Connect:Express® UNIX

Option SSL

Version 1.4.6

Sterling Commerce
An IBM Company

Connect:Express Unix
Option SSL

Version 1.4.6
Première édition

La présente documentation a pour objet d'aider les utilisateurs autorisés du système Connect:Express (ci-après le « Logiciel de Sterling Commerce »). Le Logiciel de Sterling Commerce, la documentation correspondante ainsi que les informations et le savoir-faire qu'il contient, sont la propriété de Sterling Commerce Inc. et sont confidentiels. Ils constituent des secrets commerciaux de cette dernière, de ses sociétés affiliées ou de ses/leurs concédants (ci-après dénommés collectivement « Sterling Commerce »). Ils ne peuvent pas être utilisés à des fins non autorisées ni divulgués à des tiers sans l'accord écrit préalable de Sterling Commerce. Le Logiciel de Sterling Commerce ainsi que les informations et le savoir-faire qu'il contient ont été fournis conformément à un contrat de licence qui inclut des interdictions et/ou des limitations quant à la copie, la modification et l'utilisation. La reproduction, en tout ou partie, si et lorsqu'elle est autorisée, devra inclure la présente notice d'information et la légende de copyright de Sterling Commerce Inc. Lorsqu'un Logiciel de Sterling Commerce ou un Logiciel Tiers est utilisé, reproduit ou divulgué par ou à une administration des Etats-Unis ou un cocontractant ou sous-traitant d'une telle administration, le Logiciel est assorti de DROITS LIMITES tels que définis au Titre 48 CFR 52.227-19 et est régi par les dispositions suivantes : Titre 48 CFR 2.101, 12.212, 52.227-19, 227-7201 à 227.7202-4, FAR 52.227-14 (g) (2) (6/87) et FAR 52.227-19 (c) (2) et (6/87), et le cas échéant, la licence habituelle de Sterling Commerce, tel que cela est décrit au Titre 48 CFR 227-7202-3 concernant les logiciels commerciaux et la documentation des logiciels commerciaux, y compris le DFAR 252-227-7013 (c) (1), 252.227-7015 (b) et (2), DFAR 252.227-7015 (b) (6/95), DFAR 227.7202-3 (a), selon le cas.
Le Logiciel de Sterling Commerce et la documentation correspondante sont concédés « EN L'ETAT » ou assortis d'une garantie limitée, telle que décrite dans le contrat de licence de Sterling Commerce. A l'exception des garanties limitées accordées, AUCUNE AUTRE GARANTIE EXPRESSE OU IMPLICITE N'EST CONCEDEE, Y COMPRIS LES GARANTIES DE QUALITE MARCHANDE ET DE CONVENANCE A UN USAGE PARTICULIER. La société Sterling Commerce concernée se réserve le droit de revoir cette publication périodiquement et d'effectuer des modifications quant à son contenu, sans obligation d'en informer qui que ce soit, personne physique ou personne morale.
Les références faites dans le présent manuel aux produits, logiciels ou services Sterling Commerce ne signifient pas que Sterling Commerce a l'intention de les commercialiser dans tous les pays dans lesquels elle a des activités.

TABLE DES MATIERES

Préface

Ce guide décrit l’utilisation de l’option SSL de Connect:Express afin de sécuriser des transferts de fichiers.

Chapitre 1
Présentation de l'option SSL

Ce chapitre présente les divers éléments livrés avec l'option SSL de Connect:Express.

Installation

Les éléments logiciels de l'option SSL de Connect:Express sont inclus dans la livraison standard du produit et ne nécessitent par conséquent pas d'installation supplémentaire.

L'activation ou pas de l'option SSL se fait par la clé d'autorisation de Connect:Express.

Le tableau ci-dessous résume les éléments logiciels spécifiques de l'option SSL :

Répertoire	Description
config/CERT.dat	Base de données des certificats importés
config/CERT.idx	Base de données des certificats importés
config/SSLPARM.dat	Base de données des paramètres SSL de transferts
config/SSLPARM.idx	Base de données des paramètres SSL de transferts
config/RDN.dat	Base de données des paramètres de contrôle des certificats
config/RDN.idx	Base de données des paramètres de contrôle des certificats
config/sslerr	Utilitaire d'affichage de libellés d'erreurs
config/ssl/cert_import	Répertoire d'importation des certificats
config/ssl/lcert	Base de données des certificats importés
config/ssl/priv	Répertoire d'importation des clés privées
config/ssl/dhparam	Répertoire de fichiers de paramètres Diffie-Hellman
config/ssl/ciphlist	Répertoire de listes de suites de chiffrement
config/ssl/calist	Répertoire de listes d'autorités de certification
config/ssl/openssl/bin	Commande utilitaire openssl de OpenSSL
config/ssl/openssl/man	Pages man de la commande openssl
config/ssl/.rnd	Fichier de données aléatoires

Protocoles de transferts de fichiers supportés

L'option SSL permet d'effectuer des transferts sécurisés en PeSIT et Etebac3 en s'appuyant sur les protocoles TLS v1.0, SSL v3.0 ou SSL v2.0.

Remarque importante : L'option SSL ne s'applique pas aux transferts FTP de Connect:Express.

Réseaux supportés

Seul le réseau TCP/IP est supporté.

Systèmes d’exploitation supportés

L’option SSL est disponible sur toutes les plates formes UNIX supportant Connect:Express (AIX, Solaris, HP-UX, Tru64, Linux, z/Linux …).

L’option SSL de Connect:Express s’appuie en interne sur l’API de OpenSSL qui est disponible sur la majorité des systèmes UNIX.

La version du système d’exploitation doit être suffisamment récente pour offrir les sources de bits aléatoires /dev/random et /dev/urandom. Ces devices sont disponibles sur les dernières versions des systèmes AIX, Solaris, Linux.

Il est néanmoins possible de faire fonctionner l’option SSL de Connect:Express en prenant comme source de bits aléatoires un fichier statique, en attendant une montée de niveau du système d’exploitation.

Chapitre 2

Généralités

Ce chapitre résume brièvement les éléments nécessaires à la mise en œuvre de l'option SSL de Connect:Express.

Ce chapitre ne décrit ni les protocoles SSL et TLS, ni les différentes spécifications PKI, ni les différents standards cryptographiques.

Le lecteur pourra consulter entre autre :

- RFC2246 The TLS protocol version 1.0 - Network Working Group (January 1999)
- Internet Draft The SSL Protocol Version 3.0 – Netscape communication (March 1996)
- Les différents standards PKCS – RSA Laboratories http://www.rsasecurity.com
- RFC2459 Internet X.509 Public Key Infrastructure – Network Working Group (January 1999)
- RFC1421,1422,1423,1424 Privacy enhancement for Internet Electronic Mail – Network working Group (February 1993)
- ITU-T RECOMMENDATION Information technology – ASN1 encoding rules : Specification of basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Dinstinguished Encoding Rules (DER) (07/2002)

Ainsi qu'un nombre important de documents sur le sujet.

Clés privées et demandes de certificat

Au cours du handshake protocolaire SSL/TLS, le serveur fait parvenir au client son certificat. Le client contrôle ce dernier par rapport au(x) certificats de l'autorité de certification (CA) qui a émis le certificat du serveur.

Authentification du serveur

Un serveur SSL Connect:Express doit donc disposer :
- d'au moins un certificat personnel associé à sa clé privée.

Un client SSL Connect:Express doit disposer :
- des certificats de CA ayant signé le certificat qu'enverra le serveur.

Authentification optionnelle du client

Le serveur a la possibilité de demander également au client de s'authentifier en lui demandant un certificat. Ce mode de fonctionnement est optionnel (Authentification client).

Le client doit en plus disposer dans ce cas:
- d'au moins un certificat personnel associé à sa clé privée.

Le serveur doit en plus disposer dans ce cas :
- des certificats de CA ayant signé le certificat qu'enverra le client.

Processus de génération des clés privées et des certificats

L'option SSL de Connect:Express ne prend pas en charge ce processus qui ce déroule en quatre étapes :

- Création d'une clé privée et d'une demande de certificat (CSR : Certificate signing request)
- Envoi de la CSR à une autorité de certification telle que Verisign
- Réception en retour de la CSR du certificat émis par l'autorité de certification, signé par celle-ci
- Obtention du/des certificats publics de CA permettant la vérification de la validité du certificat. Ces certificats seront à communiquer aux correspondants s'il ne les a pas.

Chaîne de certification

Les autorités de certification peuvent fonctionner sur un modèle hiérarchique, le CA racine déléguant la signature de certificats à des CA secondaires. Dans ce cas le certificat du CA secondaire est signé par le certificat du CA de rang immédiatement supérieur dans la hiérarchie.
Vérifier un certificat personnel consiste dans ce cas à vérifier toute la chaîne de certification jusqu'à aboutir au certificat du CA racine qui est auto-signé.
Pour effectuer la vérification d'un tel type de certificat, Connect:Express devra disposer de tous les certificats de la chaîne de certification.

Listes de révocation de certificats

Connect:Express ne gère pas les listes de révocation de certificats

Format d'importation dans Connect:Express

Format des certificats et des clés importés dans Connect:Express

Lorsque l'on dispose de son certificat personnel et des certificats de CA de ses correspondants, il est nécessaire de les importer dans la base de données de Connect:Express à l'aide de l'utilitaire STERM.

Les formats des fichiers de clés privées ou de certificats que peut importer Connect:Express sont de l'un des deux types suivants :

- DER : fichier binaire ASN1.
- PEM : fichier texte affichable, encodé en base64

Format des certificats PEM

Pour pouvoir être importés dans Connect:Express, les fichiers de certificats au format PEM doivent commencer et se terminer par les 2 lignes suivantes :

-----BEGIN CERTIFICATE-----
…
-----END CERTIFICATE-----

Format des clés privées PEM

Un fichier de clé privée RSA au format PEM protégé par un mot de passe (pass phrase) a une structure semblable à la structure suivante :

-----BEGIN RSA PRIVATE KEY-----
Proc-Type : 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,(…IV…)
…
-----END RSA PRIVATE KEY-----

DEK-Info indique l'algorithme de cryptage de la clé avec le mot de passe.

On a une structure similaire pour les clés DSA.

Remarque :
Les fichiers de clés privées doivent être présents physiquement dans le répertoire $TOM_DIR/config/ssl/priv de Connect:Express, même après importation. Contrairement aux fichiers au format PEM, les fichiers de clés au format DER ne sont pas protégés par mot de passe.
Il y aura donc intérêt à convertir les fichiers de clés du format DER au format PEM.
Lors de l'importation d'une clé au format PEM, le mot de passe est demandé à l'utilisateur. Afin que ce dernier n'apparaisse pas en clair dans la base de donnée de Connect:Express, ce dernier est lui-même crypté par DES triple.
Le mot de passe est décrypté dynamiquement en mémoire à chaque transfert.

Certificats auto-signés

Les certificats auto-signés ne doivent être utilisés que dans le cadre de tests.
Voir l'annexe B pour la création à l'aide de la commande openssl d'un certificat auto-signé et de sa clé privée.

Options de vérification

Le paramètre OPTIONS DE VERIFICATION des profils de transfert SSLPARM de Connect:Express permet d'indiquer le contrôle qui sera fait concernant l'authentification du correspondant. Il définit également si le serveur demande l'authentification du client ou pas.

Ce paramètre peut prendre les valeurs :

- 0 : VERIFY_NONE
- 1 : VERIFY_PEER
- 2 : VERIFY_FAIL_IF_NO_PEER_CERT

Les valeurs 0,1 et 2 peuvent s'appliquer à un serveur
Les valeurs 0 et 1 peuvent s'appliquer à un client

Le tableau suivant résume le comportement d'un serveur et d'un client pour chaque option :

Serveur

Option	Description
VERIFY_NONE	Le serveur n'envoie pas de « ClientCertificateRequest ». Le client n'enverra donc pas de certificat.
VERIFY_PEER	Le serveur envoie un « ClientCertificateRequest ». Si un certificat client est retourné, il est contrôlé. Si la vérification échoue, le handshake est terminé immédiatement.
VERIFY_FAIL_IF_NO_PEER_CERT	Le serveur envoie un « ClientCertificateRequest ». Si le client n'a pas renvoyé de certificat, le handshake est terminé immédiatement. Si un certificat client est retourné, il est contrôlé. Si la vérification échoue, le handshake est terminé immédiatement.

Client

Option	Description
VERIFY_NONE	Le serveur envoie un certificat. Le client analyse ce certificat. L'échange se poursuit indépendamment de la validité du certificat
VERIFY_PEER	Le serveur envoie un « ClientCertificateRequest ». Le certificat du serveur est vérifié. Si la vérification échoue, le handshake est terminé immédiatement.

Un fonctionnement sans authentification du client se traduira donc par :
Serveur : 0 (VERIFY_NONE), Client : 1 (VERIFY_PEER)

Un fonctionnement avec authentification du client se traduira donc par :
Serveur : 2 (VERIFY_FAIL_IF_NO_PEER_CERT), Client : 1 (VERIFY_PEER)

Note: le contrôle des certificats, qui consiste à vérifier les noms contenus dans le certificat reçu, après la phase d'authentification, est effectué quelle que soit l'option de vérification courante. (Se reporter au Chapitre 5)

Suites de chiffrement

Une suite de chiffrement indique les divers modes de chiffrements utilisés par une session SSL : Elle définit les éléments suivants :

- La version de protocole SSL
- La méthode d'échange de clés
- La méthode d'authentification
- L'algorithme de chiffrement symétrique
- L'algorithme de chiffrement MAC
- Les restrictions sur la longueur des clés
- Les restrictions d'exportation

Par exemple, TLS_RSA_WITH_DES_CBC_SHA indique :

- Le protocole TLSv1
- L'échange de clés RSA
- L'authentification RSA
- Le chiffrement symétrique DES-CBC
- L'algorithme de chiffrement Mac SHA1

OpenSSL utilise pour la dénomination des suites de chiffrement une terminologie légèrement différente de celle utilisée dans les spécifications des différents protocoles SSL et TLS.

La correspondance entre les deux terminologies est indiquée en annexe C, avec des indications de non implémentation.

Connect:Express utilise la terminologie de OpenSSL pour désigner les suites de chiffrement. Le tableau ci-dessous résume les différentes suites utilisées.

Suite	**SSLv2**	**TLSv1/ SSLv3**	**Kx**	**Au**	**Enc**	**Mac**	**Exp**
DHE-RSA-AES256-SHA		x	DH	RSA	AES 256	SHA1	
DHE-DSS-AES256-SHA		x	DH	DSS	AES 256	SHA1	
AES256-SHA		x	RSA	RSA	3DES 256	SHA1	
EDH-RSA-DES-CBC3-SHA		x	DH	RSA	3DES 168	SHA1	
EDH-DSS-DES-CBC3-SHA		x	DH	DSS	3DES 168	SHA1	
DES-CBC3-SHA		x	RSA	RSA	3DES 168	SHA1	
DES-CBC3-MD5	x		RSA	RSA	3DES168	MD5	
DHE-RSA-AES128-SHA		x	DH	RSA	AES 128	SHA1	
DHE-DSS-AES128-SHA		x	DH	DSS	AES 128	SHA1	
AES128-SHA		x	RSA	RSA	AES 128	SHA1	
IDEA-CBC-SHA		x	RSA	RSA	IDEA 128	SHA1	
IDEA-CBC-MD5	x		RSA	RSA	IDEA 128	MD5	
RC2-CBC-MD5	x		RSA	RSA	RC2 128	MD5	
DHE-DSS-RC4-SHA		x	DH	DSS	RC4 128	SHA1	
RC4-SHA		x	RSA	RSA	RC4 128	SHA1	
RC4-MD5	X	x	RSA	RSA	RC4 128	MD5	
RC4-64-MD5	x		RSA	RSA	RC4 64	MD5	
EXP1024-DHE-DSS-DES-CBC-SHA		x	DH 1024	DSS	DES 56	SHA1	x
EXP1024-DES-CBC-SHA		x	RSA 1024	RSA	DES 56	SHA1	x
EXP1024-RC2-CBC-MD5		x	RSA 1024	RSA	RC2 56	MD5	x
EDH-RSA-DES-CBC-SHA		x	DH	RSA	DES 56	SHA1	

EDH-DSS-DES-CBC-SHA		x	DH	DSS	DES 56	SHA1	
DES-CBC-SHA		x	RSA	RSA	DES 56	SHA1	
DES-CBC-MD5	x		RSA	RSA	DES 56	MD5	
EXP1024-DHE-DSS-RC4-SHA		x	DSS	DSS	RC4 56	SHA1	x
EXP1024-RC4-SHA		x	RSA 1024	RSA	RC4 56	SHA1	x
EXP1024-RC4-MD5		x	RSA 1024	RSA	RC4 56	MD5	x
EXP-EDH-RSA-DES-CBC-SHA		x	DH 512	RSA	DES 40	SHA1	x
EXP-EDH-DSS-DES-CBC-SHA		x	DH 512	DSS	DES 40	SHA1	x
EXP-DES-CBC-SHA		x	RSA 512	RSA	DES 40	SHA1	x
EXP-RC2-CBC-MD5		x	RSA 512	RSA	RC2 40	MD5	x
EXP-RC2-CBC-MD5	x		RSA 512	RSA	RC2 40	MD5	x
EXP-RC4-MD5		x	RSA 512	RSA	RC4 40	MD5	x
EXP-RC4-MD5	x		RSA 512	RSA	RC4 40	MD5	x

Kx=Méthode d'échange de clés : RSA, RSA, DH/RSA, DH/DSS avec indication de la limitation de taille de clé pour les suites « export » (par exemple RSA 512)
Au = Méthode d'authentification : RSA, DSS, DH
Enc = Méthode de chiffrement symétrique avec le nombre de bits secrets (par exemple DES 56)
Mac= Message authentication code (Message digests) : SHA1, MD5
Exp = indique les suites statisfaisant aux anciennes réglementations d'exportation US.

Lors du « Hello client », le client présente au serveur une liste des suites de chiffrement qu'il désire utiliser pour la connexion, classées par ordre de préférence.

Le serveur choisit la première suite correspondant à l'une des suites qu'il est disposé à utiliser.
Les listes de suites de chiffrements sont définies dans Connect:Express dans le paramètre « LISTE DE CHIFFREMENT » pour chaque profil SSLPARM de session SSL (Voir « Chapitre 4. Configuration des paramètres des transferts »).
Ce paramètre contient le nom d'un simple fichier texte à placer sous $TOM_DIR/config/ssl/ciphlist, qui contient la liste des suites applicables à la connexion.
Par exemple :
LISTE DE CHIFFREMENT = clist1.txt
avec $TOM_DIR/config/ssl/ciphlist/clist.txt défini de la manière suivante :

```
DES-CBC3-SHA:DES-CBC3-MD5:RC4-SHA
```

Les différentes suites y sont indiquées par ordre de préférence, séparées par « : ».

Si le paramètre facultatif LISTE DE CHIFFREMENT n'est pas renseigné dans le profil de session SSL de Connect:Express, la liste de suites de chiffrement suivante est utilisée par défaut :

```
DHE-RSA-AES256-SHA:DHE-DSS-AES256-SHA:AES256-SHA:EDH-RSA-DES-CBC3-SHA:
EDH-DSS-DES-CBC3-SHA:DES-CBC3-SHA:DES-CBC3-MD5:DHE-RSA-AES128-SHA:
DHE-DSS-AES128-SHA:AES128-SHA:IDEA-CBC-SHA:IDEA-CBC-MD5:RC2-CBC-MD5:
DHE-DSS-RC4-SHA:RC4-SHA:RC4-MD5:RC4-MD5:RC4-64-MD5:
EXP1024-DHE-DSS-DES-CBC-SHA:EXP1024-DES-CBC-SHA:EXP1024-RC2-CBC-MD5:
EDH-RSA-DES-CBC-SHA:EDH-DSS-DES-CBC-SHA:DES-CBC-SHA:DES-CBC-MD5:
EXP1024-DHE-DSS-RC4-SHA:EXP1024-RC4-SHA:EXP1024-RC4-MD5:
EXP-EDH-RSA-DES-CBC-SHA:EXP-EDH-DSS-DES-CBC-SHA:EXP-DES-CBC-SHA:
EXP-RC2-CBC-MD5:EXP-RC2-CBC-MD5:EXP-RC4-MD5:EXP-RC4-MD5
```

Listes de CA

Lors du handshake protocolaire SSL, dans le cas où le serveur demande au client de s'authentifier, le serveur indique au client la liste des DN (distinguished names) des CA dont il dispose pour contrôler le certificat du client.

La liste des CA est configurée dans le paramètre « LISTE DE CA » du profil de session SSLPARM du serveur en mentionnant :

- Soit directement l'identifiant d'un certificat de CA importé dans Connect:Express (cas d'une liste à un seul élément)
- Soit le nom d'un fichier liste d'identifiants de certificats de CA présent dans le répertoire $TOM_DIR/config/ssl/calist. Le nom de ce fichier doit commencer par le caractère # suivi d'au plus 7 caractères alphabétiques majuscules ou numériques.

Exemple 1 :

LISTE DE CA = CACERT. CACERT est l'identifiant du certificat d'un CA importé dans Connect:Express. Seul le DN sujet de CACERT sera envoyé au client.

Exemple 2 :
LISTE DE CA = #CALISTE1.
Le fichier $TOM_DIR/config/ssl/calist/#CALISTE1 contient :

```
CACERT1 :CACERT2 :… :CACERTn
```

CACERT1, CACERT2, …, CACERTn sont des identifiants de certificats de CA importés dans Connect:Express.
Les différents DN sujets de ces certificats seront envoyés au client.

Diffie-Hellman éphémère

L'échange de clés Diffie-Hellman éphémère permet au serveur et au client de générer un « pre-master secret » à partir de paramètres Diffie-Hellman (Un nombre premier p et un générateur g).
Les clés publiques échangées et les clés secrètes ont pour durée de vie la session SSL seulement (d'où le terme éphémère).

L'échange de clé peut être utilisé conjointement avec une authentification RSA ou DSS.

Dans le cas de l'authentification par DSS, l'échange de clés Diffie-Hellman est le seul échange de clés possible, car DSS/DSA ne permet que la signature.

Les paramètres DH utilisés par un serveur Connect:Express sont lus à partir de fichiers PEM de paramètres DH situés dans le répertoire $TOM_DIR/config/ssl/dhparam. Le nom du fichier de paramètres à utiliser est configurable dans le profil SSLPARM du serveur (PARAMETRE DH)

Par exemple : PARAMETRE DH = mydhparm.pem.

A l'installation de Connect:Express, des fichiers de paramètres utilisables sont livrés dans le répertoire $TOM_DIR/config/ssl/dhparam : dh512.pem, dh1024.pem, dh2048.pem et dh4096.pem pour des nombres de bits allant de 512 à 4096.

Il est toujours possible d'utiliser la commande $TOM_DIR/config/ssl/bin/openssl pour générer ses propres fichier de paramètre DH (Voir Annexe B).

Note : Les échanges de clés DH fixe et DH anonyme ne sont pas implémentés.

Chapitre 3

Configuration des certificats et des clés

Avant de pouvoir être utilisés par Connect:Express, les certificats et les clés doivent être importés à l'aide de STERM. Il s'agit de :

- Nos certificats personnels et nos clés privées associées
- Les différents certificats de CA qui permettront de contrôler la validité des certificats présentés par nos partenaires distants.

Avant de procéder à une importation dans Connect:Express :

- Le fichier PEM ou DER de clé privée doit être placé manuellement dans le répertoire $TOM_DIR/config/ssl/priv.
- Le fichier PEM ou DER de certificat doit être placé dans le répertoire $TOM_DIR/config/ssl/cert_import.

Dans le cas de l'importation d'un certificat de CA, il n'y a pas de clé privée.

Les fichiers de clés privées placés sous priv ne doivent pas être supprimés de ce répertoire suite à importation, car ils sont physiquement utilisés par Connect:Express pour les transferts.

Le menu de gestion des certificats de STERM permet les opérations suivantes :

- Importation d'un certificat personnel et de sa clé privée.
- Importation d'un certificat de CA
- Liste des certificats importés
- Modification d'un certificat
- Suppression d'un certificat de Connect:Express
- Visualisation des caractéristiques d'un certificat.

Il se présente ainsi :

Menu d'importation de certificats

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS---------------------- tom1
OPTION ===> V




                          I    IMPORTER
                          L    LISTER
                          M    MODIFIER
                          S    SUPPRIMER
                          V    VISUALISER


                               ID ===> ........




  X  EXIT                                                    -F3- FIN
```

ID est un identifiant Connect:Express du certificat.
La suppression d'un certificat de Connect:Express n'entraîne pas la suppression physique des fichiers correspondants des répertoires cert_import et priv. Il est toujours possible de les réimporter ultérieurement.

Importation d'un certificat privé et de sa clé dans Connect:Express

L'écran STERM suivant donne un exemple d'importation d'un certificat personnel de serveur. Le fichier certificat server.pem doit avoir été placé préalablement sous $TOM_DIR/config/ssl/cert_import et le fichier de clé privée serverkey.pem doit avoir été placé sous $TOM_DIR/config/ssl/priv.

Importation d'un certificat personnel

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS---------------------- tom1
OPTION ===> I


ID     : SRVCERT

TYPE   : P                      (P:PERSONNEL,C:CA)

FORMAT FICHIER CERTFICAT : 1    (1:PEM,2:DER)
NOM DU FICHIER CONTENANT LE CERTIFICAT A IMPORTER :
server.pem
FORMAT FICHIER CLE : 1          (1:PEM,2:DER)
NOM DU FICHIER CONTENANT LA CLE PRIVEE A IMPORTER (CERTIFICAT PERSONNEL) :
serverkey.pem


MOT DE PASSE DE LA CLE PRIVEE (CERTIFICAT PERSONNEL) :
********
SAISISSEZ A NOUVEAU
********

OPTION : CREER                          MAJ : .............. ........
-ENTER- CHAMP SUIVANT              -F3- ANNULATION              -F8- VALIDATION
```

L'identification du couple certificat, clé privée pour Connect:Express est SRVCERT. Les deux fichiers sont au format PEM. La clé privée est protégée par mot de passe (pass phrase) indiqué lors de la création de celle-ci.
Note : Les clés privées au format DER ne sont pas protégées par mot de passe.

Importation d’un certificat de CA

L’écran STERM suivant donne un exemple d’importation d’un certificat de CA. Le fichier certificat cacert.pem doit avoir été placé préalablement sous $TOM_DIR/config/ssl/cert_import.

Importation d’un certificat de CA

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS---------------------- tom1
OPTION ===> I


ID     : CACERT

TYPE   : C                      (P:PERSONNEL,C:CA)

FORMAT FICHIER CERTFICAT : 1    (1:PEM,2:DER)
NOM DU FICHIER CONTENANT LE CERTIFICAT A IMPORTER :
cacert.pem
FORMAT FICHIER CLE :            (1:PEM,2:DER)
NOM DU FICHIER CONTENANT LA CLE PRIVEE A IMPORTER (CERTIFICAT PERSONNEL) :



MOT DE PASSE DE LA CLE PRIVEE (CERTIFICAT PERSONNEL) :

SAISISSEZ A NOUVEAU


OPTION : CREER                            MAJ : .............. ........
-ENTER- CHAMP SUIVANT              -F3- ANNULATION              -F8- VALIDATION
```

L’identification du certificat pour Connect:Express est CACERT. Le fichier est au format PEM.

Visualisation des certificats importés dans Connect:Express

La visualisation d'un certificat par STERM affiche l'écran suivant :

Visualisation d'un certificat

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS---------------------- tom1
OPTION ===> V
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 286 (0x11e)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=FR, ST=Paris, L=Paris, O=Sterling, OU=Labs, CN=CA test
        Validity
            Not Before: Mar  7 16:02:34 2006 GMT
            Not After : Mar  7 16:02:34 2007 GMT
        Subject: C=FR, ST=Paris, L=Paris, O=tlabs, OU=tlabs01, CN=Test ssl
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (1024 bit)
                Modulus (1024 bit):
                    00:b5:b2:8f:c3:2b:e8:52:db:de:c7:19:5e:ce:0f:
                    fc:68:85:5f:ac:4d:e9:f9:b2:fc:e0:d9:c5:07:37:
                    6a:42:03:e9:88:51:94:17:64:95:91:b4:f6:32:f7:
                    ba:02:a4:d0:b6:7b:44:16:2f:79:75:63:c8:97:bb:
                    c8:1f:df:ff:63:2a:e2:71:32:85:7e:fa:5e:5d:48:
                    f9:15:72:d6:29:4f:01:c8:0e:a1:ba:7d:cf:f4:3c:

  X <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  --F11- ->
```

Le défilement peut être obtenu avec les touches <UP>, <DOWN>, <F7> ou <ARROW-UP> et <F8> ou <ARROW-DOWN>. L'affichage est tronqué à 80 caractères. On peut le décaler de 80 caractères vers la droite avec les touches <F11> ou <RIGHT> et revenir en position initiale par <F10> ou <LEFT>.

Renouvellement d’un certificat arrivant à expiration

Lorsqu’un certificat est parvenu à expiration, placer le nouveau certificat valide transmis par le CA dans $TOM_DIR/config/ssl/cert_import. Puis à l’aide de STERM utiliser l’option de modification pour mettre à jour les données afférentes au certificat dans Connect:Express.

Mise à jour d’un certificat

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS---------------------- tom1
OPTION ===> M


ID     : SRVCERT

TYPE   : P                      (P:PERSONNEL,C:CA)

FORMAT FICHIER CERTFICAT : 1    (1:PEM,2:DER)
NOM DU FICHIER CONTENANT LE CERTIFICAT A IMPORTER :
newservercert.pem
FORMAT FICHIER CLE : 1          (1:PEM,2:DER)
NOM DU FICHIER CONTENANT LA CLE PRIVEE A IMPORTER (CERTIFICAT PERSONNEL) :
serverkey.pem


MOT DE PASSE DE LA CLE PRIVEE (CERTIFICAT PERSONNEL) :
********
SAISISSEZ A NOUVEAU
********

OPTION : CREER                            MAJ : .............. ........
-ENTER- CHAMP SUIVANT              -F3- ANNULATION              -F8- VALIDATION
```

Note : Les éléments de la clé privée ne changent pas à priori.

Liste des certificats importés

L'option L permet de lister les différents certificats importés dans Connect:Express et d'en visualiser notamment les dates de validité.

Liste des certificats importés

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS---------------------- tom1
OPTION ===>
    ID         TYPE DEBUT VALIDITE       EXPIRATION           DN SUJET

    CLICERT    P    2006/02/07 16:09:58  2007/02/07 16:09:58  /C=FR/ST=Paris/L=Pa
    CERTDSAC   P    2006/05/16 15:15:06  2007/05/16 15:15:06  /C=FR/ST=Paris/L=Pa
    CERTDSAR   C    2006/05/10 16:40:28  2007/05/10 16:40:28  /C=FR/ST=Paris/L=Pa
    CERTDSAS   P    2006/05/10 16:45:26  2007/05/10 16:45:26  /C=FR/ST=Paris/L=Pa
    SRVCERT    P    2006/02/07 16:02:34  2007/02/07 16:02:34  /C=FR/ST=Paris/L=Pa
    DSASELF    P    2006/04/30 17:57:13  2007/04/30 17:57:13  /C=FR/ST=Paris/O=tl
    CACERT     C    2006/02/07 15:37:16  2016/02/04 15:37:16  /C=FR/ST=Paris/L=Pa
    SELFCERT   P    2006/04/17 09:22:11  2006/04/18 09:22:11  /C=FR/ST=Paris/L=Pa









    <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  -F11- ->
```

Décalage vers la droite par touches <F11> ou <RIGHT> :

```
C:E/UNIX 146-1 ------------IMPORTATION DE CERTIFICATS----------------------
tom1OPTION ===>
    ID         DN SUJET

    CLICERT    ris/O=tlabs/OU=tlabs01/CN=Test ssl client
    CERTDSAC   ris/O=tlabs/OU=tlabs01/CN=Test dsa1024 cli
    CERTDSAR   ris/O=tlabs/OU=tlabs01/CN=CA dsa1024 cert
    CERTDSAS   ris/O=tlabs/OU=tlabs01/CN=Test dsa1024 srv
    SRVCERT    ris/O=tlabs/OU=tlabs01/CN=Test ssl
    DSASELF    abs/OU=tlabs01/CN=Test selfdsa
    CACER      ris/O=Sterling/OU=Labs/CN=CA test
    SELFCERT   ris/O=tlabs/OU=tlabs01/CN=Test selfsigned









    <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  -F11- ->
```

Visualisation des caractéristiques d'un certificat avant importation

L'option 3 (Propriétés d'un certificat) du menu général de paramétrage SSL de STERM permet de visualiser les caractéristiques d'un certificat X509 avant de le placer dans cert_import pour l'importer. Il est nécessaire d'indiquer le chemin d'accès complet au fichier contenant le certificat.

Propriétés d'un certificat

```
C:E/UNIX 146-1 ------------PROPRIETES D'UN CERTIFICAT---------------------- tom1
OPTION ===>


FORMAT : 1          (1:PEM,2:DER)


NOM DU FICHIER CONTENANT LE CERTIFICAT :
/tmp/certificate.pem









OPTION : VISUALISER                        MAJ :
-ENTER- CHAMP SUIVANT              -F3- ANNULATION              -F8- VALIDATION
```

L'affichage est similaire à l'affichage produit par la visualisation d'un certificat déjà importé.

Chapitre 4

Configuration des paramètres SSL des transferts

Les paramètres SSL de transfert (SSLPARM) permettent d'indiquer les différents profils de session SSL utilisés pour les transferts. Ils sont de deux types : Serveurs et Clients. Ces paramètres sont définis dans Connect:Express à l'aide de l'interface STERM.

Chaque profil en mode serveur donne lieu au lancement d'un processus SSL serveur (tom_apm) au démarrage du moniteur. Chaque serveur SSL est à l'écoute sur un port réseau particulier.
Une fois défini un profil SSL serveur à l'aide de STERM, il est nécessaire d'arrêter et de redémarrer Connect:Express afin de démarrer le processus SSL serveur correspondant.

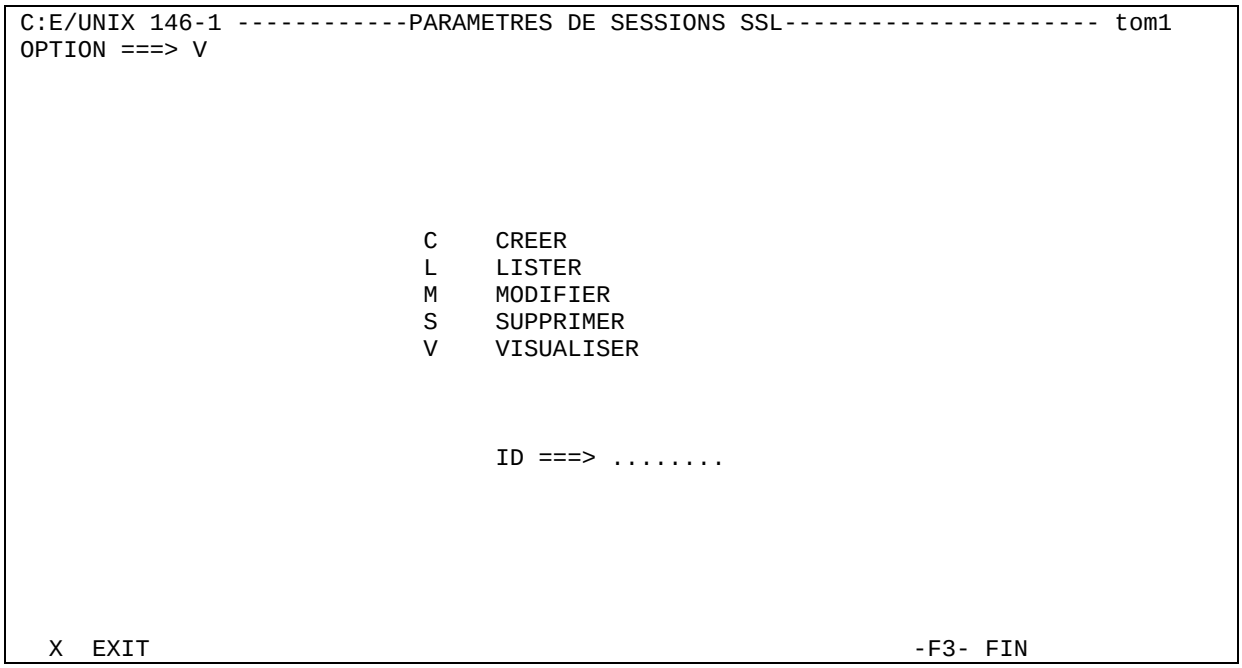
Les profils en mode client ne nécessitent pas d'arrêt relance du moniteur. Chaque transfert PeSIT demandeur est associé à un partenaire symbolique. Dans le cas d'un transfert vers un partenaire SSL, la définition du partenaire symbolique indique dans son champ SSLPARM le profil session SSL à utiliser. Un processus tom_apm client utilisant ce profil sera lancé dynamiquement pour effectuer le transfert.

Les paramètres SSL sont les suivants :

- Mode (client, serveur)
- Options de vérification et d'authentification
- Certificat et clé
- Liste de chiffrement
- Versions de protocole SSL (TLSv1,SSLv3,SSLv2)
- Liste de CA (serveurs)
- Fichier de paramètres Diffie-Hellman (serveurs)
- Adresse locale et port TCP/IP (serveurs)

Le menu de STERM de gestion des paramètres SSL se présente ainsi :

Menu des paramètres SSL



ID est un identifiant Connect:Express du profil.

SSLPARM serveur

L'écran STERM suivant donne un exemple de définition d'un profil SSLPARM serveur.

SSLPARM serveur

```
C:E/UNIX 146-1 ------------PARAMETRES DE SESSIONS SSL---------------------- tom1
OPTION ===>

ID                       : SRV01
ETAT                     : E                    (E:EN-SERVICE,H:HORS-SERVICE)
MODE                     : S                    (C:CLIENT,S:SERVEUR)
OPTIONS DE VERIFICATION  : 2                    (0:AUCUNE,1:PEER
                                                2:PEER + FAIL_IF_NO_PEER_CERT)
ID CERTIFICAT            : SRVCERT
LISTE DE CHIFFREMENT     :                      (NOM FICHIER LISTE)
VERSIONS DE PROTOCOLE SSL :
TLSV1 : O SSLV3 : O SSLV2 : O

LISTE DE CA (SERVEUR)    : CACERT               ('CACERT-ID',#LISTE)
PARAMETRES DH (SERVEUR)  :                      (NOM FICHIER)
ADRESSE RESEAU LOCALE (MODE SERVEUR) :
RESEAU (T:TCPIP)         : T
TCP/IP : ADRESSE IP      :                      PORT TCP : 06678
         ENTETE  IP      : N



OPTION : MODIFIER                          MAJ : 06/06/23 15:27 pga
```

Les divers éléments sont commentés ci-dessous :

Champ	Description
ID	Identifiant Connect:Express du profil
ETAT	Permet d'activer ou de désactiver le profil, donc le serveur SSL correspondant. L'activation ou la désactivation ne sera prise en compte qu'au prochain redémarrage du moniteur
OPTIONS DE VERIFICATION	La valeur 2 indique que l'authentification du client est demandée et que la session échouera si la vérification du certificat du client est négative (Voir le paragraphe « Options de vérification »)
ID CERTIFICAT	Indique l'identifiant Connect:Express du couple certificat/clé privée du serveur. Cet identifiant doit avoir été créé dans Connect:Express avec STERM par une opération d'importation de certificat personnel. Ce champ est obligatoire pour un profil serveur.
LISTE DE CHIFFREMENT	Permet de spécifier le nom d'un fichier du répertoire $TOM_DIR/config/ssl/ciphlist contenant une liste des suites de chiffrement admissibles. Ici, par défaut, tous les chiffrements disponibles peuvent être utilisés (Voir le paragraphe « Listes de chiffrement ») Champ facultatif
VERSIONS DE PROTOCOLE SSL	Indique la ou les versions de protocoles SSL autorisées. Si plusieurs versions sont indiquées, c'est la version de plus haut niveau possible qui sera utilisée avec le partenaire distant
LISTE DE CA	Permet au serveur , dans le cas où il demande au client de s'authentifier, d'indiquer au client la liste des CA qu'il est prêt a accepter pour contrôler le certificat du client. Ce paramètre est - soit directement l'ID d'un certificat de CA importé dans Connect:Express - soit le nom d'un fichier liste présent dans le répertoire $TOM_DIR/config/ssl/calist (Voir le paragraphe « Listes de CA »). Le nom de ce fichier doit commencer par le caractère # suivi d'au plus 7 caractères alphabétiques majuscules ou numériques(Ex : #CALIST1) Dans l'exemple ci-dessus, LISTE DE CA fait directement référence à l'identifiant du certificat de CA « CACERT » champ facultatif
PARAMETRES DH	Nom d'un fichier de paramètres Diffie-Hellman présent dans le répertoire $TOM_DIR/config/ssl/dhparam utilisé par le serveur pour des échanges de clés en mode Diffie-Hellman éphémère (Voir le paragraphe « Paramètres Diffie-Hellman ») Champ facultatif
ADRESSE IP et PORT TCP	Permettent de définir le port d'écoute du serveur Si l'adresse IP n'est pas renseignée, il y a écoute pour toutes les adresses IP locales
ENTETE IP	Cette option détermine si les messages PeSIT sont construits avec les deux octets de longueur en tête de message (comme défini par le protocole PeSIT, sans SSL), ou sans ces deux octets, rendus inutiles par le protocol SSL. L'usage courant est N (Non).

SSLPARM client

L'écran STERM suivant donne un exemple de définition d'un profil SSLPARM client.

SSLPARM client

```
C:E/UNIX 146-1 ------------PARAMETRES DE SESSIONS SSL---------------------- tom1
OPTION ===>

ID                       : CLI01
ETAT                     : E                    (E:EN-SERVICE,H:HORS-SERVICE)
MODE                     : C                    (C:CLIENT,S:SERVEUR)
OPTIONS DE VERIFICATION  : 1                    (0:AUCUNE,1:PEER
                                                2:PEER + FAIL_IF_NO_PEER_CERT)
ID CERTIFICAT            : CLICERT
LISTE DE CHIFFREMENT     :                      (NOM FICHIER LISTE)
VERSIONS DE PROTOCOLE SSL :
TLSV1 : 0 SSLV3 : 0 SSLV2 : 0

LISTE DE CA (SERVEUR)    :                      ('CACERT-ID',#LISTE)
PARAMETRES DH (SERVEUR)  :                      (NOM FICHIER)
ADRESSE RESEAU LOCALE (MODE SERVEUR) :
RESEAU (T:TCPIP)         :
TCP/IP : ADRESSE IP      :                      PORT TCP :
         ENTETE  IP      : N



OPTION : MODIFIER                          MAJ : 06/06/23 15:27 pga
```

Les divers éléments sont commentés ci-dessous :

Champ	Description
ID	Identifiant Connect:Express du profil
ETAT	Permet d'activer ou de désactiver le profil. La prise en compte est immédiate.
OPTIONS DE VERIFICATION	La valeur 1 indique que la session échouera si la vérification du certificat du serveur est négative (Voir le paragraphe « Options de vérification »)
ID CERTIFICAT	Indique l'identifiant Connect:Express du couple certificat/clé privée du client. Cet élément n'est utilisé que dans le cas où le serveur demande au client de s'authentifier. Cet identifiant doit avoir été créé dans Connect:Express avec STERM par une opération d'importation de certificat personnel. Ce champ est facultatif. Si ce paramètre n'est pas renseigné, le profil ne peut être utilisé qu'avec des serveurs qui ne demandent pas d'authentification du client.
LISTE DE CHIFFREMENT	Permet de spécifier le nom d'un fichier du répertoire $TOM_DIR/config/ssl/ciphlist contenant une liste des protocoles de chiffrement admissibles. Ici, par défaut, tous les chiffrements disponibles peuvent être utilisés (Voir le paragraphe « Listes de chiffrement ») Champ facultatif
VERSIONS DE PROTOCOLE SSL	Indique la ou les versions de protocoles SSL autorisées. Si plusieurs versions sont indiquées, c'est la version de plus haut niveau possible qui sera utilisée avec le partenaire distant
ENTETE IP	Cette option détermine si les messages PeSIT sont construits avec les deux octets de longueur en tête de message (comme défini par le protocole PeSIT, sans SSL), ou sans ces deux octets, rendus inutiles par le protocol SSL. L'usage courant est N (Non).

Liste des paramètres de transfert

L'option L permet de lister les paramètres SSL de transfert définis.

Liste des définitions SSLPARM

```
C:E/UNIX 146-1 ------------PARAMETRES DE SESSIONS SSL---------------------- tom1
OPTION ===>
    ID        MODE TLSv1 SSLv3 SSLv2 ETAT  VERIF CERTIFICAT  PORT  ENTETE

    CLI01     C    O     O     O     E     1     CLICERT           N
    CLI02     C    O     O     O     E     1     SELFCERT          N
    CLICIPH   C    O     O     O     E     1     CLICERT           N
    CLIDSA    C    O     O     O     E     1     CERTDSAC          N
    CLIE0     C    O     O     O     E     1     CLIE0             N
    SRV01     S    O     O     O     H     2     SRVCERT     06680 N
    SRV02     S    O     O     O     H     2     SELFCERT    06679 N
    SRVCIPH   S    O     O     O     E     2     SRVCERT     06678 N
    SRVDSA    S    O     O     O     H     2     CERTDSAS    06681 N
    SRVE0     S    O     O     O     H     2     SRVE0       06684 N









    <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  -F11- ->
```

Décalage vers la droite par touches <F11> ou <RIGHT> :

```
C:E/UNIX 146-1 ------------PARAMETRES DE SESSIONS SSL---------------------- tom1
OPTION ===>
    ID        ADRESSE         LISTE CHIFFREMENT LISTE CA    PARAMETRES DH

    CLI01
    CLI02
    CLICIPH                   cipher.txt
    CLIDSA                    cipher.txt
    CLIE0
    SRV01                                       CACERT
    SRV02
    SRVCIPH                                     CACERT      dh4096.pem
    SRVDSA                                      CERTDSAR
    SRVE0









    <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  -F11- ->
```

Définition d'un partenaire symbolique

Un partenaire symbolique définit le profil d'un partenaire pour la couche applicative PeSIT ou Etebac3. Si cette couche applicative utilise SSL pour communiquer avec le partenaire, la couche SSL vient s'intercaler entre la couche applicative et la couche réseau. Le comportement est dans ce cas légèrement différent du cas où la couche applicative est située directement sur la couche réseau.

Transferts PeSIT serveur sur SSL

Le mode PeSIT serveur coïncide avec le mode SSL serveur. Localement, la définition avec STERM d'un partenaire PeSIT serveur avec SSL est identique à celle d'un partenaire PeSIT sans SSL. Mais les ports TCP d'écoute locaux seront différents selon que l'on effectue un transfert PeSIT standard sans SSL (paramètre TCPORT du fichier $TOM_DIR/config/sysin) ou un transfert PeSIT avec SSL (PORT défini dans le profil SSLPARM serveur).
Nous devrons communiquer à notre correspondant l'adresse IP et le port d'écoute indiqué dans la définition SSLPARM.
De même qu'il peut y avoir plusieurs partenaires PeSIT serveurs associés à un port d'écoute local, il peut y avoir plusieurs partenaires PeSIT serveurs SSL associés au même profil SSLPARM (donc au même port d'écoute).

Transferts PeSIT client sur SSL

Le mode PeSIT client coïncide avec le mode SSL client. Pour effectuer une demande sortante d'émission ou de réception PeSIT vers un partenaire PeSIT SSL distant, il est nécessaire d'indiquer dans la définition symbolique de ce partenaire l'identifiant d'un profil SSLPARM client qui définira l'ensemble des caractéristiques de la session SSL.

L'exemple ci-dessous indique le profil SSLPARM 'CLI01', qui définit les conditions des sessions SSL client avec le serveur PeSIT SSL distant situé à l'adresse (192.168.0.12 ,7712).

Création d'un partenaire symbolique

```
C:E/UNIX 146-1 ----------- REPERTOIRE DES PARTENAIRES -------------------tom1
OPTION ===>

NOM SYMBOLIQUE       :    EUX
MOT DE PASSE ....... :    PSW             INTERNE AU MONITEUR
ETAT INITIALISATION ... : E               E:EN-SERVICE H:HORS-SERVICE
NATURE ................ : O               T/O
NUMERO PROTOCOLE ...... : 3               1:ETEBAC 3, 2:FTP, 3:PESIT
TABLE DE SESSION ...... : 1               1->9 TABLES DE SESSION
PORT X25 .............. :                 NOM DEVICE X25
NOMBRE DE SESSIONS .... : 20/10/10        01->64  TOT/IN/OUT
TYPE DE LIAISON ....... : T               X, P, T OU M
NUMERO X25 ............ :                 1-15 CAR. (NO TRANSPAC)
(SOUS)ADRESSE LOCALE .. :                 1-15 CAR. (NO TRANSPAC)
COMPLEMENT D'APPEL .... :                 'USER-DATA-FIELD'
FACILITES ............. :
HOST TCPIP ............ : localhost                        PORT . : 05090
ADRESSE TCPIP ......... :                 DEF FICHIER FTP :
DPCSID ALIAS .......... : NOUS            SSLPARM ID .... : SSLCLI
DPCPSW ALIAS .......... : PSW             CONTROLE DES CERTIFICATS : TESTDN01
NOMBRE DE REPRISES .... : 00              INTERV.SESS ,TRF  : 00, 00 MINUTES
DESIREZ-VOUS CONTINUER ?
OPTION : VISUALISER                       MAJ : 06/05/18 18:42 pga
-ENTER- CHAMP SUIVANT             -F3- ANNULATION              -F8- VALIDATION
```

Chapitre 5

Contrôle des certificats

Cette section décrit le mécanisme de contrôle des certificats, comment l'utiliser et le configurer dans Connect:Express unix.

Généralités

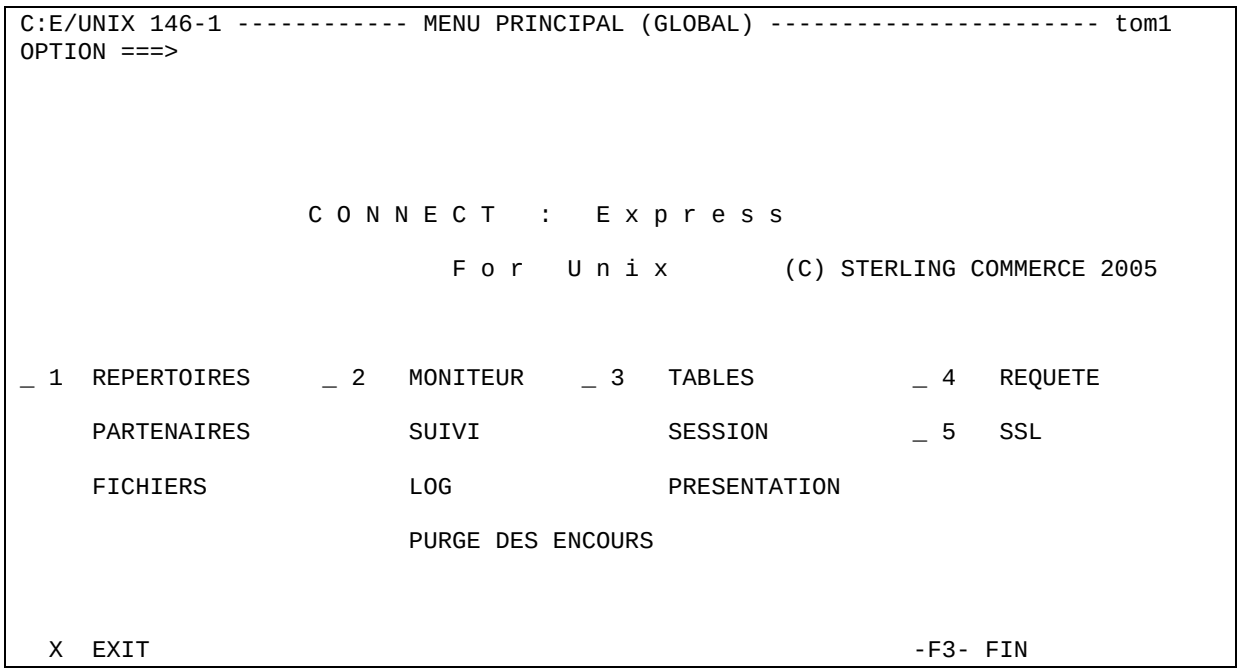
Quand Connect:Express reçoit le certificat d'un partenaire distant, l'interface SSL (openssl) contrôle l'authenticité du certificat et sa présence dans la base des certificats du moniteur. Pour pouvoir vérifier que le nom contenu dans le certificat correspond à celui qu'on attend, il faut utiliser la fonction de contrôle des certificats.

On peut définir le nom de domaine, le DN, attendu pour une session entrante (DN Client) et une session sortante (DN serveur). En mode serveur, le contrôle est lié au paramètre OPTIONS DE VERIFICATION des paramères SSL. Le DN du Client ne peut être contrôlé que si le paramètre OPTIONS DE VERIFICATION n'est pas égal à zéro. En mode client, le contrôle de DN est fait même si le paramètre OPTIONS DE VERIFICATION n'est pas égal à 2.

Mise en oeuvre

Les définitions de DN sont gérées par STERM. Il faut définir des profils de contrôle de DN, puis associer à un partenaire le profil qui convient.

L'option 4 du menu SSL permet d'accéder au CONTROL DES CERTIFICATS



```
C:E/UNIX 146-1 ----------------------SSL-------------------------------- tom1
OPTION ===>




                    1 PARAMETRES DE SESSIONS SSL

                    2 IMPORTATION DE CERTIFICATS

                    3 PROPRIETES D'UN CERTIFICAT

                    4 CONTROLE DES CERTIFICATS




  X  EXIT                                                  -F3- FIN
```

Le contrôle des certificats se fait au travers de définitions de DN (Domain Name) . Pour un certificat, on peut contrôler le 'DN objet' (l'identité du partenaire) et le 'DN Racine ' (l'identité de l'autorité qui a délivré le certificat).

Les figures ci dessous représentent les définitions identifiées par le nom symbolique TESTDN.

```
C:E/UNIX 146-1 -------------CONTROLE DES CERTIFICATS---------------------- tom1
OPTION ===> V



            C    CREER
            L    LISTER
            M    MODIFIER
            S    SUPPRIMER
            V    VISUALISER


                    ID  ===> TESTDN




  X  EXIT                                                  -F3- FIN
```

On peut afficher la liste des définitions, ajouter, modifier, supprimer, visualiser une définition.

Configurer le contrôle des certificats

Une définition de contrôle de certificat peut contenir à la fois les règles en mode serveur et en mode client. En session entrante, mode serveur, le certificat du client peut être contrôlé. En session sortante, mode client, le certificat du serveur peut être contrôlé. Les caractères '*' et ' ?' peuvent être utilisés, de la même façon que pour les noms de fichiers.

Le champ ETAT indique si le contrôle est actif ou non.

```
C:E/UNIX 146-1 -------------CONTROLE DES CERTIFICATS---------------------- tom1
OPTION ===>

ID   ===>           TESTDN01        ETAT  ===> E


DN CLIENT  DISTANT

DN OBJET    : CN=Test*

DN RACINE   : CN=CA*


DN SERVEUR DISTANT

DN OBJET    : CN=Test*

DN RACINE   : CN=CA*




OPTION : MODIFIER                          MAJ : 10/03/09 17:22 gcz
-ENTER- CHAMP SUIVANT              -F3- ANNULATION              -F8- VALIDATION
```

La liste montre les définitions en deux parties.

```
C:E/UNIX 146-1 -------------CONTROLE DES CERTIFICATS---------------------- tom1
OPTION ===>                    DN CLIENT  DISTANT

  ID         ST  DN OBJET    :                 DN RACINE   :

  TESTDN     E   CN=Test*                      CN=CA*
  TESTDN02   E   CN=test*




    <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  -F11- ->
```

F10 or F11 affiche la seconde partie.

```
C:E/UNIX 146-1 -------------CONTROLE DES CERTIFICATS---------------------- tom1
OPTION ===>                    DN SERVER  DISTANT

  ID         ST  DN OBJET    :                 DN RACINE   :

  TESTDN     E   CN=Test*                      CN=CA*
  TESTDN02   E   CN=test*




    <- -F10-  -F3- FIN  -F7- ECRAN PRECEDENT  -F8- ECRAN SUIVANT  -F11- ->
```

Contrôle des certificats d'un partenaire

On peut associer à un partenaire les règles de contrôles des certificats en mode client et serveur, au travers du paramètre CONTROLE DES CERTIFICATS.

```
C:E/UNIX 146-1 ----------- REPERTOIRE DES PARTENAIRES -------------------- tom1
OPTION ===>

NOM SYMBOLIQUE ........ : SSLBCL
MOT DE PASSE .......... : PSW             INTERNE AU MONITEUR
ETAT INITIALISATION ... : E               E:EN-SERVICE H:HORS-SERVICE
NATURE ................ : O               T/O
NUMERO PROTOCOLE ...... : 3               1:ETEBAC 3, 2:FTP, 3:PESIT
TABLE DE SESSION ...... : 1               1->9 TABLES DE SESSION
PORT X25 .............. :                 NOM DEVICE X25
NOMBRE DE SESSIONS .... : 20/10/10        01->64  TOT/IN/OUT
TYPE DE LIAISON ....... : T               X, P, T OU M
NUMERO X25 ............ :                 1-15 CAR. (NO TRANSPAC)
(SOUS)ADRESSE LOCALE .. :                 1-15 CAR. (NO TRANSPAC)
COMPLEMENT D'APPEL .... :                 'USER-DATA-FIELD'
FACILITES ............. :
HOST TCPIP ............ : localhost                      PORT . : 05090
ADRESSE TCPIP ......... :                 DEF FICHIER FTP :
DPCSID ALIAS .......... : SSLBCL          SSLPARM ID .... : SSLCLI
DPCPSW ALIAS .......... : PSW             CONTROLE DES CERTIFICATS : TESTDN
NOMBRE DE REPRISES .... : 00              INTERV.SESS ,TRF  : 00, 00 MINUTES

OPTION : MODIFIER                         MAJ : 10/04/09 09:15 gcz
```

ANNEXE A
Codes d'erreurs

L'interface STERM, en ce qui concerne l'option SSL, peut afficher :
- soit des codes d'erreur TRC (codes générés par Connect:Express)
- soit des codes SSLRC retournés par l'API OpenSSL.

Si une erreur due à la couche SSL survient au cours d'un transfert de fichiers, elle se caractérise par un code TRC d'erreur sur la connexion réseau (2044 ou 2077) et par un code SSLRC retourné par l'API d'OpenSSL. Ces codes sont affichés par STERM dans le détail d'une requête.

TRC

Les codes TRC retournés spécifiquement par l'option SSL sont les suivants :

TRC	Description
2057	DN objet invalide
2058	DN racine invalide
2059	DN définition invalide
2211	Erreur création SSLPARM
2212	Erreur lecture SSLPARM
2213	Erreur mise à jour SSLPARM
2214	Erreur suppression SSLPARM
2215	Erreur SSLPARM invalide
2216	Erreur création CERT
2217	Erreur lecture CERT
2218	Erreur mise à jour CERT
2219	Erreur suppression CERT
2220	Erreur fichier certificat invalide
2221	Erreur DN sujet existe déjà

SSLRC

Les codes d'erreur retournés par la couche SSL sont formés de 8 digits hexadécimaux.

La commande $TOM_DIR/sslerr affiche le libellé correspondant à un code erreur. Par exemple :

```
# $TOM_DIR/config/sslerr 1408A0C1
Error:1408A0C1:SSL routines:SSL3_GET_CLIENT_HELLO:no shared cipher
```

Annexe B

Exemples d'utilisation de la commande openssl

Ce chapitre donne des exemples d'utilisation de la commande openssl.

Commande openssl

Production des clés et des demandes de certificat.

Connect:Express ne prend pas en charge la production des clés et des certificats. Celle-ci doit être faite avec un logiciel adapté indépendant de Connect:Express (par exemple iKeyman, OpenSSL). Une fois les clés et les certificats obtenus, ils peuvent être importés dans Connect:Express.

Package OpenSSL.

Le package OpenSSL complet inclut une commande openssl permettant de générer clés et demande de certificat. Il permet également de créer sa propre autorité de certification, donc de signer des certificats dépendant de ce CA.
Si l'on veut créer son propre CA avec OpenSSL, il est nécessaire d'installer le package complet OpenSSL.
Pour de plus amples informations, consulter la documentation d'OpenSSL (http://www.openssl.org).

Commande openssl livrée avec Connect:Express.

Connect:Express inclut dans sa livraison, une partie seulement du package complet OpenSSL :

- La commande openssl (répertoire $TOM_DIR/config/ssl/openssl/bin)
- Les pages man afférentes à cette commande (répertoire $TOM_DIR/config/ssl/openssl/man)

La commande livrée avec Connect:Express permet, à elle seule, un grand nombre d'opérations sur les clés et certificats, à l'exclusion de la création d'un CA personnel.

Exemples d'utilisation

Le paragraphe ci-dessous indique un certain nombre d'utilisations possibles de la commande openssl.

Création d'une clé privée RSA au format PEM

La commande suivante crée une clé RSA de 1024 bits au format PEM, protégée par le mot de passe « mypass » et un cryptage DES triple en utilisant les données aléatoires contenues dans le fichier « .rnd » du répertoire courant.

```
# ./openssl genrsa –des3 [–rand .rnd] –out mykey.pem –passout pass:mypass 1024
```

Note : option –rand . Un fichier ".rnd" ne doit pas être utilisé si le système d'exploitation dispose du device /dev/random

Conversion d'une clé privée RSA du format PEM au format DER

La commande suivante convertit la clé RSA mykey.pem du format PEM au format DER :

```
# ./openssl rsa –in mykey.pem –passin pass:mypass –out mykey.der –outform DER
```

Note : La clé n'est plus protégée par mot de passe dans le fichier de sortie

Visualisation des caractéristiques d'une clé privée RSA

Format DER :

```
# ./openssl rsa –in mykey.der –inform DER –noout -text
```

Format PEM :

```
# ./openssl rsa –in mykey.pem –passin: mypass –noout -text
```

Création d'un fichier de paramètres DSA

Un fichier de paramètres DSA peut être utilisé pour faciliter la création de plusieurs clés DSA. La commande suivante génère un fichier de paramètres DSA pour une longueur de clé de 1024 en utilisant les données aléatoires contenues dans le fichier « .rnd » du répertoire courant.

```
# ./openssl dsaparam [–rand .rnd] –out dsaparam.pem 1024
```

Note : cf note sur l'option –rand ci-dessus.

Création d'une clé privée DSA

La commande suivante génère une clé DSA :

```
# ./openssl gendsa –des3 [–rand .rnd] –out mykey.pem –passout pass:mypass dsaparam.pem
```

Note : cf note sur l'option –rand ci-dessus.

Conversion d'une clé privée DSA du format PEM au format DER

La commande suivante convertit la clé RSA mykey.pem du format PEM au format DER :

```
# ./openssl dsa –in mykey.pem –passin pass:mypass –out mykey.der –outform DER
```

Note : La clé n'est plus protégée par mot de passe dans le fichier de sortie

Visualisation des caractéristiques d'une clé privée DSA

Format DER :

```
# ./openssl dsa –in mykey.der –inform DER –noout -text
```

Format PEM :

```
# ./openssl dsa –in mykey.pem –passin: mypass –noout -text
```

Création d'une demande de certificat avec une clé privée RSA ou DSA au format PEM existante

Une demande de certificat (CSR Certificate Signing Request) mycsr.pem est crée pour utilisation avec la clé privée mykey.pem

```
# ./openssl req -new –in mykey.pem –passin pass :mypass –out mycsr.pem –days 365 \
> -subj '/C=FR/ST=Paris/L=Paris/O=org/U=unit/CN=Test ssl srv'
```

Visualisation des caractéristiques d'une demande de certificat

La commande suivante permet de visualiser les caractéristiques d'une demande de certificat :

```
# ./openssl req –in mycsr.pem –noout -text
```

Visualisation d'un certificat

La commande suivante permet de visualiser les caractéristiques d'un certificat X509 retourné par le CA:

```
# ./openssl x509 –in mycert.pem –noout -text
```

Création d'un fichier PKCS#12

On suppose que l'on dispose d'un certificat personnel mycert.pem, de sa clé privée mykey.pem et du certificat du CA cacert.pem. La commande suivante génère un fichier pkcs#12 mycert.p12 regroupant les trois éléments :

```
# ./openssl pkcs12 –export –in mycert.pem –inkey mykey.pem –passin pass:mypass –certfile cacert.pem
–name 'NAME' –out mycert.p12 –passout pass:mypkcs12pass
```

mypass est le mot de passe du fichier mykey.pem.
mypkcs12pass est le mot de passe du fichier pkcs#12 généré.
NAME est un nom identifiant le fichier pkcs#12 (« friendly name »)

Extraction des certificats et de la clé privée d'un fichier PKCS#12

Connect:Express n'utilise pas les fichiers pkcs#12. Si l'on dispose d'un tel fichier, il faut en extraire les certificats et la clé privée avant de les importer par STERM dans Connect:Express.
La commande suivante extrait les différents éléments du fichier mycert.p12. Ceux-ci sont concaténés en sortie dans le fichier concat.pem. Il est ensuite nécessaire de séparer les divers éléments (certificats,clés) présents dans output.pem à l'aide d'un éditeur de texte.

```
# ./openssl pkcs12 –in mycert.p12 –passin pass:mypkcs12pass –des3 –out concat.pem –passout pass:mypass
```

La clé privée est encryptée en DES triple dans le fichier de sortie et protégée par le mot de passe mypass.

Création d'un fichier de paramètres Diffie-Hellman

La commande suivante crée un fichier de paramètres Diffie-Hellman avec une taille de clé 2048.
Ce fichier de paramètre sert à un serveur à générer à la volée les paramètres et secrets des échanges de clés Diffie-Hellman éphémère.

```
# ./openssl dhparam [–rand .rnd] –out dhparam.pem 2048
```

Note : cf note sur l'option –rand ci-dessus.

Visualisation des paramètres Diffie-Hellman

La commande suivante permet de visualiser les paramètres d'un fichier de paramètres Diffie-Hellman

```
# ./openssl dhparam –noout –text –in dhparam.pem
```

Création d'un certificat auto-signé RSA et de sa clé.

La commande suivante permet de créer un certificat auto-signé ainsi que sa clé privée :

```
# ./openssl req –x509 –days 365 \
> -subj '/C=FR/ST=Paris/L=Paris/O=org/U=unit/CN=Test selfsigned'  \
> -newkey rsa:1024 –keyout mykey.pem –passout pass:mypass \
> -out mycert.pem
```

Annexe C

Dénomination des suites de chiffrement

Le tableau suivant indique la correspondance entre les noms des suites telles que définies par les spécifications SSL et TLS (à gauche) et ceux définis par OpenSSL et utilisés par Connect:Express (à droite).

Dans la dénomination OpenSSL des listes ci-dessous, plusieurs suites n'incluent pas le mode d'authentification utilisé (par exemple DES-CBC-SHA). Dans ce cas le mode d'authentification utilisé est RSA.

Suites de chiffrement SSLv3.0

SSL_RSA_WITH_NULL_MD5	NULL-MD5
SSL_RSA_WITH_NULL_SHA	NULL-SHA
SSL_RSA_EXPORT_WITH_RC4_40_MD5	EXP-RC4-MD5
SSL_RSA_WITH_RC4_128_MD5	RC4-MD5
SSL_RSA_WITH_RC4_128_SHA	RC4-SHA
SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5	EXP-RC2-CBC-MD5
SSL_RSA_WITH_IDEA_CBC_SHA	IDEA-CBC-SHA
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA	EXP-DES-CBC-SHA
SSL_RSA_WITH_DES_CBC_SHA	DES-CBC-SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA	DES-CBC3-SHA
SSL_DH_DSS_EXPORT_WITH_DES40_CBC_SHA	Non implémenté
SSL_DH_DSS_WITH_DES_CBC_SHA	Non implémenté
SSL_DH_DSS_WITH_3DES_EDE_CBC_SHA	Non implémenté
SSL_DH_RSA_EXPORT_WITH_DES40_CBC_SHA	Non implémenté
SSL_DH_RSA_WITH_DES_CBC_SHA	Non implémenté
SSL_DH_RSA_WITH_3DES_EDE_CBC_SHA	Non implémenté
SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA	EXP-EDH-DSS-DES-CBC-SHA
SSL_DHE_DSS_WITH_DES_CBC_SHA	EDH-DSS-CBC-SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA	EDH-DSS-DES-CBC3-SHA
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	EXP-EDH-RSA-DES-CBC-SHA
SSL_DHE_RSA_WITH_DES_CBC_SHA	EDH-RSA-DES-CBC-SHA
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA	EDH-RSA-DES-CBC3-SHA
SSL_DH_anon_EXPORT_WITH_RC4_40_MD5	EXP-ADH-RC4-MD5
SSL_DH_anon_WITH_RC4_128_MD5	ADH-RC4-MD5
SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA	EXP-ADH-DES-CBC-SHA
SSL_DH_anon_WITH_DES_CBC_SHA	ADH-DES-CBC-SHA
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA	ADH-DES-CBC3-SHA
SSL_FORTEZZA_KEA_WITH_NULL_SHA	Non implémenté
SSL_FORTEZZA_KEA_WITH_FORTEZZA_CBC_SHA	Non implémenté
SSL_FORTEZZA_KEA_WITH_RC4_128_SHA	Non implémenté

Suites de chiffrement TLSv1.0

TLS_RSA_WITH_NULL_MD5	NULL-MD5
TLS_RSA_WITH_NULL_SHA	NULL-SHA
TLS_RSA_EXPORT_WITH_RC4_40_MD5	EXP-RC4-MD5
TLS_RSA_WITH_RC4_128_MD5	RC4-MD5
TLS_RSA_WITH_RC4_128_SHA	RC4-SHA
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5	EXP-RC2-CBC-MD5
TLS_RSA_WITH_IDEA_CBC_SHA	IDEA-CBC-SHA
TLS_RSA_EXPORT_WITH_DES40_CBC_SHA	EXP-DES-CBC-SHA
TLS_RSA_WITH_DES_CBC_SHA	DES-CBC-SHA

TLS_RSA_WITH_3DES_EDE_CBC_SHA	DES-CBC3-SHA
TLS_DH_DSS_EXPORT_WITH_DES40_CBC_SHA	Non implémenté
TLS_DH_DSS_WITH_DES_CBC_SHA	Non implémenté
TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA	Non implémenté
TLS_DH_RSA_EXPORT_WITH_DES40_CBC_SHA	Non implémenté
TLS_DH_RSA_WITH_DES_CBC_SHA	Non implémenté
TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA	Non implémenté
TLS_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA	EXP-EDH-DSS-DES-CBC-SHA
TLS_DHE_DSS_WITH_DES_CBC_SHA	EDH-DSS-CBC-SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	EDH-DSS-DES-CBC3-SHA
TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	EXP-EDH-RSA-DES-CBC-SHA
TLS_DHE_RSA_WITH_DES_CBC_SHA	EDH-RSA-DES-CBC-SHA
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	EDH-RSA-DES-CBC3-SHA
TLS_DH_anon_EXPORT_WITH_RC4_40_MD5	EXP-ADH-RC4-MD5
TLS_DH_anon_WITH_RC4_128_MD5	ADH-RC4-MD5
TLS_DH_anon_EXPORT_WITH_DES40_CBC_SHA	EXP-ADH-DES-CBC-SHA
TLS_DH_anon_WITH_DES_CBC_SHA	ADH-DES-CBC-SHA
TLS_DH_anon_WITH_3DES_EDE_CBC_SHA	ADH-DES-CBC3-SHA

Suites de chiffrement AES RFC3268 (extension de TLSv1.0)

TLS_RSA_WITH_AES_128_CBC_SHA	AES128-SHA
TLS_RSA_WITH_AES_256_CBC_SHA	AES256-SHA
TLS_DH_DSS_WITH_AES_128_CBC_SHA	DH-DSS-AES128-SHA
TLS_DH_DSS_WITH_AES_256_CBC_SHA	DH-DSS-AES256-SHA
TLS_DH_RSA_WITH_AES_128_CBC_SHA	DH-RSA-AES128-SHA
TLS_DH_RSA_WITH_AES_256_CBC_SHA	DH-RSA-AES256-SHA
TLS_DHE_DSS_WITH_AES_128_CBC_SHA	DHE-DSS-AES128-SHA
TLS_DHE_DSS_WITH_AES_256_CBC_SHA	DHE-DSS-AES256-SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	DHE-RSA-AES128-SHA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	DHE-RSA-AES256-SHA
TLS_DH_anon_WITH_AES_128_CBC_SHA	ADH-AES128-SHA
TLS_DH_anon_WITH_AES_256_CBC_SHA	ADH-AES256-SHA

Suites de chiffrement Export et autres

Ces suites peuvent être utilisées en SSLv3 et TLSv1

TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA	EXP1024-DES-CBC-SHA
TLS_RSA_EXPORT1024_WITH_RC4_56_SHA	EXP1024-RC4-SHA
TLS_DHE_DSS_EXPORT1024_WITH_DES_CBC_SHA	EXP1024-DHE-DSS-DES-CBC-SHA
TLS_DHE_DSS_EXPORT1024_WITH_RC4_56_SHA	EXP1024-DHE-DSS-RC4-SHA
TLS_DHE_DSS_WITH_RC4_128_SHA	DHE-DSS-RC4-SHA

Suites de chiffrement SSLv2.0

SSL_CK_RC4_128_WITH_MD5	RC4-MD5
SSL_CK_RC4_128_EXPORT40_WITH_MD5	EXP-RC4-MD5
SSL_CK_RC2_128_CBC_WITH_MD5	RC2-MD5
SSL_CK_RC2_128_CBC_EXPORT40_WITH_MD5	EXP-RC2-MD5
SSL_CK_IDEA_128_CBC_WITH_MD5	IDEA-CBC-MD5
SSL_CK_DES_64_CBC_WITH_MD5	DES-CBC-MD5
SSL_CK_DES_192_EDE3_CBC_WITH_MD5	DES-CBC3-MD5

Note :
Les modes Diffie-Hellman non éphémères ne sont pas implémentés actuellement par OpenSSL car il n'y a pas de support pour les certificats DH.