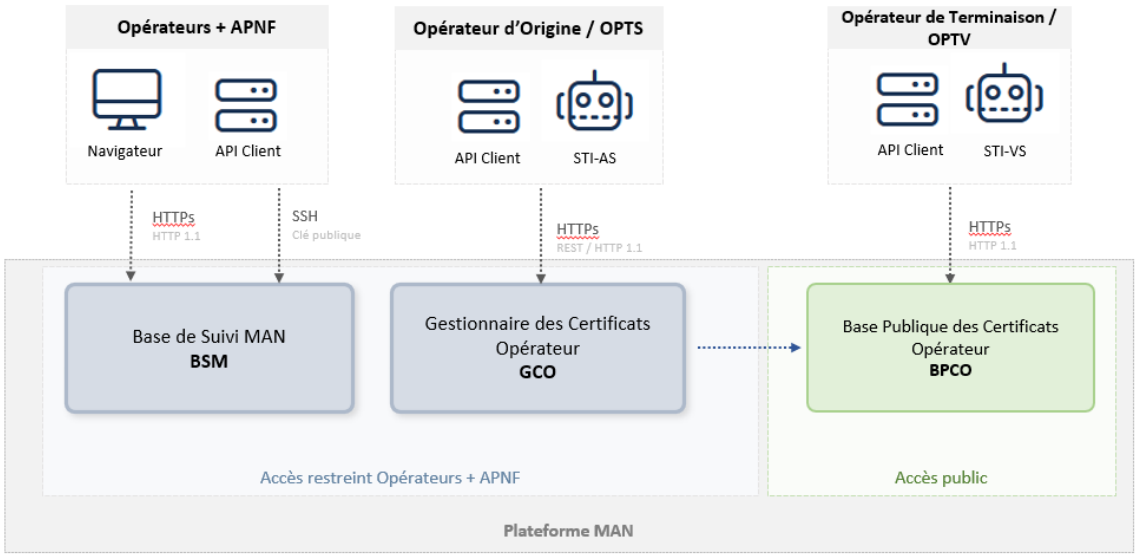
Pilotée par l'entité unique STI-CA/STI-PA, la plateforme MAN représente le socle logique et technique du **plan programme MAN**, fournissant l'ensemble des fonctionnalités prévues par celui-ci.

Cette plateforme a pour vocation d'être utilisée directement ou indirectement, par la totalité des opérateurs télécoms opérant sur le territoire français et procédant à l'acheminement des appels voix et des messages.

Les composants de la plateforme en charge de la mise en œuvre du mécanisme de confiance sont :

- Le Gestionnaire des Certificats Opérateur (**GCO**), en charge de l'ensemble des processus métier liés à la délivrance, publication et gestion des certificats opérateurs (§2.4.2.2)
- La Base Publique des Certificats Opérateur (**BPCO**), regroupant les certificats et données (§2.4.2.3, §2.7)
- L'autorité de certification en charge de l'infrastructure à clé publique utilisée par le GCO pour la délivrance des certificats opérateur (§2.5)

La plateforme fournit de plus un autre composant principal, la Base de Suivi du MAN (BSM), utilisée pour la remontée des traces, incidents, signalements et métriques des opérateurs liés au dispositif MAN. Ce module n'est pas abordé au sein de ce document. Il convient de se référer au document *« Mode opératoire des incidents, signalements et métriques du MAN »* qui lui est dédié.



2.4.2.1 Responsabilités dans le cadre du mécanisme de confiance

La plateforme MAN met à disposition des opérateurs l'ensemble des fonctionnalités nécessaires permettant aux opérateurs signataires de se voir délivrer des certificats pour signer leurs appels, et aux opérateurs de terminaison de vérifier ceux-ci :

- Enregistrement des opérateurs (§6)
- Délivrance des certificats aux opérateurs (§8)
- Gestion du cycle de vie des certificats :
    - Renouvellement des certificats (§10)

    - Révocation des certificats (§11)
    - Suppression des certificats (§12)
- Mise en place et maintien de la BPCO, comprenant :
    - La base publique des certificats opérateurs (§2.7.1)
    - La liste publique de révocation des certificats opérateurs (§2.7.2)
    - Les certificats et informations liées à l'autorité de certification en charge de délivrer les certificats (§2.5.6)
- Fonctionnalités permettant la mise en place de copies locales opérateur (§9)

2.4.2.2 Gestionnaire des Certificats Opérateur (GCO)

Le Gestionnaire des Certificats Opérateur est le composant logiciel de la plateforme MAN en charge pour les opérateurs des opérations liées à la délivrance et gestion de leurs certificats.

Accessible par l'intermédiaire d'une IHM et d'APIs dédiées (§13), cette solution propose les fonctionnalités suivantes :

- Délivrance de certificats opérateurs
- Publication des certificats auprès de la BPCO
- Renouvellement manuel ou automatique des certificats
- Révocation des certificats
- Publication et mise à jour des listes de révocation des certificats opérateur

Cette solution s'appuie sur une infrastructure à clé publique nommé autorité de certification (§2.5).

2.4.2.3 Base Publique des Certificats Opérateur (BPCO)

En accès public, elle permet de récupérer les certificats délivrés aux opérateurs par la plateforme MAN, la liste de révocation de ces certificats, ainsi que les certificats et CRL de l'autorité de certification signant les certificats. Ces informations publiques, accessibles via des URLs HTTPs, permettent d'assurer la fonctionnalité de vérification des signatures d'appels (§5). Une description détaillée ces composants de la base est disponible en section §2.7.

La BPCO est accessible indépendamment des autres modules de la plateforme MAN ; le domaine hébergeant les URLs de la BPCO est ainsi différent de celui utilisé pour les autres composants de la plateforme MAN.

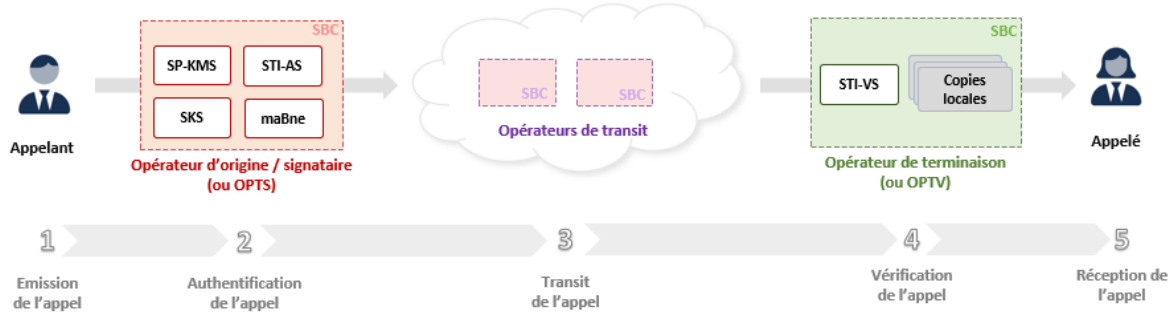
2.4.3 Rôles des Opérateurs

Lors de l'acheminement d'un appel SIP, plusieurs types d'opérateurs peuvent être distingués, chacun ayant un rôle distinct dans le cadre du mécanisme de confiance :

- Opérateur signataire et opérateur d'origine
- Opérateur de transit
- Opérateur de terminaison

A ces derniers s'ajoutent deux nouveaux rôles, représentant des opérateurs techniques mandatés par un des types d'opérateurs précédent afin d'effectuer le traitement des appels à leur place :

- L'OPTS : opérateur technique de signature (mandaté par l'opérateur signataire)
- L'OPTV : opérateur technique de vérification (mandaté par l'opérateur de terminaison)



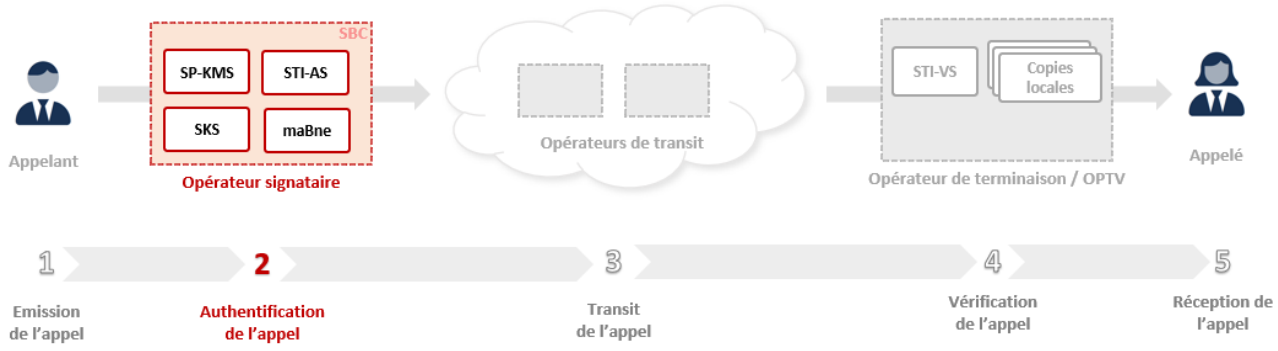
Cette section présente ces différents rôles et leur participation au sein du mécanisme de confiance.

2.4.3.1 Opérateur signataire et opérateur d'origine

L'opérateur d'origine est l'opérateur qui détient le contrat de service avec le client final émetteur de l'appel. Il collecte physiquement les appels émis par le client final. Lorsqu'un appel est émis en SIP sur le réseau public, l'opérateur signataire correspond à l'opérateur d'origine, **sauf pour les cas de mises à disposition et certains cas d'appels pour les MVNO**.

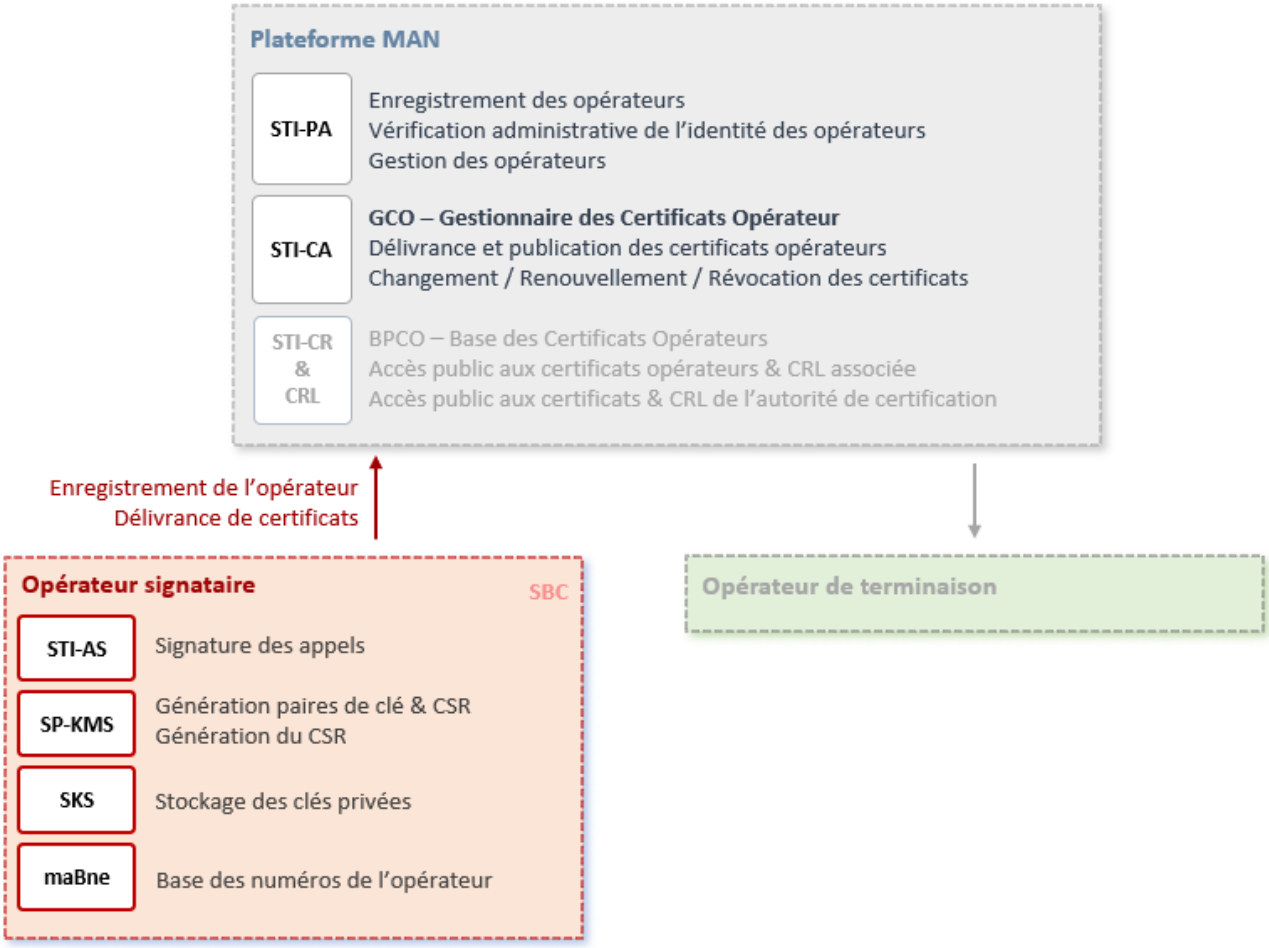
L'opérateur signataire est l'opérateur détenteur du certificat utilisé pour la signature de l'appel et qui est responsable des informations véhiculées dans le cadre du MAN (dont le niveau d'attestation shaken).

Le reste de ce document fera ainsi seulement mention de l'opérateur signataire en lieu et place d'opérateur d'origine afin de se restreindre aux acteurs effectivement en charge de la signature.



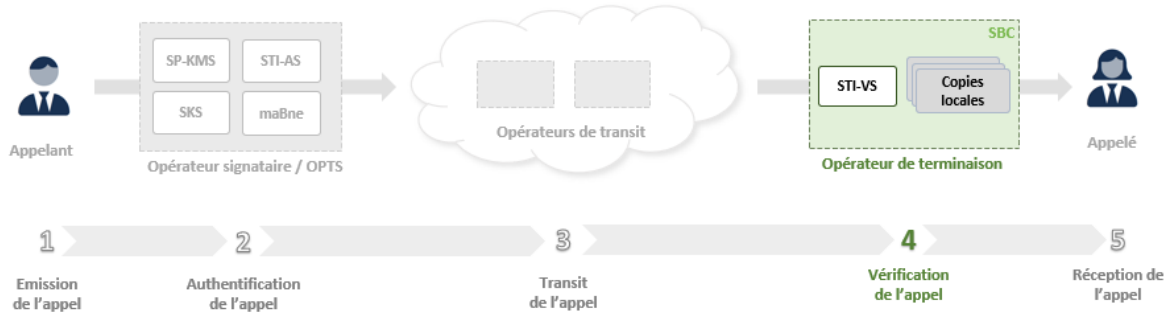
L'infrastructure de l'opérateur signataire doit comprendre les éléments suivants :

- L'équipement réseau à l'interconnexion VoIP, supportant les messages SIP INVITE entrants et sortants. Il appelle le **STI-AS** pour la signature des appels sortants. Le terme « **SBC** » sera utilisé dans la suite de ce document pour identifier cet équipement.
- Le **STI-AS** de l'opérateur, en charge de la construction et de l'insertion de l'entête *Identity* dans le message SIP INVITE, y compris la génération du token **PASSport** avec extension **SHAKEN**. Ce composant doit ainsi disposer d'une clé privée pour signer les appels et de l'URL publique du certificat associé afin de l'inclure dans l'entête Identity.
- Le **SP-KMS** de l'opérateur, en charge de la génération des clés et de la procédure de génération des certificats. Il récupère aussi l'URL publique du certificat qui lui aura été délivré par l'autorité de certification afin de la fournir au **STI-AS**.
- Le **SKS** de l'opérateur, stockant la clé privée utilisée pour signer l'appel.
- **maBNE**, base de données des numéros de l'opérateur.



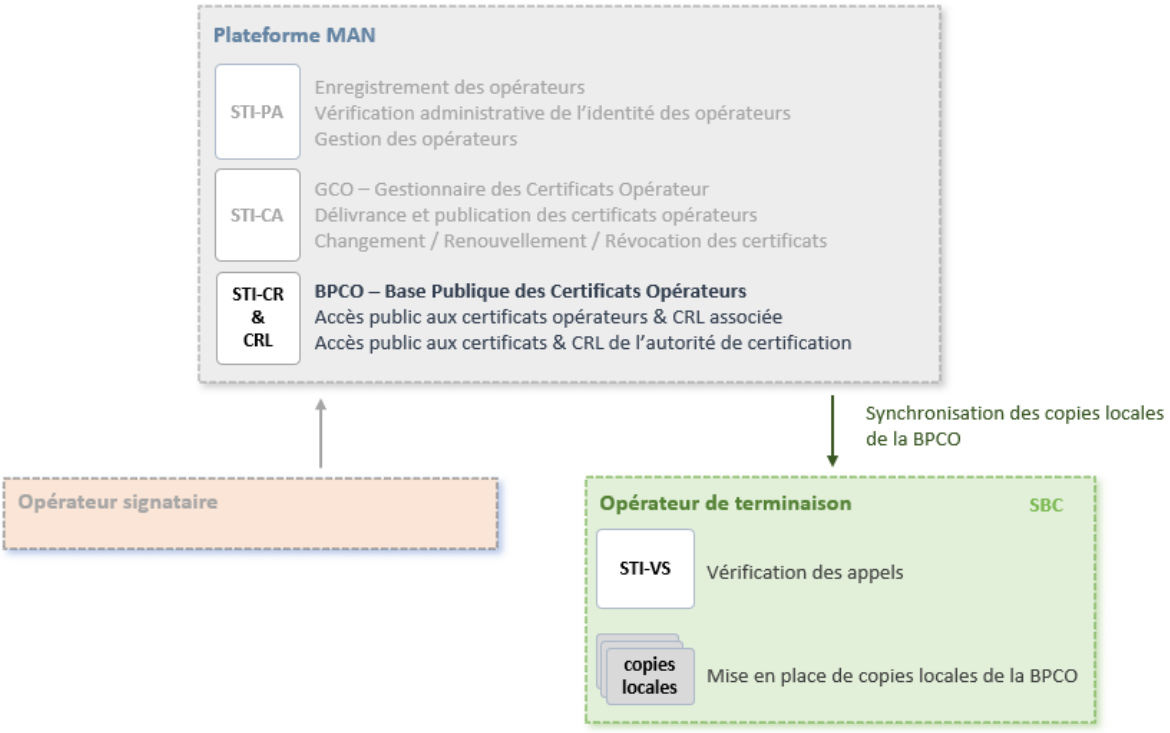
2.4.3.2 Opérateur de terminaison

Opérateur à la réception de l'appel SIP, il est en charge de vérifier la signature de l'appel et de casser l'appel en cas d'échec. La procédure à mettre en place par cet opérateur est explicitée en section 5.



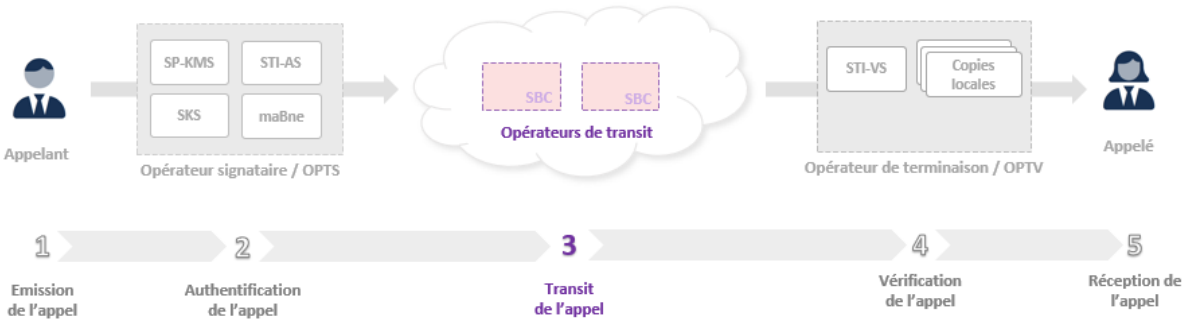
L'infrastructure de l'opérateur de terminaison doit comprendre les éléments suivants :

- L'équipement réseau à l'interconnexion VoIP, supportant les messages SIP INVITE entrants et sortants. Il appelle le **STI-VS** pour tout ou partie de la vérification de l'appel. Le terme « **SBC** » sera utilisé dans la suite de ce document pour identifier cet équipement.
- Le **STI-VS (Verification Service)**, gérant pour l'opérateur de terminaison la vérification de la présence de l'entête *Identity* dans le message SIP INVITE et de valider la signature incluse dans cet entête. L'opérateur de terminaison doit pouvoir récupérer les certificats de la BPCO, afin d'en extraire la clé publique pour vérifier les signatures. Afin d'optimiser les délais de traitement, il est demandé aux opérateurs de terminaison de mettre en place des copies locales afin de disposer directement des certificats (§9).
- Les **copies locales** des certificats opérateurs, des CRL et des données du STI-CA, afin d'optimiser les délais de traitement et de pallier à tout problème d'indisponibilité de la plateforme. Ces copies locales peuvent être assimilées à un « STI-CR local » utilisé par le **STI-VS.**



2.4.3.3 Opérateur de transit

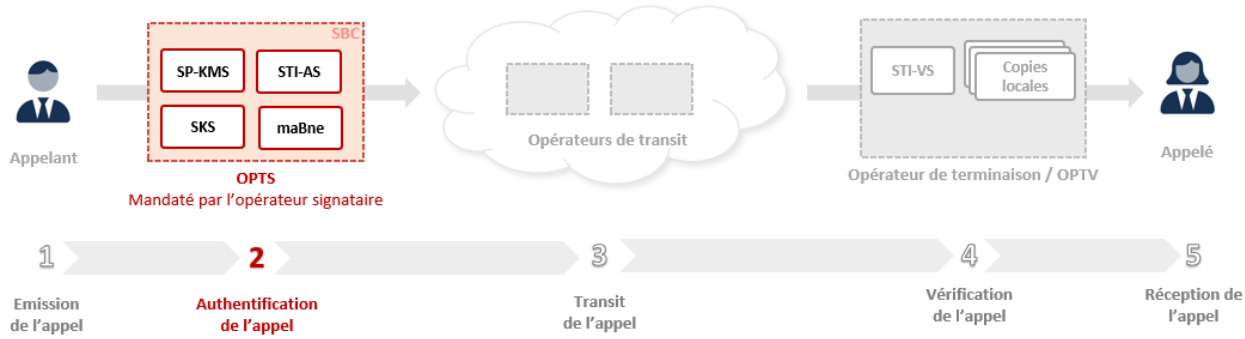
L'opérateur de transit voit l'appel SIP transiter par son réseau. Sa responsabilité est limitée à la vérification de la présence de la signature de l'appel et au format de la signature de l'appel et doit casser l'appel si celle-ci est absente ou présente un format invalide. La procédure attendue pour cet opérateur est décrite en section §4.



2.4.3.4 OPTS – OPérateur Technique de Signature

Le mécanisme de confiance permet à un opérateur signataire de mandater l'opérateur qui émet vers le réseau public ses appels pour les signer pour son compte. Ce dernier est dit « OPérateur Technique de Signature (OPTS) ».

L'OPTS se substitue par conséquent à l'opérateur signataire dans la phase d'émission de l'appel et devient responsable de la procédure à suivre (§3). Le certificat généré pour un opérateur signataire mandatant un OPTS par la plateforme MAN est appelé certificat indirect (§2.6.2) et la procédure liée à sa délivrance est sensiblement différente que pour un certificat direct, car nécessitant l'intervention de l'opérateur signataire et de l'OPTS.



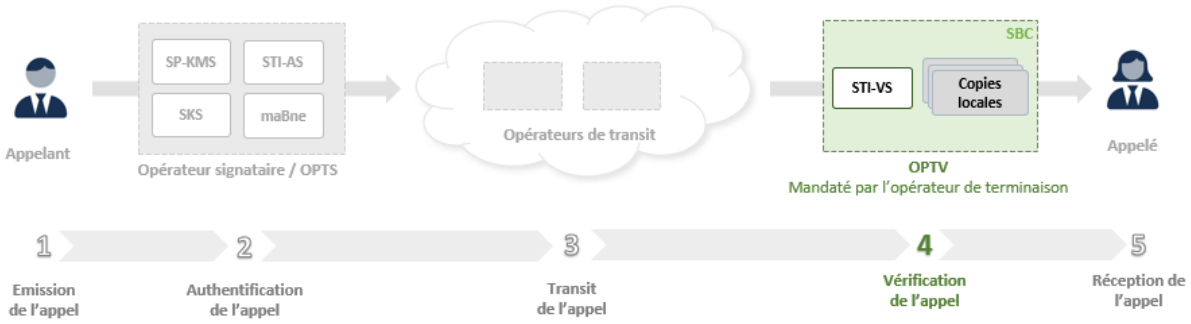
Les règles suivantes s'appliquent :

- Un OPTS est avant tout un opérateur et peut signer les appels dont il est opérateur signataire avec le certificat direct qui lui aura été délivré par la plateforme MAN
- Un OPTS peut être mandaté indépendamment par plusieurs opérateurs signataires en même temps. Un certificat indirect distinct est délivré pour chaque opérateur signataire
- Un OPTS ne peut pas mandater un autre OPTS pour la signature des appels d'un opérateur signataire dont il a le mandat
- Un opérateur signataire peut avoir plusieurs OPTS.
- L'opérateur signataire reste responsable de la signature des appels émis au vue de la communauté des opérateurs STIR.

L'infrastructure requise pour l'OPTS est par conséquent identique à celle de l'opérateur signataire (§2.4.3.1).

2.4.3.5 OPTV – OPérateur Technique de Vérification

Dans le cadre du mécanisme de confiance, un OPérateur Technique de Vérification (OPTV) est un opérateur mandaté par un opérateur de terminaison pour appliquer les règles MAN pour son compte. L'OPTV doit par conséquent effectuer les mêmes contrôles attendus de l'opérateur de terminaison et tels que définis en section 5.



La notion d'OPTV est une relation entre opérateurs et qui n'est pas ratifiée au sein de la plateforme MAN. Elle n'a par conséquent aucun impact sur le mécanisme de confiance. L'opérateur de terminaison assume dans tous les cas la responsabilité des appels cassés par son OPTV.

L'infrastructure requise pour l'OPTV est par conséquent identique à celle de l'opérateur de terminaison (§2.4.3.2).

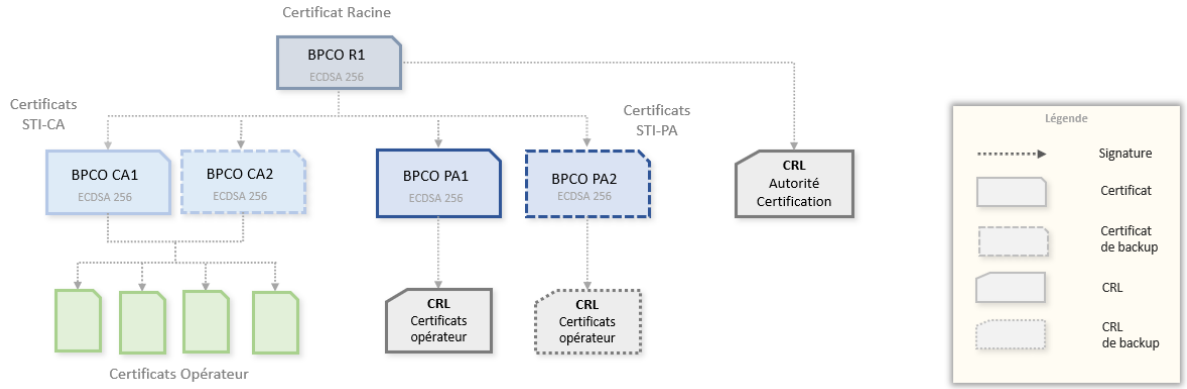
2.5 Autorité de certification

Composant critique de la plateforme MAN dans le cadre du mécanisme de confiance, l'autorité de certification a la charge de l'infrastructure à clé publique (PKI) permettant la délivrance des certificats aux opérateurs.

2.5.1 Infrastructure à clé publique

L'infrastructure à clé publique (ICP ou PKI) sécurisée mise en place par l'autorité de certification nécessite la mise en place d'une hiérarchie de certificats :

- Un **certificat racine**, chargé de signer les certificats intermédiaires, les certificats PA et la liste de révocation des certificats intermédiaires et PA
- Des **certificats intermédiaires**, correspondant au STI-CA dans le cadre STIR SHAKEN, chargés de signer les certificats opérateurs
- Des **certificats PA**, correspondant au STI-PA dans le cadre STIR SHAKEN, chargés de signer la liste de révocation des certificats opérateurs (§14.4) et des JSON Web tokens utilisés dans les APIs fournissant les informations de l'autorité de certification (§14.5, §14.6).
- Une liste de révocation (CRL) pour les certificats opérateur, signée par un certificat PA (§14.4)
- Une liste de révocation (CRL) pour les certificats intermédiaires et PA, signée par le certificat racine (§14.8)



2.5.2 Certificat racine

Certificat auto-signé, ce certificat est le certificat primaire de l'autorité de certification servant à signer les certificats intermédiaires qui seront eux-mêmes en charge de signer les certificats opérateur et la liste de révocation (CRL) des certificats de l'autorité. Le certificat racine en cours de validité est récupérable via une API BPCO dédiée, cf. §14.5.

Le certificat racine ne sert en aucun cas à signer directement les certificats opérateurs.

Durée de validité

La durée de validité du certificat racine de l'autorité de certification est de **10 ans**.

Génération et stockage des clés

Les clés sont générées par un module de cryptographie certifié.

La clé privée du certificat racine est stockée hors ligne.

Propriétés du certificat

Conformément aux standards STIR/SHAKEN, le certificat racine de l'autorité de certification se caractérise par :

- Un attribut « Common Name » contenant *SHAKEN Root* afin de confirmer son rôle
- Une extension x509 « Basic Constraints » renseignée avec *CA = true*
- Une extension x509 « Key Usage » à *Certificate Sign* et *CRL Sign* afin de lui permettre de signer respectivement les autres certificats de l'autorité et la CRL associée

Propriété	Valeur attendue
Version	3 (0x2)
Serial Number	Identifiant unique du certificat
Issuer	C=FR, O=Base des Certificats Opérateurs, OU=Certificate Authority, CN=BPCO R1 – SHAKEN Root
Nom du certificat (CN)	BPCO R1 - SHAKEN Root
Country (C)	FR
Organization (O)	Base des Certificats Opérateur
Organization Unit (OU)	Certificate Authority
Type de clé	ECDSA
Longueur de clé	256 bits
Algorithme de hachage	SHA256
Extension : X509v3 Subject Key Identifier	Identifiant unique du certificat, dérivé de la clé publique du certificat
Extension : X509v3 Key Usage	critical Certificate Sign, CRL Sign
Extension : X509v3 Basic Constraints	Critical CA:TRUE

2.5.3 Certificats intermédiaires

Ces certificats sont signés par le certificat racine **BPCO R1** et sont ceux utilisés pour la signature des certificats opérateurs. La liste des certificats intermédiaires en cours de validité est récupérable via une API BPCO dédiée, cf. §14.6.

Durée de validité

Les certificats intermédiaires ont une validité de 5 ans.

Redondance des certificats

Afin d'assurer une continuité des opérations, l'autorité de certification dispose à tout moment d'un certificat intermédiaire (**BPCO CA1**), utilisé pour la signature des certificats opérateurs, et d'un backup (**BPCO CA2**) permettant de délivrer des certificats si jamais le certificat **BPCO CA1** doit être révoqué.

Le certificat de backup est généré à partir de paires de clés différentes.

Ainsi, si la clé du certificat **BPCO CA1** est compromise, l'autorité révoque immédiatement ce certificat et utilise le certificat de backup **BPCO CA2** pour générer tout nouveau certificat opérateur en attente de création d'un nouveau certificat **BPCO CA1**.

Génération et stockage des clés

Les clés sont générées par un module de cryptographie certifié. La clé privée est stockée en ligne au sein du module de cryptographie pour permettre la signature automatisée des certificats opérateurs sans intervention manuelle.

Propriétés du certificat

Conformément aux standards STIR/SHAKEN, le certificat intermédiaire de l'autorité de certification se caractérise par :

- Un attribut « Common Name » contenant *SHAKEN Intermediate* afin de confirmer son rôle
- Une extension x509 « Basic Constraints » renseignée avec *CA = true*
- Une extension x509 « Key Usage » à *Certificate Sign* afin de lui permettre de signer les certificats opérateurs

Propriété	Valeur attendue
Version	3 (0x2)
Serial Number	Identifiant unique du certificat
Issuer	C=FR, O=Base des Certificats Opérateurs, OU=Certificate Authority, CN=BPCO R1 – SHAKEN Root
Nom du certificat (CN)	BPCO CA1 – SHAKEN Intermediate / BPCO CA2 – SHAKEN Intermediate
Country (C)	FR
Organization (O)	Base des Certificats Opérateurs
Organization Unit (OU)	Certificate Authority
Type de clé	ECDSA
Longueur de clé	256 bits
Algorithme de hachage	SHA256
Extension : X509v3 Subject Key Identifier	Identifiant unique du certificat, dérivé de la clé publique du certificat
Extension: X509v3 Authority Key Identifier	Subject Key Identifier du certificat racine ayant signé le certificat
Extension : X509v3 Key Usage	critical Certificate Sign
Extension : X509v3 Basic Constraints	Critical CA:TRUE
Extension : Authority Information Access	URL du certificat racine ayant signé le certificat (§2.5.6)
Extension : X509v3 CRL Distribution Points	Full Name: URI:https://<domaine-bpco>/ca/crl CRL Issuer : DirName: C = FR, O = Base des Certificats Opérateurs, OU = Certificate Authority, CN = BPCO R1 – SHAKEN Root

2.5.4 Certificat “PA” Policy Administrator

Ce certificat est signé par le certificat racine **BPCO R1** et est utilisé pour la signature de la CRL des certificats opérateurs et des tokens JWT utilisés pour authentifier les informations de l’autorité de certification de la plateforme MAN :

- L’identifiant de l’autorité de certification (§14.5)
- La liste des certificats intermédiaires de l’autorité de certification (§14.6)

L’URL d’accès au certificat PA utilisé dans les cas mentionnés ci-dessus est fournie soit via la propriété x5u du token JWT qu’il signe, soit par l’extension « Authority Information Access » de la CRL des certificats opérateurs.

Durée de validité

Le certificat a une validité de 5 ans.

Redondance des certificats

Afin d’assurer une continuité des opérations, un certificat de **BPCO PA2** est disponible en backup. Il peut être utilisé pour signer les JWT si jamais le certificat **BPCO PA1** doit être révoqué. De plus, il est en charge de maintenir une CRL de backup des certificats opérateurs qui pourra prendre le relai de la CRL principale signée par le certificat **BPCO PA1** si jamais ce dernier doit être révoqué.

Le certificat de backup est généré à partir de paires de clés différentes.

Ainsi, si la clé du certificat **BPCO PA1** est compromise, l’autorité révoque immédiatement ce certificat et utilise le certificat de backup **BPCO PA2** en attente de création d’un nouveau certificat **BPCO PA1**.

Génération et stockage des clés

Les clés sont générées par un module de cryptographie certifié.

La clé privée est stockée en ligne au sein du module de cryptographie pour permettre la signature automatisée des JSON Web tokens et des CRLs.

Propriétés du certificat

Conformément aux standards STIR/SHAKEN, le certificat PA de l’autorité de certification se caractérise par :

- Une extension x509 « Basic Constraints » renseignée avec *CA = false*
- Une extension x509 « Key Usage » à *CRL Sign* et *Digital Signature* afin de lui permettre de signer respectivement la CRL des certificats opérateurs et les JWT utilisés dans les réponses aux APIs retournant les données de l’autorité de certification.

Propriété	Valeur attendue
Version	3 (0x2)
Serial Number	Identifiant unique du certificat
Issuer	C=FR, O=Base des Certificats Opérateurs, OU=Certificate Authority, CN=BPCO R1 – SHAKEN Root
Nom du certificat (CN)	BPCO PA1 / BPCO PA2
Country (C)	FR
Organization (O)	Base des Certificats Opérateurs
Organization Unit (OU)	Policy Authority
Type de clé	ECDSA
Longueur de clé	256 bits
Algorithme de hachage	SHA256
Extension : X509v3 Subject Key Identifier	Identifiant unique du certificat, dérivé de la clé publique du certificat
Extension: X509v3 Authority Key Identifier	Subject Key Identifier du certificat racine ayant signé le certificat
Extension : X509v3 Key Usage	Critical Digital Signature, CRL Sign
Extension : X509v3 Basic Constraints	Critical CA:FALSE
Extension : Authority Information Access	URL du certificat racine ayant signé le certificat (§2.5.6)
Extension : X509v3 CRL Distribution Points	Full Name: URI:https://<domaine-bpco>/ca/crl CRL Issuer : DirName: C = FR, O = Base des Certificats Opérateurs, OU = Certificate Authority, CN = BPCO R1 – SHAKEN Root

2.5.5 Liste de révocation des certificats de l'autorité de la plateforme MAN

Une liste de révocation (CRL) des certificats de l'autorité de certification est disponible au sein de la BPCO afin de permettre de connaître les certificats révoqués. Elle est distincte de la CRL des certificats opérateur (§2.7.2) et signée par le certificat racine de l'autorité de certification.

Propriété	Valeur attendue
Version	2
Algorithme de signature	ECDSA / SHA256
Issuer	C=FR, O=Base des Certificats Opérateurs, OU=Certificate Authority, CN=BPCO R1 – SHAKEN Root
Last Update	Date au format UTCTime de la dernière mise à jour de la CRL
Next Update	Date au format UTCTime de la prochaine mise à jour à maxima de la CRL
CRL Extension : CRL Number	Entier incrémenté à chaque ajout dans la CRL
CRL Extension : Authority Key Identifier	Identifiant unique du certificat racine ayant signé la CRL
CRL Extension : Authority Information Access	URL d'accès au certificat racine ayant signé la CRL (non encore disponible)

2.5.6 Accès aux données de l'autorité de certification

Les certificats et la CRL de l'autorité de certifications sont disponibles en accès public via la BPCO. Les fonctions d'accès sont détaillées en sections §14.6, §14.7 et §14.8.

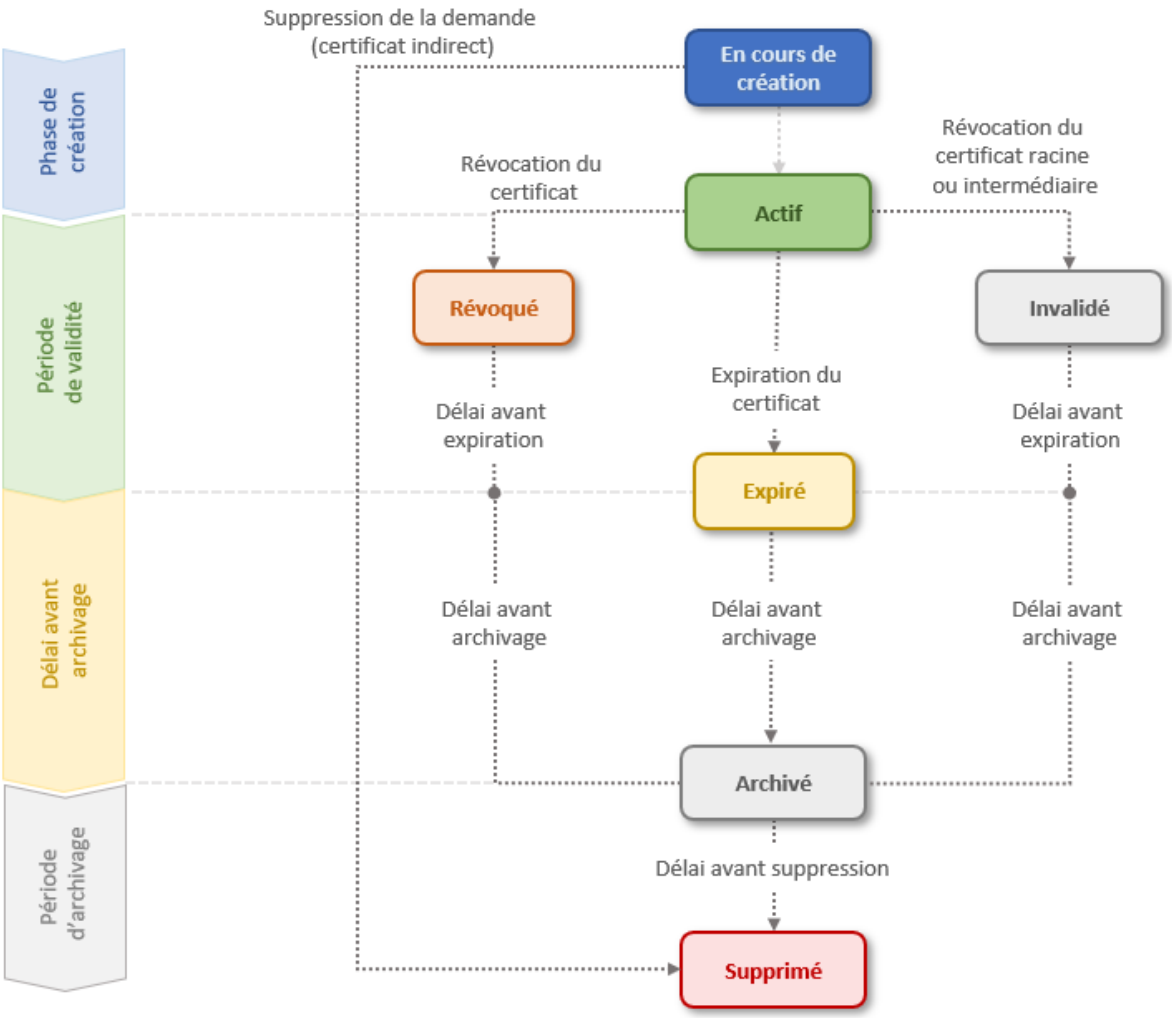
2.6 Certificats Opérateurs

Les certificats opérateurs sont générés à partir du certificat intermédiaire de l'autorité de certification (§2.5.3) et ne peuvent être utilisés que dans le cadre du mécanisme de confiance. Leur but est d'être utilisé par les opérateurs de terminaison afin d'extraire la clé publique et de vérifier la signature effectuée par l'opérateur signataire avec la clé privée correspondante.

Ce document fournit une présentation générale des certificats opérateurs, de leur cycle de vie et de leur utilisation. Il convient de se référer au code de procédures MAN pour toute propriété spécifique liée aux certificats pouvant évoluer dans le temps, telle que les durées de validité des certificats, les délais d'archivage….

2.6.1 Cycle de vie

Le statut d'un certificat opérateur peut prendre différentes valeurs au sein de la plateforme MAN. Ces changements d'état sont représentés par le schéma suivant, et peuvent intervenir automatiquement ou suite à l'action d'un opérateur ou de la plateforme MAN.



Délivrance du certificat

cf §8

La procédure de délivrance d'un certificat commence par la création d'un certificat dans un statut **En cours de Création**. Une fois la procédure de délivrance du certificat complétée, le statut du certificat passe à **Actif**. L'état En cours de Création n'est effectivement visible que dans le cas de délivrance de certificats indirects, §8.8.

Révocation d'un certificat

cf §11

Une fois le certificat créé et tant que celui-ci n'a pas expiré, la révocation d'un certificat peut être effectuée à l'initiative de l'opérateur ou suite à une décision de l'autorité de gouvernance de la plateforme MAN.

Le statut du certificat passe alors à **Révoqué**. Il n'est alors plus possible de l'utiliser pour signer des appels et tout appel signé avec ce certificat doit être rejeté et signalé par une trace d'appel cassable/cassé (§2.3.3).

Le certificat est ajouté à la CRL de la BPCO, mais reste encore disponible au sein de la base publique des certificats opérateur de la BPCO jusqu'à son archivage.

Un opérateur signataire doit avoir la possibilité de faire la demande de révocation de tous les certificats directs ou indirects qui lui ont été délivrés. Un OPTS peut quant à lui effectuer la demande de révocation pour les certificats indirects délivrés aux opérateurs signataires et dont il a le mandat.

Expiration des certificats

Une fois la date de fin de validité d'un certificat atteinte, la plateforme MAN passe le statut du certificat, s'il n'a pas été révoqué, à **Expiré**. Il n'est alors plus possible de l'utiliser pour signer des appels et tout appel signé avec ce certificat doit être rejeté et signalé (par une trace d'appel cassable/cassé), §2.3.3.

Le certificat reste encore disponible au sein de la BPCO jusqu'à son archivage.

Archivage des certificats

Un système automatique d'archivage de tous les certificats expirés après une certaine période est mis en place au niveau de la plateforme MAN. Les certificats archivés resteront disponibles pour consultation au sein de l'IHM et de l'API de la plateforme pendant une durée supplémentaire avant leur purge automatique. Les certificats de test ne sont pas archivés mais directement supprimés. Les délais d'archivage et de suppression sont précisés au sein du code de procédures MAN.

Lors de leur archivage, les certificats sont supprimés de la BPCO. Si les certificats étaient révoqués, ils sont également supprimés de la CRL de la BPCO.

Suppression de certificats archivés

cf. §12

Un certificat archivé reste disponible conformément à la durée de conservation des données définie au sein du code de procédures MAN, puis est automatiquement supprimé de la plateforme. Un certificat archivé ne peut pas être supprimé manuellement par un opérateur.

Invalidation de certificats dans le cas de compromission de l'autorité de certification

cf. §15

Dans le cas de compromission d'une des clés de l'autorité de certification, il se peut qu'un certificat intermédiaire doive être révoqué. Tous les certificats opérateurs associés ne pourront plus être utilisés pour la signature d'appel, et leur statut passe alors à **Invalidé** une fois le certificat intermédiaire effectivement révoqué.

Ils restent disponibles dans la base des certificats opérateur jusqu'à leur archivage et ne peuvent être révoqués ni renouvelés. La procédure mise en place afin de garantir la continuité de service est détaillée dans le code de procédures MAN.

2.6.1.1 Gestion du cycle de vie des certificats

Cette section présente en des diagrammes simples les changements d'état des certificats sur la plateforme MAN en fonction des événements, mais aussi les répercussions sur la BPCO.

Sans révocation

Avec révocation

Révocation de certificats de l'autorité de certification – sans révocation du certificat opérateur

2.6.2 Certificats directs vs. indirects

La plateforme MAN distingue deux types de certificats opérateurs, pour lesquels la procédure de délivrance des certificats diffère (§8). Par contre, une fois le certificat délivré, les procédures de signature d'un appel (§3) et de vérification (§5) restent inchangées.

Certificats Directs

Un certificat direct permet à un opérateur signataire de signer en son nom les appels.

Certificats Indirects

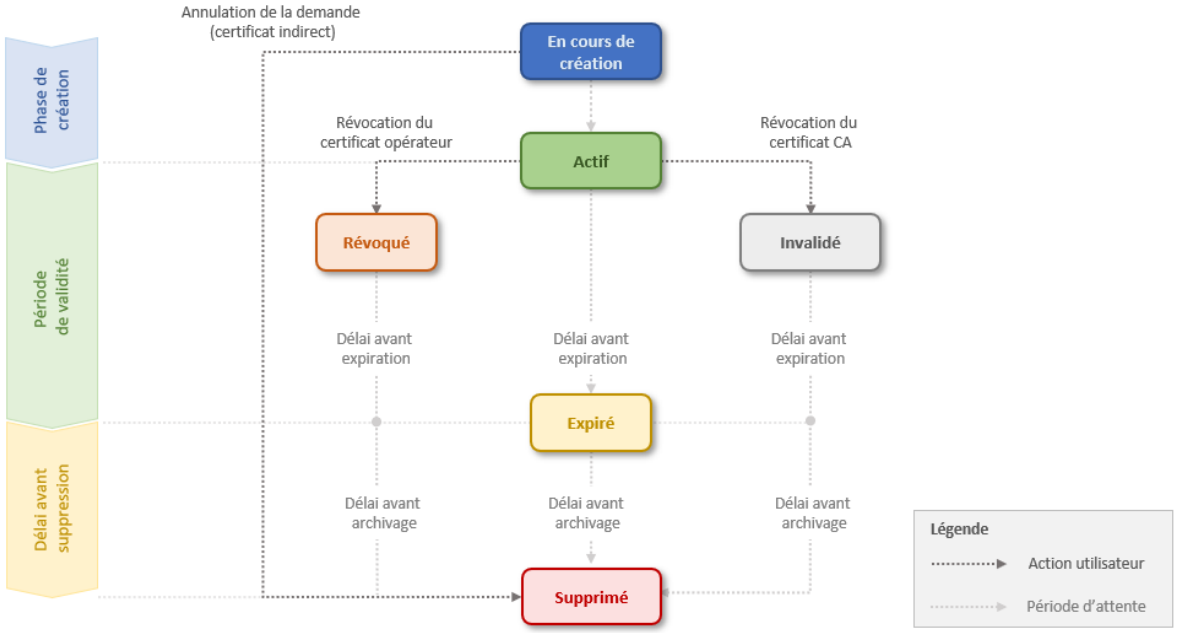
Un certificat indirect permet à un opérateur signataire de mandater un autre opérateur pour signer ses appels, dénommé OPTS (§2.4.3.4). Comme pour le certificat direct, le certificat indirect permet toujours d'identifier l'opérateur signataire dans la signature de l'appel, mais cette signature utilise une paire de clés détenue par l'OPTS. La procédure de création du certificat indirect nécessite par conséquent l'intervention des deux opérateurs :

- L'opérateur signataire, qui initie de la demande de création du certificat indirect en choisissant l'OPTS à mandater
- L'OPTS pour finaliser la création en fournissant le CSR lié à ses clés de chiffrement

2.6.3 Certificats de Test

Des certificats de test peuvent aussi être délivrés aux opérateurs. À des fins uniquement de tests occasionnels entre opérateurs, ils sont en tout point semblables dans leur contenu aux certificats standards mais présentent les différences suivantes dans la gestion de leur cycle de vie :

- Leur durée de validité est réduite
- Ils peuvent être supprimés à tout moment (§12). Dans ce cas, si le certificat n'a pas déjà expiré, il est révoqué et inscrit dans la CRL des certificats opérateur avant sa suppression.
- Ils ne peuvent être renouvelés automatiquement (§10).
- Ils ne sont pas archivés mais directement supprimés.
- Ils ne sont pas inclus dans la copie locale des certificats opérateurs (§ 9).



Un certificat de test ne doit en aucun cas être utilisé pour des appels d'utilisateurs finals. Il doit être réservé pour les tests opérateurs.

2.6.3.1 Gestion du cycle de vie des certificats de test

Cette section présente en des diagrammes simples les changements d'état des certificats de test sur la plateforme MAN en fonction des événements, mais aussi les répercussions sur la BPCO.

Sans révocation

Avec révocation

Suppression avant expiration

2.6.4 Dates & durée de validité d'un certificat

L'opérateur peut choisir, lors de la demande de délivrance d'un certificat, la date de début de validité de ce certificat. Si le certificat n'est pas un certificat de test, cette date doit être au minimum définie une semaine après la date actuelle.

La durée de validation n'est configurable que pour des certificats indirects ou de test. Les durées fixées pour les certificats directs, ainsi que les valeurs minimales et maximales pour les autres certificats sont fournies par le code de procédures MAN.

Type	Configuration possible
Direct	Non – fixée par la plateforme
Indirect	Oui – défini par l'opérateur signataire
Test	Oui – défini par l'opérateur signataire

Les paramètres à utiliser pour configurer ces propriétés sont décrits dans le chapitre dédié aux fonctions de délivrance des certificats opérateurs (§8).

2.6.5 Nombre de certificats

Un opérateur signataire peut se voir attribuer un nombre limité de certificats actifs, qu'ils soient directs ou indirects. Les certificats de test ne sont pas inclus dans cette limite et font l'objet d'une limite différenciée. Les valeurs de ces deux limites sont indiquées au sein du code de procédures.

Aucune limite n'est par contre appliquée sur le nombre de certificats indirects dont un OPTS peut recevoir le mandat.

2.6.6 Exigences algorithmiques

2.6.6.1 Opérateurs signataires & OPTS

Comme précisé dans le code de procédures MAN, seul l'algorithme ES256 est autorisé pour la signature d'appels.

Commande openssl de génération de clé privée ECDSA avec une courbe P-256

```
openssl ecparam -genkey -name prime256v1 -out ecdsa-p256.key
```

Commande openssl d'extraction de la clé publique à partir de la clé privée

```
openssl ec -in ecdsa-p256.key -pubout -out ecdsa-p256.pub
```

2.6.6.2 Opérateurs de terminaison & OPTV

Pour les opérateurs en charge de vérifier les appels, il convient de supporter les mêmes algorithmes de signature définis dans la section précédente et préconisés dans le code de procédures MAN.

2.6.7 Propriétés du certificat

Les certificats délivrés aux opérateurs ont obligatoirement comme Common Name (CN) le code APNF assigné à l'opérateur signataire, précédé de la mention SHAKEN tel que défini dans la section 6.4.1 de la spécification ATIS-1000080.

Une extension X509 *tnAuthList* est ajoutée afin de toujours désigner le code APNF de l'opérateur signataire. L'OID 1.3.6.1.5.5.7.1.26 est utilisé pour identifier cette extension dans le certificat.

Propriété	Valeur
Version	3 (0x2)
Serial Number	Valeur unique par certificat
Subject | Common Name (CN)	SHAKEN <Code APNF de l'opérateur signataire>
Issuer	C=FR, O=Base Publique des Certificats Opérateurs, OU=Certificate Authority, CN=BPCO CA1 – SHAKEN Intermediate
Pays (C)	Tel que défini dans le fichier CSR
Localité (L)	Tel que défini dans le fichier CSR
Etat (S)	Tel que défini dans le fichier CSR
Organisation (O)	Tel que défini dans le fichier CSR
Département (OU)	Tel que défini dans le fichier CSR
Type de clé	ECDSA
Longueur de clé	256 bits
Algorithme de hachage	SHA256
Extension : X509v3 Subject Key Identifier	Identifiant unique du certificat
Extension: X509v3 Authority Key Identifier	Subject Key Identifier du certificat intermédiaire ayant signé le certificat
Extension : Authority Information Access	URL dans la BPCO du certificat intermédiaire ayant signé le certificat
Extension : X509 Key Usage	Critical, Digital Signature
Extension : X509 Basic Constraints	Critical, CA: FALSE
Extension : X509v3 CRL Distribution Points	Full Name: URI:https://<domaine-bpco>/crl CRL Issuer : DirName: C=FR, O=Base des Certificats Opérateurs, OU=Policy Authority, CN=BPCO PA1 DirName: C=FR, O= Base des Certificats Opérateurs, OU=Policy Authority, CN=BPCO PA2
Extension : tnAuthList **(OID : 1.3.6.1.5.5.7.1.26)**	Code APNF de l'opérateur signataire au format DER : `1.3.6.1.5.5.7.1.26: 0.....<code APNF opérateur>`

2.7 BPCO – Base Publique des Certificats Opérateurs

Élément central de la plateforme MAN et du mécanisme de confiance, la BPCO – Base Publique des Certificats Opérateurs – correspond au service d'accès public aux certificats opérateurs et de toute autre donnée devant être disponibles aux opérateurs de terminaison afin de leur permettre de :

- récupérer les certificats opérateurs ;
- vérifier la validité de ces certificats, y compris leur chaîne de certification.

2.7.1 Base publique des certificats opérateurs

Cette base permet l'accès public aux certificats délivrés aux opérateurs par la plateforme MAN, permettant d'assurer la fonctionnalité de vérification des signatures d'appels (§5).

2.7.1.1 URLs d'accès

Dans le cadre du mécanisme de confiance, le protocole utilisé pour la mise à disposition des certificats opérateurs est le protocole HTTPs. Chaque certificat opérateur est ainsi accessible à partir d'une URL qui lui est unique et dont le format est le suivant :

`https://<domaine-bpco>/certs/<code-apnf-operateur>/<serial-number-certificat>.cer`

Cette URL est celle qui devra être utilisée par le STI-AS lors de la construction du champ Identity (§3.5). La section 14.3.1 précise les détails de la requête, du format et du contenu de la réponse.

2.7.1.2 Certificats inclus

Tous les certificats opérateurs, qu'ils soient directs, indirects, ou de test, doivent être disponibles au sein de la base publique.

2.7.1.3 Certificats expirés, révoqués et invalidés

L'URL d'un certificat expiré, révoqué ou invalidé reste accessible tant que ce certificat n'a pas été archivé par la plateforme MAN (§2.7.1.5). Lors de l'archivage du certificat, l'URL associée à ce dernier est supprimée.

2.7.1.4 Politique de publication des URLs

Pour éviter une copie facile et complète de la base des certificats et la possible récupération d'informations liées au fonctionnement des opérateurs par rapport au mécanisme de confiance, aucune URL permettant d'énumérer les certificats disponibles au sein de la base n'est publiée.

En particulier, l'accès à l'URL `https://<domaine-bpco>/certs/<code-apnf-operateur>/` n'est pas autorisée. Une erreur HTTP 403 sera remontée pour toute requête effectuée sur ce type d'URL.

2.7.1.5 Mise à jour de la base des certificats opérateurs

La base publique des certificats est mise à jour par la plateforme MAN dans les cas suivants. Voir aussi les sections §2.6.1.1 et §2.6.3.1 pour des diagrammes résumant ces mises à jour en fonction du cycle de vie des certificats.

Contexte	Mise à jour appliquée
Création d'un certificat	Publication de l'URL associée
Révocation d'un certificat	Aucune
Invalidation du certificat	Aucune
Expiration d'un certificat	Aucune
Suppression d'un certificat de test	Suppression de l'URL associée
Archivage du certificat	Suppression de l'URL associée

Il est possible de connaître la date de dernière mise à jour de la base publique des certificats par l'intermédiaire d'une fonction d'API de la plateforme, §13.8.

2.7.2 Liste publique de révocation des certificats opérateurs (CRL)

2.7.2.1 URL d'accès

Comme pour les certificats opérateurs, la liste de révocation de ces derniers est disponible en accès public via l'URL :

```
https://<domaine-bpco>/crl
```

La section §14.4.1 précise les détails de la requête, du format et du contenu de la réponse.

2.7.2.2 Propriétés de la CRL

Propriété	Valeur attendue
Version	2
Algorithme de signature	ECDSA / SHA256
Issuer	C=FR, O=Base des Certificats Opérateurs, OU=Certificate Authority, CN=BPCO PA1 – SHAKEN Root
Last Update	Date au format UTCTime de la dernière génération de la CRL
Next Update	Date au format UTCTime de la prochaine mise à jour à maxima de la CRL. Définie à une semaine après la date spécifiée dans la propriété *Last Update*.
CRL Extension : CRL Number	Entier incrémenté à chaque ajout dans la CRL
CRL Extension : Authority Key Identifier	Identifiant unique du certificat PA ayant signé la CRL
CRL Extension : Authority Information Access	URL d'accès au certificat PA ayant signé la CRL
CRL Extension : Issuing Distribution Point	Critical Indirect CRL

Pour chaque certificat révoqué sont stockés au sein de la CRL :

- Le serial number
- L'issuer, à savoir le certificat CA ayant signé le certificat.
- La date de révocation
- La raison de la révocation

2.7.2.3 Certificats expirés

Les certificats révoqués et expirés restent accessibles tant qu'ils ne sont pas archivés (§2.6.1). Lors de l'archivage des certificats, les certificats sont alors supprimés.

2.7.2.4 Mise à jour de la CRL des certificats opérateurs

La CRL est mise à jour par la plateforme MAN dans les cas suivants. Voir aussi les sections §2.6.1.1 et §2.6.3.1 pour des diagrammes résumant ces mises à jour en fonction du cycle de vie des certificats.

Contexte	Mise à jour appliquée
Création d'un certificat	Aucune
Révocation d'un certificat	Ajout du certificat dans la CRL
Invalidation du certificat	Aucune
Suppression certificat de test	Ajout du certificat dans la CRL
Expiration d'un certificat non révoqué	Aucune
Expiration d'un certificat révoqué	Aucune
Expiration d'un certificat de test supprimé manuellement	Suppression du certificat de la CRL
Archivage certificat expiré	Aucune
Archivage certificat révoqué	Suppression du certificat de la CRL

Il est possible de connaître la date de dernière mise à jour de la CRL des certificats opérateur par l'intermédiaire d'une fonction d'API de la plateforme, §13.10.

La CRL est de plus renouvelée automatiquement chaque jour pour s'assurer qu'elle reste valide par rapport à la date fournie dans la propriété *Next Update*.

Lors de sa mise à jour / renouvellement, les propriétés suivantes de la CRL sont modifiées :

- CRL Number : La valeur est incrémentée de 1
- Last Update : La valeur est renseignée avec la date et heure de génération de la CRL
- Next Update : La valeur est renseignée avec la nouvelle valeur de la propriété Last Update à laquelle est ajoutée une semaine.

2.7.3 Informations de l'autorité de certification

La BPCO permet enfin d'accéder aux données associées à l'autorité de certification :

- Son certificat racine (§14.5)
- Ses certificats intermédiaires (§14.6, §14.7)
- La liste de révocation de ses certificats (§14.8)

2.8 Spécificités du modèle français

2.8.1 Communications Opérateur / STI-PA

L'ATIS **100080** prévoit que l'opérateur communique avec le STI-PA afin de récupérer un *Service Provider Code token* fourni au STI-CA lors de la procédure de délivrance des certificats. Cette communication est assurée par un service s'inspirant du protocole OAuth, où le STI-PA est le serveur

d'autorisation qui fournit l'*access token* - appelé *Service Provider Code Token*, et le STI-CA le serveur de ressources.

Reprenant ce principe, le modèle français va plus loin et applique une solution en tout point conforme au protocole OAuth 2.0, où l'opérateur se voit délivrer un *access token* à la plateforme qui lui permettra de s'authentifier lors des appels aux APIs de cette dernière, y compris pour les demandes de certificats opérateur. L'*access token* remplace ainsi le *Service Provider Code token* et permet la mise en place du standard OAuth pour la sécurisation de l'ensemble des APIs.

Plus de détails quant à l'authentification des APIs de la plateforme MAN sont disponibles en section §13.1.

2.8.2 Protocole de délivrance des certificats

Contrairement aux préconisations de l'ATIS **100080**, le protocole ACME n'est pas utilisé dans le mode opératoire du mécanisme de confiance en France pour la procédure de délivrance des certificats.

Comme indiqué dans la section précédente, le modèle français se repose sur le protocole OAuth 2.0 pour authentifier les appels d'API effectués sur la plateforme MAN. La solution du protocole ACME où le *Service Provider Code token* doit être fourni comme challenge est ainsi remplacée par des appels API standards dont l'authentification est assurée par le protocole OAuth 2.0 et l'utilisation d'*access tokens.* Ceci permet à la plateforme MAN de ne pas multiplier les solutions d'accès à ses APIs et de garantir une cohérence.

Plus de détails quant à l'authentification des APIs de la plateforme MAN sont disponibles en section §13.1.

2.8.3 Protocole d'accès des certificats

Alors que la RFC 8226 indique que 3 protocoles peuvent être utilisés pour l'accès aux certificats opérateurs, seul le protocole HTTPs est supporté dans le modèle français (§2.7.1.1).

2.8.4 Durée de vie fixée des certificats opérateurs

Le modèle français prévoit la délivrance de certificats opérateurs ayant une durée de validité fixe (cf. code de procédures MAN). Il n'est pas prévu que l'opérateur puisse choisir la durée de validité d'un certificat, sauf pour les certificats de test (§8.7.3) et les certificats indirects (§8.8.1).

De plus, contrairement à la RFC 8226, il n'est pas prévu d'utiliser des certificats à courte durée de vie. Alors que le standard préconise l'emploi de ces certificats pour réduire les problèmes de synchronisation avec le STI-CR, la solution mise en place en France à base de copies locales (cf. §9) permet de se prémunir de ce type de problèmes. L'utilisation de certificats à longue durée de vie réduit qui plus est les contraintes de maintenance liées au renouvellement des certificats et leur installation sur chaque composant de l'opérateur.

2.8.5 Algorithmes supportés pour les certificats de l'autorité de certification

Alors que la RFC 8226, section 4, précise que les certificats opérateur peuvent être signés avec des algorithmes ECDSA P-256 ou RSA PKCS #1 v1.5, l'autorité de certification française n'utilisera que l'algorithme ECDSA P-256.

2.8.6 Valorisation du claim ORIG du token PASSPORT

Le modèle français préconise que le claim orig du token PASSport soit défini, suivant les règles définies au sein du document « *MAN_Regles techniques* », à partir de l'entête SIP « From », voire « P-Asserted-Identity » si l'entête « From » vaut *anonymous@anonymous.invalid* ou *unavailable@unknown.invalid*, contrairement à la spécification **ATIS-1000074**, section 5.2.2, où le champ « P-Asserted-Identity » est utilisé en priorité.

Ce principe est aussi appliqué lors de la procédure de vérification du message SIP INVITE par l'opérateur de terminaison (cf §5.5.2.3).

De plus, le numéro affiché au client en terminaison doit toujours être celui contenu dans le FROM. Seulement dans certains cas spécifiques, le PAI peut être utilisé pour véhiculer la valeur du FROM.

Cette décision a été entérinée par les opérateurs français et normalisée au sein du profil SIP 3.1 de la FFTélécoms pour l'interconnexion voix inter-opérateurs, section 17.1. Lors d'un appel d'interconnexion voix livré en SIP à un opérateur national, le numéro appelant que l'on veut présenter à l'appelé doit être contenu dans le header From de l'INVITE initial.

2.8.7 Non inclusion de la Certificate Policy Identifier dans les certificats

La spécification ATIS-1000080, section 6.4.1, stipule qu'une extension «Certificate Policies » doit être incluse dans les certificats opérateur et les certificats intermédiaire de l'autorité de certification :

STI intermediate and End-Entity certificates shall include a Certificate Policies extension containing a single OID value that identifies the SHAKEN Certificate Policy established by the STI-PA. The OID value is specified in the SHAKEN Certificate Policy document.

Cette extension ne sera pas incluse dans les certificats du modèle français, le temps qu'une politique de gestion des certificats soit finalisée et un OID fourni par l'organisme d'attribution des OIDs.

2.8.8 CRL de l'autorité de certification

Contrairement à la spécification **ATIS-1000080**, section 6.3.5.1, il n'est pas demandé dans la création du fichier CSR d'inclure l'extension *CRL Distribution Points* spécifiant l'URL d'accès de la CRL des certificats opérateurs, le modèle de gouvernance français n'utilisant qu'une seule plateforme pour le STI-CA et le STI-PA.

Si cette extension est incluse, elle devra être valorisée à l'URL d'accès de la CRL, §2.7.2.1.

2.8.9 Utilisation du code réponse SIP 400 par l'opérateur de terminaison

Contrairement à la spécification **ATIS-1000082**, section 8.2.4.2, il est demandé aux opérateurs d'utiliser un code réponse SIP 400 dans le cas où les informations contenues dans le message SIP INVITE reçu ne permettent pas de valider la signature de l'appel, le modèle français prévoyant de couper les appels systématiquement dans le cas où la signature ne peut être vérifiée.

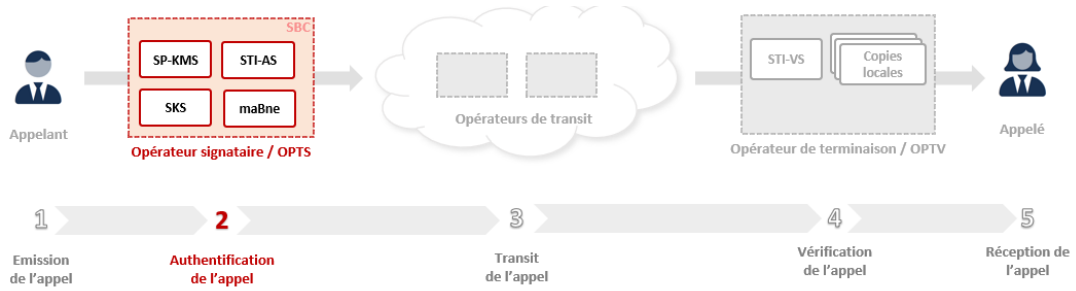
Les modalités d'utilisation de ce code réponse sont précisées en section §5.5.

3 Procédure d'émission des appels

3.1 Opérateurs concernés

- Opérateur signataire
- OPTS

3.2 Contexte d'application



Lors de l'émission d'un appel SIP, l'opérateur doit authentifier le numéro appelant en signant la requête INVITE avec son certificat opérateur qui lui aura été délivré par l'autorité de certification.

Cette signature est embarquée au sein d'un JSON Web Token appelé **PASSport** - dont le format est décrit dans le **RFC 8225**, lui-même inclus dans un entête *Identity* à ajouter par l'opérateur à la requête INVITE.

Le token **PASSport** doit de plus inclure l'extension **SHAKEN** permettant de passer le niveau d'attestation de l'appel tel que défini dans la section §2.3.2.

Si l'opérateur se retrouve dans l'incapacité de signer ses appels, suite à un incident au sein de son SI ou un incident impactant la plateforme MAN (cf. §15), il lui est possible de déclencher le débrayage de son STI-AS (cf §3.6).

3.3 Prérequis

- L'opérateur doit s'être enregistré auprès de la plateforme MAN (§6) pour accéder aux fonctionnalités de délivrance de certificats opérateurs

- L'opérateur doit avoir été vérifié administrativement pour se voir délivrer un certificat opérateur (§7).
- Le STI-AS de l'opérateur dispose d'une clé privée pour signer l'appel. Les exigences de génération des clés privées sont définies en section §2.6.6.
- Un certificat opérateur a été délivré à l'opérateur pour la clé privée mentionnée ci-dessus (§8).
- L'URL publique du certificat délivré lors de la phase précédente a été communiquée au STI-AS.
- L'appel répond aux conditions définies par les règles techniques MAN pour être signé.

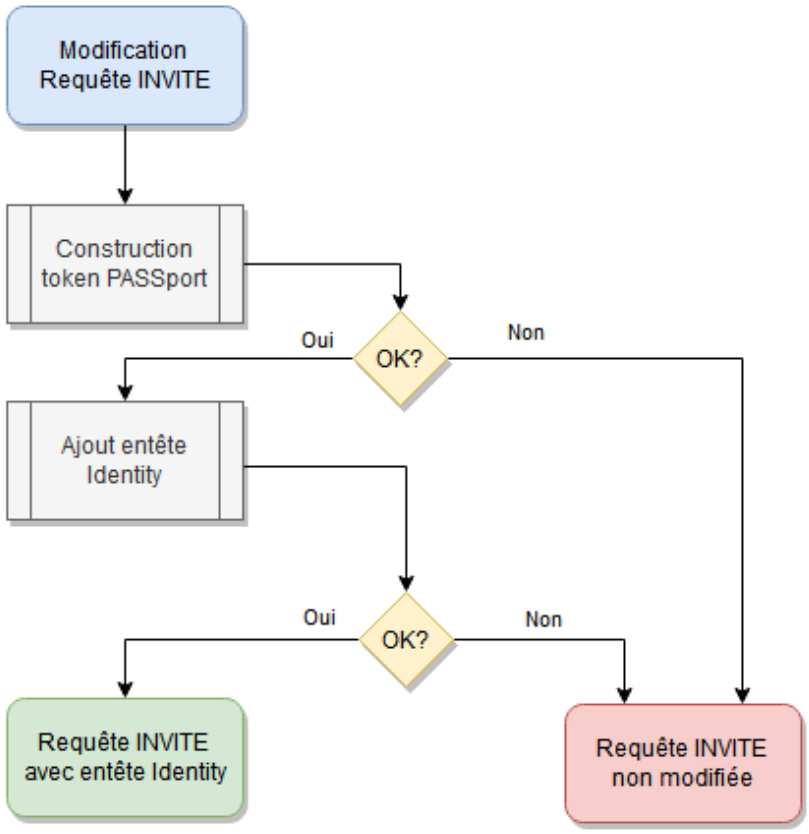
3.4 Composants impliqués

- Le **SBC**, élément réseau supportant les messages SIP INVITE entrants et sortants. Il appelle le **STI-AS** pour la signature des appels sortants.
- Le **STI-AS** de l'opérateur, en charge de la génération du token **PASSport** avec extension **SHAKEN** et son insertion dans le message SIP INVITE au sein d'un entête *Identity*.
- Le **SP-KMS** de l'opérateur, pour fournir l'URL publique du certificat qui lui aura été délivré par l'autorité de certification.
- Le **SKS** de l'opérateur, stockant la clé privée utilisée pour signer l'appel.
- **maBNE**, base de données des numéros de l'opérateur.

3.5 Procédure détaillée

L'opérateur doit inclure l'entête *Identity* au sein de la requête SIP INVITE suivant ces 3 étapes :

1. Construction du token PASSport, §3.5.1,
2. Construction et inclusion de l'entête Identity au sein du message SIP INVITE, §3.5.2,

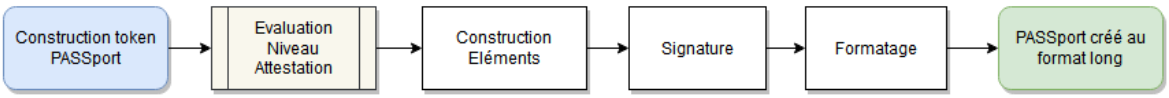


Dans le cas où une erreur intervient lors de l'une de ces phases, la requête INVITE n'est pas modifiée et est envoyée sans l'entête *Identity*. La requête sera alors rejetée en transit ou en terminaison conformément au profil SIP 3.1.

3.5.1 Construction du token PASSport

La construction du token PASSport s'effectue en 4 étapes :

- La définition du niveau d'attestation de l'appel
- La construction des éléments header et payload
- La signature du token
- Formatage final du token



3.5.1.1 Evaluation du niveau d'attestation

L'opérateur doit définir un niveau d'attestation du numéro appelant, représenté par une lettre A, B ou C, suivant les critères explicités dans le document « *MAN_Cas_Usages_Voix* ».

L'opérateur s'appuie sur tout système interne incluant **maBNE** pour établir le niveau d'attestation.

3.5.1.2 Construction des éléments du token

Le token PASSport est un JSON Web Token normalisé suivant les RFC 7515 et 7518. Il est ainsi composé de 3 éléments :

- **Header** : Un entête fournissant les informations permettant d'appréhender le contenu et le type de signature du token
- **Payload** : Le contenu effectif du token
- **Signature :** La signature créée à partir des éléments *header* et *payload*

Cette étape se focalise sur la génération du header et du payload, la génération de la signature étant explicitée en section §3.5.1.3.

Header

L'entête est un objet JSON avec obligatoirement les propriétés suivantes :

- **alg** : Définit l'algorithme de signature utilisé. Doit être valorisé à « ES256 ».
- **ppt** : Extension utilisé pour le token. Doit être valorisé à « *shaken* » pour permettre l'inclusion du niveau d'attestation dans le *payload* du token PASSport. Il est à noter que dans le futur d'autres valeurs pourront être utilisées.
- **typ** : Définit le type de JSON Web token. Doit toujours être « *passport* ».
- **x5u :** Chemin d'accès au certificat contenant la clé publique permettant de vérifier la signature. Ce chemin doit correspondre à l'URL HTTPs publique de la BPCO pour ce certificat (§2.7.1.1).

L'ordre des propriétés dans l'objet est important pour la signature du token, celles-ci devant être ordonnées suivant l'ordre alphabétique. Les propriétés doivent ainsi toujours être définies dans cet ordre : *alg*, *ppt*, *typ, x5u*.

Le header doit de plus être encodé suivant l'encodage UTF-8, le chemin du certificat pouvant contenir des caractères non-ASCII.

```
{
  "alg": "ES256",
  "ppt": "shaken",
  "typ": "passport",
  "x5u": "https://<domaine-bpco>/certs/<code-apnf-operateur>/<sn-certificat>.cer"
}
```

Payload

Le payload est un objet JSON avec obligatoirement les propriétés suivantes :

- **attest** : niveau d'attestation évalué en section §3.5.1.1. Sa valeur doit être « *A* », « *B* », ou « *C* »
- **dest** : Objet JSON composé d'une propriété unique *tn* dont la valeur est une liste de chaînes de caractères contenant une seule valeur devant correspondre au numéro utilisé dans le champ DEST reçu du composant SBC et valorisé suivant les règles émises par le document « *MAN_Regles techniques* ».

- **iat** : Date d'émission de l'appel. Nombre entier Javascript. Correspond au nombre de secondes depuis le 1er Janvier 1970 selon le fuseau horaire UTC.
- **orig** : Objet JSON composé d'une propriété unique *tn* dont la valeur est une chaîne de caractères devant correspondre, suivant les règles définies au sein du document « *MAN_Regles techniques* », au numéro reçu par le composant SBC dans l'entête SIP « From », voire « P-Asserted-Identity » si l'entête From vaut *anonymous@anonymous.invalid* ou *unavailable@unknown.invalid* (cf §2.8.6).
- **origid** : un UUID permettant à l'opérateur d'identifier le composant réseau ou logiciel intervenant dans l'acheminement de l'appel. Il est important de ne pas utiliser de références trop précises sous peu de divulguer des informations sur la topologie du réseau de l'opérateur (section 10 du RFC 8588).

L'ordre des propriétés dans l'objet est important pour la signature du token, celles-ci devant être ordonnées suivant l'ordre alphabétique. Les propriétés doivent ainsi toujours être définies dans cet ordre : *attest, dest, iat, orig*, *origid*.

Le type utilisé pour les valeurs est aussi important ; ainsi la valeur de la propriété *iat* doit être un entier et non une chaîne de caractères.

Exemple :

```
{
   "attest":"A",
   "dest":{"tn":["3900"]},
   "iat":1443208345,
   "orig":{"tn":"33123456789"},
   "origid":"123e4567-e89b-12d3-a456-426655440000"
}
```

3.5.1.3 Signature du token PASSport

La signature du token PASSport passe par les étapes suivantes :

- Création d'une empreinte du token
- Hachage de l'empreinte, en utilisant l'algorithme de hachage associé à l'algorithme de signature choisi par l'opérateur et défini dans le claim « alg » de l'entête du token PASSport.
- Chiffrement du hash de l'empreinte, en utilisant la clé privée mise à disposition du STI-AS

Note : Le token PASSPorT étant généré suivant les RFCs 7519 (JSON Web Token - JWT) et 7515 (JSON Web Signature - JWS), il est recommandé d'utiliser des librairies standards suivant ces spécifications afin de générer la valeur de ce token. Cette section décortique les opérations effectuées pour la signature du token et fournit des exemples avec l'outil openssl disponible sur la majorité des systèmes d'exploitation, mais l'utilisation de librairies permettra de simplifier l'implémentation du mécanisme de confiance.

Création de l'empreinte

Les éléments *header* et *payload* sont utilisés pour la génération de l'empreinte. Ces éléments ont été « préparés » afin de s'assurer que la valeur de l'empreinte ne change pas suivant les implémentations :

- L'ordre des priorités doit être constant et suivre l'ordre alphabétique
- L'encodage des données est en UTF-8

Les objets JSON doivent ensuite être transformés en chaîne de caractères (« sérialisés »), sans utilisation de formatage étendu. Tous les caractères d'espacement et de retour à la ligne doivent être supprimés pour éviter la création de hash différents dû à la présence de caractères parasites.

Ainsi, le header présenté dans la section précédente devient la chaîne de caractères

```
{"alg":"ES256","ppt":"shaken","typ":"passport","x5u":"https://<domaine-
bpco>/certs/<code-apnf-operateur>/<sn-certificat>.cer"}
```

et le payload devient :

```
{"attest":"A","dest":{"tn":["3900"]},"iat":1443208345,
"orig":{"tn":"33123456789"},"origid":"123e4567-e89b-12d3-a456-
426655440000"}
```

Une fois ces éléments prêts, l'empreinte peut être construite. Son format consiste en une chaîne de caractères où les valeurs du *header* et du *payload* sont encodées en base 64 URL (voir la RFC 4648) et jointes par l'utilisation du caractère point « . » (code ASCII 2E) :

```
EMPREINTE STI-AS = BASE64URL(UTF8(PASSport Header)).BASE64URL(PASSport Payload)
```

Commandes pour générer l'empreinte STI-AS à partir du header et du payload.

```
PASSPORT_HEADER_JSON='{"alg":...}'
PASSPORT_PAYLOAD_JSON='{"attest":...}'
PASSPORT_HEADER_B64URL=$(echo $PASSPORT_HEADER_JSON | base64 | tr -d '\n=' | tr --
'+/' '-_')
PASSPORT_PAYLOAD_B64URL=$(echo $PASSPORT_PAYLOAD_JSON | base64 | tr -d '\n=' | tr -
- '+/' '-_')
echo -n "$PASSPORT_HEADER_B64URL.$PASSPORT_PAYLOAD_B64URL" > EMPREINTE.STI_AS
```

Hachage & Chiffrement de l'empreinte

L'empreinte construite est alors hachée en utilisant l'algorithme choisi.

```
HASH EMPREINTE STI-AS = HASH(EMPREINTE STI-AS)
```

Il ne reste plus qu'à chiffrer le hash de l'empreinte avec la clé privée du STI-AS pour obtenir la signature du token.

```
PASSport Signature = ENCRYPT(HASH EMPREINTE STI-AS)
```

Commande openssl pour hacher en utilisant l'algorithme SHA256 et chiffrer avec la clé privée ECDSA.

```
openssl dgst -sha256 -sign private.key -out signature EMPREINTE.STI_AS
```

Dans le cas de clés ECDSA, la signature générée par OpenSSL doit encore être modifiée afin de se conformer au standard JWS. Le contenu final correspond à la valeur encodée en Base64 URL telle qu'attendue dans la section suivante.

```
openssl asn1parse -in signature -inform DER > sig.asn1
cat sig.asn1 | tail -n 2 | sed -e s/.*INTEGER\\s*:// > sig.hex
cat sig.hex | xxd -p -r | base64 | tr -d '\n=' | tr -- '+/' '-_' > sig.base64
```

3.5.1.4 Formatage du token

Le token **PASSport** doit être inclus dans la valeur de l'entête *Identity* dans un format long tel que défini par le **RFC 7515**. Ce format consiste en une chaîne de caractères où 3 valeurs sont encodées en Base64 URL et jointes par l'utilisation du caractère point « . » (code ASCII 2E) :

- Le header du token PASSport
- Le payload du token PASSport
- La signature du token PASSport

```
BASE64URL(UTF8(PASSport Header)).BASE64URL(PASSport Payload).BASE64URL(PASSport Signature)
```

Les valeurs du header et du payload préparées lors de la création de l'empreinte pour la signature du token (§3.5.1.3) sont réutilisées pour cette étape.

3.5.2 Ajout de l'entête Identity

Un entête doit être inclus à la requête SIP, dont la clé est « Identity » et la valeur est une chaîne de caractères composée des éléments suivants, chacun séparé par le caractère « ; » (ASCII code 3B) :

```
Identity: PASSport;info=<URL certificat>;alg=ES256;ppt=shaken
```

- Du token **PASSport** tel que généré en section §3.5.1
- De 3 paramètres dont les valeurs consistent en une paire de clé/valeur :
  - **info :** fournissant l'URL d'accès public au certificat associé à la clé privée utilisée pour la signature du token PASSport
  - **alg :** définit l'algorithme de signature utilisé pour le token PASSport. Sa valeur doit correspondre à la propriété *alg* utilisée dans le header du token.
  - **ppt :** précisant l'extension utilisé dans le token PASSport. Sa valeur doit correspondre à la propriété ppt utilisée dans le header du token.

Il est attendu que l'élément PASSport soit le premier élément de la chaîne, suivi du paramètre *info*. L'ordre des paramètres suivants n'est par contre pas contraint.

Exemple :

```
INVITE sip:+33623456789@10.0.0.1;user=phone SIP/2.0
Via: SIP/2.0/UDP 192.168.0.1;branch=x9hD56xHCfEER2;rport
From: <sip:+331234567890@127.0.0.1;user=phone>;
To: <sip:+33623456789@10.0.0.1;user=phone>
…
Identity: eyJhbGciOiJFUzI1NiIsInBwdCI6InNoYWtlbiIsInR5cCI6InBhc3Nwb3J0IiwieDV
1IjoiaHR0cHM6Ly9kb21haW4tYnBjby9jZXJ0cy9jb2RlLWFwbmYvc24tY2VydGlmaWNhdGUuY2Vy
In0.eyJhdHRlc3QiOiJBIiwiZGVzdCI6eyJ0biI6WyIzMzYyMzQ1Njc4OSJdfSwiaWF0IjoxNjcyN
TI0MDAwLCJvcmlnIjp7InRuIjoiMzMxMjM0NTY3ODkifSwib3JpZ2lkIjoiY2E5MTJjMGEtZTQ1ZS
00YTBjLTlmZDAtYmY5Nzc5YzI4MWNkIn0.N5o6G2GyQkcUyit62PWAltIKuOsbf9tVpHJAhWXax5L
LSFFSm_9s-_hEh7iRiIN_kOX-wwetxs-eY8RngrH8Vg;info=<https://domain-
bpco/certs/code-apnf/sn-certificate.cer>;alg=ES256;ppt=shaken
…
```

3.6 Débrayage Opérateur STI-AS

3.6.1 Procédure de débrayage

La procédure de débrayage du STI-AS remplace la procédure normale détaillée en section §3.5, et consiste pour l'opérateur à inclure au sein du message SIP INVITE de ses appels sortants un entête *P-Identity-Bypass* valorisé avec un token de débrayage fourni par la plateforme MAN.

```
INVITE sip:+33623456789@10.0.0.1;user=phone SIP/2.0
Via: SIP/2.0/UDP 192.168.0.1;branch=x9hD56xHCfEER2;rport
From: <sip:+331234567890@127.0.0.1;user=phone>;
To: <sip:+33623456789@10.0.0.1;user=phone>
…
P-Identity-Bypass: valeur du token de débrayage
```

Si des entêtes *Identity* sont déjà présents dans le message INVITE, il n'est pas nécessaire de les supprimer, les opérateurs en transit et de terminaison devant ignorer ces entêtes si l'entête *P-Identity-Bypass* est présent.

Dans le cas où le débrayage est déclenché suite à une défaillance de l'opérateur, et non à un incident plateforme, l'opérateur doit de plus déclarer un incident sur la météo de la plateforme MAN et fournir au sein du ticket météo la valeur du token associé.

3.6.2 Token de Débrayage

La plateforme MAN attribue à la création d'un opérateur signataire (avec STI-AS) ou d'un OPTS un token de débrayage ; ce token est spécifique à chaque opérateur signataire (avec STI-AS)/OPTS.

Un OPTS débrayant son STI-AS utilise le même token pour tous les appels émis par lui-même et par ses clients opérateurs qui le mandatent comme OPTS.

La valeur du token se représente suivant le format suivant :

CODE_APNF-RANDOMID

où *CODE_APNF* est le code APNF de l'opérateur pour lequel le token a été généré,

et *RANDOMID* est un champ de 1 à 16 caractères maximum parmi 0-9, A-Z, et le caractère "-"

Un token ne doit être utilisé que pour un incident donné.

L'opérateur peut à tout moment demander à la plateforme MAN, via son IHM ou une méthode de l'API GCO dédiée, la génération d'un nouveau token pour être utilisé pour le prochain incident.

3.6.3 Fin de Débrayage

Une fois l'incident terminé, l'opérateur demande un nouveau token de débrayage à la plateforme MAN utilisable en cas de nouvel incident.

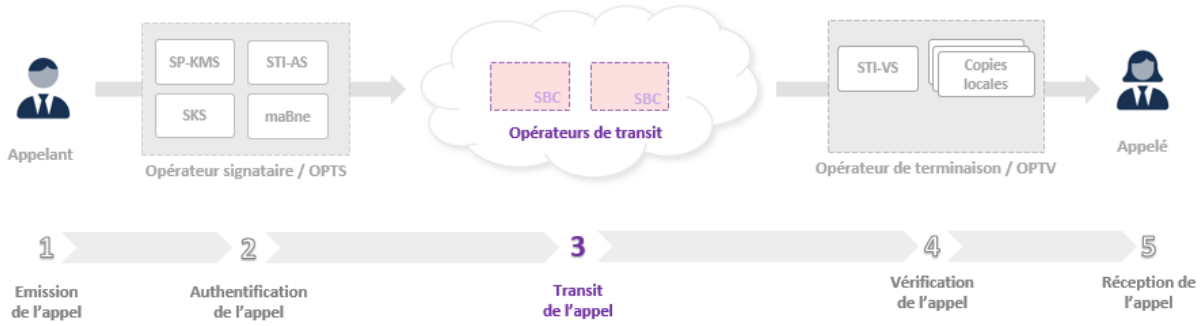
Dans le cas où l'opérateur est à l'origine du débrayage, il doit de plus procéder à la clôture de l'incident sur la plateforme MAN.

4 Procédure de transit des appels

4.1 Opérateurs concernés

- Opérateur de transit

4.2 Contexte d'application



Lors du transit d'un appel SIP, l'opérateur doit s'assurer de la présence dans la requête INVITE du champ *Identity* et de son format général. Si les vérifications échouent, l'opérateur est légitime pour couper l'appel, voir section §2.3.3.

Si l'opérateur se retrouve dans l'incapacité de vérifier les appels en transit, suite à un incident au sein de son SI, il lui est possible de déclencher le débrayage des contrôles (cf. §4.6).

Remarque : un opérateur de transit qui n'est pas en mesure de mettre en place le dispositif prévu dans les règles techniques MAN pour identifier les appels d'urgence ne doit pas, sous sa seule responsabilité, couper les appels contrairement à ce qui est indiqué dans la loi Naegelen.

4.3 Prérequis

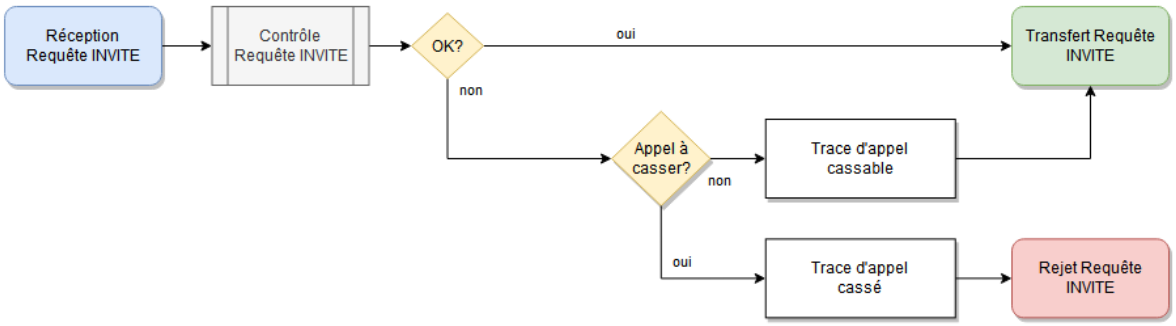
- L'opérateur doit s'être enregistré auprès de la plateforme MAN afin d'accéder aux fonctionnalités de remontée vers la BSM (traces d'appels cassables / cassés, incidents, signalements et volumétries d'appels), voir le document « *Mode opératoire des incidents, signalements et métriques du MAN* »
- L'appel SIP répond aux conditions définies par les règles techniques MAN pour être vérifié par l'opérateur de transit.

4.4 Composants impliqués

- Le **SBC**, élément réseau supportant les messages SIP INVITE entrants et sortants.

4.5 Procédure détaillée

L'opérateur en charge du transit de la requête INVITE d'un appel SIP doit suivre la procédure décrite par le diagramme ci-dessous.



1. Le contenu de la requête est contrôlé afin de vérifier la présence des éléments requis dans le message, §4.5.1. Il est notamment vérifié si l'opérateur émetteur a déclenché sa procédure de débrayage (§3.6).

2. Si le format du message est approuvé, l'opérateur valide la requête INVITE et la transfère.

3. Si le format du message est rejeté :

- l'opérateur vérifie si l'appel doit être effectivement cassé. Les cas pour lesquels l'appel ne doit pas être cassé sont définis en section §2.3.3
- l'opérateur génère une trace d'appel cassé ou cassable pour la remonter dans un second temps en batch à la plateforme MAN. Le mode opératoire des incidents, signalements et métriques du MAN fournit les détails d'une trace d'appel et des champs à renseigner.

Lors du rejet de la requête INVITE, un code erreur spécifique au contrôle doit être retourné dans la réponse SIP. Le tableau suivant liste l'ensemble des contrôles obligatoires attendus de l'opérateur de transit et le code erreur SIP associé. Tous les autres contrôles effectués par l'opérateur de terminaison et disponibles en section §5.5 sont optionnels pour l'opérateur de transit.

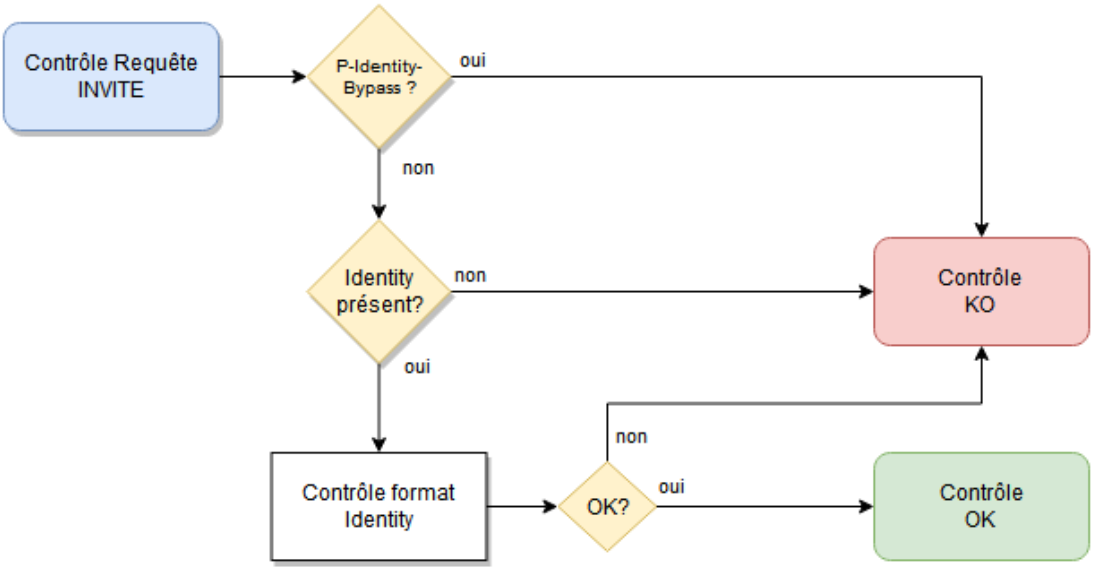
Les codes erreur sont conformes aux standards définis par le 3GPP TS 24.229, la RFC 8224 (section 6.2) et l'ATIS-100082 (section 8.2).

Erreur SIP	Contrôle	Section
428 "Use Identity Header"	L'entête *Identity* est présent	4.5.1.1
437 "Unsupported Credential"	Le paramètre *alg* de l'entête Identity est valorisé à ES256.	4.5.1.3
438 "Invalid Identity Header"	Le paramètre *ppt* de l'entête Identity est présent et est valorisé à « shaken »	4.5.1.3

4.5.1 Contrôle de la requête INVITE

La procédure de contrôle de la requête est simplifiée par rapport à la procédure mise en place lors de la réception par l'opérateur de terminaison (§5) :

- Vérification du débrayage opérateur émetteur par la présence d'un entête *P-Identity-Bypass*
- La présence de l'entête *Identity* est vérifiée
- Le format général de l'entête Identity est contrôlé



4.5.1.1 Présence de l'entête P-Identity-Bypass

L'opérateur en transit doit d'abord vérifier si un entête *P-Identity-Bypass* est présent dans le message INVITE, correspondant à un débrayage de l'opérateur émetteur de son STI-AS lors de l'émission de l'appel. Seule la présence de l'entête doit être contrôlée, sa valeur ne doit pas faire l'objet de vérification, même si celle-ci est vide.

Si c'est le cas, le contrôle du message INVITE s'arrête et l'opérateur devra laisser passer l'appel tel quel. Une trace d'appel cassable devra néanmoins être générée où il sera fait mention qu'un débrayage STI-AS a été enclenché, et les autres champs découlant du traitement de l'entête *Identity* n'auront pas besoin d'être renseignées.

4.5.1.2 Présence de l'entête Identity

Si le débrayage STI-AS n'est pas enclenché, l'entête *Identity* doit être présent dans le message INVITE conformément au profil SIP 3.1. Si ce n'est pas le cas, le message est rejeté avec un code erreur 428 *Use Identity Header*.

4.5.1.3 Contrôle du format de l'entête

Le format de l'entête est ensuite vérifié. L'opérateur de transit doit contrôler le format général de l'entête tel que construit en section §3.5.2. Il n'est pas demandé à l'opérateur de transit une vérification exhaustive de l'entête ; seuls les contrôles mentionnés dans le tableau, section §4.5, sont demandés. Si un contrôle échoue, l'appel est rejeté avec l'erreur associée définie dans le tableau.

4.6 Débrayage procédure de vérification

Dans le cas où l'opérateur n'est plus en capacité de vérifier les appels transitant par son réseau, il ne doit pas casser les appels mais les laisser passer. L'opérateur doit par contre prévenir les autres opérateurs à travers la création d'un ticket d'incident météo au niveau de la plateforme MAN.

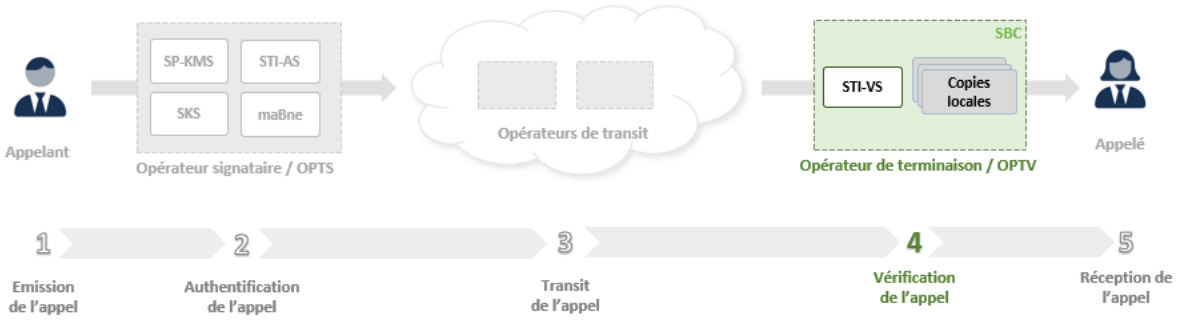
L'opérateur confirme la résolution du problème par la clôture de l'incident sur la plateforme MAN.

5 Procédure de vérification des appels (STI-VS)

5.1 Opérateurs concernés

- Opérateur de terminaison
- OPTV

5.2 Contexte d'application



Lors de la réception d'un appel SIP, l'opérateur de terminaison doit valider l'authenticité de l'appel en effectuant les vérifications suivantes :

- l'entête *Identity* doit être présent dans la requête INVITE reçue,
- la signature de l'appel doit être valide,
- les données incluses dans le token PASSport sont cohérentes par rapport aux données de la requête INVITE

Si une de ces vérifications échouent, l'opérateur est légitime à couper l'appel, voir section §2.3.3.

Si l'opérateur se retrouve dans l'incapacité de vérifier les appels reçus, suite à un incident au sein de son SI ou au niveau de la plateforme MAN (cf. §15), il lui est possible de déclencher le débrayage de son STI-VS (voir le code de procédures MAN).

5.3 Prérequis

- L'opérateur doit s'être enregistré auprès de la plateforme MAN afin d'accéder aux fonctionnalités de mise en place et de maintenance des copies locales (§6) et de remontée vers la BSM (voir le document « *Mode opératoire des incidents, signalements et métriques du MAN* »)
- Le STI-VS de l'opérateur doit avoir mis en place ses copies locales de certificats opérateur et CRL (§9)
- Le STI-VS de l'opérateur doit pouvoir accéder à la BPCO pour récupérer les certificats opérateur si jamais ceux-ci ne se trouvent pas dans la copie locale de l'opérateur
- L'appel répond aux conditions définies par les règles techniques MAN pour être vérifié par l'opérateur.

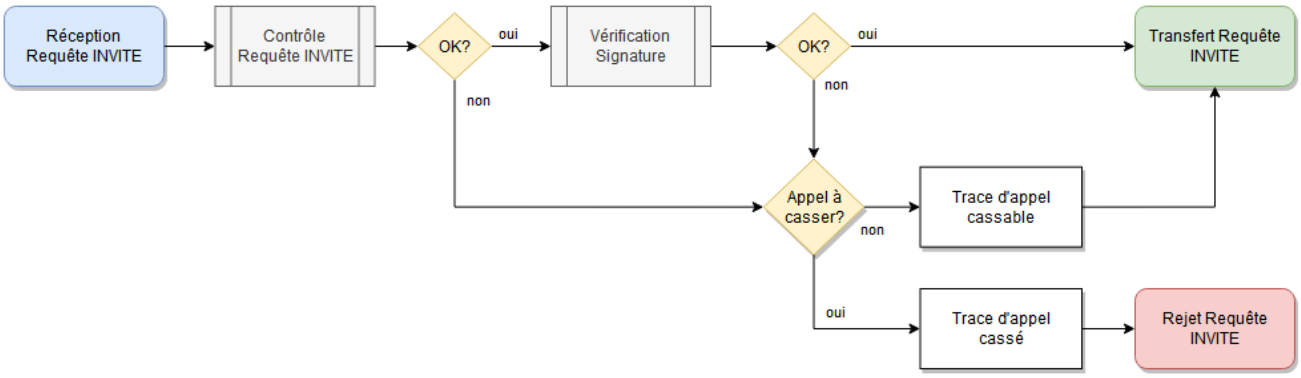
5.4 Composants impliqués

- Le **SBC**, élément réseau supportant les messages SIP INVITE entrants et sortants. Il appelle le **STI-VS** pour tout ou partie de la vérification de l'appel
- Le **STI-VS** de l'opérateur, en charge de la vérification

5.5 Procédure détaillée

Une fois les systèmes de copies locales mis en place, la procédure à effectuer par le STI-VS de l'opérateur est décrite ci-dessous. Les contrôles et codes erreur à retourner se conforment aux standards définis par le 3GPP TS 24.229, la RFC 8224 (section 6.2) et l'ATIS-100082 (section 8.2).

1. Le contenu de la requête INVITE est contrôlé afin de vérifier la présence des éléments requis dans le message, et notamment le contenu de l'entête Identity (§5.5.2). Il est notamment vérifié si l'opérateur émetteur a déclenché sa procédure de débrayage (§3.6).
2. Si les données de la requête sont validées, la signature de l'appel est vérifiée afin de valider l'authenticité de celui-ci (§5.5.3).
3. Si le format du message est approuvé, l'opérateur valide la requête INVITE.
4. Si le format du message est rejeté :
   - l'opérateur vérifie si l'appel doit être effectivement cassé. Les cas pour lesquels l'appel ne doit pas être cassé sont définis en section §2.3.3
   - l'opérateur génère une trace d'appel cassé ou cassable pour la remonter dans un second temps en batch à la plateforme MAN. Le mode opératoire des incidents, signalements et métriques du MAN explicite le format d'une trace d'appel et des champs à renseigner.



5.5.1 Codes Erreur SIP

Lors du rejet de la requête INVITE, un code erreur spécifique au contrôle doit être retourné dans la réponse SIP en tant que *SIP Status Code* afin de casser l'appel.

Le tableau suivant liste l'ensemble des contrôles attendus et le code SIP à utiliser si le contrôle échoue. Pour chaque code est fournie la *Reason Phrase* à associer par défaut au code SIP dans la réponse au message INVITE. Conformément à la RFC 3261, l'opérateur est libre d'utiliser une autre valeur que celle fournie dans ce tableau s'il souhaite apporter des détails supplémentaires quant à l'erreur remontée.

Dans cette optique, le mode opératoire fournit au sein des sections suivantes des propositions de *Reason Phrase* plus détaillées pouvant être utilisées afin de préciser le contexte de l'erreur.

Tous les contrôles listés sont obligatoires pour l'opérateur de terminaison (ou l'OPTV).

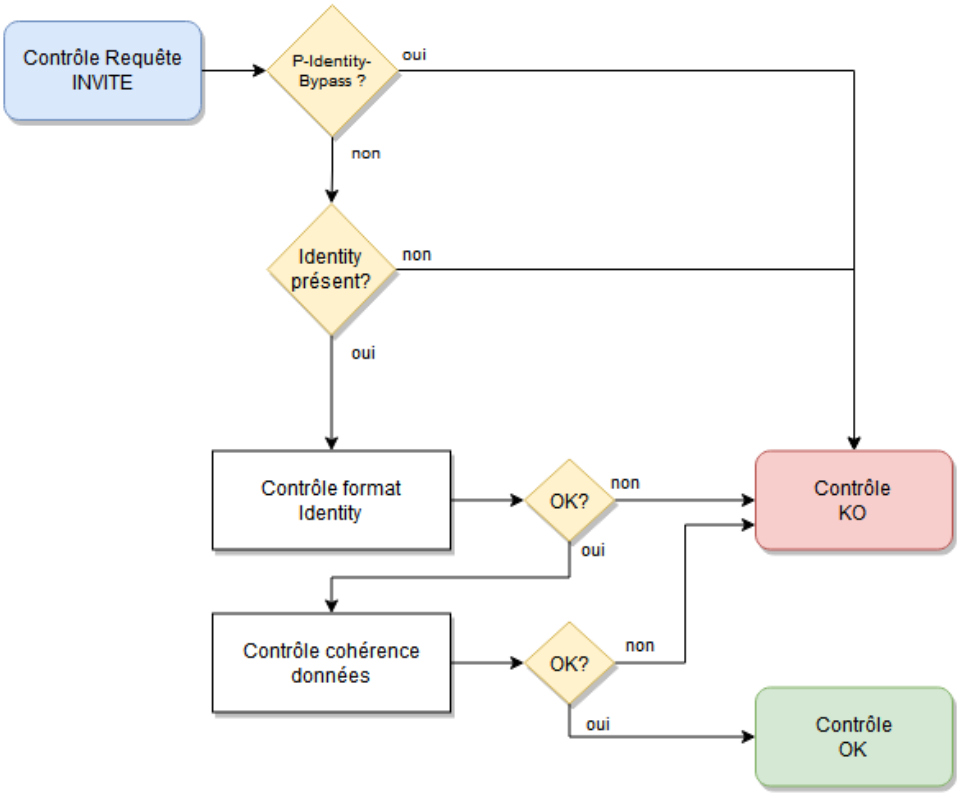
Il est à noter que dans le cas où l'appel ne doit pas être cassé, le code erreur est retourné dans un entête Reason, et non en tant que code retour SIP dont la valeur 200 est utilisé (cf. §2.3.3)

Erreur SIP **Raison par défaut**	**Contrôle**	**Section**
400 Bad Request	Le paramètre *time*, *from* et *to* peuvent être valorisés par le SBC pour créer la requête de vérification au STI-VS.	5.5.2.3
403 Stale Date	Si présent, l'entête *Date* du message SIP INVITE est égal à l'heure actuelle à plus ou moins 60s.	5.5.2.3
	Le paramètre *iat* du payload du token PASSport est égal au paramètre « time » à plus ou moins 60s.	5.5.2.4
428 Use Identity Header	L'entête *Identity* est présent dans le message SIP	5.5.2.2
436 Bad Identity Info	Le paramètre *info* de l'entête *Identity* est présent et est une URL HTTPs valide	5.5.2.3
	Le header du token PASSport contient 4 paramètres *alg, ppt, typ, x5u*	5.5.2.4
	Le paramètre *x5u* du header du token PASSport est identique au paramètre *info* de l'entête Identity	5.5.2.4
	Le certificat opérateur a pu être récupéré	5.5.3.1
437 Unsupported Credential	Le paramètre *alg* de l'entête *Identity* est présent et est valorisé à ES256.	5.5.2.3
	La valeur du paramètre *alg* du header du token PASSport est valorisé à ES256.	5.5.2.4
	La valeur du paramètre *typ* du header du token PASSport est valorisé à « passport »	5.5.2.4
	Tous les certificats de l'autorité utilisés dans la chaîne de certification sont valides, n'ont pas expiré et ne sont pas révoqués	5.5.3.3
	Le certificat opérateur n'a pas expiré	5.5.3.2
	La date de début de validité du certificat est dans le passé	5.5.3.2
	Le certificat opérateur n'est pas révoqué	5.5.3.2
438 Invalid Identity Header	La valeur de l'entête *Identity* est composée de 4 sections séparées par un point-virgule	5.5.2.3
	Le champ PASSport est présent en première position de l'entête *Identity* et utilise le format long.	5.5.2.3
	Le paramètre *ppt* de l'entête *Identity* est présent et valorisé à shaken.	5.5.2.3
	La valeur du paramètre *ppt* du header du token PASSport est valorisé à « shaken »	5.5.2.4
	Le payload du token PASSport contient 5 paramètres *attest, dest, iat, orig, origid*	5.5.2.4
	La valeur du paramètre *attest* du payload du token PASSport est valorisée à « A », « B » ou « C »	5.5.2.4

	La valeur spécifiée par la propriété *tn* contenue dans le paramètre *orig* du payload du token PASSport correspond à l'entête SIP « From » de la requête INVITE reçue.	5.5.2.5
	La valeur spécifiée par la propriété *tn* contenue dans le paramètre *dest* du payload du token PASSport correspond à l'entête SIP « To » de la requête INVITE reçue.	5.5.2.5
	La signature de l'appel a pu être vérifiée	5.5.3.4

5.5.2 Contrôle de la requête INVITE

L'entête *Identity* de la requête INVITE est vérifié de manière étendue afin de contrôler le format des données et leur cohérence avec les autres entêtes de la requête.



5.5.2.1 Présence de l'entête P-Identity-Bypass

L'opérateur doit d'abord vérifier si un entête *P-Identity-Bypass* est présent dans le message INVITE, correspondant à un débrayage de l'opérateur émetteur de son STI-AS lors de l'émission de l'appel.

Si c'est le cas, le contrôle du message INVITE s'arrête et l'opérateur devra laisser passer l'appel tel quel. Une trace d'appel cassable devra néanmoins être générée où il sera fait mention qu'un débrayage STI-AS a été enclenché, et la valeur de l'entête *P-Identity-Bypass* renseignée en tant que token de débrayage. Il n'est pas nécessaire de fournir les valeurs des autres champs découlant normalement du traitement de l'entête *Identity*. Le mode opératoire des incidents, signalements et métriques du MAN explicite le format d'une trace d'appel et des champs à utiliser.

Il est à noter qu'il n'est pas attendu que l'opérateur effectue des contrôles sur la valeur associée à l'entête *P-Identity-Bypass*, même si cette valeur est vide. Les contrôles seront effectués en aval par l'APNF lors de la revue de l'utilisation du débrayage par les opérateurs émetteurs dans les traces d'appels cassables.

5.5.2.2 Présence de l'entête Identity

Si le débrayage STI-AS n'est pas enclenché, l'entête *Identity* doit être présent dans le message INVITE conformément au profil SIP 3.1 et ultérieur. Si ce n'est pas le cas, le message est rejeté avec un code erreur 428 *Use Identity Header* à la différence de la préconisation de l'ATIS-1000082 proposant le code erreur générique 400 en attendant que tous les opérateurs savent signer leurs appels.

5.5.2.3 Contrôle du format de l'entête Identity

Le SBC passe alors la main au STI-VS afin de contrôler l'entête Identity. Les données à contrôler sont fournies par le SBC dans sa requête au STI-VS, telle que précisée par l'ATIS-100082 section 8.2.1, où doivent être présents :

- Un paramètre « from », valorisé avec le champ FROM du composant SBC, à partir de l'entête SIP « From », voire « P-Asserted-Identity » si l'entête « From » vaut *anonymous@anonymous.invalid* ou *unavailable@unknown.invalid* (cf §2.8.6)
- Un paramètre « to », valorisé avec le champ TO du composant SBC, valeur déduite de l'entête SIP To de la requête INVITE suivant les règles techniques MAN.
- Un paramètre « time », défini avec la valeur de l'entête Date du message SIP INVITE si présent, ou sinon la date et heure locale du composant SBC.
- Un paramètre « identity » contenant la valeur de l'entête Identity présent dans le message SIP INVITE.

Si les paramètres « from », « to » ou « time » ne peuvent être renseignés, un code erreur 400 doit être retourné. Le paramètre « identity » a quant à lui déjà été validé au sein de la section précédente. Afin de préciser l'erreur rencontrée, l'opérateur peut associer au code erreur 400 une *Reason Phrase* plus explicite que la valeur par défaut *Bad Request* :

Contrôle	Code SIP	Reason Phrase proposée
La valeur pour le numéro appelant ne peut être récupérée du message SIP INVITE	400	Missing parameter « from »
La valeur pour le numéro appelé ne peut être récupérée du message SIP INVITE	400	Missing parameter « to »
La valeur pour le numéro appelant n'est pas dans le format attendu par le mécanisme de confiance	400	Invalid parameter « from »
La valeur pour le numéro appelé n'est pas dans le format attendu par le mécanisme de confiance	400	Invalid parameter « to »
La valeur de l'entête *Date* est invalide	400	Invalid parameter « time »

Si présent et si l'entête Date du message SIP INVITE est différent de la date locale du composant SBC de plus de 60 secondes (dans le passé ou le futur), une erreur 403 *Stale Date* doit être retournée. **Il est par conséquent important que tous les opérateurs s'assurent que l'horloge de leurs composants soit bien à jour, sous peine de voir l'ensemble des appels coupés suite à ce contrôle.**

Le format de l'entête *Identity* est ensuite vérifié par l'envoi de la requête de vérification au STI-VS; Pour rappel, le format général de l'entête tel que construit en section 3.5.2 est le suivant :

```
Identity: PASSporT;info=<URL certificat>;alg=ES256;ppt=shaken
```

Les contrôles à effectuer par le STI-VS sont les suivants. Au premier contrôle échoué le code erreur associé est retourné. Il est à noter que pour la génération de la trace d'appel à remonter (cassé ou cassable) des informations de l'entête soit demandées, si présentes et interprétables.

Contrôle valeur	Code erreur SIP
Le champ PASSporT doit être présent et doit utiliser le format long	438 *Invalid Identity Header*
Le paramètre *ppt* doit être présent et valorisé à *shaken*.	*438 Invalid Identity Header*
Le paramètre *info* doit être présent et correspondre à une URL HTTPs valide	436 *Bad Identity Info*

5.5.2.4 Contrôle du format du token PASSport

Le token PASSport doit être un JSON Web Token au format long tel que décrit par la section §3.5.1. Il consiste en une chaîne de caractères où 3 valeurs sont encodées en base 64 URL et jointes par l'utilisation du caractère point « . » (code ASCII 2E) :

- Le header du token PASSport
- Le payload du token PASSport
- La signature du token PASSport

```
BASE64URL(UTF8(Header)).BASE64URL(Payload).BASE64URL(Signature)
```

Header

Une fois décodé, l'entête doit être un objet JSON avec obligatoirement et seulement les propriétés ci-dessous. Si une propriété n'est pas présente, le code erreur 436 *Bad Identity Info* doit être retourné.

Le contrôle de leur valeur associée est alors effectué et si un des contrôles échoue le code erreur correspondant est retourné.

Propriété	Contrôle valeur	Code erreur SIP
alg	La valeur doit correspondre à ES256.	437 *Unsupported Credential*
ppt	La valeur doit être « shaken »	438 I*nvalid Identity Header*
typ	La valeur doit être « *passport* »	437 *Unsupported Credential*

x5u	La valeur doit être identique au paramètre info de l'entête Identity	436 *Bad Identity Info*

Payload

Une fois décodé, le payload est un objet JSON avec obligatoirement les propriétés ci-dessous. Si une propriété n'est pas présente, le code erreur 438 *Invalid Identity Header* doit être retourné.

Le contrôle de leur valeur associée est alors effectué et si un des contrôles échoue le code erreur correspondant est retourné.

Propriété	Contrôle valeur	Code erreur SIP
attest	Sa valeur doit être « *A* », « *B* », ou « *C* »	438 I*nvalid Identity Header*
dest	Objet JSON composé d'une propriété unique *tn* dont la valeur est une liste de chaînes de caractères contenant une seule valeur devant correspondre à un numéro de téléphone conforme au format canonique spécifié dans la RFC 8224 (section 8.3).	Voir section §5.5.2.5
iat	Date d'émission de l'appel. Entier représentant le nombre de secondes après le 1er janvier 1970 selon le fuseau horaire UTC. Cette date doit avoir un écart inférieur à 60 secondes (dans le passé ou le futur) avec le paramètre « time » fourni dans la requête de vérification.	403 *Stale Date*
orig	Objet JSON composé d'une propriété unique *tn* dont la valeur est une chaîne de caractères devant correspondre à un numéro de téléphone conforme au format canonique spécifié dans la RFC 8224 (section 8.3).	Voir section §5.5.2.5
origid	Sa valeur doit être un UUID valide, tel que décrit dans le RFC 4122	-

Le format des paramètres *orig* et *dest* est contrôlé via la cohérence de leurs valeurs par rapport aux données véhiculées dans la requête INVITE, cf. §5.5.2.5.

Quant au paramètre *origid*, sa valeur doit être un UUID valide mais aucun contrôle n'est requis par les standards.

Signature

La signature sera validée ultérieurement, §5.5.3.

5.5.2.5 Contrôle de cohérence des données

Le STI-VS doit enfin vérifier que les données contenues dans le token PASSport sont cohérentes avec les informations de la requête INVITE et en particulier de l'entête *Identity*. Une erreur 438 *Invalid Identity Header* doit être retournée si un de ces contrôles échoue.

Paramètre « orig » du payload PASSport

La valeur spécifiée par la propriété *tn* contenue dans le claim *orig* du payload du token PASSport doit correspondre, suivant les règles définies au sein du document « *MAN_Regles techniques* », au numéro reçu par le composant SBC dans l'entête SIP « From », voire « P-Asserted-Identity » si l'entête « From » vaut *anonymous@anonymous.invalid* ou *unavailable@unknown.invalid* (cf §2.8.6).

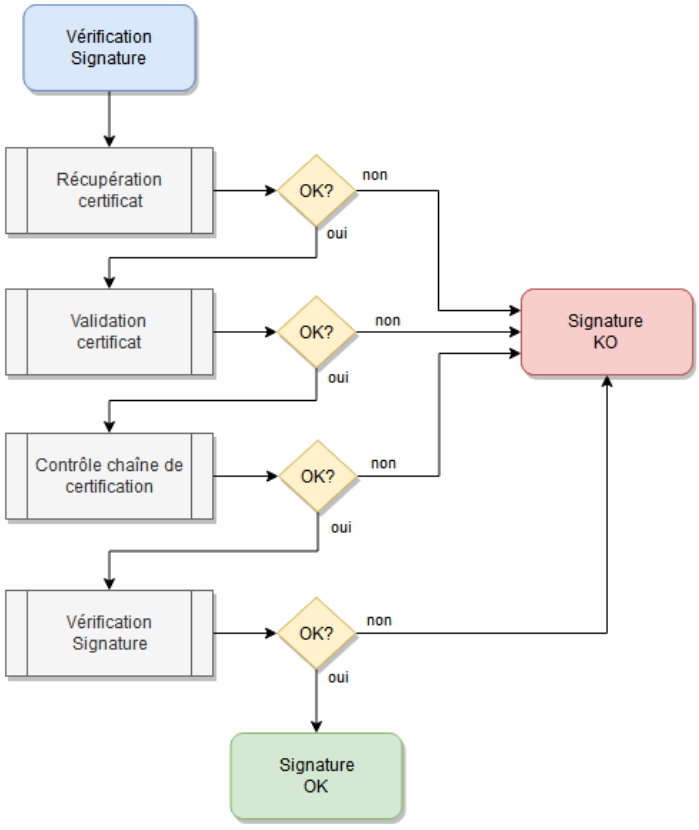
Paramètre « dest » du payload PASSport

La valeur spécifiée par la propriété *tn* contenue dans le claim *dest* du payload du token PASSport doit correspondre, suivant les règles définies au sein du document « *MAN_Regles techniques* », au numéro reçu par le composant SBC dans l'entête SIP « To ».

5.5.3 Vérification de la signature

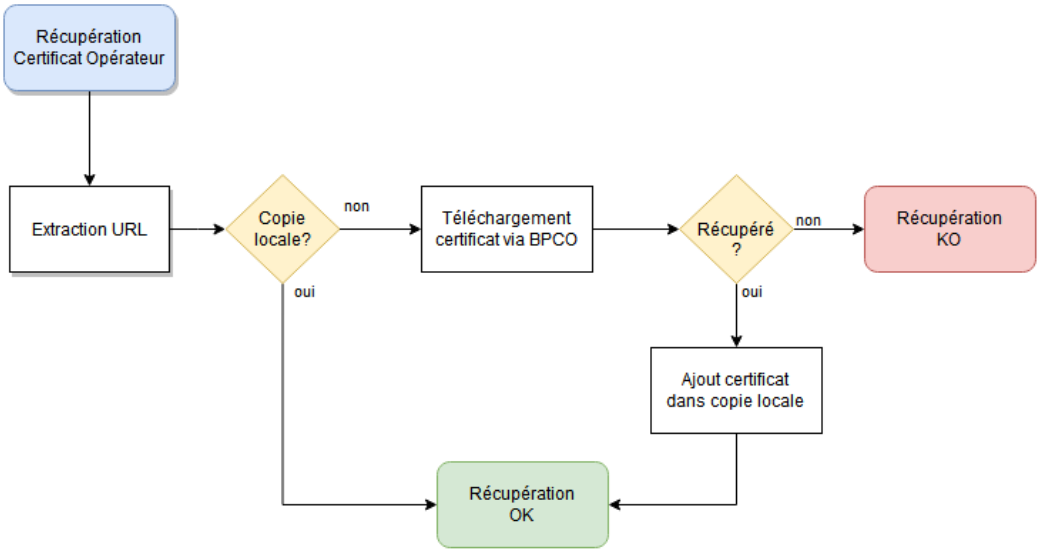
Afin de valider la signature présente dans le champ Identity, il convient de :

- Récupérer le certificat (§5.5.3.1)
- Vérifier la validité du certificat opérateur (§5.5.3.2)
- Vérifier la chaîne de certification complète du certificat (§5.5.3.3)
- Si le certificat est validé, la vérification de la signature peut être effectuée (§5.5.3.4).



5.5.3.1 Récupération du certificat opérateur

La procédure de vérification du certificat opérateur fait intervenir les copies locales mises en place au niveau du STI-VS afin d'éviter de récupérer les certificats directement de la BPCO pour chaque tentative d'appel reçu.



Extraction de l'URL

L'opérateur de terminaison extrait du paramètre *info* du champ *Identity* le chemin public d'accès au certificat de l'opérateur signataire, correspondant pour le modèle français à une URL de la BPCO (§2.7.1.1).

Récupération du certificat

Le STI-VS récupère le certificat de sa copie locale créée préalablement et tenu à jour (§9.6.1). Le mécanisme d'extraction du certificat de la copie locale n'est pas défini dans ce document, dépendant de l'implémentation faite par l'opérateur pour la mise en place de sa copie locale des certificats.

Si le certificat n'existe pas dans la copie locale de l'opérateur, le STI-VS récupère alors le certificat directement via l'URL.

- Si le certificat n'est pas accessible, c'est-à-dire si la BPCO retourne un code http 404, il est considéré que le certificat n'est pas accessible.
- Si le certificat a pu être récupéré, le STI-VS ajoute le certificat dans sa copie locale avant de continuer.

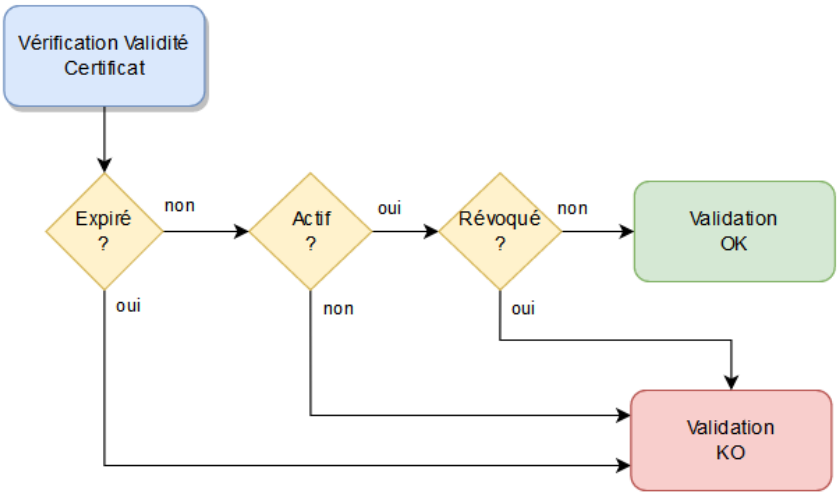
Dans le cas où le certificat ne peut être récupéré, une erreur 436 *Bad Identity Info* doit être retournée.

5.5.3.2 Vérification de la validité du certificat

Une fois le certificat récupéré, les contrôles devant s'appliquer sur ce dernier sont :

- Le certificat ne doit pas être expiré, i.e. la date de fin de validité du certificat doit être dans le futur

- La date de début de validité du certificat ne doit pas être dans le futur
- La non révocation du certificat doit être vérifiée en vérifiant que le certificat n'est pas présent dans la copie locale de la CRL des certificats opérateurs (§2.7.2).

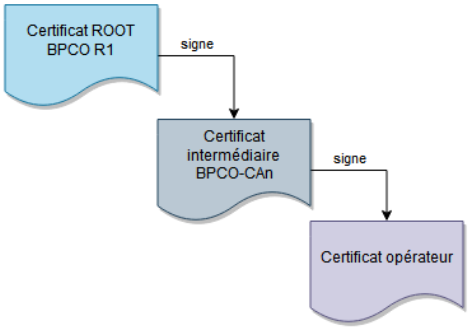


L'erreur 437 *Unsupported Credential* doit être retournée si un de ces contrôles échoue.

5.5.3.3 Contrôle de la chaîne de certification

Afin de considérer le certificat valide, il convient enfin de vérifier l'ensemble de la chaîne de certification :

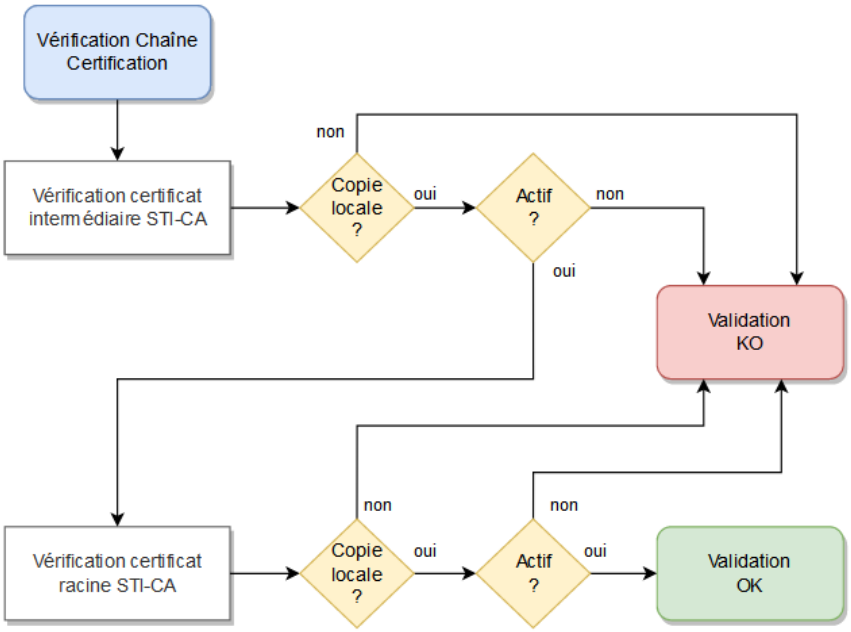
- Le certificat opérateur doit être signé par un certificat intermédiaire de l'autorité de certificat et ce dernier est toujours valide.
- Le certificat intermédiaire doit être signé par le certificat racine de l'autorité de certification et ce dernier est toujours valide



Les vérifications suivantes doivent ainsi être effectuées pour les certificats intermédiaires et racine :

- Le certificat doit être inclus dans la copie locale des certificats de l'autorité de certification (§9.6.3 & 9.7.3), pour vérifier en particulier
  - Que le certificat intermédiaire n'a pas été révoqué

- Que les certificat intermédiaire et racine n'ont pas expiré
- les certificats intermédiaire et racine doivent être actifs, c'est-à-dire que leur date de début de validité ne doit pas être dans le futur



L'erreur 437 *Unsupported Credential* doit être retournée si un de ces certificats est invalide.

5.5.3.4 Vérification de la signature

Une fois le certificat opérateur validé, la signature doit être vérifiée en utilisant les informations incluses dans le token PASSport. Pour rappel, les éléments du token PASSport sont disponibles dans la valeur de l'entête Identity suivant ce format :

```
Identity: BASE64URL(UTF8(PASSport Header)).BASE64URL(PASSport Payload).BASE64URL(PASSport Signature);info=<URL certificat>;ppt=shaken;alg=ES256
```

Exemple champ Identity complet

```
Identity: eyJhbGciOiJFUzI1NiIsInBwdCI6InNoYWtlbiIsInR5cCI6InBhc3Nwb3J0IiwieDV1IjoiaHR0cHM6Ly9kb21haW4tYnBjby9jZXJ0cy9jb2RlLWFwbmYvc24tY2VydGlmaWNhdGUuY2VyIn0.eyJhdHRlc3QiOiJBIiwiZGVzdCI6eyJ0biI6WyIzMzk4NzY1NDMyMSJdfSwiaWF0IjoxNjg0MTMwNjgyLCJvcmlnIjp7InRuIjoiMzMxMjM0NTY3ODkifSwib3JpZ2lkIjoiZGE5MTJjMGEtZTQ1ZS00YTBjLTlmZDAtYmY5Nzc5YzI4MWNkIn0.Ai3g8fXIldcerfcEq9P-4tS-fShYqgsaBkIYaZuYoeefxvckg69Qog8CbqkTj7087zvhcGZXmoufOQ7ruvsMUA;info=<https://domain-bpco/certs/code-apnf/sn-certificate.cer>;ppt=shaken;alg=ES256
```

Note : Comme indiqué en section §3.5.1.3, il est recommandé d'utiliser des librairies standards implémentant les standards JWT et JWS afin de vérifier la signature du token PASSporT. Cette section présente un procédé de vérification en décortiquant les opérations effectuées et en fournissant des exemples avec l'outil openssl disponible sur la majorité des systèmes d'exploitation, mais l'utilisation de librairies permettra de simplifier l'implémentation du mécanisme de confiance.

1. Le STI-VS recrée d'abord une empreinte qui sera utilisée comme point de comparaison. Du token PASSport sont extraites les valeurs encodées de l'entête et du payload et séparées par un « . » (code ASCII 2E).

```
EMPREINTE STI-VS = BASE64URL(UTF8(PASSport Header)).BASE64URL(PASSport Payload)
```

2. Le STI-VS récupère à présent la signature du STI AS de l'opérateur signataire, présente dans la valeur de l'entête *Identity*.

```
SIGNATURE STI_AS = BASE64URL_DECODE(BASE64URL(PASSport Signature))
```

Commande shell pour décoder la signature encodée en base64 URL :

```
echo <BASE64URL(PASSport Signature)> | awk '{ if (length($0) % 4 == 3) print $0"=";
else if (length($0) % 4 == 2) print $0"=="; else print $0; }' | tr -- '-_' '+/' |
base64 -d > signature.sti_as
```

3. Le STI-VS extrait ensuite du certificat opérateur la clé publique. Cette dernière est alors utilisée pour vérifier la signature décodée avec l'empreinte du token PASSporT.

```
VERIFY_SIGN_WITH_PUBKEY (EMPREINTE STI-VS, SIGNATURE STI_AS, PUBKEY)
```

Si cette procédure échoue, l'erreur 438 *Invalid Identity Header* doit être retournée.

Commandes pour vérifier avec openssl la signature avec l'empreinte

OpenSSL attendant une signature ECDSA avec un format différent que celui fourni par le JSON Web Token, il convient de modifier la signature récupérée du token PASSporT afin de recréer la structure attendue par openssl :

```
cat signature.sti_as | xxd -p -u -c 32 > signature.sti_as.hex

echo -n "" > .der.struct.tmp
addToSig () {
    SIG_INT_FIRST_CHAR_DEC=$(echo $((16#$(echo $1 | head -c 2))))
    if [ $(echo $(($SIG_INT_FIRST_CHAR_DEC & 128))) -eq 128 ]; then
        echo -ne "\x02\x21\x00" >> .der.struct.tmp
    else
        echo -ne "\x02\x20" >> .der.struct.tmp
    fi
    echo -n $1 | xxd -p -r >> .der.struct.tmp
}

addToSig $(head -n 1 signature.sti_as.hex)
addToSig $(tail -n 1 signature.sti_as.hex)

echo -en "\x30\x$(printf '%x' $(cat .der.struct.tmp | wc -c))" > stias.openssl.sig
cat .der.struct.tmp >> stias.openssl.sig
```

Une fois la structure recréée, openssl peut être appelé pour effectuer la vérification de signature :

```
openssl dgst -sha256 -verify public.key -signature stias.openssl.sig <EMPREINTE
STI-VS>
```

5.6 Débrayage STI-VS

Dans le cas où l'opérateur n'est plus en capacité de vérifier les appels reçus, il ne doit pas casser les appels mais les laisser passer en débrayant son STI-VS dont la procédure est à sa discrétion. L'opérateur doit par contre prévenir les autres opérateurs à travers la création d'un ticket d'incident météo au niveau de la plateforme MAN.

L'opérateur confirme la résolution du problème par la clôture de l'incident sur la plateforme MAN.

6 Enregistrement d'un opérateur

6.1 Opérateurs concernés

Tous les opérateurs de la communauté MAN sont concernés. Les critères d'appartenance à la communauté MAN, les processus d'admission et les obligations sont décrites dans le code de procédures MAN.

6.2 Contexte d'application

L'enregistrement d'un opérateur auprès de la plateforme MAN lui permet d'accéder à l'IHM de la plateforme et à l'ensemble des fonctions d'API de la solution GCO, énumérées en section §13.2, et en particulier la délivrance de certificats pour les STI-AS, la gestion des copies locales des STI-VS.

Lors de cette phase, le gestionnaire de la plateforme MAN crée l'opérateur et le compte de l'administrateur opérateur MAN; l'opérateur a ensuite la charge de la gestion de ses propres comptes utilisateurs.

6.3 Procédure détaillée

La procédure d'enregistrement d'un opérateur est détaillée au sein du code de procédures MAN. Il est néanmoins important de noter que lors cette procédure, l'opérateur devra fournir les informations nécessaires lui permettant d'utiliser le mécanisme de confiance de manière optimale. Il devra ainsi fournir parmi d'autres éléments :

- Le point de contact au niveau opérateur pouvant répondre aux questions administratives
- Si l'opérateur peut agir en tant qu'OPTS, la liste des opérateurs signataires étant autorisés à le sélectionner comme OPTS, ceci afin de s'assurer que cet OPTS ne soit accessible que par les opérateurs habilités.
- Les adresses email à utiliser pour les notifications envoyées par la plateforme :

Liste de notification	Type de notifications
Général	Notifications liées à la plateforme en général : maintenance, incidents…
Certificats	Notifications liées aux certificats de l'opérateur : demande/confirmation de création, révocation, renouvellement…
Légal	Notifications liées à la vérification administrative de l'identité de l'opérateur
Dépôt	Notifications liées au dépôt de fichiers de traces et volumétries. Voir Mode opératoire des incidents, signalements et métriques du MAN.

Une vérification des renseignements fournis et du point de contact sera entreprise avant l'approbation de l'opérateur et de son enregistrement au niveau de la plateforme. Le point de contact se verra créer un compte avec mise en place d'une double-authentification lui permettant d'ajouter de nouveaux utilisateurs et de commencer à mettre en place le mécanisme de confiance chez son opérateur.

Une fois l'opérateur créé, les administrateurs de celui-ci pourront mettre à jour les informations liées au profil opérateur (listes de diffusion, listes des opérateurs liés à l'OPTS …), et ceci en dehors de la procédure d'enregistrement initial de l'opérateur.

7 Vérification périodique d'un opérateur

7.1 Opérateurs concernés

Tous les opérateurs enregistrés sur la plateforme MAN.

7.2 Contexte d'application

En plus de l'enregistrement initial (§6), une vérification de chaque opérateur enregistré sur la plateforme MAN est réalisée annuellement afin de s'assurer que les données associées à l'opérateur soient toujours valides et que ce dernier, si son rôle l'exige, continuer à se voir délivrer des certificats.

7.3 Prérequis

- L'opérateur a été enregistré dans la plateforme MAN (§6)

7.4 Procédure détaillée

La procédure de vérification annuelle d'un opérateur est détaillée au sein du code de procédures MAN.

Elle commence 9 mois après l'enregistrement de l'opérateur ou de la dernière vérification, laissant 3 mois pour être finalisée. Afin de faciliter la procédure, il est ainsi recommandé aux opérateurs de contacter l'APNF dès la prise de connaissance de changements impactant leur gestion du mécanisme de confiance (modification du point de contact, listes de diffusions…), afin que ces changements puissent être validés et répercutés en amont de la vérification.

Il est à noter que pendant cette phase de vérification l'opérateur pourra toujours créer de nouveaux certificats et gérer ses existants.

Par contre, si cette vérification échoue, il ne sera plus possible pour un opérateur signataire ou OPTS de se voir délivrer de nouveaux certificats.

8 Délivrance des certificats opérateurs

8.1 Opérateurs concernés

- Opérateur signataire
- OPTS

8.2 Contexte d'application

Lors de la signature d'un appel SIP, l'opérateur en charge de cette signature doit fournir au sein du token **PASSport** et de l'entête Identity le chemin dans la BPCO du certificat contenant la clé publique permettant aux opérateurs de terminaison de vérifier cette signature.

L'opérateur doit par conséquent passer par la solution GCO de la plateforme MAN pour se voir délivrer un certificat, qui sera alors mis à disposition via une URL publique de la BPCO (§2.7.1.1).

8.3 Prérequis

La délivrance de certificats peut être réalisée via l'IHM ou l'API de la plateforme MAN. Les prérequis dépendent par conséquent du mode utilisé par l'opérateur, ainsi que du rôle de l'opérateur dans la phase de délivrance.

Prérequis généraux :

- L'opérateur doit s'être enregistré auprès de la plateforme MAN pour accéder aux fonctionnalités de délivrance de certificats opérateur (§6).
- L'opérateur doit avoir été vérifié administrativement pour avoir le droit de se voir délivrer un certificat opérateur (§7).
- Utilisation de l'IHM : l'utilisateur dispose d'un compte utilisateur de type « Administrateur » ou « Gestionnaire de certificats ».
- Utilisation de l'API : Un *API credential* a été généré par l'opérateur au niveau de la plateforme pour pouvoir s'authentifier auprès de ses APIs en utilisant le protocole OAuth 2 (§13.1.2).

Pour la délivrance d'un certificat direct à un opérateur signataire

- L'utilisateur ou le composant de l'opérateur signataire dispose d'une paire de clés de chiffrement. Les exigences de création des clés sont définies en section §2.6.6.

Pour la demande d'un certificat indirect par un opérateur signataire

Aucun prérequis additionnel.

Pour la délivrance d'un certificat indirect à un OPTS

- L'utilisateur ou le composant de l'OPTS dispose d'une paire de clés de chiffrement. Les exigences de création des clés sont définies en section §2.6.6.

8.4 Composants impliqués

- Le KMS de l'opérateur (**SP-KMS**), en charge de la création de la paire de clés publique/privée et de la procédure de génération du certificat auprès de l'autorité de certification.
- Le **SKS** de l'opérateur, pour stocker la clé privée utilisée pour signer l'appel.

8.5 Politique de gestion des certificats

Le code de procédures MAN fournit aux opérateurs un ensemble de directives et de préconisations quant à la bonne gestion de leurs certificats. Il convient par conséquent de s'y référer afin de s'assurer d'une mise en place la plus pérenne dans le temps et prévenir tout risque lié à des défaillances éventuelles des composants.

8.6 Mode de délivrance des certificats

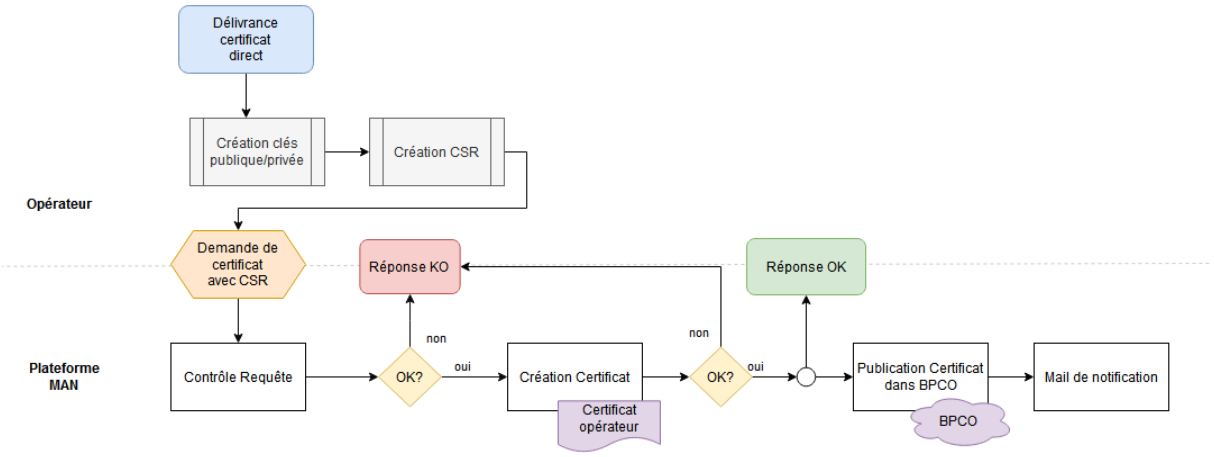
La création des certificats X.509 passe par la fourniture d'un fichier CSR, fichier contenant l'identité de l'opérateur mais aussi la clé publique qui servira à la vérification de la signature des appels.

Le fichier CSR devant être signé par la clé privée, il doit obligatoirement être généré par l'entité disposant de la clé privée. Pour des raisons de sécurité, il est convenu que la clé privée doit rester confidentielle et ne doit en aucun cas être divulguée.

L'opérateur est par conséquent responsable de la génération de sa clé privée et doit générer et fournir le fichier CSR nécessaire à la création du certificat.

8.7 Procédure de délivrance de certificats directs

Dans ce cas de figure, seul l'opérateur signataire est partie prenante. L'opérateur a la responsabilité de sa paire de clés et du fichier CSR servant à la création du certificat.



8.7.1 Opérateur : Création de la paire de clés

La clé privée à créer doit répondre aux exigences énoncées en section §2.6.6.

Commande openssl de génération de clé privée ECDSA avec une courbe P-256

```
openssl ecparam -genkey -name prime256v1 -out ecdsa-p256.key
```

Commande openssl d'extraction de la clé publique à partir de la clé privée

```
openssl ec -in ecdsa-p256.key -pubout -out ecdsa-p256.pub
```

8.7.2 Opérateur : Création du fichier CSR

Le fichier CSR doit répondre aux exigences suivantes :

- La propriété *Common Name* doit correspondre au code APNF attribué à l'opérateur
- L'algorithme de hachage doit être SHA256.
- Une extension *TN Authorization List* (OID : 1.3.6.1.5.5.7.1.26) doit être valorisée avec le code APNF de l'opérateur signataire
- La propriété *Serial Number* ne doit pas être incluse

Suivant la spécification **ATIS-100080**, section 6.4.1, plusieurs champs doivent être obligatoirement renseignés, les autres champs étant optionnels et pouvant être renseignés par l'opérateur, à l'exception du *Serial Number* comme indiqué ci-dessus.

Propriété	Obligatoire	Valeur attendue
Nom du certificat (CN)	Oui	Code APNF de l'opérateur signataire
Pays (C)	Oui	Laissé libre à l'opérateur
Localité (L)	Non	Laissé libre à l'opérateur
Etat (S)	Non	Laissé libre à l'opérateur
Organisation (O)	Oui	Laissé libre à l'opérateur
Département (OU)	Non	Laissé libre à l'opérateur
Extension : tnAuthList (OID : 1.3.6.1.5.5.7.1.26)	Oui	Code APNF de l'opérateur signataire

Exemple de génération de fichier CSR

Les commandes suivantes fournissent un exemple de procédure permettant de préparer les données au bon format pour un opérateur dont le code APNF est SPC000. Il convient de remplacer par conséquent ce code, ainsi que toutes les informations en orange par les données de l'opérateur.

1) Création de la paire de clés ECDSA P-256

```
openssl ecparam -genkey -name prime256v1 -out provider.key
```

2) Préparation du fichier de configuration openssl utilisé pour la création du CSR

```
cat > MAN_openssl.conf << EOF
[ req ]
distinguished_name      = req_distinguished_name
req_extensions          = v3_req

[ req_distinguished_name ]
commonName                  = Common Name
commonName_default          = SPC000
countryName                 = Country
countryName_default         = FR
0.organizationName          = Organization Name
0.organizationName_default = Opérateur SPC

[ v3_req ]
basicConstraints        = critical,CA:FALSE
keyUsage                = critical,digitalSignature
1.3.6.1.5.5.7.1.26      = ASN1:SEQUENCE:tn_auth_list

[tn_auth_list]
field1=EXP:0,IA5:SPC000
EOF
```

3) Génération du CSR

```
openssl req -new -sha256 -config MAN_openssl.conf -key provider.key >
provider.csr
```

8.7.3 Opérateur : Demande de délivrance de certificat

L'opérateur effectue sa demande de certificat direct à la solution GCO de la plateforme soit via :

- l'IHM en accédant au formulaire de création de certificat direct
- ou la méthode d'API GCO dédiée `POST /certificate` (§13.3).

Doivent être renseignés au sein de la page IHM de création ou de l'API :

- Le nom du certificat – valeur indicative permettant de le distinguer des autres certificats délivrés à l'opérateur
- Une description optionnelle
- Le contenu du fichier CSR au format PEM
- Si ce certificat doit être un certificat de test
- La date de début du certificat. Sauf pour une demande de certificat de test, il est requis que cette date soit au minimum une semaine dans le futur par rapport à la date actuelle.
- La date de fin du certificat, seulement si la demande concerne un certificat de test. Une valeur par défaut de 1 mois est alors utilisée.
- Si l'opérateur souhaite un renouvellement automatique du certificat. Cette option n'est pas disponible pour les certificats de test.

- Dans le cas où le renouvellement automatique est demandé, le délai en nombre de jours avant de lancer la procédure de renouvellement.

8.7.4 GCO : Contrôle de la demande

La solution GCO procède à différents contrôles avant de procéder à la création du certificat. Si la validation échoue, le module retourne l'erreur correspondante :

- Au niveau du formulaire de l'IHM soumise par l'utilisateur
- Ou par l'utilisation du code erreur spécifique au contrôle en réponse à la requête d'API. Les codes erreur sont disponibles en section §13.3.2.

1) Contrôle de la validité et de la cohérence des paramètres de la requête

Les validations effectuées par la plateforme sont les suivantes :

- Le paramètre *name* a été renseigné
- Le paramètre *type* a été renseigné à la valeur « DIRECT »
- Le paramètre *csr* a été renseigné
- Le paramètre *csr* correspond à une demande CSR au format PEM
- Si le paramètre *test_certificate* a été renseigné, vérifie que la valeur est *true* ou *false*
- Si le paramètre *valid_from* a été renseigné, vérifie que
    - la valeur est au format ISO 8601
    - la date est dans le futur
    - si le certificat n'est pas un certificat de test, la date doit être au minimum une semaine dans le futur par rapport à la date actuelle
- Si le paramètre *valid_to* a été renseigné, vérifie que
    - le paramètre *test_certificate* est renseigné à *true*
    - la valeur est au format ISO 8601
    - la date est dans le futur
    - la date est postérieure à *valid_from*
- Si le paramètre *renewal_auto* a été renseigné, vérifie que
    - le paramètre *test_certificate* est valorisé à *false*
    - la valeur est *true* ou *false*
- Si le paramètre *renewal_after* a été renseigné, vérifie que
    - Le paramètre *renewal_auto* est valorisé à *true*
    - La valeur est *true* ou *false*
    - La valeur est inférieure à la période de validité du certificat, définie en fonction des valeurs de *valid_from* et *valid_to*

La solution GCO valide aussi que le nombre de certificats délivrés à l'opérateur n'atteint pas la limite autorisée, à savoir 100 certificats (§2.6.5).

2) Contrôle de la demande CSR

La solution GCO valide la demande de certificat CSR en examinant les paramètres suivants :

- Le paramètre *Common Name* correspond au code APNF attribué à l'opérateur

- L'extension *TN Authorization List* doit être valorisée avec le code APNF de l'opérateur signataire selon le format prévu par la RFC 8226
- L'algorithme de hachage doit être « SHA256 »
- La demande CSR est conforme au format défini dans la section §8.7.2

8.7.5 GCO : Création du certificat

Le GCO crée le certificat correspondant au CSR fourni par l'opérateur et le signe avec le certificat intermédiaire de son autorité de certification (§2.5.3).

- Le Common Name du certificat est dérivé de la valeur fournie dans le CSR, précédée de "SHAKEN"
- Un serial number unique lui est attribué, permettant de le distinguer de l'ensemble des certificats délivrés par l'autorité de certification.
- Une liste d'extensions est incluse au certificat, fournissant des informations complémentaires quant à l'autorité de certification ayant signé le certificat, la CRL associée, et en particulier l'extension **tnAuthList** avec le code APNF de l'opérateur signataire, indiquant que ce certificat est habilité à signer les appels provenant de cet opérateur.

Le tableau ci-dessous rassemble les informations incluses dans le certificat généré (§2.6.7) :

Propriété	Valeur
Version	3 (0x2)
Serial Number	Valeur unique par certificat
Subject | Common Name (CN)	SHAKEN <Code APNF de l'opérateur>
Issuer	Certificat intermédiaire autorité de certificat (BPCO CA1 ou BPCO CA2 (§2.5.3)) : `C=FR, O=Base Publique des Certificats Opérateurs, CN=BPCO CA1 – SHAKEN Intermediate`
Pays (C)	Tel que défini dans le fichier CSR
Localité (L)	Tel que défini dans le fichier CSR
Etat (S)	Tel que défini dans le fichier CSR
Organisation (O)	Tel que défini dans le fichier CSR
Département (OU)	Tel que défini dans le fichier CSR
Date de début de validité	Telle que défini dans la requête API ou l'IHM
Date de fin de validité	- Telle que défini dans le code de procédures MAN pour les certificats de production - Telle que défini dans la requête API ou l'IHM pour les certificats de test
Type de clé	ECDSA-256
Algorithme de hachage	SHA256
Extension : X509v3 Subject Key Identifier	Identifiant unique du certificat
Extension: X509v3 Authority Key Identifier	Subject Key Identifier du certificat intermédiaire ayant signé le certificat
Extension : X509 Key Usage	`Critical, Digital Signature`
Extension : X509 Basic Constraints	`Critical, CA: FALSE`

Extension : X509v3 CRL Distribution Points	`Full Name:`
	`URI:https://<domaine-bpco>/crl`
	`CRL Issuer :`
	`DirName: C=FR, O=Base Publique des Certificats Opérateurs, OU=Policy Authority, CN=BPCO PA1`

Extension : tnAuthList
(OID : 1.3.6.1.5.5.7.1.26)
	Code APNF de l'opérateur signataire au format DER :
	`1.3.6.1.5.5.7.1.26:`
	`0.....<code APNF opérateur>`

8.7.6 GCO : Confirmation de la création du certificat

Afin de répondre au plus vite à l'opérateur signataire, la plateforme confirme la création du certificat sans attendre les tâches de publication et de notification pouvant être effectuées en parallèle.

- Si la demande a été formulée via l'IHM, la solution GCO retourne les données du certificat et permet à l'utilisateur de le télécharger
- Si la demande a été faite via une requête API, la solution GCO répond avec un code retour HTTP 200 et un objet JSON contenant l'ensemble des données liées au certificat et à sa création, ainsi que son contenu effectif au format PEM. Il convient de se référer aux « Guides de référence des APIs de la plateforme MAN » pour les détails de cette réponse.

8.7.7 GCO : Publication du certificat dans la BPCO

Une fois le certificat généré, la solution GCO le publie au sein de la BPCO via l'URL spécifiée en section §2.7.1.1. La date de modification de la BPCO (§2.7.1.5) est modifiée avec la date à laquelle le certificat est publié.

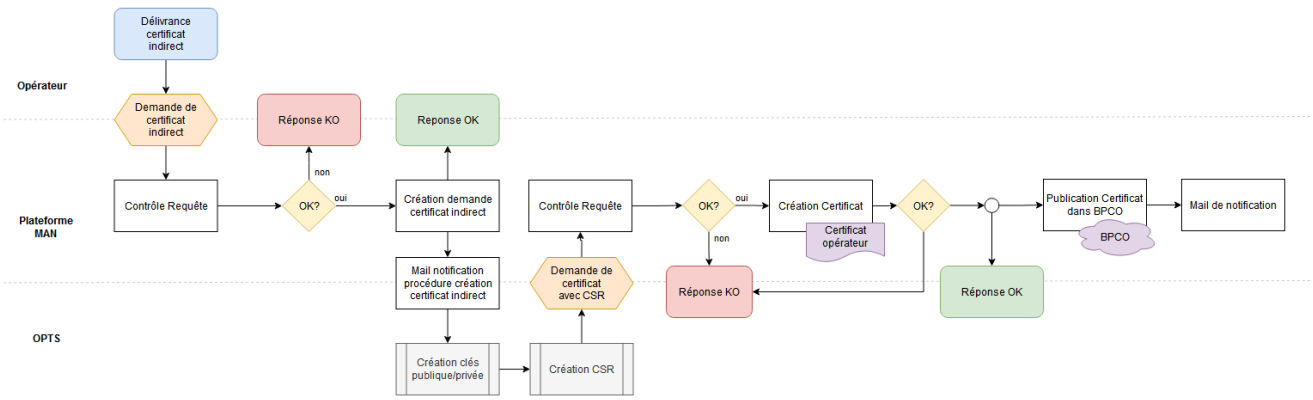
8.7.8 GCO : Mail de notification

La solution GCO envoie enfin un mail à la liste de notification « certificats » de l'opérateur afin de confirmer la délivrance du certificat. Ce mail inclut :

- Le nom du certificat et la description tels que fournis par l'opérateur dans sa requête
- Le serial number du certificat
- Les dates de début et de fin de validité du certificat
- La liste des propriétés du certificat
- L'URL publique d'accès au certificat dans la BPCO

8.8 Procédure de délivrance de certificats indirects

Pour les certificats indirects, l'opérateur signataire initie la procédure de création du certificat, mais c'est à l'OPTS de finaliser la création par la fourniture du fichier CSR. L'OPTS doit effectuer la finalisation avant que la date de début de validité du certificat ne commence.



8.8.1 Opérateur signataire : Demande de délivrance de certificat indirect

L'opérateur signataire effectue sa demande de certificat indirect à la solution GCO de la plateforme soit via :

- l'IHM en accédant au formulaire de création de certificat indirect
- ou la méthode d'API GCO dédiée `POST /certificate` (§13.4).

Doivent être renseignés au sein de la page IHM de création ou de l'API :

- Le nom du certificat – valeur indicative permettant de le distinguer des autres certificats délivrés à l'opérateur
- Une description optionnelle
- L'OPTS mandaté : l'opérateur doit sélectionner parmi la liste des opérateurs s'étant enregistrés comme OPTS dans leur fiche de renseignement et ayant confirmé que cet opérateur puisse le sélectionner (§6.3)
- Si ce certificat doit être un certificat de test
- La date de début de validité du certificat. Sauf pour une demande de certificat de test, il est requis que cette date soit au minimum une semaine dans le futur par rapport à la date actuelle.
- La date de fin de validité du certificat
- Si l'opérateur souhaite un renouvellement automatique du certificat. Cette option n'est pas disponible pour les certificats de test.
- Dans le cas où le renouvellement automatique est demandé, le délai en nombre de jours avant de lancer la procédure de renouvellement

La liste des OPTS possibles est disponible au niveau de l'IHM. Pour un appel API, il conviendra par conséquent de vérifier en amont via l'IHM quel opérateur à mandater comme OPTS.

8.8.2 GCO : validation de la demande de certificat indirect

La solution GCO procède à différents contrôles avant de procéder à la création du certificat. Si la validation échoue, elle retourne l'erreur correspondante :

- Au niveau du formulaire de l'IHM soumise par l'utilisateur
- Ou par l'utilisation du code erreur spécifique au contrôle en réponse à la requête d'API. Les codes erreur sont disponibles en section §13.4.2.

Les validations effectuées par la solution GCO sont les suivantes :

- Le paramètre *name* a été renseigné
- Le paramètre *type* a été renseigné à la valeur « INDIRECT »
- Dans le cas de certificat indirect, l'OPTS défini par le paramètre *opts* a bien confirmé via sa fiche de renseignement qu'il peut être OPTS pour l'opérateur signataire
- Si le paramètre *test_certificate* a été renseigné, vérifie que la valeur est *true* ou *false*
- Si le paramètre *valid_from* a été renseigné, vérifie que
    - la valeur est au format ISO 8601
    - la date est dans le futur

    - Si le certificat n'est pas un certificat de test, la date doit être au minimum une semaine dans le futur par rapport à la date actuelle
- Si le paramètre *valid_to* a été renseigné, vérifie que
    - la valeur est au format ISO 8601
    - la date est dans le futur
    - la date est postérieure à *valid_from*
- Si le paramètre *renewal_auto* a été renseigné, vérifie que
    - Le paramètre *test_certificate* est valorisé à *false*
    - la valeur est *true* ou *false*
- Si le paramètre *renewal_after* a été renseigné, vérifie que
    - Le paramètre *renewal_auto* est valorisé à *true*
    - La valeur est *true* ou *false*
    - La valeur est inférieure à la période de validité du certificat, définie en fonction des valeurs de *valid_from* et *valid_to*

La solution GCO valide aussi que le nombre de certificats délivrés à l'opérateur signataire n'atteint pas la limite autorisée, à savoir 100 certificats (§2.6.5).

8.8.3 GCO : initialisation de la demande de certificat indirect

Dans le cas où les vérifications ont été effectuées avec succès, la solution GCO crée un certificat indirect dans un statut *En Cours de Création* et confirme à l'opérateur signataire de la bonne prise en compte de la demande.

Elle envoie alors une notification à l'OPTS en utilisant sa liste de notification « certificats » pour indiquer l'initialisation d'une procédure de création de certificat indirect initié par un opérateur signataire.

Le mail contient :

- Le nom de l'opérateur signataire
- L'identifiant du certificat en cours de création
- Les dates de début et de fin du certificat définies par l'opérateur signataire

Il est alors à la responsabilité de l'OPTS de continuer la procédure de création du certificat. L'opérateur signataire peut à tout moment supprimer sa demande de certificat indirect tant que celui-ci n'a pas été finalisé.

L'OPTS doit quant à lui finaliser le certificat indirect avant la date de début de validité du certificat défini par l'opérateur signataire. Si la date de début de validité est passée, la finalisation sera refusée et il sera demandé à l'opérateur signataire de modifier la date de début de validité avant de redemander à l'OPTS de finaliser le certificat.

Si des demandes de certificats indirects ne sont pas finalisées après un délai de 3 mois, celles-ci seront automatiquement supprimées par la plateforme. Afin d'éviter ce cas, des rappels par email sont régulièrement envoyées à l'OPTS et à l'opérateur signataire afin de s'assurer que l'OPTS effectue les opérations demandées.

8.8.4 OPTS : Création de la clé privée

L'OPTS devant signer les appels pour l'opérateur signataire, il a la charge de créer la clé privée suivant les exigences énoncées en section §2.6.6.

Commande openssl de génération de clé privée ECDSA avec une courbe P-256

```
openssl ecparam -genkey -name prime256v1 -out ecdsa-p256.key
```

Commande openssl d'extraction de la clé publique à partir de la clé privée

```
openssl ec -in ecdsa-p256.key -pubout -out ecdsa-p256.pub
```

8.8.5 OPTS : Création du fichier CSR

Le fichier CSR doit répondre aux exigences suivantes :

- La propriété *Common Name* doit correspondre exactement au code APNF de l'opérateur signataire
- L'algorithme de hachage doit être SHA256.
- Une extension TN Authorization List doit être valorisée avec le code APNF de l'opérateur signataire
- La propriété *Serial Number* ne doit pas être incluse

Suivant la spécification **ATIS-100080**, section 6.4.1, plusieurs champs doivent être obligatoirement renseignés, les autres champs étant optionnels et pouvant être renseignés par l'opérateur.

Propriété	Obligatoire	Valeur attendue
Nom du certificat (CN)	Oui	Code APNF de l'opérateur signataire
Pays (C)	Oui	Laissé libre à l'opérateur
Localité (L)	Non	Laissé libre à l'opérateur
Etat (S)	Non	Laissé libre à l'opérateur
Organisation (O)	Oui	Laissé libre à l'opérateur
Département (OU)	Non	Laissé libre à l'opérateur
Extension : tnAuthList (OID : 1.3.6.1.5.5.7.1.26)	Oui	Code APNF de l'opérateur signataire

La section 8.7.2 fournit un exemple de procédure de création de fichier CSR.

8.8.6 OPTS : Demande de délivrance de certificat

L'OPTS finalise la demande de certificat indirect auprès de la solution GCO de la plateforme soit via :

- l'IHM en accédant au formulaire de création de certificat indirect
- ou la méthode d'API GCO dédiée `POST /certificate/:id` (§13.5), où `:id` est l'identifiant du certificat créé par la solution GCO lors de l'initialisation du processus.

Doit être renseigné dans le corps de la requête la demande CSR au format PEM. Une description propre à l'OPTS peut être spécifiée, accessible seulement de l'OPTS

Paramètre	Requis	Format	Valeur attendue
description	Optionnel	Chaîne de caractères	Description optionnelle spécifique à l'OPTS
csr	Obligatoire	Chaîne de caractères	Contenu de la demande CSR au format PEM

8.8.7 GCO : Contrôle de la demande

La solution GCO procède à différents contrôles avant de procéder à la finalisation du certificat. Si la validation échoue, elle retourne l'erreur correspondante à l'OPTS :

- Au niveau du formulaire de l'IHM soumise par l'OPTS
- Ou par l'utilisation du code erreur spécifique au contrôle en réponse à la requête d'API. Les codes erreur sont disponibles en section §13.5.2.

1) Contrôle de la validité et de la cohérence des paramètres de la requête

Les validations effectuées par la solution GCO sont les suivantes :

- Le paramètre *csr* a été renseigné
- Le paramètre *csr* correspond à une demande CSR au format PEM

2) Contrôle de la demande CSR

La solution valide la demande de certificat CSR en examinant les paramètres suivants :

- Le paramètre *Common Name* correspond au code APNF attribué à l'opérateur signataire
- L'extension *TN Authorization List* doit être valorisée avec le code APNF de l'opérateur signataire
- L'algorithme de hachage doit être « SHA256 »
- La demande CSR est conforme au format défini dans la section §8.8.5

8.8.8 GCO : Création du certificat

La solution GCO crée le certificat correspondant au CSR fourni par l'OPTS et le signe avec le certificat intermédiaire de son autorité de certification (§2.5.3). Un serial number unique lui est attribué, permettant de le distinguer de l'ensemble des certificats délivrés par l'autorité de certification.

- Le Common Name du certificat est dérivé de la valeur fournie dans le CSR, précédée de « SHAKEN ».
- Un serial number unique lui est attribué, permettant de le distinguer de l'ensemble des certificats délivrés par l'autorité de certification.

- Une liste d'extensions est incluse au certificat, fournissant des informations complémentaires quant à l'autorité de certification ayant signé le certificat, la CRL associée, et en particulier l'extension **tnAuthList** avec le code APNF de l'opérateur signataire, indiquant que ce certificat est habilité à signer les appels provenant de cet opérateur.

Le tableau ci-dessous rassemble les informations incluses dans le certificat généré :

Propriété	Valeur
Version	3 (0x2)
Serial Number	Identifiant unique du certificat
Subject | Common Name (CN)	SHAKEN <Code APNF de l'opérateur signataire>
Issuer	Certificat intermédiaire autorité de certificat (BPCO CA1 ou BPCO CA2 (§2.5.3)) : `C=FR, O=Base Publique des Certificats Opérateurs, CN=BPCO CA1 – SHAKEN Intermediate`
Pays (C)	Tel que défini dans le fichier CSR
Localité (L)	Tel que défini dans le fichier CSR
Etat (S)	Tel que défini dans le fichier CSR
Organisation (O)	Tel que défini dans le fichier CSR
Département (OU)	Tel que défini dans le fichier CSR
Date de début de validité	Telle que défini dans la requête API ou l'IHM
Date de fin de validité	Telle que défini dans la requête API ou l'IHM
Type de clé	ECDSA-256
Algorithme de hachage	SHA256
Extension : X509v3 Subject Key Identifier	Identifiant unique du certificat
Extension: X509v3 Authority Key Identifier	Subject Key Identifier du certificat intermédiaire ayant signé le certificat
Extension : X509 Key Usage	`Critical, Digital Signature`
Extension : X509 Basic Constraints	`Critical, CA: FALSE`

Extension : X509v3 CRL Distribution Points	`Full Name:`
	`URI:https://<domaine-bpco>/crl`
	`CRL Issu er :`
	`DirName: C=FR, O=Base Publique des Certificats Opérateurs, OU=Policy Authority, CN=BPCO PA1`

Extension : tnAuthList **(OID : 1.3.6.1.5.5.7.1.26)**	Code APNF de l'opérateur signataire au format DER : `1.3.6.1.5.5.7.1 .26: 0.....<code APNF opérateur>`

8.8.9 GCO : Confirmation de la création du certificat

Afin de répondre au plus vite à l'OPTS, la solution GCO confirme la création du certificat sans attendre les tâches de publication et de notification pouvant être effectuées en parallèle.

- Si la demande a été formulée via l'IHM, la solution retourne les données du certificat et permet à l'utilisateur de le télécharger
- Si la demande a été faite via une requête API, la solution retourne une réponse avec un code retour HTTP 201 et un objet JSON contenant l'ensemble des données liées au certificat et à sa création, ainsi que son contenu effectif au format PEM. Il convient de se référer aux « Guides de référence des APIs de la plateforme MAN » pour les détails de cette réponse.

L'opérateur signataire peut accéder via l'IHM ou l'API le certificat finalisé, ses données et le télécharger.

8.8.10 GCO : Publication du certificat dans la BPCO

Une fois le certificat généré, la solution le publie au sein de la BPCO via l'URL définie en section §2.7.1.1. La date de modification de la BPCO (§2.7.1.5) est modifiée avec la date de publication.

8.8.11 GCO : Mail de notification

La solution GCO envoie un mail à la liste de notification « certificats » de l'OPTS et de l'opérateur signataire afin de leur confirmer la délivrance du certificat indirect. Ce mail inclut :

- Le nom du certificat et la description telle que fournie par chaque opérateur dans leur requête
- Le serial number du certificat
- Les dates de début et de fin du certificat
- La liste des propriétés du certificat
- L'URL publique d'accès au certificat dans la BPCO

8.8.12 Suppression de demande de certificats indirects

L'opérateur signataire a la possibilité via l'IHM de la plateforme ou d'une méthode d'API GCO dédiée (§13.13) de supprimer la demande de certificat indirect tant que celui-ci n'a pas été finalisé par l'OPTS.

8.9 Récupération de l'URL du certificat délivré

Une fois le certificat délivré, l'opérateur signataire peut récupérer à tout moment l'URL publique associée soit en visualisant les données du certificat via l'IHM de la plateforme MAN, soit en effectuant une requête à la méthode d'API de la plateforme précisée en section §13.6.

9 Mise en place des copies locales opérateurs

9.1 Opérateurs concernés

- Opérateur de terminaison
- OPTV

9.2 Contexte d'application

La mise en place de copies locales est une procédure obligatoire pour le STI-VS de l'opérateur avant de pouvoir vérifier les appels SIP reçus (§5).

Le mécanisme de confiance introduit en effet des délais supplémentaires dans la procédure de vérification des appels dû à la récupération des certificats et de la CRL. Afin d'optimiser ces délais, mais aussi de minimiser l'impact d'une indisponibilité de la BPCO, il est demandé aux opérateurs devant vérifier les signatures d'appel de créer et de synchroniser régulièrement une copie locale des certificats (opérateurs et autorité de certification) et des CRLs associées.

De plus, un système de rate limiting étant appliqué sur les requêtes effectuées sur la BPCO - afin de protéger cette dernière d'attaques de type Denial of Service (DoS) ou Denial of Service Distribué (DDoS), il ne sera pas possible pour un opérateur recevant un grand nombre d'appels d'effectuer une vérification en récupérant à chaque fois le certificat à partir de la BPCO. Une copie locale devra ainsi obligatoirement être mise en place pour limiter le nombre d'appels effectués auprès de la BPCO.

Il est à noter que seuls les certificats standards peuvent être automatiquement synchronisés. Les certificats de test ne sont pas inclus dans les copies locales et doivent être récupérés directement de la BPCO.

9.3 Prérequis

- Un opérateur doit être enregistré auprès de la plateforme MAN afin d'accéder aux fonctions d'API de téléchargement de copie et mises à jour différentielles des données de la BPCO (§6).
- Un *API credential* a été généré par l'opérateur au niveau de la plateforme pour pouvoir s'authentifier auprès de ses APIs en utilisant le protocole OAuth (§13.1.2).

9.4 Composants impliqués

- Le **STI-VS** de l'opérateur, devant utiliser les copies locales lors de la procédure de vérification des appels
- Le **KMS** de l'opérateur, si celui-ci prend en charge la gestion des copies locales des certificats et CRL

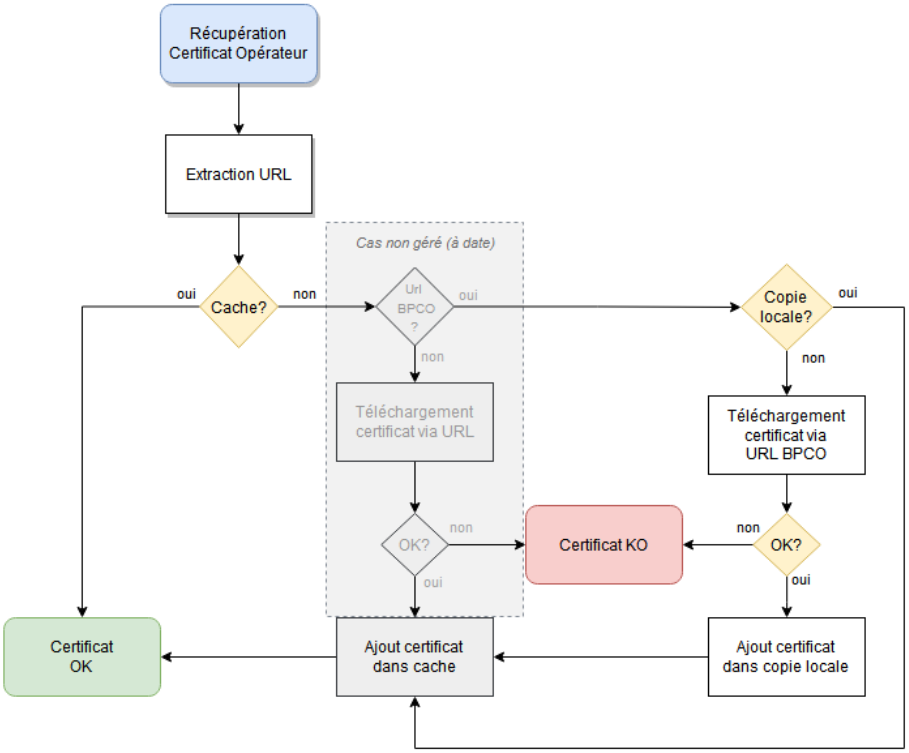
9.5 Copie locale vs cache

Une mise en place de caches peut être envisagée en complément de copies locales. L'utilisation d'une copie locale reste obligatoire afin de pallier aux problèmes d'indisponibilité qui pourraient affecter la BPCO et donc la récupération des certificats nécessaires à la vérification des signatures.

En effet, alors que les systèmes de cache purgent automatiquement leurs données en fonction d'un délai d'expiration, la copie locale garde l'ensemble des certificats valides. Ainsi, si jamais la BPCO n'est pas disponible pendant une durée significative, le STI-VS aura toujours à disposition dans sa copie locale le certificat nécessaire, alors que le cache a déjà pu le supprimer si celui-ci n'a pas été utilisé pendant un certain temps ou si celui-ci n'a pas encore été employé lors de la signature d'un appel.

Néanmoins, cette solution n'exclut pas la mise en place d'un cache en première ligne avec la copie locale en seconde ligne. Dans le cas où le cache ne dispose pas du certificat demandé :

- si le certificat est un certificat opérateur français – à savoir si l'URL du certificat est une URL de la BPCO telle que définie en section §2.7.1.1, le STI-VS va d'abord récupérer le certificat de la copie locale avant d'essayer de le récupérer de la BPCO (§5.5.3.1) ;
- si le certificat provient d'un opérateur international, le STI-VS récupère directement le certificat du chemin fourni dans l'entête Identity. Ce cas n'est pas envisagé à date mais pourra s'appliquer dans le futur.

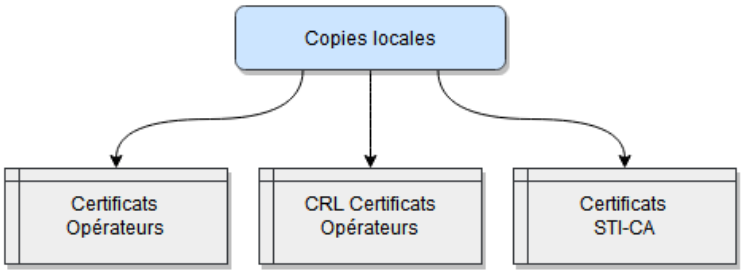


9.6 Création des copies locales

L'opérateur doit obligatoirement mettre en place 2 copies locales :

- Des certificats opérateurs présents dans la BPCO
- De la liste de révocation des certificats opérateurs disponible dans la BPCO

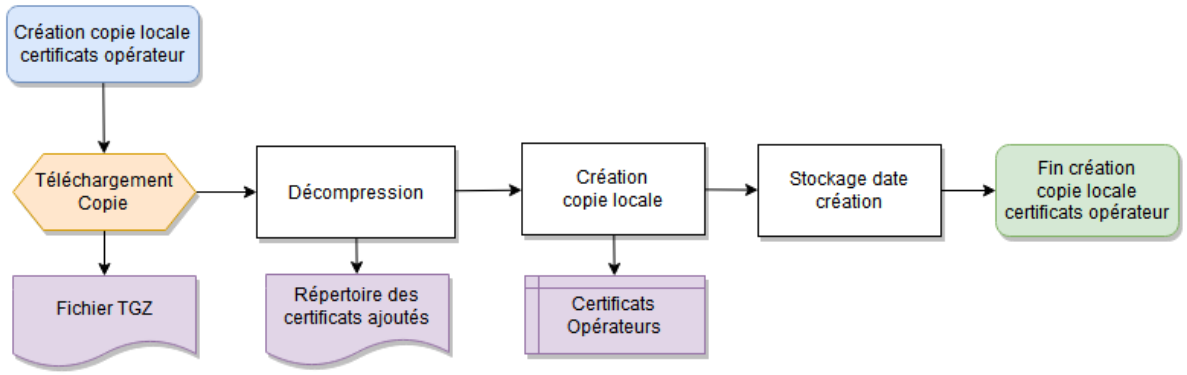
A ces 2 copies peuvent s'ajouter une supplémentaire pour la vérification des certificats racine et intermédiaires de l'autorité de certification.



L'implémentation de ces copies locales et leur méthode de stockage sont laissées libres à l'opérateur. Ce document se focalise sur les moyens fournis par la plateforme MAN pour permettre aux opérateurs de créer et de maintenir à jour leurs copies locales.

Le module GCO de la plateforme MAN met à disposition de tous les opérateurs enregistrés des APIs pour effectuer ces tâches, §13.2. Il n'est pas possible d'accéder à ces fonctionnalités depuis l'IHM de la plateforme.

9.6.1 Création de la copie locale des certificats opérateur



1. Le STI-VS/KMS récupère la base complète des certificats via la méthode d'API GCO dédiée `GET /bpco/certs` (§13.7)
2. Un fichier TAR GZippé est retourné, contenant, une fois décompressé, la liste des certificats opérateurs standards délivrés par la plateforme MAN. L'arborescence utilisée reprend le format de l'URL d'accès du certificat (§2.7.1.1), où :
    - les certificats d'un opérateur sont rassemblés dans un répertoire dont le nom est le code APNF de l'opérateur

- les certificats sont stockés dans des fichiers dont le nom est le serial number du certificat et l'extension de fichier *cer*.

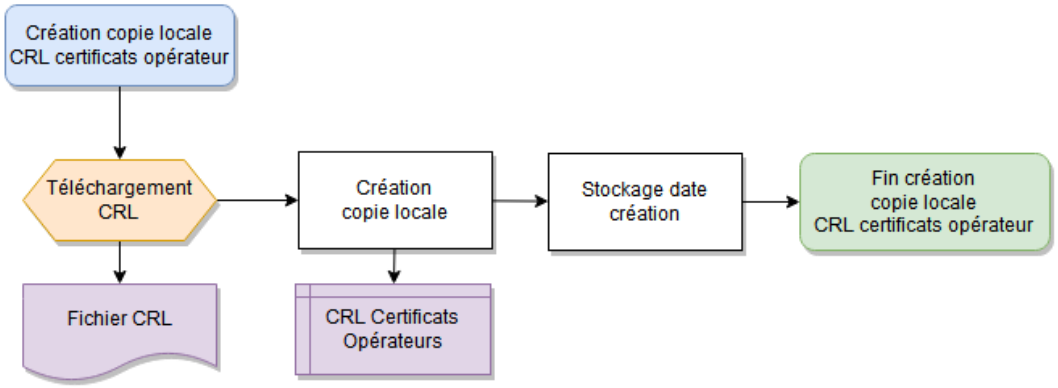
```
.
|- <code apnf opérateur A>/
|    |- <serial number certificat #1>.cer
|    |- ...
|    `- <serial number certificat #N>.cer
`- < code apnf opérateur X>/
     ...
```

Le serial number est au format hexadécimal sur exactement 16 caractères ou plus.

Il est à noter que les certificats de test ne sont pas inclus dans l'archive.

3. Le STI-VS/KMS crée sa copie locale à partir de cette arborescence de fichiers suivant l'implémentation définie par l'opérateur.
4. Le STI-VS/KMS récupère la date de mise à jour de la BPCO via l'entête *Last-Modified* de la réponse à la requête de l'API envoyée et la sauvegarde.

9.6.2 Création de la copie locale de la CRL certificats opérateur

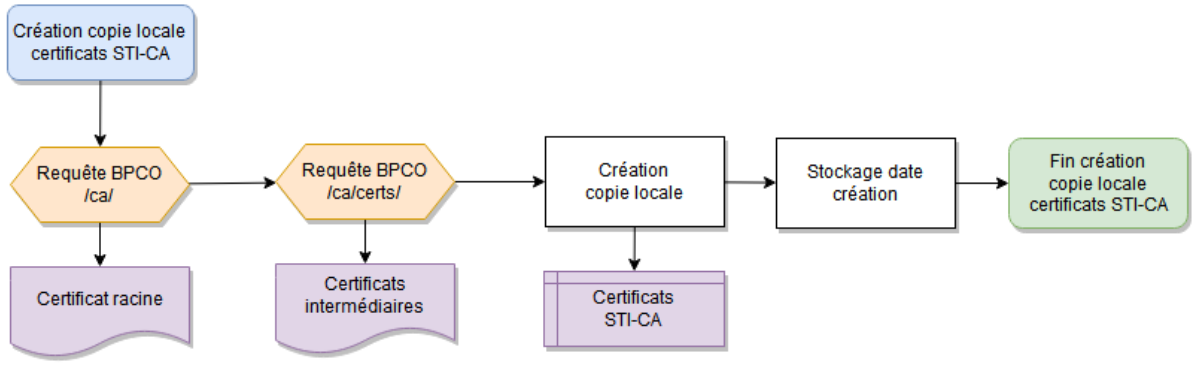


1. Le STI-VS/KMS récupère la CRL via la méthode d'API GCO dédiée `GET /bpco/certs/crl` (§13.9)
2. Le STI-VS/KMS stocke ce fichier en local pour créer sa copie locale
3. Le STI-VS/KMS récupère à partir de l'entête *Last-Modified* de la réponse de l'API la date de mise à jour de la CRL et la sauvegarde en local

9.6.3 Création de la copie locale des certificats STI-CA

1. Le STI-VS/KMS récupère la liste des certificats STI-CA de la BPCO à partir de 2 URLs publiques :
   - `GET /ca` pour le certificat racine (§14.5)
   - `GET /ca/certs` pour les certificats intermédiaires (§14.6)
2. Le STI-VS/KMS stocke la liste des certificats récupérés en local

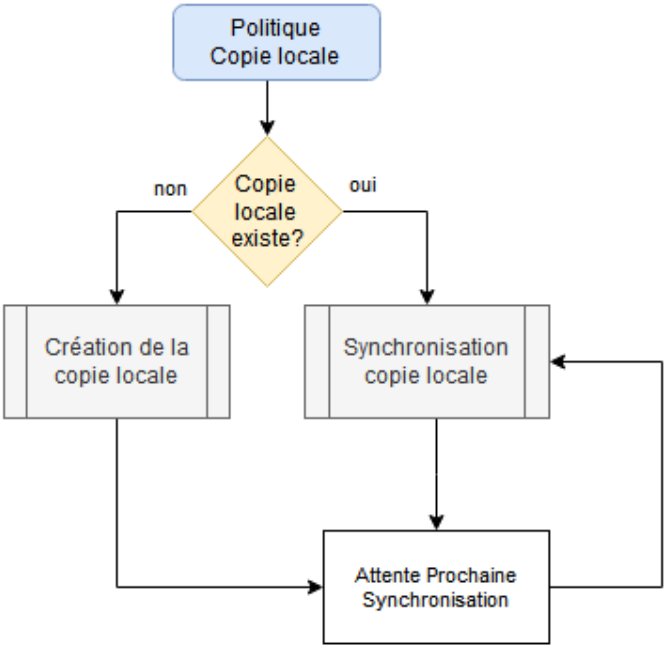
Ces APIs retournent les certificats valides, non expirés et non révoqués. Il n'est donc pas nécessaire de récupérer la CRL des certificats du STI-CA – disponible via la requête `GET /ca/crl` - pour vérifier si les certificats sont révoqués.



9.7 Synchronisation des copies locales

Une fois les copies locales créées, une politique de synchronisation doit être mise en place par l'opérateur afin de garder les données le plus à jour possible avec la BPCO et ainsi éviter un traitement erroné des appels pouvant intervenir si :

- Le certificat opérateur utilisé pour signer l'appel n'existe pas dans la copie locale des certificats opérateurs, obligeant le STI-VS à récupérer le certificat directement de la BPCO
- Le certificat opérateur a été révoqué mais la copie locale de la CRL des certificats opérateurs ne contient pas de référence à ce certificat
- Le certificat utilisé par l'autorité de certification pour signer le certificat opérateur n'est pas disponible dans la copie locale des certificats du STI-CA
- Le certificat utilisé par l'autorité de certification est révoqué mais la copie locale des certificats du STI-CA n'est pas à jour et contient toujours ce certificat



La fréquence de synchronisation préconisée dépend du type de copie et du type de synchronisation utilisée (dans le cas de la copie des certificats opérateur). Le tableau ci-dessous fournit des recommandations par type de copie locale, mais l'opérateur est libre d'appliquer une politique différente.

Type de copie locale	Synchronisation	Fréquence attendue
Certificats opérateur	Complète	24h
Certificats opérateur	Partielle	4h
CRL certificats opérateur	Complète	4h
Certificats STI-CA	Complète	1 semaine

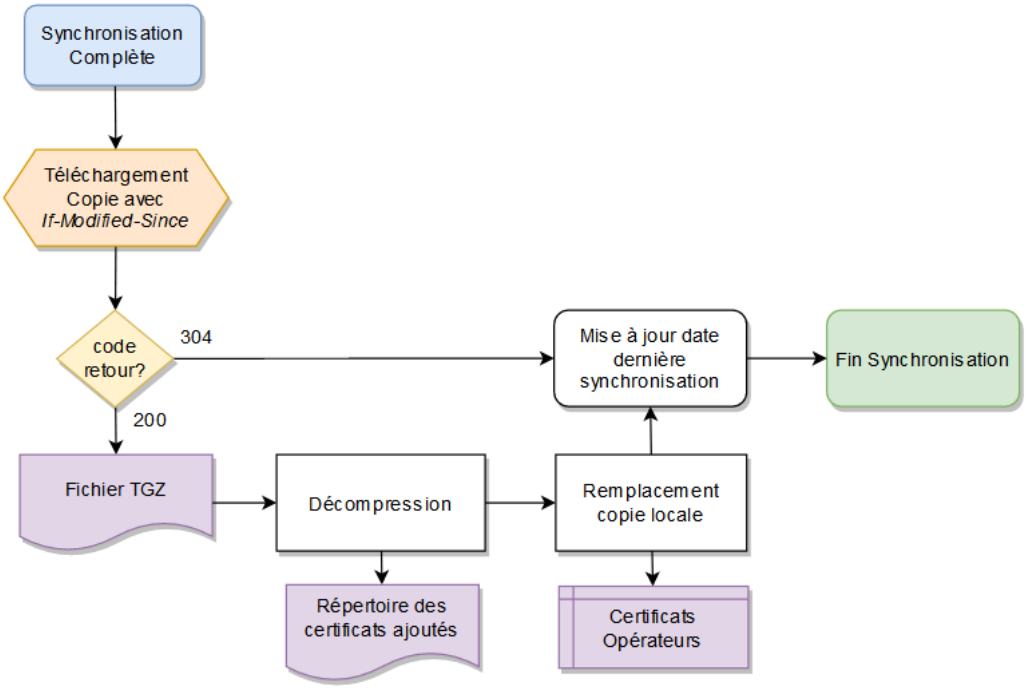
9.7.1 Synchronisation de la copie locale des certificats opérateurs

2 modes de synchronisation sont possibles :

- **Complète**, où l'opérateur télécharge la base complète des certificats
- **Partielle**, où l'opérateur ne récupère que les modifications effectuées depuis une date à définir par l'opérateur

La synchronisation partielle est recommandée, impliquant une utilisation moindre de la bande passante tout en permettant une fréquence de synchronisation plus élevée.

Synchronisation complète

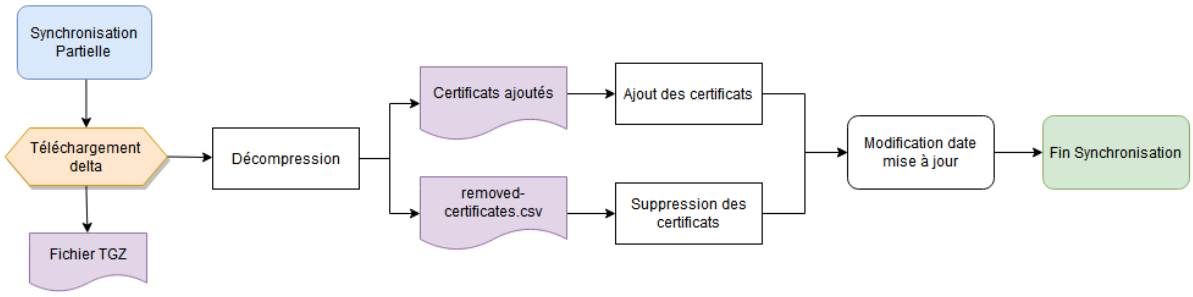


1. Le STI-VS/KMS récupère la base des certificats via l'API GCO dédiée `GET /bpco/certs` (§13.7) en passant la date d'initialisation ou de dernière synchronisation de la copie locale dans l'entête *If-Modified-Since* de la requête.
2. Si le composant GCO retourne un code erreur 304, aucune action est requise, la base des certificats n'ayant pas été modifiée depuis la date d'initialisation ou de dernière synchronisation de la copie locale.
3. Sinon, le STI-VS/KMS décompresse le fichier TAR GZippé renvoyé et remplace sa copie locale avec la liste des certificats contenus dans le fichier utilisant la même installation que celle utilisée pour la création initiale de la copie locale, §9.6.1.

   Tout comme pour la récupération initiale de la copie locale, l'archive retournée ne contient pas les certificats de test ayant pu être créés par les opérateurs.

4. Le STI-VS/KMS sauvegarde la date de l'entête *Last-Modified* comme date de synchronisation

Synchronisation partielle



1. Le STI-VS/KMS récupère un différentiel de la base des certificats opérateur en appelant la méthode d'API GCO dédiée `GET /bpco/certs?since=<DATE>` (§13.7) où le paramètre *since* correspond à la date d'initialisation ou de dernière synchronisation de la copie locale de l'opérateur. Cette date ne peut être antérieure à plus de 15 jours par rapport à la date actuelle.
2. Le STI-VS/KMS décompresse le fichier TAR GZippé récupéré, afin d'obtenir :
   - une arborescence de certificats créés depuis la date spécifiée dans le paramètre *since*
   - un fichier `removed-certificates.csv` contenant la liste des certificats ayant expirés et donc supprimés de la base depuis la date spécifiée dans le paramètre *since*.

```
.
|- removed-certificates.csv
|- <code apnf opérateur A>/
|    |- <serial number certificat #1>.cer
|    `- <serial number certificat #2>.cer
`- <code apnf opérateur B>/
     ...
```

   Tout comme pour la récupération initiale de la copie locale, les certificats de test ayant pu être créés par les opérateurs ne sont pas pris en compte.

3. Le STI-VS/KMS ajoute les certificats présents dans l'arborescence dans la copie locale des certificats opérateur. Comme indiqué en section §9.6, cette procédure de mise à jour dépend du type d'implémentation choisi par l'opérateur pour la mise en place de sa copie locale.
4. Le STI-VS/KMS supprime les certificats contenus dans le fichier `removed-certificates.csv`.
   Ce fichier est un fichier CSV utilisant comme caractère de fin de ligne et séparateur de colonne respectivement les caractères LineFeed (`\n`) et point-virgule (`;`). Chaque ligne du fichier correspond à un certificat supprimé, et est composée de 2 colonnes :
   - code APNF de l'opérateur associé
   - Serial number du certificat, au format hexadécimal sur 16 caractères ou plus.

```
<code apnf opérateur A>;<serial number certificat #1>
<code apnf opérateur B>;<serial number certificat #2>
<code apnf opérateur B>;<serial number certificat #3>
```

5. Le STI-VS/KMS sauvegarde la date de l'entête *Last-Modified* de la réponse de l'API appelée en point 1.

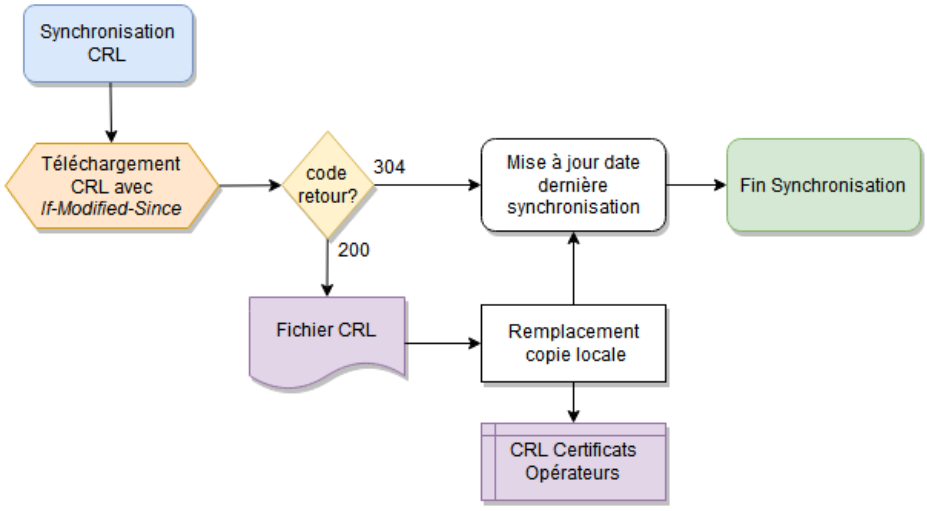
9.7.2 Synchronisation de la copie locale de la CRL certificats opérateur

La CRL consistant en un seul fichier, la synchronisation consiste à remplacer complètement la copie locale de la CRL.

1. Le STI-VS/KMS récupère la CRL via l'API GCO dédiée `GET /bpco/crl` (§13.9) en passant la date d'initialisation ou de dernière synchronisation de la CRL dans l'entête *If-Modified-Since* de la requête.
2. Si le composant GCO retourne un code erreur 304, aucune action est requise, la CRL n'ayant pas été modifiée depuis la date d'initialisation ou de dernière synchronisation de la copie locale.

3. Sinon, le STI-VS remplace la copie locale de la CRL avec le fichier récupéré dans le corps de la réponse

4. Le STI-VS sauvegarde la date de l'entête *Last-Modified* comme date de synchronisation.



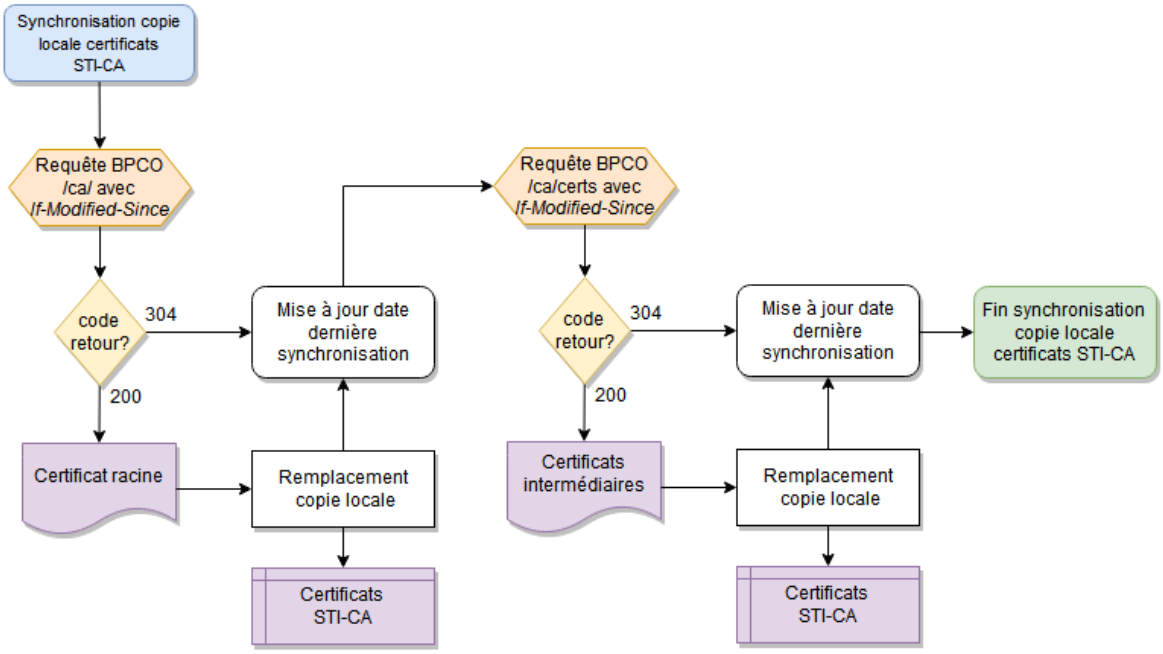
9.7.3 Synchronisation de la copie locale des certificats STI-CA

Le nombre de certificats racine et intermédiaires du STI-CA étant très limité, la procédure de synchronisation de la copie locale des certificats du STI-CA se limite à remplacer complètement la copie locale avec la liste récupérée suivant le même principe que lors de la création de la copie locale.

1. Le STI-VS récupère la liste des certificats CRL STI-CA de la BPCO à partir de 2 URLs publiques :
   - `GET /ca` pour le certificat racine (§14.5)
   - `GET /ca/certs` pour les certificats intermédiaires (§14.6)

   Pour ces deux requêtes, le STI-VS passe la date d'initialisation ou de dernière synchronisation de ces données dans l'entête *If-Modified-Since*.

2. Si la requête retourne un code retour 304, aucune synchronisation pour la donnée associée, celle-ci n'ayant pas été modifiée
3. Sinon, le STI-VS stocke la liste des certificats récupérés en local


Synchronisation copie locale certificats STI-CA
Requête BPCO /ca/ avec If-Modified-Since
code retour?
304
200
Mise à jour date dernière synchronisation
Certificat racine
Remplacement copie locale
Certificats STI-CA
Requête BPCO /ca/certs avec If-Modified-Since
code retour?
304
200
Mise à jour date dernière synchronisation
Fin synchronisation copie locale certificats STI-CA
Certificats intermédiaires
Remplacement copie locale
Certificats STI-CA

10 Renouvellement de certificats

10.1 Opérateurs concernés

- Opérateur signataire

10.2 Contexte d'application

Lors de la création de son certificat – direct ou indirect, un opérateur signataire a la possibilité de spécifier à la plateforme s'il souhaite que le certificat soit renouvelé automatiquement (§8.7.3 et §8.8.1). Il est de plus possible pour l'opérateur de lancer manuellement et à tout moment la procédure de renouvellement d'un certificat.

Seul l'opérateur signataire peut être à l'initiative du renouvellement d'un certificat indirect. De plus, afin qu'un certificat indirect puisse être renouvelé, il convient que l'OPTS associé au certificat ait toujours l'opérateur signataire dans sa liste des contrats OPTS.

Enfin, seuls les certificats standards peuvent bénéficier de la fonctionnalité de renouvellement automatique. Les certificats de tests ne peuvent être seulement renouvelés manuellement.

Il est à noter que le renouvellement d'un certificat implique toujours la création d'un nouveau certificat, et non la modification du certificat existant, cette opération étant techniquement impossible. La différence alors entre la procédure de création d'un nouveau certificat et la procédure de renouvellement est que cette dernière crée un nouveau certificat en utilisant la même paire de clés de chiffrement que le certificat initial. L'opérateur n'a ainsi pas besoin de fournir un nouveau fichier CSR. Par contre, comme pour une création, le serial number du certificat créé lors du renouvellement est différent de celui associé au certificat initial, le serial number étant unique par certificat.

La propriété *renewed_by* du certificat est utilisée afin de connaître le certificat créé lors d'une procédure de renouvellement. Lorsque cette dernière est effectuée – de façon manuelle ou automatique – la propriété est ainsi renseignée avec l'identifiant du certificat créé.

10.3 Prérequis

Le renouvellement d'un certificat, qu'il soit manuel ou automatique, n'est possible que si les conditions suivantes sont réunies :

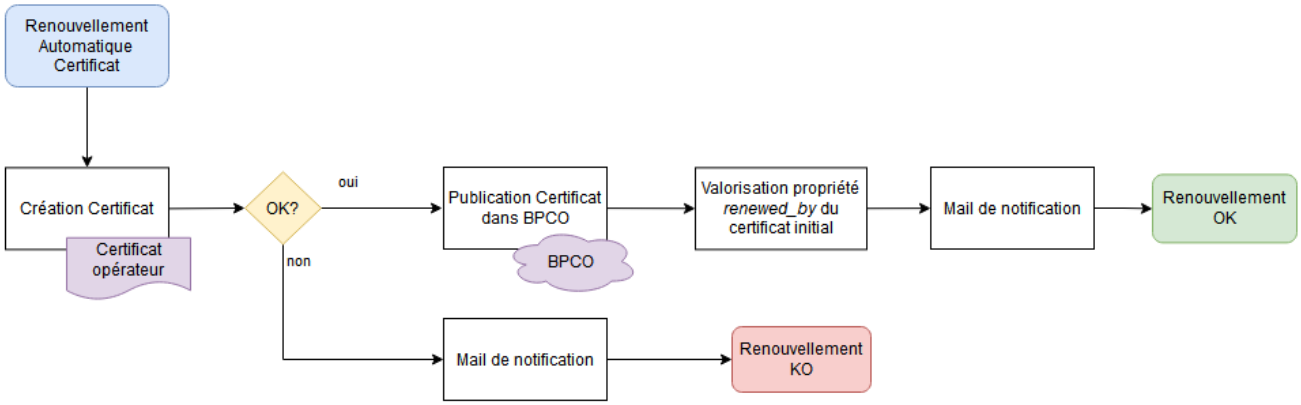
- Un certificat direct ou indirect a été délivré (§8). Dans le cas du certificat indirect, il doit avoir été finalisé par l'OPTS (§8.8).
- Pour un certificat indirect, l'opérateur signataire est toujours renseigné dans la liste des contrats de l'OPTS
- Le certificat n'a pas été révoqué.
- Le certificat n'a pas été invalidé.
- Le certificat n'a pas été archivé.

10.4 Procédure automatique

10.4.1 Prérequis

Le renouvellement automatique de certificats ne nécessite aucune intervention de l'opérateur. Il suffit qu'un certificat direct ou indirect ait été délivré ou configuré avec l'option de renouvellement automatique. Les certificats de test sont par conséquent exclus de cette procédure, l'option de renouvellement automatique leur étant indisponible.

La plateforme déclenche la procédure de renouvellement après le nombre de jours spécifié par l'option *renewal_after* par rapport à la date de génération du certificat. Pour un certificat indirect cette date correspond à sa date de finalisation par l'OPTS.



10.4.2 GCO : création d'un nouveau certificat

Le module génère un nouveau certificat en utilisant le fichier CSR fourni par l'opérateur lors de la demande de création du certificat initial. Les mêmes informations que le certificat précédent sont ainsi utilisées (nom, description…), à l'exception de la date de début de validité qui sera fixée à la date actuelle incrémentée d'une semaine pour être conforme aux règles décrites en section §2.6.4. La durée du nouveau certificat sera quant à elle identique à celle du certificat existant.

10.4.3 GCO : Valorisation de la propriété renewed_by du certificat initial

L'identifiant du nouveau certificat créé lors de cette procédure de renouvellement sert alors à renseigner la propriété *renewed_by* du certificat initial.

10.4.4 GCO : Publication du certificat dans la BPCO

Une fois le certificat généré, le module le publie au sein de la BPCO via l'URL définie en section §2.7.1.1. La date de modification de la BPCO (§2.7.1.5) est modifiée avec la date de publication de ce certificat.

10.4.5 GCO : Notification

Une fois le nouveau certificat créé, le module notifie les opérateurs associés au certificat via leur liste de notification « certificats » :

- L'opérateur signataire pour un certificat direct
- L'opérateur signataire et l'OPTS pour un certificat indirect

Le mail inclut les informations suivantes :

- Serial number du nouveau certificat
- Nom de l'opérateur signataire
- Nom de l'OPTS, si certificat indirect
- Dates de validité du nouveau certificat
- URL du certificat

Dans le cas où une erreur est intervenue lors de la procédure, ces mêmes opérateurs sont notifiés par mail de l'erreur survenue.

10.5 Procédure manuelle

10.5.1 Prérequis

Le renouvellement manuel de certificats peut être réalisée via l'IHM ou l'API de la plateforme. À la différence de la procédure automatique, il est possible lors de la procédure manuelle de modifier un certain nombre d'informations pour le certificat nouvellement créé. Ces propriétés sont détaillées dans le guide de référence de l'API GCO de la plateforme MAN.

Les prérequis dépendent du mode utilisé par l'opérateur pour cette tâche.

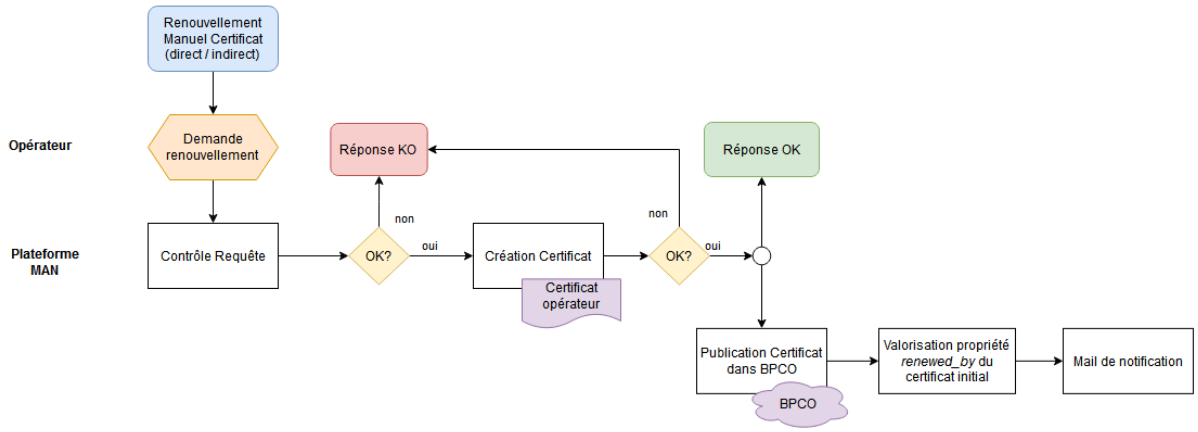
- Un certificat direct ou indirect a déjà été délivré à l'opérateur signataire (§8)
- Utilisation de l'IHM : l'utilisateur dispose d'un compte utilisateur de type « Administrateur » ou « Gestionnaire de certificats ».
- Utilisation de l'API : Un *API credential* a été généré par l'opérateur au niveau de la plateforme pour pouvoir s'authentifier auprès de ses APIs en utilisant le protocole OAuth (§13.1.2).

10.5.2 Déclenchement de la procédure de renouvellement

Le renouvellement est déclenché par l'opérateur à partir de l'IHM de la plateforme ou de la méthode d'API GCO dédiée `POST /certificate/:id/renew` (§13.11), où `:id` est l'identifiant du certificat créé par la plateforme MAN.

L'opérateur peut spécifier les informations suivantes afin d'être associées au nouveau certificat :

Propriété	Libellé API
Nom	name
Description	description
Date de début de validité	valid_from
Date de fin de validité	valid_to
Renouvellement automatique **(seulement si le certificat n'est pas un certificat de test)**	renewal_auto
Nombre de jours avant renouvellement **(par rapport à la date de génération du certificat)**	renewal_after



10.5.3 GCO : création d'un nouveau certificat

Le module génère un nouveau certificat en utilisant le fichier CSR fourni par l'opérateur lors de la demande de création du certificat initial.

Pour les propriétés non renseignées par l'opérateur, les mêmes informations que le certificat précédent sont utilisées. De plus, la date de début de validité est fixée à la date d'appel de la requête ajoutée d'une semaine pour les certificats de production et la date d'expiration est valorisée afin que la durée de validité soit identique à celle du certificat existant.

10.5.4 GCO : Valorisation de la propriété renewed_by du certificat initial

L'identifiant du nouveau certificat créé lors de cette procédure de renouvellement sert alors à renseigner la propriété *renewed_by* du certificat initial.

10.5.5 GCO : Confirmation de la création du certificat

Afin de répondre au plus vite à l'opérateur signataire, la plateforme confirme la création du nouveau certificat sans attendre les tâches de publication et de notification pouvant être effectuées en parallèle.

- Si la demande a été formulée via l'IHM, la solution GCO retourne les données du certificat et permet au client de le télécharger.
- Si la demande a été faite via une requête API, la solution GCO répond avec un code retour HTTP 201 et un objet JSON contenant l'ensemble des données liées au certificat et à sa création, ainsi que son contenu effectif au format PEM. Il convient de se référer aux « Guides de référence des APIs de la plateforme MAN » pour les détails de cette réponse.

10.5.6 GCO : Publication du certificat dans la BPCO

Une fois le certificat généré, le module le publie au sein de la BPCO via l'URL définie en section §2.7.1.1. La date de modification de la BPCO (§2.7.1.5) est modifiée avec la date de publication de ce certificat.

10.5.7 GCO : Notification

Une fois le nouveau certificat créé, le module notifie les opérateurs associés au certificat via leur liste de notification « certificats » :

- L'opérateur signataire pour un certificat direct
- L'opérateur signataire et l'OPTS pour un certificat indirect

Le mail inclut les informations suivantes :

- Serial number du nouveau certificat
- Nom de l'opérateur signataire
- Nom de l'OPTS, si certificat indirect
- Dates de validité du nouveau certificat
- URL du certificat

11 Révocation de certificats

11.1 Opérateurs concernés

- Opérateur signataire
- OPTS

11.2 Contexte d'application

Un opérateur signataire doit avoir la possibilité de faire la demande de révocation de tous les certificats directs ou indirects qui lui ont été délivrés par la plateforme MAN.

De même, un OPTS peut effectuer la demande de révocation pour un certificat indirect demandé par l'opérateur signataire.

11.3 Prérequis

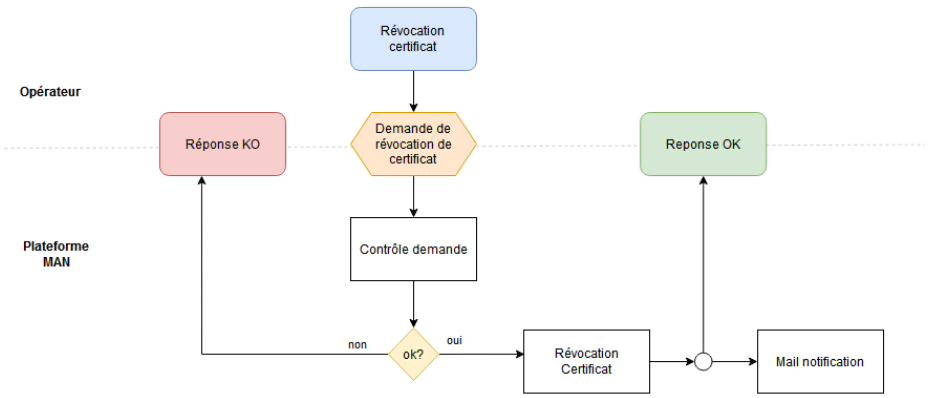
La révocation de certificats peut être réalisée via l'IHM ou l'API de la plateforme. Les prérequis dépendent par conséquent du mode utilisé par l'opérateur pour cette tâche.

- Un certificat direct ou indirect a été délivré à un opérateur (§8)
- Le certificat n'est pas expiré, révoqué ou invalidé
- Utilisation de l'IHM : l'utilisateur dispose d'un compte utilisateur de type « Administrateur » ou « Gestionnaire de certificats ».
- Utilisation de l'API : Un *API credential* a été généré par l'opérateur au niveau de la plateforme pour pouvoir s'authentifier auprès de ses APIs en utilisant le protocole OAuth (§13.1.2).

11.4 Procédure détaillée

La procédure de révocation des certificats opérateur se résume aux étapes suivantes :

- L'opérateur effectue la demande auprès de la plateforme MAN
- Le GCO valide la requête
- Le GCO révoque effectivement le certificat et notifie les opérateurs concernés



11.4.1 Opérateur : Demande de Révocation

L'opérateur peut effectuer la demande de révocation d'un certificat à partir de l'IHM ou de la méthode d'API GCO dédiée `POST /certificate/:id/revoke` (§13.12), où `:id` est l'identifiant du certificat créé par la plateforme.

L'opérateur doit inclure à sa demande la raison pour laquelle il souhaite que ce certificat soit révoqué, choisie parmi une liste finie d'options autorisées par la RFC 5280. L'opérateur peut fournir de plus un texte libre en tant qu'informations complémentaires pour justifier la procédure.

11.4.2 GCO : Contrôle de la requête

Le module valide la requête en vérifiant les points suivants :

- Le certificat est bien associé à l'opérateur, en tant qu'opérateur signataire ou OPTS
- Le certificat n'est pas expiré
- Le certificat n'a pas été révoqué

11.4.3 GCO : Application de la révocation

Une fois la requête validée, le module révoque le certificat en l'ajoutant à la CRL des certificats opérateurs ; La date de modification de la CRL (§2.7.2.4) est modifiée pour correspondre à la date d'application de ces changements.

Le certificat révoqué reste disponible en consultation seule sur la plateforme MAN suivant les règles de cycle de vie des certificats (§2.6.1). L'URL du certificat reste ainsi disponible dans la BPCO jusqu'à l'expiration de ce dernier.

11.4.4 GCO : Confirmation de la révocation du certificat

Afin de répondre au plus vite à l'opérateur signataire, la plateforme confirme la révocation du certificat sans attendre les tâches de notification pouvant être effectuées en parallèle.

- Si la demande a été formulée via l'IHM, la solution GCO retourne les données du certificat et permet au client de le télécharger.
- Si la demande a été faite via une requête API, la solution GCO répond avec un code retour HTTP 204 sans corps de réponse.

11.4.5 GCO : Notification

Une fois la révocation appliquée, le module notifie les opérateurs associés au certificat via leur liste de notification « certificats » :

- L'opérateur signataire pour un certificat direct
- L'opérateur signataire et l'OPTS pour un certificat indirect

Le mail inclut les informations suivantes :

- Serial number du nouveau certificat
- URL du certificat
- Nom de l'opérateur signataire
- Nom de l'OPTS, si certificat indirect
- Date effective de révocation
- Raison de la révocation

12 Suppression de certificats

12.1 Opérateurs concernés

- Opérateur signataire

12.2 Contexte d'application

Suivant le cycle de vie des certificats (§2.6.1), il n'est pas possible pour un opérateur de supprimer un certificat une fois délivré à deux exceptions près :

- Le certificat est un certificat de test
- Un opérateur signataire a initié la demande de délivrance d'un certificat indirect, mais la procédure n'a pas encore été finalisée par l'OPTS

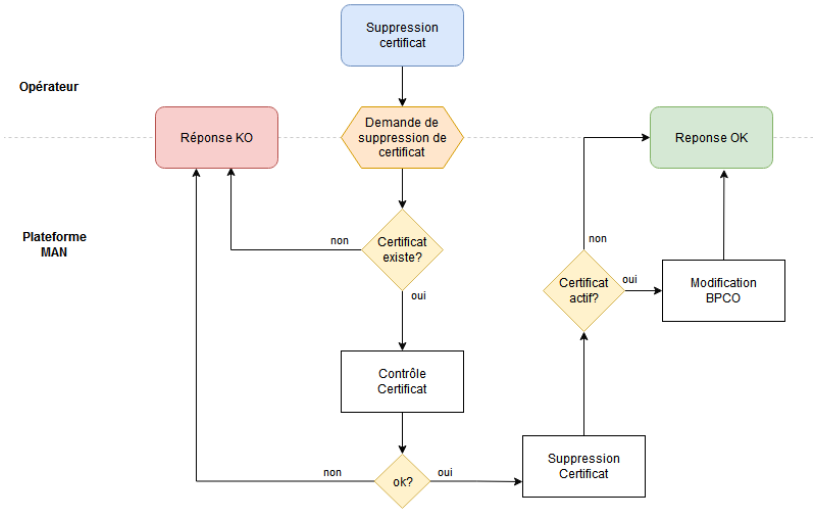
Seul l'opérateur signataire peut être à l'initiative de la suppression d'un certificat.

12.3 Prérequis

La suppression de certificats peut être réalisée via l'IHM ou l'API de la plateforme MAN. Les prérequis dépendent par conséquent du mode utilisé par l'opérateur pour cette tâche.

- Un certificat de test, direct ou indirect, a été délivré à un opérateur signataire (§8). Ce certificat peut être dans un statut révoqué.
- Un certificat indirect a été demandé par un opérateur et la procédure de finalisation de création du certificat n'est pas encore achevée (§8.8)
- Utilisation de l'IHM : l'utilisateur dispose d'un compte utilisateur de type « Administrateur » ou « Gestionnaire de certificats ».
- Utilisation de l'API : Un *API credential* a été généré par l'opérateur au niveau de la plateforme pour pouvoir s'authentifier auprès de ses APIs en utilisant le protocole OAuth (§13.1.2).

12.4 Procédure détaillée



12.4.1 Opérateur signataire : Demande de Suppression

L'opérateur peut effectuer la demande de suppression d'un certificat à partir de l'IHM ou de la méthode d'API dédiée `DELETE /certificate/:id` (§13.13), où `:id` est l'identifiant du certificat créé par la plateforme MAN.

12.4.2 GCO : Contrôle du certificat

La plateforme valide la requête en vérifiant les points suivants :

- Le certificat est bien associé à l'opérateur, en tant qu'opérateur signataire
- Le certificat remplit une de ces deux conditions :
    - Le certificat est un certificat de test
    - Le certificat est un certificat indirect en cours de création

12.4.3 GCO : Modification de la BPCO

Une fois la requête validée, la plateforme supprime le certificat suivant les règles suivantes :

- Si le certificat est un certificat de test, L'URL du certificat est supprimée de la BPCO. La date de modification de la base des certificats (§2.7.1.5) est modifiée pour correspondre à la date d'application de ces changements. Si le certificat de test n'était pas révoqué, le GCO révoque de plus automatiquement le certificat de test afin qu'il soit inscrit dans la CRL des certificats opérateurs et qu'il ne puisse plus être utilisé même si encore présent dans les copies locales des opérateurs.
- Dans le cas d'un certificat indirect en cours de création, aucune modification de la BPCO n'est nécessaire, le certificat n'ayant pas encore été publié.

13 Fonctions de la solution GCO

Le module GCO de la plateforme MAN met à disposition de tous les opérateurs les fonctionnalités de délivrance et gestion des certificats via l'IHM de la plateforme et des APIs dédiées.

Cette section se focalise sur les méthodes d'API utilisées pour le mécanisme de confiance et sur leur utilisation générale. Il convient de se référer aux « guides de référence des APIs de la plateforme MAN » afin de consulter l'ensemble des APIs mises à disposition par la plateforme et des détails quant au format des requêtes et réponses des APIs.

13.1 Protocole d'échange

Les APIs dédiées pour le module GCO de la plateforme MAN sont disponibles via une API REST mise à disposition via un service HTTPs. Le chemin racine d'accès aux différentes méthodes d'API est représenté dans ce document par :

```
https://<domaine-plateforme-man>/
```

13.1.1 Format des requêtes

Les appels API prennent la forme de requêtes HTTPs, dont le corps est au format JSON.

Le service peut vérifier en fonction des API appelées les entêtes suivants :

Entête http	Requis	Utilisation
Authorization	Oui	Token d'autorisation d'accès à l'API
Accept	Non	Format souhaité pour la réponse. Si non précisé, le service retourne dans la réponse le champ *Content-Type* précisant le format utilisé pour la réponse.
Content-Type	Non	Format du corps de la requête. A préciser pour les méthodes POST et PUT. Si non précisé, le service essaiera de lire le corps de la requête par rapport au format attendu par la méthode, mais peut retourner l'erreur 415 si le corps de la requête ne peut être lu.

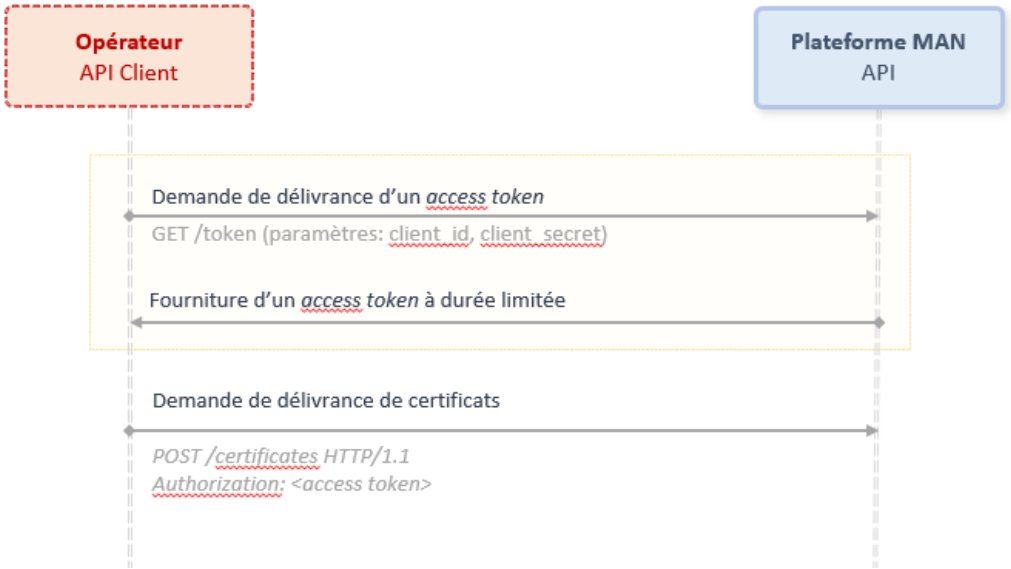
13.1.2 Authentification

L'accès aux méthodes d'API du module GCO requiert une authentification de type OAuth 2.0, où le client d'API doit d'abord se voir délivrer un *access token* par la plateforme, token qui sera alors véhiculé dans les appels aux APIs de la plateforme.

La plateforme MAN permet à l'opérateur de créer via son IHM des *API credentials* contenant les paramètres *client_id* et *client_secret* qui devront être fournis à la requête de délivrance d'un *access token*.

L'*access token* a une durée de vie limitée, de l'ordre de quelques minutes, nécessitant par conséquent au composant en charge de communiquer avec les APIs de récupérer un access token (ou rafraîchir un existant) avant chaque appel d'API.

Exemple dans le cas d'une demande de certificat



13.1.3 Codes réponse HTTP

Les différents codes réponse HTTP pouvant être retournés par le service sont énumérés dans le tableau ci-dessous. En cas d'erreur (code erreur différent de 2XX), le corps de la réponse contient un objet JSON avec l'erreur rencontrée.

Code Réponse HTTP	Contexte
200	Requête traitée avec succès. Le corps de la réponse peut potentiellement contenir les informations requises par la requête.
201	Requête traitée avec succès. Utilisé dans le cadre de création effective de certificats. Le corps de la réponse contient les informations demandées.
202	Requête traitée avec succès. Utilisé dans le cadre de demande de création de certificats indirects. Le corps de la réponse contient les informations demandées.
204	Requête traitée avec succès. Aucun corps de réponse n'est retourné.
400	Les paramètres de la requête sont invalides.
401	L'authentification n'est pas valide.
403	- L'accès à cette URL n'est pas autorisé. - L'utilisateur n'a pas les autorisations pour accéder à cette API.
404	L'objet demandé n'existe pas.
405	La méthode HTTP utilisée n'est pas autorisée.
406	Le format précisé dans l'entête *Accept* de la requête n'est pas supporté.
409	Conflit détecté par rapport au statut de l'objet manipulé ou de l'opérateur.
415	Le format précisé dans l'entête *Content-Type* de la requête n'est pas supporté par le service.
429	Trop de requêtes dans le délai imparti.
500	Une erreur interne est survenue.
503	Le service est indisponible.

13.1.4 Nombre d'appels aux APIs

Un système de *rate limiting* est mis en place, limitant le nombre de requêtes pouvant être envoyées dans un laps de temps donné. Si le client d'API atteint le nombre de requêtes autorisés, toute nouvelle requête sera rejetée temporairement avec une erreur HTTP 429.

Les seuils de déclenchement et les délais appliqués pour cette fonctionnalité seront définis ultérieurement.

13.2 Liste des fonctions

L'accès aux différents APIs dépend du rôle de l'opérateur tel que précisé par ce dernier lors de son enregistrement à la plateforme (§6.3). Le tableau ci-dessous énumère les fonctions publiées par le module GCO et le rôle opérateur requis pour accéder à l'API correspondante.

Fonction	Rôle Opérateur	API
Demande de certificat opérateur direct	Signataire	POST /certificates/
Demande de certificat opérateur indirect	Signataire	POST /certificates/
Finalisation de création de certificat opérateur indirect	OPTS	POST /certificates/:id
Récupération URL publique d'un certificat	Signataire / OPTS	GET /certificates/:id
Téléchargement de la base des certificats opérateurs	Tous	GET /bpco/certificates/
Téléchargement différentiel de la base des certificats opérateurs	Tous	GET /bpco/certificates/?since=DATE
Identification date mise à jour de la base des certificats opérateurs	Tous	HEAD /bpco/certificates/
Téléchargement de la CRL des certificats opérateurs	Tous	GET /bpco/crl/
Identification date mise à jour de la CRL des certificats opérateurs	Tous	HEAD /bpco/crl/
Renouvellement de certificat opérateur	Signataire	POST /certificates/:id/renew
Révocation de certificat opérateur	Signataire / OPTS	POST /certificates/:id/revoke
Suppression de certificat opérateur de test	Signataire	DELETE /certificates/:id

13.3 Demande de certificats opérateurs directs

Cette API est utilisée par les opérateurs signataires afin d'effectuer leur demande de certificats opérateurs directs (§8.7).

13.3.1 Requête

```
POST /certificates
```

Doivent être renseignés dans le corps de la requête :

- Le nom du certificat – valeur indicative permettant de le distinguer des autres certificats délivrés à l'opérateur

- Une description optionnelle
- Type de certificat : **DIRECT**
- Le contenu du fichier CSR au format PEM. La procédure de création du CSR est décrite en section §8.7.2.
- Si ce certificat doit être un certificat de test
- Si l'opérateur souhaite un renouvellement automatique du certificat. Ne peut pas être utilisé pour les certificats de test.
- La date de début du certificat. Si le certificat n'est pas un certificat de test, la date doit être au minimum une semaine après la date actuelle.
- La date de fin du certificat, seulement si un certificat de test est demandé. Voir le code de procédures MAN pour les valeurs minimum et maximum autorisées pour la durée de validité des certificats indirects.

Paramètre	Requis	Format ou Valeur	Valeur par défaut
name	Oui	Chaîne de caractères	
description	Non	Chaîne de caractères	-
type	Oui	DIRECT	
csr	Oui	Chaîne de caractères	
test_certificate	Non	Booléen	false
valid_from	Oui	Date au format ISO-8601	
valid_to	Requis si certificat de test	Date au format ISO-8601	
renewal_auto	Non	Booléen	false
renewal_after	Non	Entier représentant le nombre de jours après la date de génération du certificat pour enclencher la procédure de renouvellement automatique	-

13.3.2 Réponse

Si le certificat peut être délivré, l'API retourne un code HTTP 201 et les informations du certificat dans le corps de la réponse.

Entête	Valeur
Content-Type	application/json
Content-Length	Taille du corps de la réponse

Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
400	La requête est invalide, c'est-à-dire qu'une ou plusieurs propriétés spécifiées sont invalides.

	Ex : La date de début de validité demandée est inférieure à une semaine par rapport à la date actuelle (Ne s'applique que dans le cas d'une demande de certificat de test).
403	- L'opérateur n'est pas renseigné en tant qu'opérateur signataire - ou l'opérateur n'a pas été vérifié.
409	L'opérateur a atteint sa limite de 100 certificats.

13.4 Demande de certificats opérateur indirects

Cette API est utilisée par les opérateurs signataires afin d'effectuer leur demande de certificats opérateurs indirects (§8.8.1).

13.4.1 Requête

```
POST /certificates
```

Doivent être renseignés dans le corps de la requête :

- Le nom du certificat – valeur indicative permettant de le distinguer des autres certificats délivrés à l'opérateur
- Une description optionnelle
- Type de certificat : **INDIRECT**
- L'OPTS mandaté : l'opérateur doit sélectionner parmi la liste des opérateurs s'étant enregistrés comme OPTS dans leur fiche de renseignement (§6.3)
- Si ce certificat doit être un certificat de test
- Si l'opérateur souhaite un renouvellement automatique du certificat. Ne peut pas être utilisé pour les certificats de test.
- La date de début de validité du certificat. Si le certificat n'est pas un certificat de test, la date doit être au minimum une semaine après la date actuelle.
- La date de fin de validité du certificat. Voir le code de procédures MAN pour les valeurs minimum et maximum autorisées pour la durée de validité des certificats indirects.

Paramètre	Requis	Format ou Valeur	Valeur par défaut
name	Oui	Chaîne de caractères	
description	Non	Chaîne de caractères	-
type	Oui	INDIRECT	
opts	Oui	Code APNF opérateur	
test_certificate	Non	Booléen	false
valid_from	Oui	Date au format ISO-8601	
valid_to	Oui	Date au format ISO-8601	
renewal_auto	Non	Booléen	false
renewal_after	Si renewal_auto = true	Entier représentant le nombre de jours après la date de génération du certificat pour enclencher la procédure de renouvellement automatique	-

13.4.2 Réponse

Si la demande de certificat indirect est validée, l'API retourne un code HTTP 202 et dans le corps de la réponse des informations telles que l'ID du certificat en cours de création.

Entête	Valeur
Content-Type	`application/json`
Content-Length	Taille du corps de la réponse

Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
400	- L'opérateur choisi comme OPTS ne s'est pas renseigné en tant qu'OPTS - ou la date de début de validité demandée est inférieure à une semaine par rapport à la date actuelle. Ne s'applique que dans le cas d'une demande de certificat de test. - toute autre erreur dans les paramètres d'entrée
403	- L'opérateur n'est pas renseigné en tant qu'opérateur signataire. - ou l'opérateur n'a pas été vérifié.
409	L'opérateur a atteint sa limite de 100 certificats

13.5 Finalisation de la création du certificat opérateur indirect par l'OPTS

Cette API est utilisée par les OPTS afin de finaliser la création d'un certificat indirect (§8.8.6).

13.5.1 Requête

```
POST /certificates/:id
```

où `:id` est l'identifiant du certificat créé par le GCO lors de l'initialisation du processus (§13.4.2). Doit être renseigné dans le corps de la requête la demande CSR au format PEM.

La procédure de création du CSR est décrite en section §8.8.5.

Une description propre à l'OPTS peut être spécifiée, accessible uniquement de l'OPTS.

Paramètre	Requis	Format ou Valeur	Valeur par défaut
description	Non	Chaîne de caractères	-
csr	Oui	Chaîne de caractères	Contenu de la demande CSR au format PEM

13.5.2 Réponse

Si le certificat peut être délivré, l'API retourne un code HTTP 201 et les informations du certificat, ainsi que son contenu au format PEM, dans le corps de la réponse.

Entête	Valeur
Content-Type	`application/json`
Content-Length	Taille du corps de la réponse

Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
404	Aucun certificat n'existe pour l'identifiant renseigné

13.6 Récupération de l'URL publique d'un certificat

Afin de récupérer l'URL publique dans la BPCO d'un certificat, il convient de faire appel à l'API de récupération des détails d'un certificat, et de récupérer la valeur de la propriété *url* de l'objet JSON défini en corps de la réponse.

13.6.1 Requête

```
GET /certificates/:id
```

Le tableau ci-dessous renseigne les paramètres attendus pour cette requête d'API :

Entête	Emplacement	Présence	Valeur
:id	URL	Obligatoire	UUID du certificat
Accept	Entête requête	Optionnel	`application/json`

13.6.2 Réponse

L'API retourne un code HTTP 200 et le corps de la réponse contient des données du certificat dans la platforme MAN.

Entête	Valeur
Content-Type	`application/json`
Content-Length	Taille de l'objet JSON

L'URL du certificat est disponible dans la propriété **url** :


```
{
  "id": "bf00aee4-71d1-4649-adce-c77bf40dd47c",
  "type": "DIRECT",
  "name": "Main Certificate",
  "valid_from": "2022-01-17T10:12:25Z",
  "valid_to": "2022-01-17T10:12:25Z",
```

```
  "url": "https://domaine-bpco/certs/code-apnf-opérateur/sn-certitificate.cer",
  ...
}
```

Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
404	Aucun certificat n'existe pour l'identifiant renseigné

13.7 Téléchargement des certificats opérateurs

13.7.1 Requête

Une API est disponible afin de récupérer l'ensemble des certificats via un fichier TAR compressé au format gzip.

```
GET /bpco/certificates
```

Seuls les certificats standards sont pris en compte. Les certificats de test ne sont jamais inclus.

Un paramètre d'entrée `since` pourra être spécifié afin de récupérer seulement un delta de la base, à savoir la liste des certificats ajoutés et supprimés de la BPCO depuis la date définie dans ce paramètre. La valeur ne peut pas être antérieure à 15 jours.

```
GET /bpco/certificates?since=2022-01-01T00:00:00Z
```

Il est à noter que les secondes et minutes ne sont pas pris en compte. Ainsi une valeur `2022-01-01T13:45:22Z` sera interprété par l'API en `2022-01-01T13:00:00Z.`

Un entête *If-Modified-Since* peut aussi être inclus à la requête avec une date demandant au composant GCO de ne retourner un résultat que si des certificats ont été ajoutés ou supprimés de la BPCO depuis la date spécifiée.

Le tableau ci-dessous renseigne les paramètres attendus pour cette requête d'API :

Paramètre	Emplacement	Présence	Valeur
since	query	Optionnel	Date au format ISO 8601. Ne peut être antérieur à 15 jours.
If-Modified-Since	Entête requête	Optionnel	Date au format http-date tel que décrit dans les RFC 7231 et 7232

13.7.2 Réponse

Les codes retour possibles pour la réponse à cette API sont les suivants :

Code Réponse HTTP	Contexte
200	La réponse contient le fichier à télécharger
304	Aucune modification n'a été effectuée depuis la date spécifiée dans l'entête *If-Modified-Since* ou le paramètre *since*.

Dans le cas où des données sont disponibles, l'API retourne un code HTTP 200 et le corps de la réponse contient des données binaires au format TAR Gzip.

Entête	Valeur
Content-Type	`application/gzip`
Content-Length	Taille du fichier TaR GZip
Last-Modified	Date de dernière modification de la base des certificats dans la BPCO

Une fois le fichier TAR décompressé, une arborescence de répertoires est disponible, dont la structure correspond à celle utilisée pour les URLs de la base publique des certificats (§14.3) :

- les certificats d'un opérateur sont rassemblés dans un répertoire dont le nom est le code APNF de l'opérateur.
- les certificats sont stockés dans des fichiers dont le nom est le serial number du certificat et l'extension de fichier *cer*. Le serial number est au format hexadécimal sur 16 caractères ou plus.

```
.
|- <code apnf opérateur A>/
|   |- <serial number certificat #1>.cer
|   |- <serial number certificat #2>.cer
|   `- <serial number certificat #3>.cer
|
|- < code apnf opérateur B>/
|   ...
|
`- < code apnf opérateur C>/
    ...
```

Tous les certificats standards non-archivés sont présents au sein de cette arborescence.

Si le paramètre `since` a été renseigné à la requête, un fichier `removed-certificates.csv` est inclus afin de lister la liste des certificats supprimés depuis la date définie dans le paramètre.

```
.
|- removed-certificates.csv
|- <code apnf opérateur A>/
|   |- <serial number certificat #1>.cer
|   `- <serial number certificat #2>.cer
|
|- < code apnf opérateur B>/
|   ...
|
`- < code apnf opérateur C>/
```

Ce fichier est un fichier CSV utilisant comme caractère de fin de ligne et séparateur de colonne respectivement les caractères LineFeed (\n) et point-virgule (;). Chaque ligne du fichier correspond à un certificat supprimé, et est composée de 2 colonnes :

- code APNF de l'opérateur associé
- Serial number du certificat, au format hexadécimal sur 16 caractères ou plus.

```
<code apnf opérateur A>;<serial number certificat #1>
<code apnf opérateur B>;<serial number certificat #2>
<code apnf opérateur C>;<serial number certificat #3>
```

13.8 Identification de la date de mise à jour de la base des certificats opérateurs

Afin de permettre à l'opérateur d'optimiser les procédures de mise à jour de la base en local, il doit lui être possible de récupérer la date de dernière mise à jour de la base des certificats dans la BPCO.

13.8.1 Requête

La même URL pour télécharger les certificats est utilisée, mais cette fois-ci avec la méthode HTTP HEAD.

```
HEAD /bpco/certificates
```

13.8.2 Réponse

L'API retourne un code HTTP 204 et le corps de la réponse est vide. La date de mise à jour de la base des certificats de la BPCO est disponible dans l'entête *Last-Modified* au format HTTP-Date (RFC 7231).

Entête	Valeur
Last-Modified	Date de mise à jour de la base des certificats `Exemple: Mon, 04 Jul 2022 14:53:25 GMT`

13.9 Téléchargement de la CRL des certificats opérateurs

Comme pour la base publique des certificats, il est possible de récupérer la CRL par l'intermédiaire d'une API dédiée.

13.9.1 Requête

```
GET /bpco/crl
```

Un entête *If-Modified-Since* peut être inclus à la requête avec une date demandant au composant GCO de ne retourner la CRL que si cette dernière a été modifiée depuis la date spécifiée.

Paramètre	Emplacement	Présence	Valeur
If-Modified-Since	Entête requête	Optionnel	Date au format HTTP-Date tel que décrit dans les RFC 7231 et 7232

13.9.2 Réponse

Les codes retour possibles pour la réponse à cette API sont les suivants :

Code Réponse HTTP	Contexte
200	La réponse contient la CRL à télécharger
304	Aucune modification n'a été effectuée depuis la date spécifiée dans l'entête *If-Modified-Since*

Dans le cas où le code retour est 200 et le corps de la réponse contient le contenu de la CRL au format DER. Cette CRL est signée avec le certificat PA de la plateforme (§2.5.4).

Entête	Valeur
Content-Type	`application/pkix-crl`
Content-Length	Taille de la CRL
Last-Modified	Date de dernière modification de la CRL

13.10 Identification de la date de mise à jour de la CRL des certificats opérateurs

Afin de permettre à l'opérateur d'optimiser les procédures de jour de la CRL en local, il doit lui être possible de récupérer la date de dernière mise à jour de celle-ci en effectuant une requête HTTP HEAD sur la même API d'accès à la copie.

13.10.1 Requête

```
HEAD /bpco/crl
```

13.10.2 Réponse

L'API retourne un code HTTP 204 et le corps de la réponse est vide. La date de mise à jour de la CRL est disponible dans l'entête *Last-Modified* au format HTTP-Date (RFC 7231).

Entête	Valeur
Last-Modified	Date de mise à jour de la CRL `Exemple: Mon, 04 Jul 2022 14:53:25 GMT`

13.11 Renouvellement de certificats

13.11.1 Requête

```
POST /certificates/:id/renew
```

Le tableau ci-dessous renseigne les paramètres attendus pour cette requête d'API :

Paramètre	Emplacement	Présence	Valeur
:id	URL	Obligatoire	UUID du certificat
name	Corps requête	Optionnel	Nouveau nom pour le certificat créé.
description	Corps requête	Optionnel	Description associée au certificat
valid_from	Corps requête	Optionnel	Date de validité. Si non fournie, la date actuelle + 1 semaine sera utilisée.
valid_to	Corps requête	Optionnel	Date d'expiration. Si non fournie, elle sera définie par la plateforme pour que la durée de validité soit identique au certificat à renouveler.
renewal_auto	Corps requête	Optionnel	Option de renouvellement automatique. Peut être valorisé à *true* seulement si le certificat n'est pas un certificat de test
renewal_after	Corps requête	Optionnel	Nombre de jours après la date de génération du certificat pour lancer la procédure de renouvellement. Ne peut être spécifiée que si le paramètre *renewal_auto* est renseigné à *true*.

13.11.2 Réponse

Si le certificat peut être délivré, l'API retourne un code HTTP 201 le corps de la réponse contient les informations du certificat et son contenu au format PEM.

Entête	Valeur

Content-Type	`application/json`
Content-Length	Taille du corps de la réponse

Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
404	Aucun certificat n'existe avec l'identifiant fourni.
403	Un OPTS a essayé de renouveler un certificat indirect

13.12 Révocation de certificats

13.12.1 Requête

`POST /certificates/:id/revoke`

Le tableau ci-dessous renseigne les paramètres attendus pour cette requête d'API. Le champ reason attend une des valeurs définies dans la RFC 5280.

Paramètre	Emplacement	Présence	Valeur
:id	URL	Obligatoire	UUID du certificat
reason	Corps requête	Obligatoire	Raison de révocation du certificat telle que définie dans la RFC 5280. Peut prendre une des valeurs suivantes : *unspecified, keyCompromise, affiliationChanged, superseded, cessationOfOperation, certificateHold, privilegeWithdrawn*
comments	Corps requête	Optionnel	Informations additionnelles

13.12.2 Réponse

En cas de succès, l'API retourne un code HTTP 204 dans le corps de réponse afin de confirmer le traitement de la requête.

Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
404	Aucun certificat n'existe pour cet opérateur avec l'identifiant fourni.
409	- Le certificat a expiré - Le certificat a déjà été révoqué

13.13 Suppression de certificats

13.13.1 Requête

```
DELETE /certificates/:id
```

Le tableau ci-dessous renseigne les paramètres attendus pour cette requête d'API :

Paramètre	Emplacement	Présence	Valeur
:id	URL	Obligatoire	UUID du certificat

13.13.2 Réponse

En cas de succès, l'API retourne un code HTTP 204 sans corps de réponse. Les erreurs spécifiques pouvant être remontées par cette API sont les suivantes :

Code Réponse HTTP	Contexte
404	Aucun certificat n'existe pour cet opérateur avec l'identifiant fourni.
409	Le certificat n'est pas un certificat de test et le certificat n'est pas en cours de création.

14 Fonctions de la BPCO

La BPCO est le service public d'accès :

- aux certificats opérateur
- à la liste de révocation des certificats opérateurs
- à l'identification de l'autorité de certification
- aux certificats de l'autorité de certification
- à la liste de révocation des certificats de l'autorité de certification

14.1 Protocole d'échange

La BPCO est accessible via un service HTTPs, où différentes URLs sont disponibles pour accéder aux données de la base. Le chemin racine d'accès à ces URLs est le suivant :

```
https://<domaine-bpco>/
```

Il est à noter que ce chemin d'accès est différent de celui utilisé pour les APIs GCO (§ 13.1).

14.1.1 Format des requêtes

Le chemin de l'URL de chaque requête HTTP précise l'objet de la BPCO à récupérer. Le service peut vérifier en fonction des URLs appelées les entêtes suivants :

Entête http	Requis	Utilisation
Accept	Non	Format souhaité pour la réponse. Si non précisé, le service retourne dans la réponse le champ *Content-Type* précisant le format utilisé pour la réponse.

14.1.2 Authentification

Aucune authentification n'est requise ; toutes les URLs sont en accès public.

14.1.3 Codes réponse HTTP

Les différents codes réponse HTTP pouvant être retournés par le service BPCO sont énumérés dans le tableau ci-dessous.

Code Réponse HTTP	Contexte
200	Requête traitée avec succès. Le corps de la réponse contient l'objet demandé.
400	Les paramètres de la requête sont invalides.

404	L'objet demandé n'existe pas. Le corps de la réponse est vide.
405	La méthode HTTP utilisée n'est pas autorisée.
406	Le format précisé dans l'entête *Accept* n'est pas supporté. Le corps de la réponse est vide.
429	Trop de requêtes dans le délai imparti. Le corps de la réponse est vide.
500	Une erreur interne est survenue. Le corps de la réponse est vide.
503	Le service est indisponible. Le corps de la réponse est vide.

14.1.4 Limitation d'accès aux URLs

Un système de *rate limiting* est mis en place, limitant le nombre de requêtes pouvant être envoyées dans un laps de temps donné. Si un client atteint le nombre de requêtes autorisés, toute nouvelle requête sera rejetée pendant ce laps de temps avec une erreur HTTP 429.

14.2 Liste des fonctions

Fonction	Utilisation
Accès certificat opérateur	`GET /certs/<code-apnf-operateur>/<sn-certificate>.cer`
Accès CRL certificats opérateurs	`GET /crl`
Identification du STI-CA	`GET /ca`
Accès certificats STI-CA	`GET /ca/certs`
Accès certificat STI-CA	`GET /ca/certs/<sn-certificat>.cer`
Accès CRL des certificats STI-CA	`GET /ca/crl`

14.3 Accès à un certificat opérateur

14.3.1 Requête

L'URL pour télécharger un certificat est la suivante :

```
GET /certs/<code-apnf-operateur>/<sn-certificate>.cer
```

où *sn-certificate* est le serial number attribué au certificat opérateur, au format hexadécimal sur 16 caractères ou plus.

14.3.2 Réponse

Les codes retour HTTP pouvant être remontées par cette API sont les suivants :

Code Réponse HTTP	Contexte
200	Le certificat est retourné dans la réponse
404	Aucun certificat n'existe pour cet opérateur avec l'identifiant fourni.

Si le certificat existe, le corps de la réponse contient le certificat dans le format PEM. Les entêtes attendus à minima sont :

Entête	Valeur
Content-Type	`application/x-pem-file`
Content-Length	Taille du fichier PEM à télécharger

14.4 Accès à la liste de révocation des certificats opérateur

14.4.1 Requête

L'URL pour télécharger la liste de révocation des certificats opérateurs est la suivante :

```
GET /crl
```

Un entête *If-Modified-Since* peut être inclus à la requête avec une date demandant au composant BPCO de ne retourner la CRL que si cette dernière a été modifiée depuis la date spécifiée.

Paramètre	Emplacement	Présence	Valeur
If-Modified-Since	Entête requête	Optionnel	Date au format HTTP-date tel que décrit dans les RFC 7231 et 7232

14.4.2 Réponse

Les codes retour possibles pour la réponse à cette API sont les suivants :

Code Réponse HTTP	Contexte
200	La liste des certificats est retournée
304	Aucune modification n'a été effectuée depuis la date spécifiée dans l'entête *If-Modified-Since*.

Dans le cas où le code retour est 200, le corps de la réponse contient la liste de révocation dans le format DER, signée par le certificat PA de la plateforme MAN. Les entêtes attendus à minima sont :

Entête	Valeur
Content-Type	`application/pkix-crl`
Content-Length	Taille du fichier CRL à télécharger

Last-Modified	Date de dernière mise à jour (format HTTP-Date)

14.5 Identification du STI-CA

Cette URL permet de vérifier l'autorité de certification de la plateforme MAN. Elle permet à la solution française d'être conforme avec le standard ATIS-1000084.v2, où il est attendu que le STI-PA fournisse la liste des STI-CA approuvés pour un pays.

14.5.1 Requête

```
GET /ca
```

Un entête *If-Modified-Since* peut être inclus à la requête avec une date demandant au composant BPCO de ne retourner une réponse que si les données ont été modifiées depuis la date spécifiée.

Paramètre	Emplacement	Présence	Valeur
If-Modified-Since	Entête requête	Optionnel	Date au format HTTP-date tel que décrit dans les RFC 7231 et 7232

14.5.2 Réponse

Les codes retour possibles pour la réponse à cette API sont les suivants :

Code Réponse HTTP	Contexte
200	La liste des certificats est retournée
304	Aucune modification n'a été effectuée depuis la date spécifiée dans l'entête *If-Modified-Since*.

Dans le cas où le code retour est 200, le corps de la réponse contient un JSON Web Token fournissant les certificats racines de l'autorité de certification de la plateforme MAN (§2.5.2). Ce token est signé par le certificat PA (§2.5.4) de la plateforme MAN. Les entêtes retournés sont :

Entête	Valeur
Content-Type	`application/jose+json`
Content-Length	Taille du corps de la réponse
Last-Modified	Date de dernière mise à jour (format HTTP-Date)

Le header fournit au sein de la propriété x5u l'URL d'accès au certificat PA ayant signé le token.

Les données du payload du token sont les suivantes :

- **exp** : timestamp indiquant la date d'expiration du token. Défini à 1 semaine.
- **sequence** : incrémenté à chaque fois que le certificat racine de l'autorité change
- **trustList** : objet JSON contenant la liste des certificats racines au format PEM

- **version** : doit rester à 1.0. Pourra être incrémenté si le format du payload change

```
"protected": base64url({
  "alg" : "ES256",
  "typ" : "JWT",
  "x5u" : "https://<domaine-bpco>/ca/certs/sn-bpco-pa1.cer"
})


"payload": base64url({
  "exp": 1300819380,
  "sequence": 1,
  "trustList": [
"-----BEGIN CERTIFICATE-----
BPCO CR 1 certificate contents
-----END CERTIFICATE-----"
  ],
  "version": 1
})
```

14.6 Accès à la liste des certificats intermédiaires du STI-CA

Cette URL publique permet de récupérer la liste des certificats intermédiaires de l'autorité de certification de plateforme MAN pouvant être utilisés pour signer les certificats opérateur (§2.5.3).

14.6.1 Requête

```
GET /ca/certs
```

Un entête *If-Modified-Since* peut être inclus à la requête avec une date demandant au composant BPCO de ne retourner des données que si celles-ci ont été modifiées depuis la date spécifiée.

Paramètre	Emplacement	Présence	Valeur
If-Modified-Since	Entête requête	Optionnel	Date au format HTTP-date tel que décrit dans les RFC 7231 et 7232

14.6.2 Réponse

Les codes retour possibles pour la réponse à cette API sont les suivants :

Code Réponse HTTP	Contexte
200	La liste des certificats est retournée
304	Aucune modification n'a été effectuée depuis la date spécifiée dans l'entête *If-Modified-Since*.

Dans le cas où le code retour est 200, la réponse doit être un message au format JSON Web Token contenant la liste de l'ensemble des certificats intermédiaires. Ce token est signé par le certificat PA de la plateforme MAN (§2.5.4). Les entêtes retournés sont :

Entête	Valeur
Content-Type	`application/jose+json`
Content-Length	Taille du corps de la réponse
Last-Modified	Date de dernière mise à jour (format HTTP-Date)

Le header fournit au sein de la propriété x5u l'URL d'accès au certificat PA ayant signé le token.

Les données du payload du token sont les suivantes :

- **certList** : objet JSON contenant les certificats intermédiaires au format PEM
- **exp** : timestamp indiquant la date d'expiration du token. Défini à 1 semaine.
- **sequence** : incrémenté à chaque fois que la liste change
- **version** : doit rester à 1.0. Pourra être incrémenté si le format du payload change

```
"protected": base64url({
  "alg": "ES256",
  "typ": "JWT",
  "x5u": "https://<domaine-bpco>/ca/certs/sn-bpco-pa1.cer"
})
```

```
"payload":base64url({
  "certList": [
"-----BEGIN CERTIFICATE-----
BPCO CA1 certificate contents
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
BPCO CA2 certificate contents
-----END CERTIFICATE-----"
  ],
  "exp": 1300819380,
  "sequence": 1,
  "version": 1
})
```

14.7 Accès à un certificat de l'autorité de certification

14.7.1 Requête

L'URL pour télécharger un des certificats de l'autorité de certification, qu'il soit racine (§2.5.2), intermédiaire (§2.5.3) ou PA (§2.5.4), est la suivante :

```
GET /ca/certs/<sn-certificat>.cer
```

où *sn-certificat* est le serial number attribué au certificat, au format hexadécimal sur 16 caractères ou plus.

14.7.2 Réponse

Les codes retour HTTP pouvant être remontées par cette API sont les suivants :

Code Réponse HTTP	Contexte
200	Le certificat est retourné dans la réponse
404	Aucun certificat n'existe avec l'identifiant fourni.

Si le certificat existe, le corps de la réponse contient le certificat dans le format PEM. Les entêtes attendus à minima sont :

Entête	Valeur
Content-Type	`application/x-pem-file`
Content-Length	Taille du fichier PEM à télécharger

14.8 Accès à la liste de révocation des certificats de l'autorité de certification

14.8.1 Requête

L'URL pour télécharger la liste de révocation des certificats de l'autorité de certification de la plateforme MAN est la suivante :

```
GET /ca/crl
```

Un entête *If-Modified-Since* peut être inclus à la requête avec une date demandant au composant BPCO de ne retourner la CRL que si cette dernière a été modifiée depuis la date spécifiée.

Paramètre	Emplacement	Présence	Valeur
If-Modified-Since	Entête requête	Optionnel	Date au format HTTP-date tel que décrit dans les RFC 7231 et 7232

14.8.2 Réponse

Les codes retour possibles pour la réponse à cette API sont les suivants :

Code Réponse HTTP	Contexte
200	La réponse contient le fichier à télécharger
304	Aucune modification n'a été effectuée depuis la date spécifiée dans l'entête *If-Modified-Since*.

Si la CRL est retournée, l'API retourne un code 200 avec la liste de révocation au format DER, signée par le certificat racine de la plateforme MAN. Les entêtes attendus à minima sont :

Entête	Valeur
Content-Type	`application/pkix-crl`
Content-Length	Taille du fichier CRL à télécharger

15 Procédures en cas d'incident

Le code de procédures MAN décrit les procédures mises en place en cas d'incident :

- Sur la vérification des appels (STI-VS) chez un opérateur (transit/OPTV/terminaison)
- Sur la signature des appels (STI-AS) chez un opérateur (signataire / OPTS)
- Sur la plateforme MAN