RFC 2829 – Méthodes d'Authentification pour LDAP

La version 3 de LDAP est un protocole puissant d'accès aux annuaires.

Elle offre des moyens de recherche et de manipulation du contenu d'un annuaire, et les manières d'accéder grâce à des fonctions de sécurité.

Afin d'avoir le meilleur fonctionnement sur Internet, il est essentiel que ces fonctions de sécurité soient interopérables ; donc il doit y avoir un sous-ensemble minimum de fonctions de sécurité qui est commun à toutes les applications qui réclament la conformité LDAPv3.

Les menaces basiques pour un annuaire LDAP sont :

(1) accès non autorisé aux données par l'intermédiaire des opérations de recherche de donnée

(2) accès non autorisé à l'information d'authentification de client en surveillant d'autres accès

(3) accès non autorisé aux données en surveillant d'autres accès

(4) modification non autorisée des données

(5) modification non autorisée de configuration

(6) utilisation non autorisée ou excessive des ressources et

(7) Spoofing d'annuaire : Duper un client dans le but de faire croire que l'information est venue de l'annuaire alors quand fait elle ne le sont pas , en modifiant des données en transit ou désorienter la connection du client.

Les menaces (1), (4), (5) et (6) sont dues aux clients hostiles. Les menaces (2), (3) et (7) sont dues aux « agents » hostiles qui peuvent intervenir entre le client et le serveur.

Le protocole LDAP peut être protégée avec les mécanismes de sécurité suivants :

(1) l'authentification de client au moyen d'un mécanisme SASL , possibilité d'ajouter le mécanisme d'échange TLS

(2) autorisation de client au moyen de contrôle d'accès basé sur l'identité authentifiée du demandeur,

(3) protection de l'intégrité des données au moyen des mécanismes de protocole TLS ou de crypatge de donnée SASL

(4) protection contre snooping au moyen des mécanismes de protocole TLS ou de crypatge de donnée SASL

(5) limitation de ressource au moyen de limites administratives sur les controles de service

(6) authentification serveur grace aux protocoles TLS ou mécanisme SASL.

A ce moment, l'imposition des contrôles d'accès est faite en dehors du protocole LDAP (de la portée).

Dans ce document, le terme "utilisateur" représente n'importe quelle application qui est un client LDAP (utilisant l'annuaire pour rechercher ou stocker l'information) .

2. Exemple de scénario de déploiement:

Les scénarios suivants sont typiques pour les annuaires LDAP sur Internet, et ont différentes conditions de sécurité. ("sensibles" signifie que les données une grande importance pour leur propriétaires si il est indiqué (ne pas endommager) ; il peut y avoir des données protégées mais pas sensible. Cette section n'est pas prévue pour être une liste exhaustives, d'autres scénarios sont possible, particulièrement sur les réseaux physiquement protégés.

(1) un annuaire en lecture seul, ne contenant aucune donnée sensible, accessible à "n'importe qui", et la connection TCP ou le spoofing IP n'est pas un problème. Cet annuaire n'exige aucune fonction de sécurité excepté des limites administratives de service.

(2) un annuaire en lecture seul ne contenant aucune donnée sensible ; l'accès en lecture est basé sur l'identité. La connection TCP détourné n'est pas un problème. Ce scénario exige des fonctions d'authentification.

(3) un annuaire en lecture seul ne contenant aucune donnée sensible ; et le client doit s'assurer que les données de l'annuaire sont authentifiées par le serveur et pas modifiées jusqu'au retour au serveur.

(4) un annuaire en lecture/écriture, ne contenant aucune donnée sensible ; l'accès en lecture est disponible par "n'importe qui", accès à la mise à jour aux personnes correctement autorisées. La connection TCP détourné n'est pas actuellement un problème. Ce scénario exige des fonctions d'authentification.

(5) un annuaire contenant des données sensibles. Ce scénario exige une protection de confidentialité de session ET l'authentification.

3. Authentification and Autorisation: Definitions et Concepts

Cette section définit des termes de bases, les concepts, et des corrélations concernant l'authentification, l'autorisation, les qualifications, et l'identité. Ces concepts sont utilisés en décrivant comment de diverses approches de sécurité sont mises en place dans l'authentification et l'autorisation de client.

3.1 Contrôle d'accès

Une politique de contrôle d'accès est un ensemble de règles définissant la protection des ressources, généralement en termes de possibilités des personnes ou d'autres entités d'accéder à des ressources. Une expression commune d'une politique de contrôle d'accès est sous forme d'une Liste des Contrôle d'Accès. Les objets et les mécanismes de sécurité, comme ceux décrits ici, permettent de définir les politiques de contrôle d'accès . Les politiques de contrôle d'accès sont typiquement exprimées en termes d'attributs de contrôle d'accès comme décrites ci-dessous.

3.2. Facteurs de Contrôle d'accès

Une demande, quand elle est traitée par un serveur, peut être associée à une grande variété de facteurs de sécurité. Le serveur emploie ces facteurs pour déterminer comment traiter la demande. Ceux-ci s'appellent les facteurs de contrôle d'accès (ACFs). Ils pourraient inclure la source des adresses IP, la force de cryptage (en fonction de l'importance de la donnée), le type d'opération étant demandée, l'heure, etc... Certains facteurs peuvent être spécifiques à la demande elle-même, d'autres peuvent être associés à la connection par l'intermédiaire duquel la demande est transmise, d'autres (par exemple l'heure) peuvent être "environmentale".

Des politiques de contrôle d'accès sont exprimées en termes de facteurs de contrôle d'accès. Par exemple, une demande ayant ACFs i, j, k peut effectuer l'opération Y sur la ressource Z. L'ensemble d'ACFs qu'un serveur rend disponible pour de telles expressions dépend de la spécifité de l'implémentation (implémentation-spécific).

3.3 Authentification, Qualifications, Identité

L'authentification doit être assurée d'une partie à l'autre, affirmant l'identité de la première partie (par exemple un utilisateur) qui essaye d'établir un lien

avec l'autre partie (typiquement un serveur).
L'authentification est le processus de produire, de transmettre, et de vérifier les qualifications et ainsi l'identité qu'ils affirment.

Une identité d'authentification est le nom présenté dans un 'credential' (?) .

Il y a beaucoup de formes d'authentification -- la forme utilisée dépend du mécanisme particulier d'authentification communes aux parties. Par exemple : Certificats X.509, billets de Kerberos, identité simple et paires de mot de passe. Notez qu'un mécanisme d'authentification peut contraindre la forme d'identités d'authentification utilisées avec celui-ci.

3.4 Identité d'Autorisation

Une identité d'autorisation est un genre de facteur de contrôle d'accès. C'est le nom de l'utilisateur ou d'une autre entité qui demande que des opérations soit exécutée. Les politiques de contrôle d'accès sont souvent exprimées en termes d'identités d'autorisation ; par exemple, l'entité X peut effectuer l'opération Y sur la ressource Z.

La limite d'identité d'autorisation à une association est souvent identique à l'identité d'authentification présentée par le client, mais elle peut être différente. SASL permet aux utilisateurs d'indiquer une identité d'autorisation distincte de celle du client. Ceci permet à des agents tels que des serveurs proxy de s'authentifier en utilisant leurs propres paramètres, mais demande les privilèges d'accès de l'identité pour laquelle ils est en train de se procurer « proxying ». En outre, la forme d'identité d'authentification fournie par un service comme TLS peut ne pas correspondre aux identités d'autorisation employées pour exprimer la politique de contrôle d'accès d'un serveur, exigeant un « server-specific mapping » d'être fait. La méthode par laquelle un serveur compose et valide une autorisation des paramètres d'authentification fournies par un client est une spécifité d'implémentation.

4. Les mécanismes de sécurité

Il est clair que ne permettre aucune exécution, confrontée aux conditions ci-dessus, de sélectionner et choisir parmi les solutions de rechange possibles ne soit pas une stratégie qui est susceptible de mener à l'interopérabilité.
Les clients qui ne soutiennent aucune fonction de sécurité soutenue par le serveur, ou qui soutiennent seulement des mécanismes comme les mots de passe en clair ont la sécurité la plus insatisfaisante.

Il est plus difficile pour un hacker d'attaquer entre le serveur et le client. Les méthodes qui nous protègent contre les clients hostiles et les attaques d'écoute passives sont les meilleurs dans les situations où le coût de protection contre des attaques intermédiaires actives n'est

pas justifié.

Etant donné la présence de l'annuaire, il est prégérable de voir des mécanismes où les identités prennent la forme d'un Distinguished Name et des données d'authentification qui peuvent être stocké dans l'annuaire ; cela signifie que ces données sont inutiles pour tromper l'authentification (comme Unix le format de dossier "/etc/passwd"), ou son contenu n'est jamais passé à travers un chemin non protégé (wire unprotected) - c.-à-d., soit il est mis à jour en dehors du protocole soit il est seulement mis à jour en sessions bien protégées contre le snooping.
Il est également souhaitable de permettre à des méthodes d'authentification de pouvoir connaitre « des identités d'autorisation » basées sur les formes existantes d'identités d'utilisateur pour améliorer la compatibilité avec des services non-LDAP.

Par conséquent, les conditions suivantes de conformité d'exécution sont en mise en place :

(1)pour la lecture seule, l'annuaire public, authentification anonyme, décrite dans la section 5, peut être utilisé.

(2) (2) les réalisations fournissant l'accès authentifié par mot de passe DOIVENT supporter l'authentification utilisant le mécanisme de DIGEST-MD5 SASL , comme décrit dans la section 6.1. Ceci fournit à l'authentification du client la protection contre des attaques d'écoute passives, mais n'assure pas la protection contre des attaques intermédiaires actives.

(3)pour un annuaire ayant besoin de la protection et de l'authentification de session, les opérations étendues TLS [5], et le choix d'authentification simple ou le mécanisme d'external de SASL, doivent être utilisés ensemble. L'implémentations DEVRAIENT supporter l'authentification avec mot de passe comme décrit dans la section 6.2, et DEVRAIENT supporter l'authentification avec un certificat comme décrit dans la section 7.1. A eux-deux, ils peuvent assurer la protection d'intégrité et la non-révélation des données transmises, et l'authentification du client et du serveur, y compris la protection contre des attaques intermédiaires actives (entre le client et le serveur).

Si TLS est choisi, le client DOIT rejeter toutes les informations des recherches sur le serveur avant le choix de TLS. En particulier, la valeur des « supportedSASLMechanisms » PEUT être différente après que TLS ait été validé (spécifiquement, le mécanisme externe ou le mécanisme PLAT proposé sont susceptibles d'être énumérés seulement après que le choix de TLS ait été fait).

Si une couche de sécurité de SASL est choisit, le client DOIT rejeter de la même manière toutes les informations recherché avant SASL. En particulier, si le client est configuré pour supporter les mécanismes multiples de SASL, elle DEVRAIT chercher des supportedSASLMechanisms avant et après que la

couche de sécurité de SASL est été choisi et vérifier que
la valeur n'a pas changé après que la couche de sécurité de
SASL ait été mis en place. Ceci détecte les attaques actives
qui enlèvent les mécanismes soutenus de SASL des listes de
MechanismsSASLSupportés, et permettent au client de
s'assurer qu'elles emploient le meilleur mécanisme soutenu par le
client et le serveur .

5. Authentification Anonyme

Les opérations sur les annuaires qui modifient des entrées ou l'accès auxattributs protégés ou des entrées exigent généralement l'authentification de client.

Les clients qui n'ont l'intention d'effectuer aucune de ces
opérations utilisent le plus souvent l'authentificatio n anonyme.

L'implémentation LDAP DOIVENT supporter l'authentification
anonyme, comme défini dans la section 5.1.

L'implémentation LDAP DEVRAIT supporter l'authentification anonyme avec TLS, comme défini dans la section 5.2.

Tandis qu'il PEUT y avoir des
restrictions de contrôle d'accès pour empêcher l'accès aux
entrées de répertoire, un serveur LDAP DEVRAIT permettre
une connection anonyme au client pour rechercher l'attribut
« supportedSASLMechanisms » à la racine DSE.

Un serveur LDAP PEUT utiliser d'autres informations sur le client provenant d'une couche inférieure ou des moyens externes d'accorder ou nier l'accès même aux clients anonyme authentifiés.

5.1.Le procédé d'authentification anonyme

Un client de LDAP qui n'a pas réussi avec succès sa connection est authentifié en anonyme.

Un client de LDAP PEUT également spécifier l'authentification anonyme pour une requête en utilisant une CHAINE d'OCTET de longueur zéro avec le choix d'authentification.

5.2.Authentification anonyme et TLS

Un client de LDAP DEVRAIENT utiliser les opérations TLS [5] pour justifier l'utilisation de la sécurité de TLS [6]. Si le client n'est pas encore connecté, puis jusqu'à ce que le client emploie le mécanisme ecterne SASL pour vérifier l'identification du client, le client est authentifié en anonyme.

Des recommandations concernant des ciphersuites
de TLS sont données dans la section 10.

Un serveur LDAP qui demande que les clients fournissent leur certificat pendant la vérification TLS PEUT employer une politique locale de sécurité pour déterminer si il doit accomplir avec succès la vérification de TLS si le client ne présentait pas un certificat qui pourrait être validé.

6. Authentification par Mot de passe

L'implémentation LDAP DOIVENT supporter l'authentification avec mot de passe en utilisant le mécanisme de DIGEST-MD5 SASL comme défini dans la section 6.1.

L'implémentation LDAP DEVRAIENT supporter l'authentification avec le choix "simple" de mot de passe quand la connection est protégé contre le système TLS (d'écoute clandestine, lol), comme défini dans la section 6.2.

6.1.Authentification sommaire

Un client LDAP PEUT vérifier si le serveur supporte son mécanisme en exécutant une recherche sur la racine DSE, et en vérifiant si "DIGEST-MD5" est présent comme valeur de l'attribut de supportedSASLMechanisms.

A la première étape de l'authentification, quand le client effectue "une authentification initiale" comme défini dans la section 2.1 de [4], le client envoie une requête de connection dans laquelle le numéro de version est 3, le choix d'authentification est sasl, le nom de mécanisme de sasl est le "DIGEST-MD5", et les qualifications sont absentes. Le client attend alors une réponse du serveur à cette demande. Le serveur donnera une réponse avec un resultCode qui est saslBindInProgress, et le champ de serverSaslCreds est présent. Le contenu de ce champ est une chaîne défini par "digest-challenge" dans la section 2.2.1 de [4].Le serveur DEVRAIT inclure une 'realm' indication et DOIT indiquer le support d'UTF-8.

Le client enverra une requête avec une identification distincte de message, dans laquelle le numéro de version est 3, le choix d'authentification est sasl, le nom de mécanisme de sasl est "DIGEST-MD5", et les qualifications contiennent la corde définie par « digest-response » dans la section 2.1.2 de [4]. Le service-type est "ldap".

Le serveur répondra avec une requête de connection dans laquelle le resultCode est soit succès soit une indication sur l'erreur.

Si l'authentification est réussie et le serveur ne supportent pas l'authentification subsequent, alors le domaine des qualifications est absent. Si l'authentification est réussie et le serveur supportent l'authentification subsequent, alors le domaine de qualifications contient la chaine défini par "response-auth" dans la section 2.1.3

de [4]. Le support de l'authentification subsequent est FACULTATIF dans les clients et les serveurs.

6.2. choix d'authentification «simple » avec le crypatage TLS

Un utilisateur qui a une entrée d'annuaire qui contient un attribut « userPassword » DEVRAT s'authentifier à l'annuaire en exécutant une séquence de connection par mot de passe simple suivant le fonctionnement d'un ciphersuite TLS fournissant une connection confidentiel.

Le client utilisera l'opération TLS [5] pour justifier l'utilisation de la sécurité TLS [6] sur la connection au serveur LDAP. Le client n'a pas besoin d'avoir eu une connection à l'annuaire par avance.

Pour que ce procédé d'authentification soit réussi, le client et le serveur DOIVENT négocier (s'arranger) un ciphersuite qui contient un algorithme de crypatage appropriée. Des recommandations concernant cypher suites sont données dans la section 10.

Après l'accomplissement réussi de la négociation de TLS, le client DOIT envoyer une demande de connection LDAP avec le numéro de version de 3, la nom de domaine contenant le nom de l'entrée d'utilisateur, et le choix "simple" d'authentification, contenant le mot de passe.

Le serveur , pour chaque valeur de l'attribut d'userPassword dans l'entrée d'utilisateur appelé, compare ces derniers au mot de passe entré par le client. S'il n'y as pas de différence, alors le serveur répondra avec le resultCode « succès », autrement le serveur répondra par InvalidCredentials dans resultCode.

6.3. D'autres choix d'authentification avec TLS

Il est également possible, suivant le paramètrage de TLS, d'effectuer une authentification de SASL qui ne comporte pas l'échange des mots de passe réutilisables en text (plaintext). Dans ce cas-ci le client et le serveur n'auront pas besoin de se mettre d'accord sur un ciphersuite qui fournit la confidentialité si le seul service exigé est l'intégrité des données.

7. authentification basée sur les Certificats

Les réalisations de LDAP DEVRAIENT sur l'authentification par l'intermédiaire d'un certificat de client dans TLS, comme défini dans la section 7.1.

7.1. authentification basée sur les Certificat avec TLS

Un utilisateur qui a une paire de clef publique/privée dans laquelle la clef publique a été signée par une Autorité de Certification peut utiliser cette paire de clef pour s'authentifier au serveur d'annuaire si le certificat de l'utilisateur est demandé par le serveur. Le domaine du certificat de l'utilisateur DEVRAIT être le nom de l'entrée du répertoire d'utilisateur, et l'Autorité de Certification doit être suffisamment sure par le serveur d'annuaire pour que le serveur puisse traiter le certificat.
Le moyen par lequel les serveurs valident des chemins de certificat ne correspond pas au but de ce document.

Un serveur PEUT supporter le tracés (mapping) pour les certificats dans lesquels le nom de domaine est différent du nom de l'entrée du répertoire d'utilisateur. Un serveur qui supporte les tracés des noms DOIT être capable d'être configurer pour supporter les certificats pour lesquels aucun tracer n'est exigé.

Le client utilisera l'opération TLS [5] pour négocier l'utilisation de la sécurité de TLS [6] sur la connection au serveur LDAP. Le client n'a pas besoin de se connecter à l'annuaure préalablement.

Dans la négociation de TLS, le serveur DOIT demander un certificat. Le client fournira son certificat au serveur, et DOIT effectuer un crypatage de la clef privé.

Comme les déploiements exigeront la protection des données sensibles en transit, le client et le serveur DOIVENT négocier un ciphersuite qui contient un algorithme de crypatage en bloc de force appropriée (en fonction de l'importance des données). Des recommandations sur les cipher suites sont données dans la section 10.

Le serveur DOIT vérifier que le certificat du client est valide. Le serveur vérifiera normalement que le certificat est délivré par un CA connu, et qu'aucun des certificats sur la chaîne des certificats du client n'est inadmissible ou retiré. Il y a plusieurs procédures par lesquelles le serveur peut exécuter ces contrôles.

Après l'accomplissement réussi de la négociation de TLS, le client enverra une requête de connection LDAP avec mécanisme de SASL "externe".

8. D'autres mécanismes

Le choix d'authentification LDAP "simple" n'est pas approprié à l'authentification sur l'Internet où il n'y a aucune confidentialité sur la couche transport.

Comme LDAP inclut des méthodes anonymes et d'authentification en plaintext, les mécanismes "ANONYMES" et "PLATS"

de SASL ne sont pas employés avec LDAP. Si une identité d'autorisation d'une forme différente d'un DN est demandée par le client, un mécanisme qui protège le mot de passe en transit DEVRAIT être employé.

Les mécanismes SASL suivants ne sont pas considérés dans ce document : KERBEROS_V4, GSSAPI et SKEY.

Le mécanisme de SASL "externe" peut être utilisé pour demander au serveur LDAP de se servir des paramètres de sécurité échangées par une couche inférieure. Si une session TLS n'a pas été établie entre le client et le serveur avant de faire la connection externe SASL et si il n'y a aucune autre source extérieure de paramètrage d'authentification (par exemple sécurité d'IP-level [8]), ou si, pendant le processus d'établissement de la session TLS, le serveur ne demandait pas les paramètres de l'authentification du client, la connection externe SASL DOIT échouer avec un code résultat « inappropriateAuthentication ». Aucune authentification client et aucun état d'autorisation l'associationLDAP est perdu, ainsi l'association de LDAP est dans un état anonyme après l'échec.

9. Identité d'Autorisation

L'identité d'autorisation est une part du domaine de qualifications de SASL et dans les requêtes LDAP (demande et la réponse).

Quand le mécanisme "externe" est en train d'être choisi, si le domaine de qualifications est présent, il contient une identité d'autorisation de la forme d'authzId décrite ci-dessous.

D'autres mécanismes définissent l'endroit de l'identité d'autorisation dans le domaine de qualifications.

L'identité d'autorisation est une chaine dans le jeu de caractères UTF-8, correspondant à l'ABNF suivant [7] :

```
; Specific predefined authorization (authz) id schemes are
; defined below -- new schemes may be defined in the future.

authzId    = dnAuthzId / uAuthzId

; distinguished-name-based authz id.
dnAuthzId  = "dn:" dn
dn         = utf8string    ; with syntax defined in RFC 2253

; unspecified userid, UTF-8 encoded.
uAuthzId   = "u:" userid
userid     = utf8string    ; syntax unspecified
```

Un utf8string est défini pour être le codage UTF-8

d'un ou plusieurs caractères d'ISO 10646.

Tous les serveurs qui supportent le stockage des qualifications d'authentification, telles que des mots de passe ou des certificats, dans l'annuaire DOIVENT supporter le choix de dnAuthzId.

Le choix d'uAuthzId tient compte de la compatibilité avec les applications clientes qui souhaitent authentifier à un annuaire local mais ne connaissent pas leur propre Distinguished Name ou ont une entrée de répertoire. Le format de la chaine est défini comme une seul séquence de caractères UTF-8 codé en ISO 10646, et davantage d'interprétation est sujette à l'accord antérieur entre le client et le serveur.

Par exemple, l'identification de l'utilisateur (userid) pourrait identifier un utilisateur d'un service spécifique d'annuaire, ou soit un nom de login ou la partie locale d'une adresse mail . En général , uAuthzId NE DOIT PAS etre unique.

Des arrangements additionnels d'identité d'autorisation PEUVENT être définis dans de futures versions de ce document.

10.TLS Cypher Suites

Les cypher suites suivantes définies dans la section 6 NE DOIVENT PAS être utilisé pour une protection confidentiel de mot de passe ou de données:

TLS_NULL_WITH_NULL_NULL
TLS_RSA_WITH_NULL_MD5
TLS_RSA_WITH_NULL_SHA

Les Cypher suites suivantes définies dans la section 6 peuvent être cracké facilement (moins d'une semaine de travail d'un processeur dattant de 1997). Le client et le serveur DEVRONT considérés leurs mots de passes et données étant protégées avant d'utiliser ces Cypher Suites:

TLS_RSA_EXPORT_WITH_RC4_40_MD5
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5
TLS_RSA_EXPORT_WITH_DES40_CBC_SHA
TLS_DH_DSS_EXPORT_WITH_DES40_CBC_SHA
TLS_DH_RSA_EXPORT_WITH_DES40_CBC_SHA
TLS_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA
TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
TLS_DH_anon_EXPORT_WITH_RC4_40_MD5
TLS_DH_anon_EXPORT_WITH_DES40_CBC_SHA

Ces Cypher Suites suivantes sont vulnérables aux attaques, et ne DEVRAIENT PAS être utilisé pour protéger les mots de passes et données sensibles, à moins que la configuration du réseau souhaite laisser un risque que certaines attaques soient tolérer.

TLS_DH_anon_EXPORT_WITH_RC4_40_MD5
TLS_DH_anon_WITH_RC4_128_MD5
TLS_DH_anon_EXPORT_WITH_DES40_CBC_SHA
TLS_DH_anon_WITH_DES_CBC_SHA
TLS_DH_anon_WITH_3DES_EDE_CBC_SHA

Un client ou un serveur qui supporte TLS doit au moins supporter TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA.

11.Service SASL pour LDAP

Pour l'utiliser avec SASL, le protocole doit spécifier un nom de service qui serat utiliser avec les différents mécanismes SASL, comme par exemple GSSAPI. Pour LDAP, le nom de service est « ldap », ce dernier à été enregistré à l'IANA comme un nom de service GSSAPI.

12.Sécurité

Les issues de sécurité ont été définient tout au long de cette documentation; la conlusion que nous pouvons en tirer est que la sécurité n'est vraiment pas à négliger, le crypatage des sessions est quasiment exigeable pour éviter certains problèmes (snooping).

Il est préférable (meme tres fortement conseillé) que les sereurs évitent les modifications par les utilisateurs anonymes.

Une connection sur lequelle le client n'as pas éxécuté une opération TLS ou négocier un mécanisme SASL approprié et les services ne sont pas cryptés peut être sujet à des attaques pour voir et modifier les informations qui y transitent.

Les sécurités additionnelles concernant les mécanismes TLS peuvent être trouvé dans la section 2, 5 et 6.