EBICS - Guide de mise en œuvre en France (Implementation Guidelines)

Version 2.1.5

AVIS AU LECTEUR*

La documentation relative au protocole EBICS, conçue dans sa version initiale par le ZKA (équivalent allemand du CFONB) a été rédigée en allemand puis traduite en anglais.

Sa version 2.4 constitue la première version commune Franco-Allemande. Une mise à jour V2.4.1 a été publiée en septembre 2009.

Néanmoins la mise en œuvre d'EBICS en France doit être adaptée au contexte national (migration des ETEBAC 3&5, instruments de paiements nationaux, etc..).

La dernière version des spécifications EBICS V2.4.2, publiée en février 2010, complète ou clarifie la version 2.4.1 pour l'implémentation de la signature personnelle jointe. Une liste exhaustive des clarifications est disponible dans les spécifications, ce guide correspond à la version V2.4.2.

Le CFONB a élaboré ce guide d'implémentation d'EBICS en France, pour les raisons suivantes :

1. afin d'assurer le remplacement progressif des protocoles ETEBAC 3 puis 5, il convenait de prendre en compte, pour donner le maximum de souplesse à cette migration, les « habitudes » d'utilisation des entreprises françaises en matière de transfert de fichiers,
2. pour le remplacement d'ETEBAC 3, des adaptations étaient nécessaires par rapport aux fonctionnalités offertes par le protocole en matière de sécurité de transport,
3. pour le remplacement d'ETEBAC 5, des adaptations étaient nécessaires pour la gestion des ordres d'exécution signés électroniquement et transportés avec les données à exécuter.
4. la liste des commandes figurant dans le protocole comprend bien évidemment les commandes utilisées dans les contextes allemands et français. Ce guide identifie avec précision le sous-ensemble recommandé pour les commandes utilisées en France,
5. le protocole EBICS est en particulier destiné à l'acheminement d'instructions de paiements européens (SEPA) mais aussi nationaux (VO, LCR,..) ainsi que des restitutions clientèles (extraits de comptes,..). La description de leur paramétrage était nécessaire.
   Il peut également être employé pour d'autres usages bancaires.

TABLEAU D'EVOLUTIONS

Version	Date IG	Chapitre IG	Type	Description des évolutions
VO 1.0	**02/2009**			Création du document
VF 1.1	**05/2009**	Tous	C	La dénomination « Signature Bancaire » est remplacé par « Ordre d'Exécution ». La dénomination « Fichier Métier » est replacée par « Remise d'Ordre ».
		Tous	C	Insertion des réponses aux questions reçues de la FAQ
		2.1.1.1 et 2.1.1.2	C	Précisions sur l'initialisation des paramètres sécuritaires et le calcul la signature du A006
		2.1.4	C	Test : précision sur le format du paramètre TEST
		2.2	C	Echanges de flux : restructuration du chapitre
		3	C	Modalités d'utilisation du poste client : mise en conformité avec le contrat type
		Annexe 2	M	Afin de faciliter la mise à jour, ces documents sont extraits des IG et publiés sur le site du CFONB.
VF 1.2	**09/2009**	Tous		Mise en cohérence avec la version 2.4.1 des spécifications
		1.1	C	Complémentarité des phases
		1.2.5	A	Codage ASCCI / EBCDIC
		Tous	C	Le PSR est mis à disposition et non pas envoyé
		2.1.2.2	C	Remarque : Calcul de la signature : Précisions suite aux réponses données à des questions
		2.1.4	C	Cas des prestataires de services
		A2	A	Nommage des fichiers : ajout du PSR de niveau protocolaire (type pain.002.001.02.ack)
		A.5	A / M	Format du Payment Status Report
VF 2.0	**03/2010**	tous	A / M	Intégration des modalités de gestion de la signature personnelle. Abandon de la notion de phase pour les appellations EBICS profil T et EBICS profil TS
		2.1.5	A	Gestion de plusieurs utilisateurs chez un même abonné
		A.5	M	Format du Payment Status Report : Externalisation dans un document spécifique
VF 2.1	**09/2010**	tous	C	Clarifications par rapport à la V2.0
		1.2.5	M	Ne pas rejeter une remise lorsque ES quantity =1 et réception de 2 signatures Rejet lorsque plus de 2 signatures sont reçues
VF2.1.1	**01/02/2011**	A3.	M	KeyUsage, nouvelle spécification
VF2.1.2	**21/06/2011**	2.1.2.1	C	Ces clarifications permettent de respecter les règles sécuritaires d'usage en matière de certificats personnels.

VF2.1.3	**13/10/2011**	1.2.9	C	Gestions des séparateurs fin de ligne
VF2.1.4	**24/02/2012**	1.2.8	C	Précision dans le texte « ….dans le sens banque/client uniquement… »
VF2.1.5	**27/05/2014**	2.1.2.5	A	Vérification automatique des certificats serveurs

[1] E : Error ; M : Modification ; C : Clarification ; S : suppression ; A : Addition/Extension

[*] Nota :

Afin de faciliter leur actualisation et le chargement par les éditeurs, les listes de données suivantes ne figurent pas en annexe de ce document mais sont disponibles en tant que documents distincts sur le site du CFONB : www.cfonb.org dans la même rubrique que ce guide d'implémentation (Documentation : Migration ETEBAC vers EBICS et SWIFTNet) :

- Annexe A2 dans son intégralité : FileFormat/Request Type – Nommage des fichiers
- Annexe A3.1: Liste des messages d'erreurs liés aux certificats
- Annexe A5 : Format du Payment Status Report
- Annexe A6 : Exemple de calcul de Hash

SOMMAIRE

1 INTRODUCTION

1.1 *Objet et périmètre du guide*

Ce guide d'implémentation est destiné aux développeurs des postes clients et des serveurs EBICS. Il ne s'agit pas d'un manuel utilisateur destiné aux clients. Il a pour but principal de préciser les modalités et paramètres d'implémentation.

Il est de la responsabilité des fournisseurs de logiciel de prévoir un manuel utilisateur. Ce manuel utilisateur devra contenir notamment la liste des codes retours EBICS ainsi que les codes retours spécifiques au logiciel avec pour chacun un libellé explicite.

Ce guide d'implémentation est un complément de la documentation EBICS :

- Spécifications générales version 2.4.2 et ses annexes Order Type et Codes Retour,
- Guide d'implémentation version 1.7.
  en Allemand http://www.ebics.de/index.php?id=93
  ou traduites en anglais http://www.ebics.org/index.php?id=93

La lecture de cette documentation est un préalable nécessaire à la bonne compréhension de ce présent guide.

Ce guide s'appuie sur la version 2.4.2 d'EBICS, commune à la France et à l'Allemagne, et codifiée H003 dans les messages. Cette version 2.4.2 apporte des clarifications par rapport à la version 2.4.1 qui a été implémentée dans les deux pays à partir de l'automne 2009.

En cible, l'utilisation du protocole EBICS, sera identique en France et en Allemagne. Cependant compte-tenu de l'existant des deux pays, les modalités d'implémentation de cette version pourront être légèrement différentes d'un pays à l'autre.

De plus, l'utilisation de certaines fonctionnalités optionnelles étant du domaine concurrentiel, il est de la responsabilité de l'utilisateur de s'assurer du niveau de service offert par ses établissements bancaires et de paramétrer le logiciel en fonction de ce niveau.

Le périmètre de ce guide CFONB est limité aux échanges de flux entre postes clients localisés en France et serveurs des banques localisées en France. Le choix de l'extension aux échanges transfrontaliers du poste client et/ou du serveur est du ressort des éditeurs de logiciels ou des banques[2].

Ce guide est plus particulièrement destiné à décrire les paramétrages des postes clients mais il donne également des recommandations pour le paramétrage des serveurs bancaires.

L'implémentation devra respecter les préconisations et règlementations existantes, notamment celles relatives aux services bancaires et/ou financiers sur Internet (SBFI) (voir site : www.ssi.gouv.fr/site_documents/pp/ppcr0401.pdf).

Ce guide décrit les modalités de déploiement du protocole EBICS en tant que

- remplaçant d'ETEBAC3 avec une confirmation, disjointe d'EBICS, de l'ordre d'exécution. Dans la suite du document, ces modalités seront décrites sous le vocable EBICS profil T (Transport)

ou

- remplaçant d'ETEBAC5 avec une signature électronique du client jointe à l'ordre. Dans la suite du document, ces modalités seront décrites sous le vocable EBICS profil TS (Transport et Signature personnelle).

[2] Le terme « banque » utilisé dans ce document doit être entendu comme Prestataire de Services de paiement (PSP) au sens de la directive 2007/064/CE sur les services de paiements du 17 novembre 2007, transposée dans l'ordonnance 2009-866 du 15 juillet 2009.

Remarque : Le protocole EBICS permet l'acheminement d'une signature de l'ordre de façon disjointe via EBICS, cette fonction n'est pas abordée dans cette version de guide mais figurera dans un profil « DS Signature Distribuée » ultérieur.

1.2 *Règles communautaires d'implémentation*

1.2.1 Multi-bancarité des postes clients :

Chaque poste client doit pouvoir se connecter à divers serveurs bancaires respectant les préconisations de ce guide. Il devra être possible de rajouter ou de supprimer une connexion bancaire sur un poste installé.

1.2.2 Les modalités d'implémentation :

Elles peuvent être de deux natures :

- Soit liées aux recommandations interbancaires françaises,
- Soit liées à l'offre de service d'un ou plusieurs établissements bancaires.

➢ Les recommandations interbancaires françaises concernent principalement :

- Les commandes utilisées (Order type – voir annexe A1)
- Le nommage des fichiers (FileFormat/Request Type – voir annexe A2)
- Les types de certificats (Certificats - voir annexe A4) en fonction du profil T ou TS

Pour limiter la liste des commandes, il n'y a pas autant de commandes que de types de fichiers (contrairement à l'implémentation en Allemagne), mais le type de fichier est une valeur de l'un des paramètres (FileFormat/request type) de la commande.

➢ Les offres de services bancaires peuvent porter sur :

- Les contrôles complémentaires à effectuer par le protocole sur les données bancaires (ex : contrôle du montant ou du compte),
- Les niveaux spécifiques de sécurité,
- Les types de fichiers propriétaires,
- Les flux d'informations propriétaires,
- Les choix bancaires de rendre obligatoire ou d'interdire un paramétrage optionnel
- La profondeur de l'historique des fichiers mis à disposition sur le serveur
- Les modalités de récupération de ces fichiers par le client.

Les modalités spécifiques liées à des offres bancaires ne sont pas couvertes par ce guide.

Afin de pouvoir paramétrer l'installation, chaque banque doit fournir au préalable à son client les informations décrivant son abonnement.

1.2.3 Sécurisation des échanges :

Le protocole EBICS assure la sécurisation des échanges à deux niveaux :

- Au niveau réseau avec le protocole https basé sur un certificat serveur au niveau du SI de la banque.
- Au niveau applicatif protocolaire par l'utilisation de certificats dédiés aux usages authentification, confidentialité et contrôle d'intégrité/signature.

Au niveau réseau :
La confidentialité des échanges sur le réseau internet est établi au travers d'une connexion https (couche TLS Transport Layer Security). Basée sur l'utilisation d'un certificat serveur au niveau du SI de la banque, ce certificat, préalablement récupéré et accepté par le poste client avant tout échange protocolaire permettra un premier niveau d'authentification du serveur de la banque par le client. Après établissement du canal https, toutes les données (commandes EBICS, données clients et signatures) seront transportées dans ce réseau sécurisé TLS.

Au niveau applicatif, le protocole EBICS assure :

1. l'authentification réciproque de l'abonné et du serveur de la banque par échanges signés à l'aide de certificats, client et serveur, dédiés à ce service d'authentification
2. La confidentialité, en plus de celle assurée par la couche TLS de https, par chiffrement des données basé sur des certificats, client et serveur, dédiés à ce service de chiffrement.
3. en EBICS T : pour assurer l'intégrité des données transmises par le poste client au serveur bancaire, les données sont scellées par signature électronique en utilisant un certificat dédié à cette fonctionnalité. L'ordre d'exécution est transmis sur un canal de communication distinct d'EBICS.
4. en EBICS TS : pour assurer l'intégrité et la non répudiation des données transmises par le poste client au serveur bancaire, les données sont signées électroniquement en utilisant un certificat dédié à cette fonctionnalité. La signature a valeur d'ordre d'exécution.

1.2.4 Implémentation d'un contrat :

Il est recommandé dans un contrat donné de toujours définir un « UserId » (personne physique ou service) de transport. Ce « UserId » de transport a, a priori, l'ensemble des droits de transport pour tous les flux du contrat et notamment pour l'ensemble des flux de reporting. Ce « UserId » de transport, comme son nom l'indique, n'a, bien sûr, aucun droit de signature bancaire sur les flux.

Tous les utilisateurs (UserId) ayant pouvoir de signature personnelle doivent figurer dans le contrat. Pour chaque flux (défini par le File format) , sont listés ensuite les droits de chaque signataire sur le flux en question (simple signature, double signature, double signature optionnelle, éventuellement plafond d'ordres ou de remises).

En EBICS TS, chaque signataire doit disposer d'un certificat de signature personnelle sur support physique dont le gabarit est conforme à ce qui figure en annexe A3. Ce certificat est obligatoirement délivré par une AC reconnue par la banque.

Les données associées au signataire sont :
- son nom et prénom,
- le nom de l'AC utilisé par le signataire
- une donnée assurant l'unicité du certificat au sein d'une AC. Selon les AC cette donnée peut être : DN, SAN,.., voire numéro de série.

En EBICS T, ces signataires sont rattachés à un user EBICS de classe T.
En EBICS TS, ces signataires sont rattachés à un user EBICS de classe E.

Rappel :

- Chaque utilisateur (UserId) (transport ou signataire) doit être détenteur de 3 certificats (Authentification/Chiffrement/Signature)
- Pour un utilisateur (UserId) de signature, seul le certificat de signature personnel doit être obligatoirement un certificat délivré par une AC.
- Lors de l'initialisation de l'abonnement (envoi des commandes INI et HIA) 2 cas se présentent pour valider l'utilisateur (UserId):
  - Si le certificat est un certificat auto généré, par vérification du courrier, envoyé à la banque, contenant les hash des certificats et signé par une personne habilitée dans l'entreprise.
  - Si le certificat est délivré par une AC, par comparaison entre les données du contrat et celles issues de la chaine de certification.

1.2.5 Intégrité et niveau de signature des échanges :

Le paramètre ES Quantity[3] est défini sur le serveur banque en fonction du contrat et en fonction du FileFormat.
Suivant les modalités contractuelles définies entre la banque et son client, le poste client doit être paramétré :

- **EBICS profil T**

  En mode Transport, pour EBICS profil T, le client confirme ses ordres d'exécution par un autre canal de communication. Dans ce cas, le poste est paramétré avec « classe de signature des utilisateurs » = T (Transport), ES quantity = 0.

Cas Signature disjointe (supportée sans le mode DS/VEU) : EBICS profil T				
Signatures Personnelles	**0**	**1**	**2 et +**	
Attendues ES = 0	**OK**	**REJ**	**REJ**	**OrderAttributes = DZHNN, toutes les signatures sont considérées comme des signatures de transport (quelle que soit la classe A, B, E ou T affectée à l'utilisateur).**

REJ = rejet

- **EBICS profil TS**

  En EBICS profil TS (Signature jointe à l'ordre d'exécution), le client utilisateur signe ses ordres électroniquement conjointement à l'envoi des ordres via EBICS.

  Dans ce cas, la « classe de signature des utilisateurs » est codifiée E (signature électronique) La fonction de signature de cet utilisateur est assurée conformément au contrat en utilisant le certificat personnel qui lui a été délivré par l'AC et reconnu par la banque.
  Suivant la quantité de signature attendue (fonction du FileFormat et décrit dans le contrat de service conclu entre l'utilisateur et la banque), la signature a pouvoir d'exécution seule ou associée à une seconde signature. L'indicateur ES quantity = 1 ou 2 définit le nombre minimal de signataires attendus.
  Pour ES quantity = 1, les ordres reçus avec 2 signatures sont acceptés. Cette fonctionnalité permet, par exemple, de traiter le cas d'un nombre de signatures variant selon le montant de l'ordre transmis.
  Dans l'implémentation d'EBICS en France, la validation finale n'est pas obligatoirement traitée directement au niveau du protocole EBICS, il est donc nécessaire de laisser passer tous ces ordres sur le serveur EBICS, et de confier la vérification du nombre de signatures attendues à l'applicatif bancaire « métier » en aval.

[3] Ce paramètre représente le nombre de signatures personnelles (voir spécifications EBICS 2.4.2)

Note :

- Le contrat décrit, pour chaque FileFormat, le nombre de signatures requises :
  - Simple signature (ES quantity = 1),
  - Une ou deux (ES quantity = 1) selon le montant par exemple,
  - Deux obligatoirement (ES quantity = 2).

- Un rejet de l'ordre, par excès ou manque de signature, peut donc se produire en aval du serveur EBICS après une acceptation au niveau du protocole EBICS.
- Pour éviter les rejets, l'applicatif client doit contrôler le nombre de signatures attendues en fonction du contrat avant d'autoriser l'envoi.
- Dans tous les cas, il n'est pas accepté plus de 2 signatures.
- Le protocole EBICS permet l'utilisation de 2 autres classes de signature A et B, mais cette utilisation n'est pas retenue, à ce jour, en France.
- Tout envoi ne comportant pas de signature ou comportant plus de 2 signatures est rejeté au niveau du serveur EBICS.

Cas Signature jointe (supportée sans le mode DS/VEU) : EBICS profil TS					
Signature Personnelle	**0**	**1**	**2**	**>2**	
Attendues ES = 1	**REJ**	**OK**	**OK**	**REJ**	**Une ou deux signatures de classe "E" sont attendues**
Attendues ES = 1	**REJ**	**REJ**	**REJ**	**REJ**	**Toute autre combinaison avec A, B ou T entraîne un REJET**
Attendues ES = 2	**REJ**	**REJ**	**OK**	**REJ**	**double-signature obligatoire : Deux signatures de classe "E" sont attendues**
Attendues ES = 2	**REJ**	**REJ**	**REJ**	**REJ**	**Double-signature obligatoire : Toute autre combinaison (A, B ou T) entraîne un "REJET"**

Exemple avec 1 signature personnelle attendue

Dans ce cas, 1 signature est requise pour les prélèvements nationaux (FileFormat ddd).
Contractuellement, l'association pour le fichier FileFormat ddd est ES quantity = 1.
Cependant, l'envoi n'est pas rejeté par EBICS s'il comporte 2 signatures, mais peut être rejeté par l'applicatif métier.

Signatures Personnelles	**0**	**1**	**2**	**>2**	**Remarque**
Attendue ES = 1	**REJ**	**OK**	**OK**	**REJ**	**une signature de classe "E" est attendue**
Attendue ES = 1	**REJ**	**REJ**	**REJ**	**REJ**	**Toute autre combinaison avec A, B ou T entraîne un REJET**

Rejet si signature type Transport, mais pas si double-signature.

Exemple avec 1 ou 2 Signatures Personnelles attendues

Dans ce cas, 1 signature est requise pour les SCT de petits montants et 2 signatures pour les SCT de gros montants.
Contractuellement, l'association pour le fichier FileFormat sct est ES quantity = 1.
Tous les envois comportant 1 ou 2 signatures sont acceptés au niveau du serveur EBICS. Les SCT de gros montants sont rejetés au niveau de l'applicatif métier s'ils ne comportent pas 2 signatures.

Signatures Personnelles	0	1	2	>2	remarque
Attendue ES = 1	REJ	OK	OK	REJ	Indicateur "double-signature optionnelle" : Une ou deux signatures de classe "E" ou «E+E » sont attendues
Attendue ES = 1	REJ	REJ	REJ	REJ	Indicateur "double-signature optionnelle" : Toute autre combinaison avec A, B ou T entraîne un REJET

Exemple avec 2 Signatures personnelles 'Double-Signature Obligatoire'

Dans ce cas, 2 signatures sont requises pour les Virements de Trésorerie (Fileformat : ict)
Contractuellement, l'association pour le fichier FileFormat ict est ES quantity = 2.

Signatures Personnelles	0	1	2	>2	remarque
Attendue ES = 2	REJ	REJ	OK	REJ	double-signature obligatoire : Deux signatures de classe "E" sont attendues
Attendue ES = 2	REJ	REJ	REJ	REJ	Indicateur Double-signature obligatoire : Toute autre combinaison (A, B ou T) entraîne un "REJET"

1.2.6 Les caractères spécifiques :

Les caractères autorisés au niveau de l'échange protocolaire en France ou en Allemagne sont définis par le schéma XSD. Les caractères spécifiques (par exemple é, è, à, ç, œ, ü, ß, Ä) sont exclus par le schéma XSD.
Par contre, dans les fichiers échangés, les caractères à utiliser sont ceux définis dans les standards de messages.

1.2.7 Codage ASCII / EBCDIC :

Il est recommandé que les fichiers échangés soient en ASCII. Si la banque le propose, pour l'envoi de fichiers en EBCDIC, l'indication « EBCDIC » doit figurer dans les balises «FULOrderParam » sur le modèle suivant :

```
<FULOrderParams>
      <Parameter>
            <Name>EBCDIC</Name>
            <Value>TRUE</Value>
      </Parameter>
      <FileFormat CountryCode="FR">pain.xxx.cfonb160.dct</FileFormat>
</FULOrderParams>
```

En acquisition, le serveur doit accepter les fichiers avec ou sans séparateur, et en restitution, c'est le poste client qui doit savoir le faire.

1.2.8 Parseur :

Les fichiers au format fixe peuvent contenir plusieurs itérations logiques (remises ou lots) à condition qu'elles soient avec un FileFormat identique. Dans ce cas, elles sont mises bout à bout pour être fusionnées dans un seul fichier physique.

Les fichiers au format variable (XML) suivent le même mécanisme dans le sens banque/client uniquement : un fichier physique peut contenir plusieurs messages logiques qui sont assemblés sans aucune balise. En conséquence, avant tout traitement, il est nécessaire de désassembler ces messages pour les intégrer dans un parseur.

1.2.9 Gestion des séparateurs fin de ligne (CR, LF, CRLF, ou absence) :

Dans le sens client vers banque, le serveur banque doit accepter tous les types de séparateurs, y compris l'absence de séparateur.

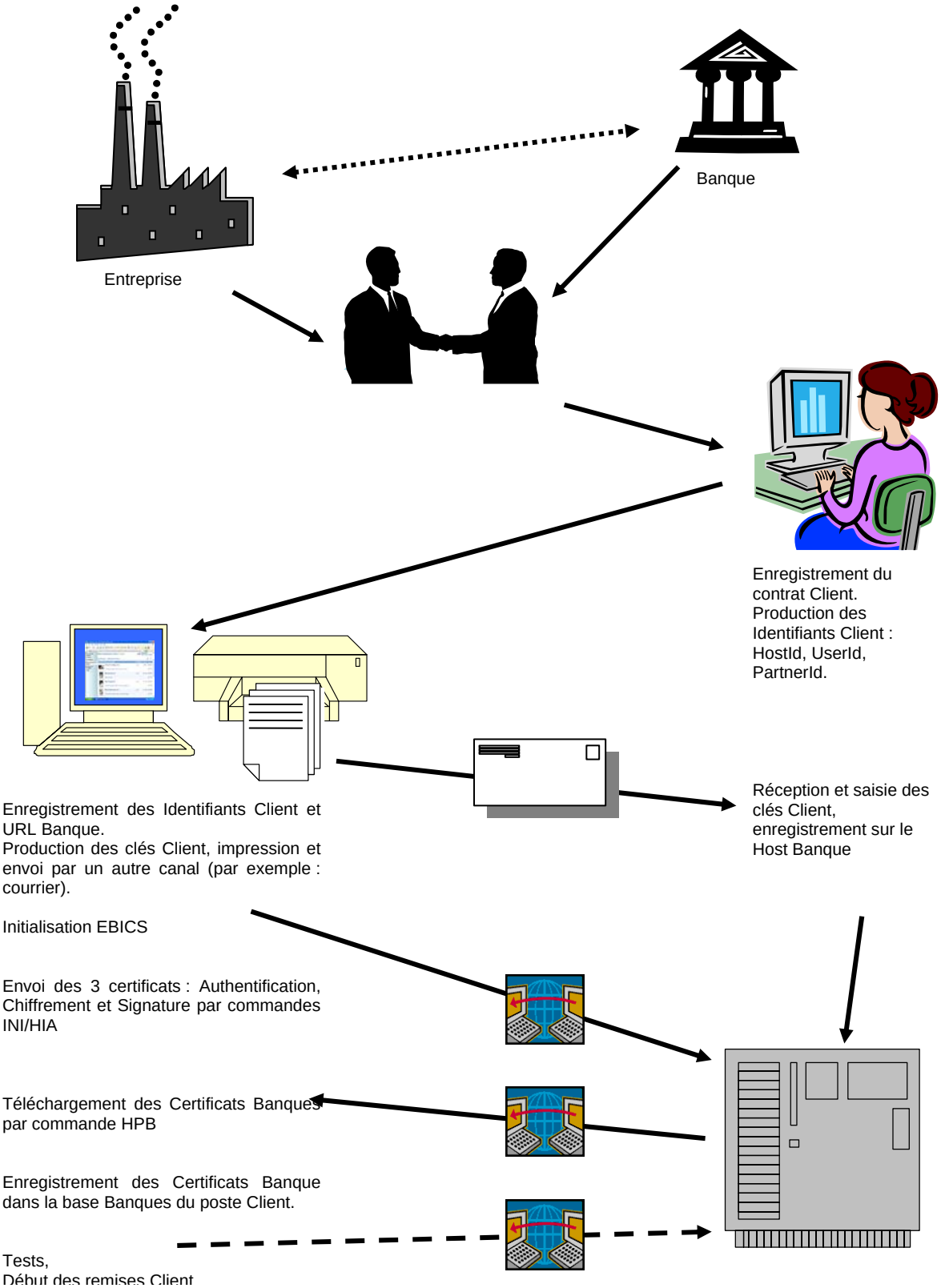
Dans le sens banque vers client, le poste client doit aussi savoir traiter les fichiers contenant tout type de séparateur, y compris l'absence de séparateur.

2 IMPLEMENTATION

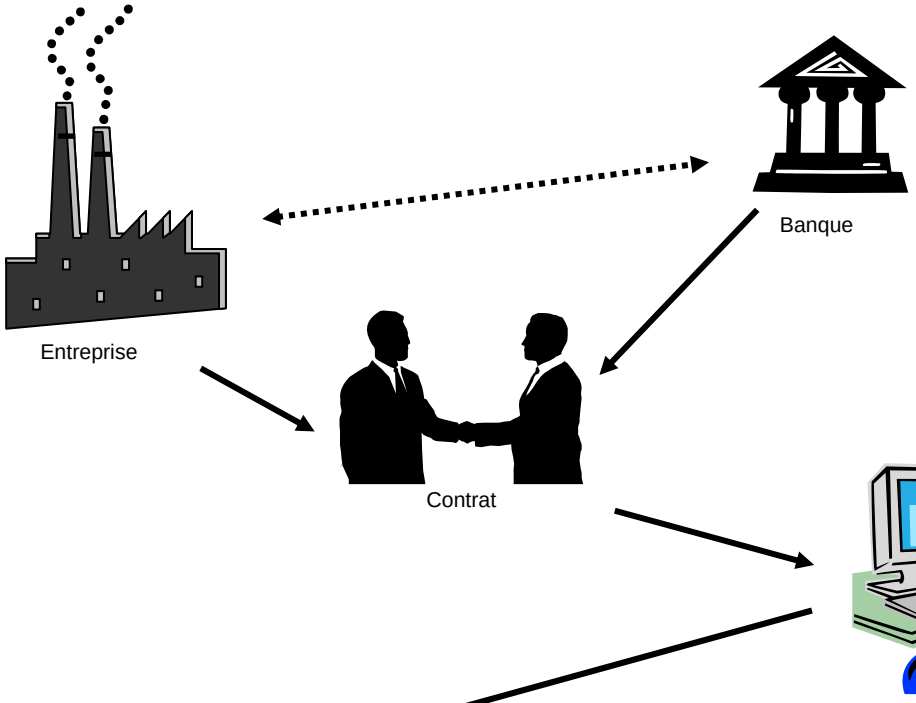
2.1 *Initialisation*

2.1.1 Schéma général de l'initialisation des paramètres sécuritaires

2.1.1.1 Exemple avec EBICS profil T : ordre d'exécution sur canal disjoint et utilisation de certificats auto-générés.



2.1.1.2 Exemple avec EBICS profil TS (ordre d'exécution par signature personnelle jointe) et utilisation de certificats d'AC (Authentification, Chiffrement et Signature).

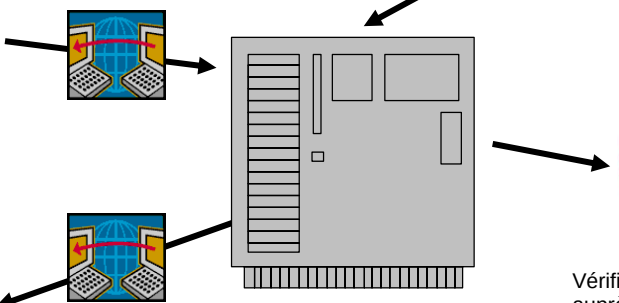


Enregistrement du contrat Client.
Production des Identifiants Client : HostId, UserId, PartnerId.

Enregistrement sur le Host Banque des paramètres Certificat Client (AC, DN, …)

Enregistrement des Identifiants Client et URL Banque.
Chargement des certificats Clients.

Initialisation EBICS
Envoi des 3 certificats : Authentification, Chiffrement et Signature par commandes INI/HIA
Il peut également y avoir une implémentation d'EBICS TS avec des certificats d'authentification et de chiffrement auto-générés et donc avec échange par un autre canal (courrier,..) et un certificat de signature délivré par une AC.



Vérification du certificat auprès de l'AC
DN, Validité, Usage, …

Rapprochement avec le contrat

Téléchargement des Certificats Banques par commande HPB

Enregistrement des Certificats Banque dans la base Banques du poste Client

Tests,

2.1.2 Initialisation des paramètres sécuritaires

2.1.2.1 Les certificats

Les échanges de flux avec les serveurs des banques françaises définis par le CFONB sont basés sur l'utilisation de certificats électroniques X509 permettant la garantie de l'intégrité en EBICS T et l'ordre d'exécution dans le cas d'utilisation d'une signature personnelle jointe en EBICS TS.

Les caractéristiques détaillées de ces certificats (gabarits) sont décrites en annexe A3. La séparation des usages (Authentification, Chiffrement et Signature (scellement en cas de signature disjointe en EBICS T et bancaire en cas de signature personnelle jointe en EBICS TS) est obligatoire, ce seront donc trois clés (une pour chacun des usages) qui seront utilisées par le poste client (et par le serveur de la banque).

Deux personnes physiques ne peuvent pas avoir de certificats en commun en ce qui concerne les certificats d'authentification et de chiffrement et de signature (chaque élément de ce triplet doit être différent).
Il est rappelé que les certificats d'authentification, de chiffrement et de signature doivent être différents les uns des autres pour un UserId.

Nota Bene1 : Dans deux contrats différents, une personne physique peut avoir deux UserId différents utilisant le même triplet.

Nota Bene 2 : En ce qui concerne les rapprochements, il est rappelé que les postes clients doivent être en mesure d'imprimer toutes les lettres d'initialisation (pour chaque certificat de chaque UserId).

2.1.2.2 Caractéristiques des certificats éligibles

Lorsque les certificats sont émis par une AC reconnue de la banque, l'algorithme utilisé par l'Autorité de Certification pour signer ces certificats est :

- Soit algorithme RSA SHA 1 et cela à titre transitoire et non renouvelable pendant une durée de 3 ans maximum (et remplacé par la cible RSA SHA2 256),
- Soit directement avec un algorithme RSA SHA2 256 (cible de tous les certificats émis par une AC à 3 ans).

Dans tous les cas, les messages EBICS sont signés avec un algorithme SHA2. L'algorithme SHA1 n'est autorisé que pour la signature de l'AC.

Les certificats pourront être placés sur différents supports :

Côté poste client :

- Certificat d'authentification :
  - sur support matériel ou logiciel,
  - auto-signé ou généré par une AC reconnue par la banque.

- Certificat de chiffrement :
  - sur support matériel ou logiciel,
  - auto-signé ou généré par une AC reconnue par la banque.

  Les certificats auto-signés doivent être renouvelés à l'issue d'une période de 5 ans au plus. Sauf changement suite à une alerte de sécurité, ils pourront être renouvelés, à l'issue de

cette période soit à nouveau par des certificats auto-signés ou par des certificats d'AC. Les certificats d'AC seront renouvelés suivant les indications de la Politique de Certification de l'AC et communiqués au serveur avec les commandes protocolaires EBICS prévues à cet effet : « PUB » et « HCS » (décrites au chapitre 10 des Spécifications EBICS 2.4.2).

- Certificat de Signature :
  - En EBICS T : Auto-signé ou généré par une AC reconnue par la banque, et sur support matériel ou logiciel, pour le scellement des messages
  - En EBICS TS : Obligatoirement généré par une AC reconnue par la banque et sur support matériel pour la signature personnelle pour EBICS profil TS avec les données à exécuter.

  Le type de support sera déduit de la valeur de l'identifiant de la politique de certification (OID) du certificat.

  Le certificat de signature sera soit à usage exclusif de signature, soit multi-usage mais exclusivement utilisé à des fins de signature. Dans tous les cas, il sera donc différent des certificats authentification et chiffrement.

Côté serveur banque :

Deux certificats auto-signés ou générés par une AC avec un algorithme RSA SHA 256 et longueur de clé 2048 bits. Les certificats auto-signés auront une durée de validité de 5 ans au plus.

2.1.2.3 Initialisation des certificats clients et envoi des clés publiques au serveur banque

Il est nécessaire pour le serveur banque de disposer des trois clés publiques du poste client, une par usage (authentification, chiffrement et scellement).

Chaque clé publique est stockée dans un certificat sur support matériel ou logiciel selon les caractéristiques d'éligibilité (cf. § 2.1.2.2). Chacun des trois certificats correspondant aux trois usages est transmis au serveur dans un fichier d'initialisation via le protocole EBICS, selon le schéma XSD d'EBICS v2.4.

Une procédure de validation est nécessaire pour la banque pour contrôler l'authentification de l'émetteur de chacun de ces trois certificats.

La procédure de rapprochement sera différente selon les cas. On pourra distinguer deux cas :
- le cas d'utilisation d'un certificat auto-signé,
- le cas d'un certificat délivré par une AC.

- **Cas d'utilisation d'un certificat auto-signé :**

Dans le cas d'utilisation d'un certificat auto-signé, le contrôle par la chaîne de certification n'est pas possible. L'authentification doit donc être assurée par un second mécanisme externe au fichier d'initialisation généré sur le poste client.

Ceci est réalisé par l'envoi à la banque, en parallèle de celui du certificat via EBICS, d'un document de confirmation par un autre canal (impression et envoi d'un pdf par courrier, fax, téléchargement). Le choix du canal est hors périmètre de ce guide mais doit être précisé dans le contrat client entreprise/banque.

En France, l'envoi de trois documents (un document par certificat) est obligatoire.

Cette authentification peut être faite par signature manuelle du client sur la version imprimée du fichier d'initialisation (voir annexe A4 – format imprimable du certificat). Ce document comprend le

certificat au format DER, ainsi que les informations d'identification de l'utilisateur (user ID, partner ID, et éventuellement UserName) et du sceau (hash) du certificat dans un format imprimable en vue du rapprochement.

Dans la banque, la validation du certificat s'effectue par le rapprochement de ces données.

Optionnellement, si la banque offre le service, ce fichier de confirmation du certificat peut être transmis par un autre canal électronique et sécurisé qu'EBICS, car il est conçu pour pouvoir être intégré automatiquement sur le serveur de la banque.

Le contrôle de l'authentification par mail simple n'est pas garanti. L'envoi par mail simple de ce fichier **n'est pas recommandé**.

- **Cas d'utilisation d'un certificat délivré par une AC :**

Dans le cas d'utilisation d'un certificat délivré par une AC, le contrôle de la chaîne de certification du certificat permet une automatisation complète du rapprochement suivant les procédures internes de chaque établissement bancaire.

En EBICS T, ce certificat d'AC est considéré comme auto-généré.

En EBICS TS, si l'AC ayant émis le certificat de signature personnelle, n'est pas reconnue par la banque, le certificat est rejeté dans la phase d'initialisation.

2.1.2.4 Récupération sur le poste client des clés publiques des certificats de la banque

Les gabarits de certificats préconisés sont en 2048 bits (pour la longueur de clé RSA) et RSA2048-SHA2 (pour l'algorithme de signature), le démarrage d'EBICS en France s'effectue donc avec ces caractéristiques de certificat.

Actuellement des AC ne sont pas encore compatibles avec cette cible. Il est donc autorisé d'utiliser provisoirement pour les serveurs des certificats auto-signés ou issus d'AC privatives (en RSA 2048 bits et RSA2048-SHA2 pour l'algorithme de signature) et de migrer ensuite vers des certificats émis par des AC (bancaires ou non bancaires) habilitées ayant ces caractéristiques.

Les postes clients devront télécharger les clés publiques du serveur mises à disposition par la banque, par la commande protocolaire HPB (download public Bank Key) puis les contrôler (voir spécifications EBICS 2.4.2 chapitre 4.4.2). La commande HPB **ne doit pas être rendue systématique** lors de chaque échange mais déclenchée uniquement en cas de réception d'une anomalie invitant à renouveler le certificat de la banque.

Dans la commande HPB, la banque doit envoyer les certificats X509 en plus des clés publiques.

Comme pour les clés publiques des postes clients, ce contrôle pourra être fait par l'envoi (en parallèle de celui via EBICS) d'un document de confirmation par un autre canal (ce document peut être le format imprimable du certificat – voir annexe A4).

Le poste client devra être en mesure de permettre une procédure simple de rapprochement entre les clés publiques transmises via EBICS et celles reçues par un autre canal.

2.1.2.5 Vérification automatique des certificats serveurs

Dans le cadre du renouvellement des certificats serveurs, la vérification des certificats peut se faire de façon automatique tel que décrit ci-après.

Lorsqu' un nouveau certificat serveur EBICS est envoyé à un client EBICS (via une requête HPB), le client EBICS devra :

1- vérifier le chemin d'accès de certification du nouveau certificat

2- s'assurer de la véracité du nom d'objet du certificat avec l'URL utilisé pour atteindre le serveur EBICS

3- vérifier que la racine de l'AC qui a délivré le certificat serveur soit différente de celle qui a délivré le certificat SSL. La différenciation de ces AC est obligatoire pour accepter le nouveau certificat serveur.

Si ces 3 étapes sont vérifiées, le client EBICS validera automatiquement le nouveau certificat serveur.

Si l'une de ces étapes n'est pas positive, la vérification visuelle du sceau (hash) entre les certificats reçus via l'ordre HPB et la lettre d'initialisation du certificat reçue par la banque sera utilisée, comme prévu par la procédure manuelle.

2.1.2.6 Les certificats révoqués

Les certificats délivrés par une AC peuvent être révoqués. La recherche de LCR / CRL (Listes de Certificats Révoqués / Certificate Revocation List) ou le contrôle de l'état du certificat en OCSP (Online Certificate Status Protocol) doit se faire pour chaque AC acceptée par la banque, donc uniquement si la chaîne de confiance est enregistrée, et dans les conditions de publication de la CRL par l'AC.

Si le certificat est révoqué, indépendamment de l'utilisation de la commande SPR par l'utilisateur, le serveur met à jour l'état de l'utilisateur à l'état 8-Suspended by user (SPR) qui ne permet plus les échanges. Tous les échanges initiés par l'utilisateur auront pour code erreur 091004 EBICS_INVALID_USER_STATE

2.1.2.7 Messages d'erreurs liés aux certificats

Toute erreur sur certificat est signalée par un message. La liste des messages d'erreurs liés aux certificats figure sur le site du CFONB : http://www.cfonb.org

2.1.3 Signature électronique et chiffrement

2.1.3.1 Principes :

La version EBICS v2.4 supporte les versions de la signature électronique codifiées A004, A005 ou A006. Mais la version A004 de la signature ne sera pas utilisée en France car elle n'est pas compatible avec l'utilisation des certificats. L'utilisation de la signature A006 n'est pas recommandée pour des problèmes de disponibilité actuelle des supports hardware.

Généralement, seulement la version A005 sera utilisée en France.

L'algorithme de chiffrement **AES** est retenu.

2.1.3.2 Calcul de la signature :

- Les caractères systèmes tels que (CR, LF and Ctrl-Z) ne sont pas inclus dans le calcul du hash dans les versions A005 et A006.

- Avec A006, il y a un double calcul de hash[4], la signature n'est pas calculée sur le message lui-même ; donc explicitement Si(Hash(M)) ; mais sur le hash du message ; donc explicitement Si(Hash(Hash(M)).
- La signature s'applique sur le tag <SignedInfo> passé à la canonisation C14N puis la canonisation est hachée.
- Il faut hasher le certificat préalablement encodé au format DER - **sans les caractères (CR, LF CTRL-Z)**
- Les 2 « paddings » suivants peuvent être utilisés :
  "ANSIX923 The ANSIX923 padding string consists of a sequence of bytes filled with zeros before the length.
  ISO10126 The ISO10126 padding string consists of random data before the length. "
  La détection du mode de « padding » se fait en constatant le type de « padding » : des zéros ou des données aléatoires.
- Il convient d'utiliser PKCS1 V1.5 pour chiffrer la clé de chiffrement
- La canonisation doit ajouter les valeurs par défaut. Dans un document XML, seuls les namespaces utilisés dans le document XML doivent être indiqués. Les autres namespaces ne doivent pas être indiqués. Le lien pour la canonisation est :
  http://www.ebics.org/index.php?id=38
- L'XML/DSIG distingue entre documents XML avec ou sans commentaires. Dans EBICS :" *the algorithm for canonicalization is defined via http://www.w3.org/2006/12/xml-c14n11 .*
  This is the algorithm, where the XML-comments are erased.
  The identifier for the algorithm which does not erase the comments would be http://www.w3.org/2006/12/xml-c14n11#WithComments »

Un exemple de calcul de hash est donné dans l'annexe 6.

[4] Voir l'explication des motifs dans la FAQ

2.1.4 Initialisation des identifiants

Chaque banque doit communiquer à ses clients abonnés les informations suivantes nécessaires au paramétrage du poste client.

Le HostID (Identifiant la banque) : il est recommandé aux banques d'utiliser un code BIC sur 11 positions (BIC 8 complété avec XXX éventuellement).

Le UserID, (Nom de l'utilisateur ou du service) : la syntaxe est libre, dans le respect du format spécifié [a-zA-Z0-9,=]{1,35}

Le PartnerID (Numéro de contrat/abonnement) : la syntaxe est libre, dans le respect du format spécifié [a-zA-Z0-9,=]{1,35} }

Les deux derniers identifiants seront communiqués par la banque lors de la signature du contrat.

Il est recommandé de ne pas renseigner le SystemID.

Nota :

Partner ID = Abonnement pour une Société
User ID = Utilisateur dans cet abonnement.

Il y a toujours au moins un utilisateur UserID dans un abonnement. Les certificats sont attachés à l'utilisateur.

En EBICS profil T, dans la majorité des cas, un client (société) n'aura qu'un utilisateur. Néanmoins, certaines entreprises plus complexes peuvent faire le choix d'avoir plusieurs utilisateurs dans le cadre d'un abonnement, par exemple un user ID pour un département Compta, et un autre user ID pour un autre département (Achats par exemple).

Un FileFormats/RequestType donné aura le même nombre de signatures attendues pour les différents UserID du PartnerID considéré.

2.1.5 Gestion de plusieurs utilisateurs chez un même abonné

Cette partie décrit la recommandation pour la gestion de plusieurs utilisateurs (UserId) chez un même abonné (PartnerId).

L'identification des messages doit être unique. Cette identification est réalisée, au niveau du serveur, par construction d'une référence constituée du numéro de transaction (OrderId) avec l'abonné (PartnerId) et du type d'ordre (OrderType). Il est donc indispensable de savoir gérer des numéros de transaction (OrderId) indépendamment de l'utilisateur (UserId).

Lors de l'initialisation des utilisateurs d'un abonné, il est recommandé, pour une bonne gestion des numéros de transaction, d'attribuer des tranches de numéros de transaction (OrderId) par utilisateurs (UserId). Par exemple, la tranche A000 à AZZZ pour le premier utilisateur, B000 à BZZZ pour le second utilisateur, etc…(dans le cas de la gestion de 26 utilisateurs maximum par abonné).

Pour rappel OrderIDtype = [A-Z]{1}[A-Z0-9]{3}

Cette précaution permet d'éviter le risque de doublons OrderId et son rejet par le serveur EBICS de la banque.

2.1.6 Prestataires de services

Ce type de société génère des fichiers (par exemple : paye,..) pour de multiples clients et doit les transmettre aux banques de ces clients.

Dans le cas d'EBICS profil T, le client signe un contrat d'échange avec sa banque, mais il peut déléguer, de manière transparente pour celle-ci ou non, les échanges à un prestataire.

De ce fait, un prestataire qui a plusieurs clients, recevra un PartnerID/UserID de chacun de ses clients. Il devra s'initialiser avec chaque banque pour chaque client.

En EBICS profil T, le prestataire de service prend à sa charge la gestion des certificats d'authentification, de chiffrement et de signature.
En EBICS profil TS (i.e. en cas de signature personnelle jointe), le certificat de signature bancaire est sous la responsabilité du client final et donc sous son contrôle exclusif permanent. Le prestataire fournit un environnement sécurisé au client final sur lequel il pourra lui-même signer chaque fichier avant remise à la banque.

2.1.7 Tests :

Un échange de fichiers de tests est nécessaire avant tout échange de fichiers réels. De ce fait, chaque serveur doit être en mesure de recevoir des flux de tests et de production.

L'obligation sera faite contractuellement au client d'être en mesure d'effectuer des tests et donc celui-ci devra disposer d'un logiciel apte à les faire. De plus il doit être possible d'être en mode test avec une banque et en mode opérationnel avec une autre.

La recommandation suivante a pour objet de préciser les conditions de ces tests.

Le poste client doit disposer d'un paramétrage permettant de basculer du mode Test au mode Production. Son utilisation ne peut être qu'à l'initiative du client en accord avec sa banque.

Compte tenu des risques de confusion entre flux de tests et flux opérationnels, il n'est pas souhaitable que cette distinction résulte d'une intervention manuelle au niveau serveur.

En France, il est recommandé d'utiliser un paramètre complémentaire aux commandes FUL ou FDL pour distinguer le flux de Test ou de Production. La présence du paramètre appelé « TEST » et sa valeur à True signifiera que le flux est en test. L'absence du paramètre équivaudra à un flux de Production.

L'indication « test » doit figurer dans les balises «FULOrderParam » sur le modèle suivant :

```
<FULOrderParams>
      <Parameter>
            <Name>TEST</Name>
            <Value>TRUE</Value>
      </Parameter>
      <FileFormat CountryCode="FR">pain.xxx.cfonb160.dct</FileFormat>
</FULOrderParams>
```

2.2 *Echanges de flux*

Toute transaction EBICS est composée au minimum de deux messages, l'un d'initialisation et l'autre de transfert (en upload ou en download).

Deux Order Type spécifiques seront utilisés pour les échanges de flux : FUL et FDL, qui sont les commandes d'échanges de fichiers.

FUL (Upload) pour l'envoi d'un fichier de remise d'ordres par le client vers sa banque

FDL (Download) pour la récupération par le client d'un fichier de données (par exemple : relevé de comptes) mis à disposition par sa banque.

2.2.1 Paramétrages liés aux flux

Dans le cas d'EBICS profil T, seule la valeur « T (transport) » est préconisée dans le référentiel du serveur pour la classe de signature. Dans ce cas le contrôle de la signature de transport est obligatoire par le serveur (voir paragraphes suivants).

Le poste client doit préciser dans le message, par la valeur D du premier champ de l'OrderAttribute que le flux est en classe de signature transport.

Dans le cas d'EBICS profil TS, le champ devra être à O

Rappel : Pour les autres commandes, la valeur du premier champ de l'OrderAttribute sera conforme aux spécifications EBICS 2 .4.2 (voir tableau page 275 du document des spécifications détaillées).

2.2.2 Traitement des remises d'ordres

Dans le cas de l'OrderType FUL, la classe de signature est associée à un FileFormat,

2.2.2.1 Liste des contrôles en pré-validation

En Upload, dans le premier message, celui d'initialisation, un certain nombre de contrôles dits de pré-validation sont prévus :

- Contrôle des certificats (3 clés publiques) sur la base des informations transmises par le poste client lors de l'initialisation,
- Contrôle de montants (limites),
- Contrôle de comptes

Parmi ces contrôles, celui portant sur les certificats est fortement recommandé pour les serveurs en France.

Les contrôles de montants et de comptes sont optionnels pour les serveurs EBICS en France et laissés au libre choix de chaque établissement bancaire. S'il ne permet pas ces contrôles, le serveur répond par la négative au poste client.

2.2.2.2 Contrôles de sécurité des fichiers de remise d'ordres

Le premier message, celui d'initialisation, contient le hash du fichier de remise d'ordres calculé par le poste client ainsi que le fichier comportant la (ou les) signature(s). A cette étape, le fichier de remise d'ordres n'est pas encore transféré, donc le hash ne peut pas être calculé sur le serveur.

Or ce contrôle en synchrone n'assure pas toute garantie pour la non répudiation du fichier de remise d'ordres puisqu'elle se fonde sur un hash communiqué par le client.

Le protocole EBICS garantit l'intégrité de chaque requête EBICS par la signature électronique d'authentification de l'utilisateur émetteur du message.

La signature est à deux niveaux :

- Tous les fichiers échangés sont signés électroniquement. Le résultat de cette signature est stocké dans une structure communiquée dans le message d'initialisation, avec le hash du fichier calculé sur le poste client. La classe de cette signature est « bancaire » (ordre d'exécution) ou « transport » selon le profil déclaré de l'utilisateur dans le référentiel du serveur.
- Chaque message EBICS est authentifié avec la clé d'authentification de l'utilisateur émetteur du message. La signature utilise un mécanisme de signature XMLDsig. La liste complète des champs du message entrant dans le calcul de cette signature se trouve dans les spécifications. On peut rappeler que les données signées sont généralement sensibles, par exemple le hash et le fichier comportant les signatures.

Le contrôle de signature des fichiers échangés s'appuie sur le hash transmis par l'utilisateur et celui calculé par la banque destinataire. Ce dernier s'effectue sur le fichier reçu par paquets de 1Mo, le calcul pourrait donc commencer, en mode synchrone, avant la fin de la réception du dernier paquet, ou en mode asynchrone après avoir terminé la communication avec l'utilisateur. C'est au serveur de choisir son mode de fonctionnement : asynchrone ou synchrone. Le choix est laissé libre pour l'implémentation.

En synchrone, il est possible de décompresser, déchiffrer et contrôler la signature du fichier avant d'envoyer la trame de réponse au client distant, c'est à dire avant la fin de la transaction EBICS. En effet la dernière trame envoyée par le client EBICS contient le dernier segment de donnée repéré par la balise <SegmentNumber lastSegment= « true »>. Le serveur a donc l'opportunité de faire la vérification de la signature et de renvoyer, le cas échéant, une erreur (EBICS_SIGNATURE_VERIFICATION_FAILED) dans la trame de réponse à la réception de ce dernier segment.

En asynchrone, lorsque la transaction EBICS se termine, le fichier reçu n'a pas été décompressé, déchiffré et contrôlé intègre par sa (ou ses) signature(s). Or, les erreurs éventuelles détectées sur le fichier échangé, doivent être mises à disposition du client EBICS par PSR ou PTK, décrits au paragraphe suivant.

2.2.2.3 Information client sur la réception du fichier de remise d'ordres

Pour les traitements en mode asynchrone, le client doit être informé par le serveur de la bonne réception ou non du fichier et surtout des contrôles ou rejets décrits dans le paragraphe précédent et qui sont principalement la vérification de signature sur le fichier métier.

Les acquittements négatifs doivent obligatoirement être transmis au client. La mise à disposition des acquittements positifs est recommandée et dépend de l'offre de service bancaire. Le poste client doit être en mesure de récupérer les deux types d'acquittement.

En Allemagne, cette fonctionnalité est réalisée, à ce jour, par l'utilisation du Kunden Protokoll (OrderType PTK) décrit au chapitre 10 des Spécifications EBICS 2.4.2.

En France, la cible est l'utilisation du PSR qui est récupéré par la commande FDL (Download) ou tout autre canal. L'utilisation du PTK peut rester une implémentation temporaire à condition de tenir compte de la longueur maximum des identifiants limitée à 8 caractères par rapport au PSR qui admet des longueurs d'identifiants sur 35 caractères.

Utilisation du Payment Status Report :

- Le Payment Status Report (Format ISO 20022 XML – pain.002.001.xx, disponible sur le site ISO www.iso20022.org) doit être fait au niveau fichier. De ce fait, les items de niveau 3.0 et au delà ne seront pas valorisés, seuls les items de niveau GroupHeader et OriginalGroupInformationAndStatus seront renseignés (voir annexe A5 – Format du Payment Status report).
- A défaut, la mise à disposition ou l'envoi d'un Payment Status Report peut s'effectuer par un autre canal (mail, portail internet) en utilisant une feuille de style afin de le rendre lisible.
- En annexe 5 **Format du Payment Status Report** figure la description des données permettant de rapprocher un PSR avec la remise d'ordres correspondante.
- La génération d'un Payment Status Report est obligatoire pour tout échec de transaction. Il n'est pas obligatoire de créer un PSR par fichier transmis en échec, mais dans le cas d'un envoi multiple, le serveur peut effectuer, à la fin de cet envoi une concaténation technique de plusieurs fichiers XML de compte rendu (présence de plusieurs header XML dans le même fichier) dans un seul PSR. Dans ce cas, le PSR ne pourra pas être déparsé directement par le poste client. Il devra auparavant être découpé en fichiers XML unitaires avant de parser chaque compte rendu.

Le PSR ne porte que sur la commande FUL (UPLOAD) et ne doit pas être généré pour les autres commandes (INI, HIA ,..).

2.2.3 Récupération des fichiers clients : la commande Download (FDL)

L'utilisation de la commande de Download (FDL) sans paramétrage de date permet de récupérer l'ensemble des fichiers disponibles (non encore récupérés). Sur cette commande, le serveur met à disposition tous les fichiers en stock, ceux de même type sont concaténés avant d'être compressés et chiffrés pour l'échange, ne formant plus ainsi qu'un seul fichier.

Après envoi, les fichiers sont systématiquement archivés sur le serveur. Ils peuvent alors être à nouveau récupérés en utilisant la commande de Download (FDL) avec un paramètre correspondant à une date ou un intervalle de dates en fonction de l'offre de service bancaire.

L'intervalle de dates possible dépend de la profondeur d'archivage mis en place sur le serveur. Chaque établissement bancaire peut fixer une durée d'historisation pour chaque type de fichier et doit le faire figurer dans le contrat d'abonnement.

La remise à disposition de fichiers au-delà de la période d'historique en ligne n'est pas du domaine protocolaire.

Remarques :

- Certaines banques peuvent proposer une offre de services spécifique de mise à disposition d'information (fréquence de relevés,…).
  La méthode décrite pour nommer les fichiers permet de couvrir les besoins relatifs à ces services spécifiques. Les banques pourront utiliser des extensions propriétaires dans le nom du fichier utilisé dans le FileFormat. Ce type de paramétrage spécifique à un établissement doit être indiqué dans les annexes du contrat d'abonnement.
- La commande FDL ne comporte qu'un seul paramètre de type FILE FORMAT. Donc seuls des fichiers de même type seront à concaténer. Le client EBICS devra gérer ce type de cas.
- L'utilisation de la commande de Download (FDL) sans paramétrage de date permet de récupérer l'ensemble des fichiers disponibles (non encore récupérés).

3 MODALITES D'UTILISATION DU POSTE CLIENT

La mise en œuvre de la connexion d'un poste client avec une banque requiert au préalable la signature d'un contrat entre le client et cette banque. Ce contrat doit comporter un document indiquant les données spécifiques à cet abonnement.

Il y a donc un contrat à signer et une initialisation à réaliser pour chaque banque.

Il est du ressort des éditeurs de logiciels de mettre à disposition les moyens nécessaires aux clients pour sécuriser l'accès au poste client et aux données sensibles.

Le client devra disposer des outils permettant de sécuriser sa connexion Internet (firewall, anti-virus,..) et les tenir à jour.

Lorsque le poste client stocke les certificats dans une base de données, celle-ci devra offrir un niveau de sécurité suffisant (chiffrement des données) et équivalent à celui du magasin Windows ou Java Key Store.

Chaque poste client et chaque serveur doivent avoir des Log permettant de faire le suivi des échanges (horodatage, montant, ….). Il doit être possible de rechercher la log de bout en bout de chaque transaction et de pouvoir la transmettre à la banque (mail,..).

Il est recommandé que les interfaces homme-machine et les éditions soient en Français.

Afin de pouvoir servir de preuve éventuelle en cas de conflit, le client doit conserver les fichiers signés tels que remis à la banque, ainsi que toutes les autres données nécessaires.

4 ANNEXES

A1 - Order Type

Les spécifications 2.4 d'EBICS s'appliquent en France et en Allemagne mais les commandes (OrderType) couplées à un format de fichier spécifique ne seront pas utilisées en France.

Pour cette fonctionnalité, en France on utilise l'OrderType FUL associé au FileFormat spécifique.

Les commandes du protocole sont classées en deux catégories :
- Les « system orders » liés à la gestion du référentiel EBICS ou au protocole lui-même
- Les « bank-technical orders » liés à un format

La liste des commandes supportées par les serveurs français est un sous-ensemble des commandes EBICS et sont plutôt considérées comme des system orders, à l'exception des commandes « FUL » et « FDL », qui sont les seules commandes d'upload et de download de fichiers autorisées en France.

La liste ci-dessous constitue la liste exhaustive des OrderTypes nécessaires en France.
En dehors de cette liste, les autres commandes sont optionnelles.
L'implémentation de la commande HTD (Download subscriber's customer and subscriber Data) est recommandée.

Un serveur qui reçoit une commande qu'il ne peut pas traiter doit répondre le code retour prévu : EBICS_UNSUPPORTED_ORDER_TYPE

Identification	Nom	Format
HCA	Send amendment of the subscriber key for identification and authentication and encryption	
HCS	Transmission of the subscriber key for ES, identification and authentication and encryption	
HIA	Transmission of the subscriber key for identification and authentication and encryption within the framework of subscriber initialisation	
HPB	Transfer the public bank key (download)	
HPD	Download bank parameters	
INI	Send password initialisation	Customer's public key for the ES
HEV	Download supported EBICS versions	
PUB	Send public key for signature verification	Customer's public key for the ES (see Appendix Chapter 15)
SPR	Suspension of access authorisation	Transmission of an ES file with a signature for a dummy file that only contains a space
FUL	File Upload	Upload de fichiers dont le type est en paramètre
FDL	File Download	Download de fichiers dont le type est en paramètre
PTK	Kunden Protokoll	

A2 - FileFormat/Request Type – Nommage des fichiers

Avertissement :
le contenu de cette annexe dénommée EBICS SWIFNet - Nommage Fichiers
est à consulter sur le site du CFONB : www.cfonb.org
dans la rubrique Documentation : Migration ETEBAC vers EBICS et SWIFTNet

A3 - Certificats

Messages d'erreurs liées aux certificats

Avertissement :
le contenu de cette partie est à consulter sur le site du CFONB : www.cfonb.org
dans la rubrique Documentation : Migration ETEBAC vers EBICS et SWIFTNet

Gabarit certificats porteurs EBICS

Un certificat porteur pour EBICS peut être auto-signé ou importé sur le poste client s'il est délivré par une AC privée. Trois usages sont définis pour EBICS et trois certificats sont requis.

CAS de certificats auto-signés :

Certificat dédié à la signature (EBICS T uniquement)

Champ X509	Valeur	Obligatoire
Version	=2	oui
serialNumber	nombre aléatoire sur 35 octets max si auto signé.	oui
Signature Algorithm	RSA-SHA2 (256)	oui
Issuer	=subject	oui
Validity	durée de validité : 5 ans	oui
subject (objet ou DN)	L'attribut utilisé est le « commonname »	oui
subjectPublicKeyInfo	clé RSA de longueur 2048 bits - rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier	=SubjectKeyIdentifier de l'AC ou du présent certificat	oui
KeyUsage	NonRepudiation	oui
ExtendedKeyUsage		non
CRLDistributionPoints		non

certificat dédié à l'authentification (EBICS T ou TS)

Champ X509	Valeur	Obligatoire
Version	=2	oui
serialNumber	nombre aléatoire sur 35 octets max si auto signé	oui
Signature Algorithm	RSA-SHA2 (256)	oui
Issuer	=subject	oui
Validity	durée de validité : 5 ans 5	oui
subject (objet ou DN)	L'attribut utilisé est le « commonname »	oui
subjectPublicKeyInfo	Clé RSA de longueur 2048 bits - rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier	=SubjectKeyIdentifier de l'AC ou du présent certificat	oui
KeyUsage	DigitalSignature	oui
ExtendedKeyUsage		non
CRLDistributionPoints		non

certificat dédié au chiffrement (EBICS T ou TS)

Champ X509	Valeur	Obligatoire
Version	=2	oui
serialNumber	nombre aléatoire sur 35 octets max si auto signé	oui
Signature Algorithm	RSA-SHA2 (256)	oui
Issuer	=subject	oui
Validity	durée de validité : 5 ans [5]	oui
subject (objet ou DN)	L'attribut utilisé est le « commonname »	oui
subjectPublicKeyInfo	Clé RSA de longueur 2048 bits - rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier	=SubjectKeyIdentifier de l'AC ou du présent certificat	oui
KeyUsage	keyEncipherment ou keyAgreement	oui
ExtendedKeyUsage		non
CRLDistributionPoints		

[5] Cette durée de validité est valable uniquement pour les certificats autosignés. Dans le cas de certificats d'AC la durée de validité sera celle de la politique de certification correspondante au certificat.

CAS de certificats d'AC :

Chaque banque détermine les certificats, conformes aux gabarits ci-après, qu'elle accepte au niveau signature personnelle.

Certificat de signature d'AC (obligatoire sur support matériel pour le profil TS)

Champ X509	Valeur	Obligatoire
version	=2 (pour X509V3)	oui
serialNumber	Unique pour l'AC longueur max 20 octets	oui
Signature Algorithm	RSA-SHA2 (256) ou SHA1 (160) en phase transitoire pendant 3 ans.	oui
issuer	=DN de l'AC	oui
validity	durée de validité : 3 ans	oui
subject (objet ou DN)	Identifiant le porteur. Le DN doit comporter le « CommonName »	oui
subjectPublicKeyInfo	clé RSA de longueur 2048 bits - rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier	=SubjectKeyIdentifier de l'AC	oui
SubjectKeyIdentifier		oui
KeyUsage	Le bit NonRepudiation ou ContentCommitment=1	oui
ExtendedKeyUsage	id-kp-emailProtection	non
Subject Alternative Name	(Peut contenir l'email) Attention à la criticité	Non, mais Non Critique si présent
Issuer Alternative Name	Attention à la criticité	Non, mais Non Critique si présent
CRLDistributionPoints	Completer éventuellement par AuthorityInformation access pour l'OCSP	Oui
Freshest CRL	Si utilisation de DeltaCRL	Oui si DeltaCRL Non Critique
Authority Information Access	Si service OCSP.	Oui si OCSP Non Critique
QCStatement	Si certificat qualifié, contient OID indiquant que le certificat est qualifié et que la clé privé correspondante est stockée dans un SSCD.	Oui si Certificat Qualifié

Certificat d'authentification d'AC sur support matériel ou logiciel

Champ X509	Valeur	Obligatoire
version	=2 (pour X509V3)	oui
serialNumber	Unique pour l'AC longueur max 20 octets	oui
Signature Algorithm	RSA-SHA2 (256) ou SHA1 (160) en phase transitoire pendant 3 ans.	oui
issuer	=DN de l'AC	oui
validity	durée de validité : 3 ans	oui
subject (objet ou DN)	Identifiant le porteur. Le DN doit comporter le « CommonName »	oui
subjectPublicKeyInfo	clé RSA de longueur 2048 bits - rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier	=SubjectKeyIdentifier de l'AC	oui
SubjectKeyIdentifier		oui
KeyUsage	Le bit DigitalSignature=1	oui
ExtendedKeyUsage	id-kp-clientAuth	non
Subject Alternative Name	(Peut contenir l'email) Attention à la criticité	Non, mais Non Critique si présent
Issuer Alternative Name	Attention à la criticité	Non, mais Non Critique si présent
CRLDistributionPoints	Completer éventuellement par AuthorityInformation access pour l'OCSP	Oui
Freshest CRL	Si utilisation de DeltaCRL	Oui si DeltaCRL Non Critique
Authority Information Access	Si service OCSP.	Oui si OCSP Non Critique

Certificat de chiffrement d'AC sur support matériel ou logiciel

Champ X509	Valeur	Obligatoire
version	=2 (pour X509V3)	oui
serialNumber	Unique pour l'AC longueur max 20 octets	oui
Signature Algorithm	RSA-SHA2 (256) ou SHA1 (160) en phase transitoire pendant 3 ans.	oui
issuer	=DN de l'AC	oui
validity	durée de validité : 3 ans	oui
subject (objet ou DN)	Identifiant le porteur. Le DN doit comporter le « CommonName »	oui
subjectPublicKeyInfo	clé RSA de longueur 2048 bits - rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier	=SubjectKeyIdentifier de l'AC	oui
SubjectKeyIdentifier		oui
KeyUsage	Le bit KeyEncipherment=1	oui
ExtendedKeyUsage	id-kp-emailProtection	non
Subject Alternative Name	(Peut contenir l'email) Attention à la criticité	Non, mais Non Critique si présent
Issuer Alternative Name	Attention à la criticité	Non, mais Non Critique si présent
CRLDistributionPoints	Completer éventuellement par AuthorityInformation access pour l'OCSP	Oui
Freshest CRL	Si utilisation de DeltaCRL	Oui si DeltaCRL Non Critique
Authority Information Access	Si service OCSP.	Oui si OCSP Non Critique

Gabarit certificats serveurs EBICS

Il est nécessaire de disposer d'un certificat par usage, soit dans la version actuelle 2.4.2 : 2 certificats par serveur banque (authentification et chiffrement).

Les deux certificats serveurs sont assimilés à des certificats SSL TLS et donc doivent avoir à la fois le KeyUsage de DigitalSignature et de KeyEncipherment.

Le certificat de signature n'est pas prévu dans la version EBICS 2.4.2 qui couvre le remplacement de ETEBAC 3 et ETEBAC 5.

Certificat serveur dédié à l'authentification

Champ X509	Valeur	Obligatoire
Version	=2	oui
serialNumber		oui
Signature Algorithm	RSA-SHA2 (256)	oui
Issuer		oui
validity	durée de validité : 5 ans	oui
subject (objet ou DN)	L'attribut utilisé est le « commonname »	oui
subjectPublicKeyInfo	clé RSA de longueur 2048 bits – rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier		oui
KeyUsage	DigitalSignature ;keyEncipherment	oui
CertificatePolicies		oui
CRLDistributionPoints		oui
FreshestCRL		non
ExtendedKeyUsage		non

certificat serveur dédié au chiffrement

Champ X509	Valeur	Obligatoire
Version	=2	oui
serialNumber		oui
Signature Algorithm	RSA-SHA2 (256)	oui
Issuer		oui
validity	durée de validité : 5 ans[6]	oui
subject (objet ou DN)	L'attribut utilisé est le « commonname »	oui
subjectPublicKeyInfo	clé RSA de longueur 2048 bits – rsaEncryption	oui
extensions :		
AuthorityKeyIdentifier		oui
KeyUsage	DigitalSignature ;keyEncipherment	oui
CertificatePolicies		oui
CRLDistributionPoints		oui
FreshestCRL		non
ExtendedKeyUsage		non

[6] Cette durée de validité est valable uniquement pour les certificats autosignés. Dans le cas de certificats d'AC la durée de validité sera celle de la politique de certification correspondante au certificat.

A4 - Format imprimable du certificat

En France, l'envoi de trois documents : un document par certificat, est obligatoire.

Chaque certificat est imprimé au format PEM.

Le hash imprimé est celui du certificat construit selon la norme X509 avec l'algorithme SHA2 (256). L'impression du hash est en hexadécimal et en majuscules.

Les lettres suivantes ne sont données qu'à titre d'exemple de présentation. Pour trouver un exemple de calcul de hash se reporter à l'annexe 6.

Lorsque le certificat a été délivré par une autorité de certification, pour permettre de faire le rapprochement, il est recommandé de faire apparaitre les données du certificat suivantes sur les lettres d'initialisation ou dans le contrat en cas d'utilisation d'un processus de rapprochement automatique :

« Certificat délivré à : NOM–PRENOM / ID » (Objet - Champ CN)

« Certificat délivré par : NOM DE L'AUTORITE DE CERTIFICATION » (Emetteur - Champ CN)

Lettre d'initialisation du certificat de signature

Date : TT.MM.JJJJ
Time : HH:MM:SS
Host Id : "BIC11 de la banque"
Banque : « nom de la banque »
User-ID : Xxxxxxxx
Partner-ID : Yyyyyyyy
Version : Signature A005

Certificat de signature électronique
Type : A005

(Si le certificat a été généré par une AC :

Certificat délivré à : NOM–PRENOM ou identifiant

Certificat délivré par : AUTORITE DE CERTIFICATION)

-----BEGIN CERTIFICATE-----
MIICcjCCAdugAwIBAgIBDzANBgkqhkiG9w0BAQQFADA6MQswCQYDVQQGEwJGUjEY
MBYGA1UECxMPYmFucXVlcG9wdWxhaOdrLaNGZtYDVQQDEwggdHVyYm9zYTAeFw0w
ODA5MTAxMzI0MzZaFw0xODA5MDgxMzI0MzZaMDQxCzAJBgNVBAYTAmZyMRgwFgYD
VQQLEw9iYW5xdWVwb3B1bGFpcmUxCzAJBgNVBAMTAmpmMIGfMA0GCSqGSIb3DQEB
AQUAA4GNADCBiQKBgQCkscideEsfU0+UPqM13kPUQVBFYB4xOcHCYqzr6PPgl7Co
GwsjK5o4CKUm/7qWS0BdnqNOdrLaNGZ4kCXlXDg1SemWMlOgPtWl9T3XAiyyr88L
Ei+9sisIUA/JE/3leQWuk0gJXohtxKUwR/fbsWrQjqLspxNK09Urbqz8hwehPQID
AQABo4GNMIGKMA4GA1UdDwEB/wQEAwIE8DA4BgNVHR8EMTAvMC2gK6AphidodHRw
Oi8vODYuNjQuMTAuMTM4LytDU0NPQ0ErL2FzYV9jYS5jcmwwHwYDVR0jBBgwFoAU
zM7nNDE4VQkAUz33C9ztXhG9P3gwHQYDVR0OBBYEFNd6cAJ8L04eB7TiCzpcumIn
gFSsMA0GCSqGSIb3DQEBBAUAA4GBAEm2OLlVyMIzf7Bk7ZUNBCQacvUEdl2o58Pg
py+CMN+K1OdrLaNGZ77TIVKbydqwl2t7hIpuC81c8D3O9r3LiYSDrgxMFhxeUKLD
sIo1dusXjV8nHm5V2zu4hOdrLaNGZix3bEEEFH+cpzOp5y/ogwHWVpz6h3r36Lgo
Vl1S6JU6
-----END CERTIFICATE-----

Hash du certificat de signature (SHA-256):

B8 3C B0 19 66 C9 9C 6E
2C A5 BA 6A 2B 56 01 92
35 2A B4 91 53 E9 0B BA
34 C1 5E B5 9F 4A 64 F7

Date : **Signature :**

Lettre d’initialisation du certificat d’authentification

Date : TT.MM.JJJJ
Time : HH :MM :SS
Host Id : “BIC11 de la banque”
Banque : « nom de la banque »
User-ID : Xxxxxxxx
Partner-ID : Yyyyyyyy
Version : Authentification X002

Certificat d’authentification
Type : X002

(Si le certificat a été généré par une AC :

Certificat délivré à : NOM–PRENOM ou identifiant

Certificat délivré par : AUTORITE DE CERTIFICATION

-----BEGIN CERTIFICATE-----
MIICcjCCAdugAwIBAgIBDzANBgkqhkiG9w0BAQQFADA6MQswCQYDVQQGEwJGUjEY
zM7nNDE4VQkAUz33C9ztXhG9P3gwHQYDVR0OBBYEFNd6cAJ8L04eB7TiCzpcumIn
MBYGA1UECxMPYmFucXVlcG9wdWxhaOdrLaNGZtYDVQQDEwggdHVyYm9zYTAeFw0w
ODA5MTAxMzI0MzZaFw0xODA5MDgxMzI0MzZaMDQxCzAJBgNVBAYTAmZyMRgwFgYD
VQQLEw9iYW5xdWVwb3B1bGFpcmUxCzAJBgNVBAMTAmpmMIGfMA0GCSqGSIb3DQEB
AQUAA4GNADCBiQKBgQCkscideEsfU0+UPqM13kPUQVBFYB4xOcHCYqzr6PPgl7Co
GwsjK5o4CKUm/7qWS0BdnqNOdrLaNGZ4kCXlXDg1SemWMlOgPtWl9T3XAiyyr88L
Ei+9sisIUA/JE/3leQWuk0gJXohtxKUwR/fbsWrQjqLspxNK09Urbqz8hwehPQID
AQABo4GNMIGKMA4GA1UdDwEB/wQEAwIE8DA4BgNVHR8EMTAvMC2gK6AphidodHRw
Oi8vODYuNjQuMTAuMTM4LytDU0NPQ0ErL2FzYV9jYS5jcmwwHwYDVR0jBBgwFoAU
gFSsMA0GCSqGSIb3DQEBBAUAA4GBAEm2OLlVyMIzf7Bk7ZUNBCQacvUEdl2o58Pg
py+CMN+K1OdrLaNGZ77TIVKbydqwI2t7hIpuC81c8D3O9r3LiYSDrgxMFhxeUKLD
sIo1dusXjV8nHm5V2zu4hOdrLaNGZix3bEEEFH+cpzOp5y/ogwHWVpz6h3r36Lgo
Vl1S6JU6
-----END CERTIFICATE-----

Hash du certificat d’authentification (SHA-256) :

2C A5 BA 6A 2B 56 01 92
35 2A B4 91 53 E9 0B BA
B8 3C B0 19 66 C9 9C 6E
34 C1 5E B5 9F 4A 64 F7

Date : **Signature :**

Lettre d'initialisation du certificat de chiffrement

Date : TT.MM.JJJJ
Time : HH:MM:SS
Host Id : "BIC11 de la banque"
Banque : « nom de la banque »
User-ID : Xxxxxxxx
Partner-ID : Yyyyyyyy
Version : Chiffrement E002

Certificat de chiffrement
Type : E002

(Si le certificat a été généré par une AC :

Certificat délivré à : NOM–PRENOM ou identifiant

Certificat délivré par : AUTORITE DE CERTIFICATION

-----BEGIN CERTIFICATE-----
MIICcjCCAdugAwIBAgIBDzANBgkqhkiG9w0BAQQFADA6MQswCQYDVQQGEwJGUjEY
zM7nNDE4VQkAUz33C9ztXhG9P3gwHQYDVR0OBBYEFNd6cAJ8L04eB7TiCzpcumIn
MBYGA1UECxMPYmFucXVlcG9wdWxhaOdrLaNGZtYDVQQDEwggdHVyYm9zYTAeFw0w
ODA5MTAxMzI0MzZaFw0xODA5MDgxMzI0MzZaMDQxCzAJBgNVBAYTAmZyMRgwFgYD
VQQLEw9iYW5xdWVwb3B1bGFpcmUxCzAJBgNVBAMTAmpmMIGfMA0GCSqGSIb3DQEB
AQUAA4GNADCBiQKBgQCkscideEsfU0+UPqM13kPUQVBFYB4xOcHCYqzr6PPgl7Co
GwsjK5o4CKUm/7qWS0BdnqNOdrLaNGZ4kCXlXDg1SemWMlOgPtWl9T3XAiyyr88L
Ei+9sisIUA/JE/3leQWuk0gJXohtxKUwR/fbsWrQjqLspxNK09Urbqz8hwehPQID
AQABo4GNMIGKMA4GA1UdDwEB/wQEAwIE8DA4BgNVHR8EMTAvMC2gK6AphidodHRw
Oi8vODYuNjQuMTAuMTM4LytDU0NPQ0ErL2FzYV9jYS5jcmwwHwYDVR0jBBgwFoAU
gFSsMA0GCSqGSIb3DQEBBAUAA4GBAEm2OLlVyMIzf7Bk7ZUNBCQacvUEdl2o58Pg
py+CMN+K1OdrLaNGZ77TIVKbydqwI2t7hIpuC81c8D3O9r3LiYSDrgxMFhxeUKLD
sIo1dusXjV8nHm5V2zu4hOdrLaNGZix3bEEEFH+cpzOp5y/ogwHWVpz6h3r36Lgo
Vl1S6JU6
-----END CERTIFICATE-----

Hash du certificat de chiffrement (SHA-256) :

2C A5 BA 6A 2B 56 01 92
35 2A B4 91 53 E9 0B BA
B8 3C B0 19 66 C9 9C 6E
34 C1 5E B5 9F 4A 64 F7

Date : **Signature :**

A5 - Format du Payment Status Report

Avertissement :
le contenu de cette annexe dénommée « EBICS Payment Status Report » est à consulter sur le site du CFONB : www.cfonb.org
dans la rubrique Documentation : Migration ETEBAC vers EBICS et SWIFTNet

A6 - Exemple de calcul de Hash

Le calcul de hash est mis en œuvre tant sur les certificats porteur que les certificats serveur.

Le hash peut être contrôlé par la commande « openssl » suivante :
« `openssl x509 -sha256 -fingerprint -in cert.pem`

```
results into this output:
SHA256 Fingerprint =
A6:16:4F:86:65:AF:84:D5:84:AB:70:51:19:37:2F:4D:61:36:AE:69:C2:6A:F6:AF:3
1:79:CD:01:37:3C:D4:81″
```


cert.pem


ExempleSignatureTransport.doc

A7 - Glossaire

Sigle	Définition
AC	Autorité de Certification
API	Application Program Interface (interface de programmation)
Authentification	Procédure permettant de vérifier l'identité d'une entité (personne ou système)
Banque	Prestataire de Services de paiement (PSP) au sens de la directive 2007/064/CE sur les services de paiements du 17 novembre 2007, transposée dans l'ordonnance 2009-866 du 15 juillet 2009.
CFONB	Comité Français d'Organisation et de Normalisation Bancaires
Certificat	Standard permettant de stocker une clé publique
Certificat auto-signé	Certificat généré par l'émetteur du message
Chaine de certification / chaine de confiance (	Vérification de la certification en remontant jusqu'à l'autorité racine (reconnue comme autorité de confiance).
Chiffrement	Processus de transformation des données à l'aide d'un algorithme cryptographique
DER	Standard de présentation de certificat (Distinguished Encoding Rules)
EBICS	Electronic Banking Internet Communication Standard
ETEBAC	Echanges Télématiques Entre Banques et Clients
Fichier d'ordres Remise d'ordres	Fichier contenant les instructions bancaires
FileFormat	Généralement pour la France il s'agit de la nature de la transaction bancaire dans les commandes FUL et FDL
Hash, empreinte sceau Sceau du certificat	Valeur numérique associée à un message pour s'assurer de son intégrité
IP	Internet Protocol
Order Type :	Nature de la transaction EBICS incluant pour l'Allemagne la nature des transactions bancaires
Ordre d'exécution	Voir signature de l'ordre
PEM	Privacy Enhanced Mail. Le format PEM est du DER encodé en base64 auquel sont ajoutées des en-têtes en ASCII. Il peut contenir des clés privées, des clés publiques et des certificats X509.
Request Type :	Nature de la demande d'information
Scellement	Fonction mathématique permettant d'obtenir le sceau
Signature de transport	Signature permettant de s'assurer de l'origine et de l'intégrité des données d'un message. Elle n'a pas de valeur personnelle
Signature de l'ordre /Ordre d'exécution /Confirmation	La signature de l'ordre est également appelée ordre d'exécution ou signature personnelle électronique. Elle permet d'authentifier personnellement l'émetteur d'un message
Signature disjointe	Les instructions et la signature de l'ordre ne sont pas transmises dans le même flux mais de façon asynchrone. Elles peuvent être transmises par le même canal ou par canal différent ainsi qu'au même moment ou décalées dans le temps.
SWIFT	Society for Worldwide Interbank Financial Telecommunication
TLS	Transport Layer Security
VEU :	Verteilte Elektronische Unterschrift (Signature Electronique Disjointe)
X25	Protocole de communication par commutation de paquets, commercialisé par France Telecom, utilisé par les protocoles

	ETEBAC
ZKA	Zentraler KreditAusschuss Nouveau nom en 2011 : DK =“Die Deutsche Kreditwirtschaft“