Groupe de travail Réseau
Request for Comments: 4422
RFC rendue obsolète : 2222
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

A. Melnikov, éd., Isode Limited
K. Zeilenga, éd., OpenLDAP Foundation

juin 2006

juin 2008

Authentification simple et couche de sécurité (SASL)

Statut du présent mémoire
Le présent document spécifie un protocole Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et des suggestions d'amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé
Authentification simple et couche de sécurité (SASL, *Simple Authentication and Security Layer*) est un cadre de travail pour la fourniture de services d'authentification et de sécurité des données dans les protocoles orientés connexion via des mécanismes remplaçables. Il fournit une interface structurée entre protocole et mécanismes. Le cadre qui en résulte permet à de nouveaux protocoles de réutiliser les mécanismes existants et permet aux anciens protocoles de faire usage des nouveaux mécanismes. Le cadre permet aussi un protocole pour la sécurisation des échanges de protocole ultérieurs au sein d'une couche de sécurité des données.

Le présent document décrit comment est structuré un mécanisme SASL, comment les protocoles incluent la prise en charge de SASL, et le protocole de portage d'une couche de sécurité des données sur une connexion. De plus, le présent document définit un mécanisme SASL, le mécanisme EXTERNAL.

Le présent document rend obsolète la RFC 2222.

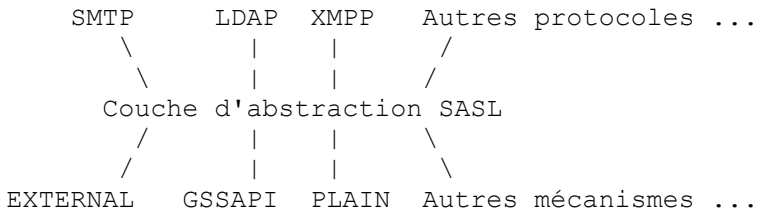
Table des Matières

1. Introduction

L'authentification simple et couche de sécurité (SASL) est un cadre de travail pour la fourniture de services d'authentification et de sécurité des données dans des protocoles orientés connexion via des mécanismes de remplacement. SASL fournit une interface structurée entre les protocoles et les mécanismes. SASL fournit aussi un protocole pour la sécurisation des échanges de protocole ultérieurs au sein d'une couche de sécurité des données. La couche de sécurité des données peut fournir la protection de l'intégrité des données, de la confidentialité des données, et d'autres services.

La conception de SASL est destinée à permettre que de nouveaux protocoles réutilisent les mécanismes existants sans exiger une nouvelle conception des mécanismes et à permettre aux protocoles existants de faire usage de nouveaux mécanismes sans changer la conception des protocoles.

SASL est conceptuellement un cadre de travail qui fournit une couche d'abstraction entre les protocoles et les mécanismes comme illustré par le diagramme suivant.



C'est à travers les interfaces de cette couche d'abstraction que le cadre de travail permet à tout protocole d'utiliser tout mécanisme. Bien que cette couche cache généralement les particularités des protocoles aux mécanismes et les particularités des mécanismes aux protocoles, cette couche ne cache généralement pas les particularités des mécanismes aux mises en œuvre de protocole. Par exemple, des mécanismes différents exigent des informations différentes pour fonctionner, certains d'entre eux utilisent l'authentification fondée sur le mot de passe, certains autres exigent des informations de domaine, d'autres utilisent des tickets Kerberos, des certificats, etc. Aussi, afin de d'effectuer l'autorisation, les mises en œuvre de serveur ont généralement à effectuer une transposition d'identité entre les identités d'authentification, dont la forme est spécifique du mécanisme, et les identités d'autorisation, dont la forme est spécifique du protocole d'application. La Section 2 discute des concepts d'identité.

Il est possible de concevoir et mettre en œuvre ce cadre de travail de façon à faire abstraction des particularités de mécanismes similaires. De telles mises en œuvre du cadre de travail, aussi bien que les mises en œuvre des mécanismes, pourraient être conçues non seulement pour être partagées par de multiples mises en œuvre d'un protocole particulier mais aussi pour être partagées par les mises en œuvre de plusieurs protocoles.

Le cadre de travail incorpore des interfaces avec des protocoles et des mécanismes dans lesquels les échanges d'authentification sont pris en charge. La Section 3 discute des échanges d'authentification SASL.

Pour utiliser SASL, chaque protocole fournit (entre autres choses) une méthode d'identification du mécanisme à utiliser, une méthode d'échange de défis du serveur et des réponses client spécifique du mécanisme, et une méthode pour communiquer le résultat de l'échange d'authentification. La Section 4 discute des exigences de protocole de SASL.

Chaque mécanisme SASL définit (entre autres choses) une série de défis du serveur et de réponses du client qui fournit des services d'authentification et négocie les services de sécurité des données. La Section 5 discute des exigences de mécanisme SASL. La Section 6 discute des considérations de sécurité. La Section 7 discute des considérations relatives à l'IANA. L'Appendice A définit le mécanisme SASL EXTERNAL.

1.1. Public visé par le document

Le présent document a été écrit pour servir plusieurs publics différents :
- les concepteurs de protocole qui utilisent cette spécification pour la prise en charge de l'authentification dans leur protocole,
- les concepteurs de mécanismes qui définissent de nouveaux mécanismes SASL,
- les développeurs de clients ou serveurs pour les protocoles qui prennent en charge SASL.

Bien que l'organisation du document soit destinée à permettre au lecteur de se concentrer sur les détails pertinents pour leur domaine, les lecteurs sont invités à lire et comprendre tous les aspects de ce document.

1.2. Relations avec d'autres documents

Le présent document rend obsolète la RFC 2222. Il remplace toutes les portions de la RFC 2222 sauf les paragraphes 7.1 (mécanisme KERBEROS_IV), 7.2 (mécanisme GSSAPI), 7.3 (mécanisme SKEY). Les mécanismes KERBEROS_IV et SKEY sont maintenant considérés comme obsolètes et leur spécification fournie dans la RFC 2222 a un caractère historique. Le mécanisme GSSAPI est maintenant spécifié séparément par la [RFC 4752].

L'appendice B fournit un résumé des changements depuis la RFC 2222.

1.3. Conventions

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" dans le présent document sont à interpréter comme décrit dans le BCP 14 [RFC2119].

Les noms de caractères dans le présent document utilisent la notation des codets et des noms tirés de la norme Unicode [Unicode]. Par exemple, la lettre "a" peut être représentée par <U+0061> ou par <LATIN SMALL LETTER A>.

Note : un glossaire des termes utilisés dans Unicode se trouve dans [Glossary]. On peut trouver les informations sur le modèle de codage de caractère Unicode dans [CharModel].

Dans les exemples, "C:" et "S:" indiquent les lignes de données envoyées respectivement par le client et le serveur. Les lignes font des retours à la ligne pour améliorer la lisibilité.

2. Concepts d'identité

En pratique, authentification et autorisation peuvent impliquer plusieurs identités, éventuellement sous différentes formes (simple nom d'utilisateur, principal Kerberos, nom distinctif X.500, etc.) éventuellement avec différentes représentations (par exemple, une chaîne de caractères Unicode codée en UTF-8 décrite en ABNF, nom distinctif codé en BER). Bien que les spécifications techniques prescrivent souvent à la fois la forme d'identité et la représentation utilisée dans le réseau, différentes formes et/ou représentations d'identité peuvent être (et sont souvent) utilisées dans les mises en œuvre. Comment des identités de différentes formes se rapportent les unes aux autres est, en général, une affaire locale. De plus, les formes et représentations utilisées dans une mise en œuvre sont une affaire locale.

Cependant, conceptuellement, le cadre de travail SASL implique deux identités :
1) une identité associée aux accréditifs d'authentification (appelée identité d'authentification), et
2) une identité pour agir (appelée identité d'autorisation).

Les spécifications de mécanisme SASL décrivent la ou les formes d'accréditifs (par exemple, certificats X.509, tickets Kerberos, simple nom d'utilisateur/mot de passe) utilisées pour authentifier le client, incluant (lorsque approprié) la syntaxe et la sémantique des identités d'authentification portées dans les accréditifs. Les spécifications du protocole SASL décrivent la ou les formes d'identité utilisées dans l'autorisation et, en particulier, prescrivent la syntaxe et la sémantique de la chaîne de caractères d'identité d'autorisation à transférer par les mécanismes.

Le client fournit ses accréditifs (qui incluent ou impliquent une identité d'authentification) et, facultativement, une chaîne de caractères représentant l'identité d'autorisation demandée au titre de l'échange SASL. Lorsque cette chaîne de caractères est omise ou vide, le client demande à agir comme l'identité associée aux accréditifs (par exemple, l'usager demande à agir comme l'identité d'authentification).

Le serveur est chargé de vérifier les accréditifs du client et de vérifier que l'identité qu'il associe aux accréditifs du client

(par exemple, l'identité d'authentification) est autorisée à agir comme identité d'autorisation. Un échange SASL échoue si l'une ou/et l'autre de ces vérifications échoue. (L'échange SASL peut échouer pour d'autres raisons, comme un échec d'autorisation de service.)

Cependant, la ou les formes précises des identités d'authentification (utilisées au sein du serveur dans ses vérifications, ou autrement) et la ou les formes précises des identités d'autorisation (utilisées pour prendre les décisions d'autorisation, ou autrement) sortent du domaine d'application de SASL et de la présente spécification. Dans certaines circonstances, les formes d'identité précises utilisées dans des contextes en dehors de l'échange SASL peuvent être dictées par d'autres spécifications. Par exemple, une spécification de politique d'autorisation par hypothèses d'identité (autorisation mandataire) peut dicter comment les identités d'authentification et d'autorisation sont représentées dans les déclarations de politique.

3. L'échange d'authentification

Chaque échange d'authentification consiste en un message du client au serveur demandant l'authentification via un mécanisme particulier, suivi d'une ou plusieurs paires de défis provenant du serveur et de réponses du client, suivies d'un message du serveur indiquant le résultat de l'échange d'authentification. (Noter que les échanges peuvent aussi être interrompus comme exposé au paragraphe 3.5.)

L'illustration suivante fournit une vue d'ensemble d'échange d'authentification.

C: Demande l'échange d'authentification
S: Défi initial
C: Réponse initiale
<messages supplémentaires de défi /réponse>
S: Résultat de l'échange d'authentification

Si le résultat est une réussite et si une couche de sécurité a été négociée, cette couche est alors installée (paragraphe 3.7). Cela s'applique aussi aux illustrations suivantes.

Certains mécanismes spécifient que les premières données envoyées dans l'échange d'authentification sont du client au serveur. Les protocoles peuvent fournir un champ facultatif de réponse initiale dans le message de demande pour porter ces données. Si le mécanisme spécifie que les premières données envoyées dans l'échange sont du client au serveur, le protocole fournit un champ facultatif de réponse initiale, et le client utilise ce champ, l'échange est alors abrégé d'un aller-retour :

C: Demande d'échange d'authentification + réponse initiale
<messages supplémentaires de défi/réponse>
S: Résultat de l'échange d'authentification

Si le mécanisme spécifie que les premières données envoyées dans l'échange sont du client au serveur et si ce champ est indisponible ou inutilisé, la demande du client est suivie d'un défi vide.

C: Demande d'échange d'authentification
S: Défi vide
C: Réponse initiale
<messages supplémentaires de défi/réponse>
S: Résultat de l'échange d'authentification

Si un client incluait une réponse initiale dans sa demande alors que le mécanisme ne permet pas au client d'envoyer d'abord les données, l'échange d'authentification échoue.

Certains mécanismes spécifient que le serveur va envoyer des données supplémentaires au client lorsque il indique un résultat réussi. Les protocoles peuvent fournir un champ facultatif de données supplémentaires dans le message de résultat pour porter ces données. Si le mécanisme spécifie que le serveur va retourner des données supplémentaires avec le résultat de succès, le protocole fournit un champ facultatif de données supplémentaires dans le message de résultat, et le serveur utilise ce champ, l'échange est abrégé d'un aller-retour:

C: Demande d'échange d'authentification
S: Défi initial
C: Réponse initiale
<messages supplémentaires de défi/réponse>

S: Résultat de l'échange d'authentification avec des données supplémentaires avec succès

Si le mécanisme spécifie que le serveur va retourner des données supplémentaires au client avec un résultat de succès et si ce champ est indisponible ou inutilisé, les données supplémentaires sont envoyées comme un défi dont la réponse est vide. Après avoir reçu cette réponse, le serveur indique alors le résultat réussi.

C: Demande d'échange d'authentification
S: Défi initial
C: Réponse initiale
<messages supplémentaires de défi/réponse>
S: Données supplémentaires de défi
C: Réponse vide
S: Résultat de l'échange d'authentification

Si les mécanismes spécifient que les premières données envoyées dans l'échange sont du client au serveur et si des données supplémentaires sont envoyées au client avec l'indication d'un résultat de réussite, et si le protocole fournit des champs qui prennent les deux en charge, l'échange prend alors deux allers-retours de moins :

C: Demande d'échange d'authentification + réponse initiale
<messages supplémentaires de défi/réponse>
S: Résultat d'échange d'authentification avec des données supplémentaires de succès

au lieu de :

C: Demande d'échange d'authentification
S: Défi vide
C: Réponse initiale
<messages supplémentaires de défi/réponse>
S: Données supplémentaires de défi
C: Réponse vide
S: Résultat de l'échange d'authentification

3.1. Dénomination des mécanismes

Les mécanismes SASL sont nommés par des chaînes de caractères, longues de 1 à 20 caractères, consistant en majuscules ASCII [ASCII], chiffres, traits d'union, et/ou soulignés. Dans la grammaire suivante en forme Backus-Naur augmenté (ABNF) [RFC4234], la production <sasl-mech> définit la syntaxe d'un nom de mécanisme SASL.

```
sasl-mech    = 1*20mech-char
mech-char    = UPPER-ALPHA / DIGIT / HYPHEN / UNDERSCORE
; mech-char est restreint à A-Z (seulement des majuscules), 0-9, -, et _ du jeu de caractères ASCII.
UPPER-ALPHA  = %x41-5A  ; A-Z (seulement des majuscules)
DIGIT        = %x30-39  ; 0-9
HYPHEN       = %x2D ; trait d'union (-)
UNDERSCORE   = %x5F ; souligné (_)
```

Les noms du mécanisme SASL sont enregistrés comme exposé au paragaphe 7.1.

3.2. Négociation de mécanisme

La négociation du mécanismes est spécifique du protocole.

Généralement, un protocole va spécifier que le serveur annonce les mécanismes pris en charge et disponibles pour le client via des facilités fournies par le protocole, et le client va alors choisir le "meilleur" mécanisme dans cette liste qu'il prend en charge et trouve convenable.

Noter que la négociation du mécanisme n'est pas protégée par l'échange d'authentification suivant et est donc sujette à des attaques en dégradation si elle n'est pas protégée par d'autres moyens.

Pour détecter les attaques en dégradation, un protocole peut permettre au client de découvrir les mécanismes disponibles suite à l'échange d'authentification et l'installation d'une couche de sécurité des données avec au moins la protection de l'intégrité des données. Cela permet au client de détecter les changements à la liste des mécanismes acceptés par le serveur.

3.3. Échange de demandes d'authentification

L'échange d'authentification est initié par le client qui demande l'authentification via un mécanisme qu'il spécifie. Le client envoie un message qui contient le nom du mécanisme au serveur. Les particularités du message sont spécifiques du protocole.

Noter que le nom du mécanisme n'est pas protégé par le mécanisme, et est donc sujet à altération par un attaquant si il n'est pas protégé en intégrité par d'autres moyens.

Si le mécanisme est défini comme permettant au client d'envoyer d'abord les données, et si le message de demande du protocole inclut un champ facultatif de réponse initiale, le client peut inclure la réponse au défi initial dans le message de demande d'authentification .

3.4. Défis et réponses

L'échange d'authentification implique une ou plusieurs paires de défis du serveur et de réponses du client, dont les particularités sont spécifiques du mécanisme. Ces défis et réponses sont inclus dans les messages du protocole, dont les particularités sont spécifiques du protocole.

Par ces défis et réponses, le mécanisme peut :
- authentifier le client auprès du serveur,
- authentifier le serveur auprès du client,
- transférer une chaîne d'identité d'autorisation,
- négocier une couche de sécurité, et
- fournir d'autres services.

La négociation de la couche de sécurité peut implique la négociation des services de sécurité à fournir dans la couche, comment ces services seront fournis, et la négociation d'une taille maximum de mémoire tampon de texte chiffré que chaque côté est capable de recevoir dans la couche (voir le paragraphe 3.6).

Après la réception d'une demande d'authentification ou de toute réponse de client, le serveur peut produire un défi, interrompre l'échange, ou indiquer le résultat d'un échange. Après la réception d'un défi, un mécanisme de client peut produire une réponse ou interrompre l'échange.

3.4.1. Chaîne d'identité d'autorisation

La chaîne d'identité d'autorisation est une séquence de zéro, un ou plusieurs caractères Unicode [Unicode], à l'exclusion du caractère NUL (U+0000), qui représente l'identité comme laquelle agir.

Si la chaîne d'identité d'autorisation est absente, il est demandé au client d'agir comme l'identité que le serveur associe aux accréditifs du client. Une chaîne vide est équivalente à une identité d'autorisation absente.

Une chaîne d'identité d'autorisation non vide indique que le client souhaite agir comme l'identité représentée par la chaîne. Dans ce cas, la forme de l'identité représentée par la chaîne, ainsi que la syntaxe et la sémantique précise de la chaîne sont spécifiques du protocole.

Bien que le schéma de codage de caractères utilisé pour transférer la chaîne d'identité d'autorisation dans l'échange d'authentification soit spécifique du mécanisme, les mécanismes sont supposés être capables de porter le répertoire Unicode entier (à l'exception du caractère NUL).

3.5. Interruption d'échange d'authentification

Un client ou serveur peut désirer interrompre un échange d'authentification si il ne veut pas ou n'est pas capable de continuer (ou commence à l'être).

Un client peut interrompre l'échange d'authentification en envoyant au serveur un message, dont les particularités sont spécifiques du protocole, indiquant que l'échange est interrompu. Le serveur peut être obligé par le protocole de retourner un message en réponse au message d'interruption du client.

De même, un serveur peut interrompre l'échange d'authentification en envoyant au client un message, dont les

particularités sont spécifiques du protocole, indiquant que l'échange est interrompu.

3.6. Résultat d'authentification

À la conclusion de l'échange d'authentification, le serveur envoie au client un message, dont les particularités sont spécifiques du protocole, indiquant le résultat de l'échange.

Le résultat n'est pas un succès si :
- l'échange d'authentification a échoué pour toute raison,
- les accréditifs du client n'ont pas pu être vérifiés,
- le serveur ne peut pas associer une identité aux accréditifs du client,
- la chaîne d'identité d'autorisation fournie par le client est mal formée,
- l'identité associée aux accréditifs du client n'est pas autorisée à agir comme identité d'autorisation demandée,
- la couche de sécurité négociée (ou son absence) ne convient pas, ou
- le serveur ne veut pas fournir de service au client pour une raison quelconque.

Le protocole peut inclure un champ facultatif de données supplémentaires dans ce message de résultat. Ce champ peut seulement inclure des données supplémentaires lorsque le résultat est un succès.

Si le résultat est un succès et si une couche de sécurité a été négociée, cette couche est alors installée. Si le résultat est un échec, ou si une couche de sécurité n'a pas été négociée, toute sécurité existante reste en place.

Le message de résultat fourni par le serveur peut donner au client le moyen de distinguer les erreurs qui seront mieux traitées en invitant l'usager à présenter à nouveau ses accréditifs, les erreurs qui seront mieux traitées en disant à l'usager de réessayer plus tard, et les erreurs où l'usager doit contacter un administrateur système pour les résoudre (voir à titre d'exemple la spécification des codes de réponse POP SYS et AUTH [RFC3206]). Cette distinction est particulièrement utile durant les périodes programmées de maintenance du serveur car cela réduit les coûts de prise en charge. Il est aussi important que le serveur puisse être configuré de façon telle que le message de résultat ne fasse pas de distinction entre un usager valide avec des accréditifs invalides et un usager invalide.

3.7 Couches de sécurité

Les mécanismes SASL peuvent offrir une large gamme de services dans les couches de sécurité. Les services typiques incluent l'intégrité des données et la confidentialité des données. Les mécanismes SASL qui ne fournissent pas de couche de sécurité sont traités comme la négociation de pas de couche de sécurité.

Si l'utilisation d'une couche de sécurité est négociée dans l'échange de protocole d'authentification, la couche est installée par le serveur après avoir indiqué le résultat de l'échange d'authentification, et installée par le client à réception de l'indication de résultat. Dans les deux cas, la couche est installée avant le transfert d'autres données de protocole. La position précise à laquelle la couche prend effet dans le flux de données de protocole est spécifique du protocole.

Une fois que la couche de sécurité est effective dans le flux des données de protocole, elle reste en effet jusqu'à ce qu'une couche de sécurité négociée ultérieurement soit installée ou que la connexion de transport sous-jacente soit close.

Quand elle est en effet, la couche de sécurité traite les données de protocole dans des mémoires tampon de données protégées. Si à un moment donné, la couche de sécurité est incapable de continuer, ou ne veut plus continuer à produire des mémoires tampon de protection des données de protocole, la connexion de transport sous-jacente DOIT être close. Si la couche de sécurité n'est pas capable de décoder une mémoire tampon reçue, la connexion sous-jacente DOIT être close. Dans les deux cas, la connexion de transport sous-jacente DEVRAIT être close en douceur.

Chaque mémoire tampon de données protégées est transférée sur la connexion de transport sous-jacente comme une séquence d'octets précédée d'un champ de quatre octets dans l'ordre des octets du réseau qui représente la longueur de la mémoire tampon. La longueur de la mémoire tampon de données protégées DOIT être pas plus grande que la taille maximum qu'attend l'autre côté. À réception d'un champ Longueur dont la valeur est supérieure à la taille maximum, le receveur DEVRAIT clore la connexion, car ce pourrait être le signe d'une attaque.

La taille maximum que chaque côté attend est fixée par le mécanisme, soit par négociation, soit par sa spécification.

3.8. Authentifications multiples

Sauf explicitement permis par le protocole (comme établi par la spécification technique du protocole) un seul échange d'authentification SASL réussi peut se produire dans une session de protocole. Dans ce cas, une fois l'échange

d'authentification achevé avec succès, les autres tentatives d'initier un échange d'authentification échouent.

Lorsque plusieurs échanges d'authentification SASL réussis sont permis dans le protocole, cela ne signifie en aucun cas que plusieurs couches de sécurité SASL peuvent être simultanément en effet. Si une couche de sécurité est en effet et qu'une négociation ultérieure choisit une seconde couche de sécurité, la seconde couche de sécurité remplace alors la première. Si une couche de sécurité est en effet et si une négociation SASL ultérieure ne choisit pas de couche de sécurité, la couche de sécurité d'origine reste en effet.

Lorsque plusieurs négociations SASL réussies sont permises dans le protocole, l'effet d'un échec d'échange d'authentification SASL sur l'état d'authentification et d'autorisation antérieurement établi est spécifique du protocole. La spécification technique du protocole devrait être consultée pour déterminer si l'état d'authentification et d'autorisation antérieur reste en effet, ou s'est changé en un état anonyme, ou a été affecté autrement. Sans considération de l'effet spécifique du protocole sur l'état antérieurement établi d'authentification et d'autorisation, la couche de sécurité négociée précédemment reste en effet.

4. Exigences du protocole

Pour qu'un protocole offre les services SASL, sa spécification DOIT fournir les informations suivantes :

1) Un nom de service, à choisir dans le registre des éléments "service" pour la forme de nom de service fondé sur l'hôte d'interface de programme d'application de service générique de sécurité (GSSAPI, *Generic Security Service Application Program Interface*) comme décrit au paragraphe 4.1 de la [RFC2743]. Noter que ce registre est partagé par toutes les GSSAPI et tous les mécanismes SASL.

2) Détailler toute facilité de négociation de mécanisme que le protocole fournit (paragraphe 3.2).
   Un protocole DEVRAIT spécifier une facilité par laquelle le client puisse découvrir, à la fois avant l'initiation de l'échange SASL et après l'installation de la couche de sécurités négociée par l'échange, les noms des mécanismes SASL que le serveur met à la disposition du client. Cela est important pour permettre au client de détecter les attaques en dégradation. Cette facilité est normalement fournie à travers les extensions ou la facilité de découverte de capacités du protocole.

3) Définir les messages nécessaires pour l'échange d'authentification, incluant les suivants :
   a) Un message pour initier l'échange d'authentification (paragraphe 3.3).
   Ce message DOIT contenir un champ pour porter le nom du mécanisme choisi par le client.
   Ce message DEVRAIT contenir un champ facultatif pour porter une réponse initiale. Si le message est défini avec ce champ, la spécification DOIT décrire comment les messages avec une réponse initiale vide sont distingués des messages sans réponse initiale. Ce champ DOIT être capable de porter des séquences arbitraires d'octets (incluant des séquences de longueur zéro et des séquences contenant des octets de valeur zéro).
   b) Des messages pour transférer les défis du serveur et les réponses du client (paragraphe 3.4).
   Chacun de ces messages DOIT être capable de porter des séquences arbitraires d'octets (incluant des séquences de longueur zéro et des séquences contenant des octets de valeur zéro).
   c) Un message pour indiquer le résultat de l'échange d'authentification (paragraphe 3.6).
   Ce message DEVRAIT contenir un champ facultatif pour porter des données supplémentaires avec un résultat de succès. Si le message est défini avec ce champ, la spécification DOIT décrire comment les messages avec des données supplémentaires vides sont distinguées des messages sans données supplémentaires. Ce champ DOIT être capable de porter des séquences arbitraires d'octets (incluant des séquences de longueur zéro et des séquences contenant des octets de valeur zéro).

4) Prescrire la syntaxe et la sémantique des chaînes d'identité d'autorisation non vides (paragraphe 3.4.1).
   Pour éviter les problème d'interopérabilité dus à des normalisations différentes, la spécification du protocole DOIT détailler précisément comment et où (chez le client ou chez le serveur) sont préparées les chaînes d'identité d'autorisation non vides, incluant toute normalisation, pour comparaison et autres fonctions applicables pour assurer un fonctionnement approprié.
   Les spécifications sont encouragées à prescrire l'utilisation des formulaires existants d'identité d'autorisation ainsi que les représentations de chaîne existantes, comme les noms de simple utilisateur [RFC4013].
   Si la spécification ne prescrit pas de façon précise comment les identités dans SASL se rapportent aux identités utilisées ailleurs dans le protocole, par exemple, dans les déclarations de politique de contrôle d'accès, il peut être approprié que le protocole fournisse une facilité par laquelle le client puisse découvrir les informations (comme la représentation de l'identité utilisée pour la prise des décisions de contrôle d'accès) sur les identités établies pour ces usages.

5) Détailler toutes les facilités que le protocole fournit qui permettent au client et/ou au serveur d'interrompre l'échange

d'authentification (paragraphe 3.5).
Les protocoles qui acceptent plusieurs authentifications permettent normalement au client d'interrompre un échange d'authentification en cours en initiant un nouvel échange d'authentification. Les protocoles qui n'acceptent pas plusieurs authentifications peuvent exiger que le client close la connexion et recommence pour interrompre un échange d'authentification en cours.
Les protocoles permettent normalement au serveur d'interrompre un échanges d'authentification en cours en retournant un message de résultat d'échec.

6) Identifier précisément où une couche de sécurité nouvellement négociée commence à prendre effet, dans les deux directions (paragraphe 3.7).
Normalement, les spécifications exigent qu'une couche de sécurité commence à prendre effet sur le premier octet qui suit le message de résultat pour ce qui concerne les données envoyées par le serveur et sur le premier octet envoyé après réception du message de résultat pour ce qui concerne les données envoyées par le client.

7) Si le protocole prend en charge d'autres services de sécurité en couches, comme la sécurité de la couche transport (TLS, *Transport Layer Security*) [RFC4346], la spécification DOIT prescrire l'ordre dans lequel les couches de sécurité sont appliquées aux données de protocole.
Par exemple, lorsque un protocole prend en charge à la fois les couches de sécurité TLS et SASL, la spécification pourrait prescrire que :
a) une couche de sécurité SASL est toujours appliquée d'abord aux données envoyées, et donc appliquée ensuite aux données reçues ;
b) une couche de sécurité SASL est toujours appliquée en dernier aux données envoyées et donc appliquée en premier aux données reçues ;
c) les couches sont appliquées dans l'ordre dans lequel elles ont été installées ;
d) les couches sont appliquées dans l'ordre inverse de celui dans lequel elles ont été installées ; ou
e) les couches de sécurité TLS et SASL ne peuvent pas être toutes deux installées.

8) Indiquer si le protocole accepte des authentifications multiples (paragraphe 3.8). Si il les accepte, le protocole DOIT détailler l'effet d'un échec d'échange d'authentification SASL sur un état antérieurement établi d'authentification et d'autorisation.

Les spécifications de protocole DEVRAIENT éviter de déclarer des exigences de mise en œuvre qui entraveraient le remplacement des mécanismes applicables. En général, les spécifications de protocole DEVRAIENT être neutres à l'égard des mécanismes. Il y a un certain nombre d'exceptions raisonnables à cette recommandation, en autres :
- de préciser comment les accréditifs (qui sont spécifiques du mécanisme) sont gérés dans le protocole,
- de préciser comment les identités d'authentification (qui sont spécifiques du mécanisme) et d'autorisation (qui sont spécifiques du protocole) se relatent l'une à l'autre, et
- de préciser quels mécanismes sont applicables au protocole.

5. Exigences du mécanisme

Les spécifications de mécanisme SASL DOIVENT fournir les informations suivantes :

1) Le nom du mécanisme (paragraphe 3.1). Ce nom DOIT être enregistré comme exposé au paragraphe 7.1.

2) Une définition des défis du serveur et des réponses du client à l'échange d'authentification, ainsi que :
   a) Une indication de si le mécanisme est client d'abord, variable, ou serveur d'abord. Si un mécanisme SASL est défini comme client d'abord et si le client n'envoie pas une réponse initiale dans la demande d'authentification, le premier défi du serveur DOIT alors être vide (le mécanisme EXTERNAL est un exemple de ce cas). Si un mécanisme SASL est défini comme variable, la spécification doit alors déclarer comment le serveur se comporte lorsque la réponse initiale du client est omise dans la demande d'authentification (le mécanisme DIGEST-MD5 [RFC2831] est un exemple de ce cas). Si un mécanisme SASL est défini comme serveur d'abord, le client NE DOIT alors PAS envoyer une réponse initiale de client dans la demande d'authentification (le mécanisme CRAM-MD5 [CRAM-MD5] est un exemple de ce cas).
   b) Une indication de si le serveur est supposé fournir des données supplémentaires lorsque il indique un résultat de succès. Si il l'est, et si le serveur envoie des données supplémentaires comme défi, la spécification DOIT indiquer que la réponse à ce défi est une réponse vide.

   Les mécanismes SASL DEVRAIENT être conçus de façon à minimiser le nombre de défis et réponses nécessaire pour achever l'échange.

3) Une indication de si le mécanisme est capable de transférer une chaîne d'identité d'autorisations (paragraphe 3.4.1). Bien que certains mécanismes traditionnels soient incapables de transmettre une identité d'autorisation (ce qui signifie que pour ces mécanismes, l'identité d'autorisation est toujours la chaîne vide) les mécanismes nouvellement définis DEVRAIENT être capables of transférer des chaînes d'identité d'autorisation. Le mécanisme NE DEVRAIT PAS être capable de transférer à la fois une chaîne d'identité d'autorisation et une identité d'autorisation vide.

   Les mécanismes qui sont capables de transférer une chaîne d'identité d'autorisation DOIVENT être capables de transférer des séquences arbitraires non vides de caractères Unicode, à l'exclusion de celles qui contiennent le caractère NUL (U+0000). Les mécanismes DEVRAIENT utiliser le format de transformation UTF-8 [RFC3629]. La spécification DOIT détailler comment tout codet Unicode spécifique du mécanisme qui pourrait apparaître dans la chaîne d'identité d'autorisation est échappé pour éviter des ambiguïtés durant le décodage de la chaîne d'identité d'autorisation. Normalement, les mécanismes qui ont des caractères spéciaux exigent que ceux-ci soient échappés ou codés dans la chaîne de caractères (après l'avoir codée dans un format de transformation Unicode particulier) en utilisant un schéma de codage de données tel que Base64 [RFC3548].

4) La spécification DOIT préciser si le mécanisme offre une couche de sécurité. Si le mécanisme le fait, la spécification DOIT préciser la sécurité et les autres services offerts dans la couche ainsi que comment ces services sont à mettre en œuvre.

5) Si la technologie cryptographique sous-jacente utilisée par un mécanisme prend en charge la protection de l'intégrité des données, la spécification du mécanisme DOIT alors protéger l'intégrité de la transmission d'une identité d'autorisation et la négociation de la couche de sécurité.

Les mécanismes SASL DEVRAIENT être neutres quant au protocole.

Les mécanismes SASL DEVRAIENT réutiliser les accréditifs et les formulaires d'identité existants, ainsi que la syntaxe et la sémantique associées.

Les mécanismes SASL DEVRAIENT utiliser le format de transformation UTF-8 [RFC3629] pour coder les codets Unicode [Unicode] pour le transfert.

Afin d'éviter les problèmes d'interopérabilité dus à des différences de normalisation, lorsqu'un mécanisme appelle à utiliser des données de caractères (autres que la chaîne d'identité d'autorisation) comme entrée à une fonction cryptographique et/ou de comparaison, la spécification DOIT détailler précisément comment et où (client ou serveur) les données de caractère seront préparées, incluant toutes normalisations, pour les entrer dans la fonction afin d'en assurer le bon fonctionnement.

Pour les noms d'utilisateur et/ou mots de passe simples dans les accréditifs d'authentification, SASLprep [RFC4013] (profil de l'algorithme de préparation de StringPrep [RFC3454]), DEVRAIT être spécifié comme algorithme de préparation.

Le mécanisme NE DEVRAIT PAS utiliser la chaîne d'identité d'autorisation pour générer de clés ou hachages cryptographiques à long terme car il n'est pas exigé que la chaîne d'identité d'autorisation soit canonique. Long terme signifie ici un terme plus long que la durée de l'échange d'authentification dans lequel ils ont été générés. C'est-à-dire, comme différents clients (du même protocole ou de protocoles différents) peuvent fournir des chaînes d'identité d'autorisation différentes qui sont sémantiquement équivalentes, l'utilisation de chaînes d'identité d'autorisation pour générer des clés et hachages cryptographiques va probablement conduire à des problèmes d'interopérabilité et autres.

6. Considérations pour la sécurité

Les questions de sécurité sont discutées tout au long du présent mémoire.

De nombreux mécanismes SASL existants ne fournissent pas une protection adéquate contre les attaques passives, sans parler des attaques actives, dans l'échange d'authentification. De nombreux mécanismes SASL existants n'offrent pas de couche de sécurité. On espère que les mécanismes SASL de l'avenir fourniront une protection forte contre les attaques passives et actives dans l'échange d'authentification, ainsi que des couches de sécurité avec des services forts de dispositifs de sécurité des données de base (par exemple, d'intégrité des données et de confidentialité des données). On espère aussi que de futurs mécanismes fourniront des services de sécurité des données plus évolués comme de changement de clés (voir au paragraphe 6.3).

Toutefois, le cadre de travail SASL est susceptible de subir des attaques en dégradation. Le paragraphe 6.1.2 présente diverses approches pour la prévention ou la détection de ces attaques. Dans certains cas, il est approprié d'utiliser des

services de protection de l'intégrité extérieurs à SASL (par exemple, TLS) pour protéger contre les attaques en dégradation dans SASL. L'utilisation de services externes de protection de la sécurité est aussi importante lorsque les mécanismes disponibles n'offrent pas par eux-mêmes une protection adéquate de l'intégrité et/ou de la confidentialité de l'échange d'authentification et/ou des données du protocole.

6.1. Attaques actives

6.1.1. Attaques par capture

Lorsque le client choisit une couche de sécurité SASL avec au moins la protection de l'intégrité, cette protection sert de contre mesure contre une attaque active de capture de la connexion et de modification des données du protocole envoyées après l'établissement de la couche de sécurité. Les mises en œuvre DEVRAIENT clore la connexion lorsque les services de sécurité d'une couche de sécurité SASL font rapport d'un manque d'intégrité des données.

6.1.2. Attaques en dégradation

Il est important que toute négociation de protocole sensible à la sécurité soit effectuée après l'installation d'une couche de sécurité avec la protection de l'intégrité des données. Les protocoles devraient être conçus de façon telle que les négociations effectuées avant cette installation soient revalidées après l'achèvement de l'installation. La négociation du mécanisme SASL est sensible à la sécurité.

Lorsque un client négocie avec le serveur le mécanisme d'authentification et/ou d'autres dispositifs de sécurité, il est possible qu'un attaquant actif cause l'utilisation par une partie des services de sécurité disponibles les moins sûrs. Par exemple, un attaquant peut modifier la liste des mécanismes annoncés par le serveur ou peut modifier la liste des dispositifs de sécurité annoncés par le client au sein d'une réponse au mécanisme. Pour se protéger contre cette sorte d'attaque, les mises en œuvre NE DEVRAIENT PAS annoncer les mécanismes et/ou dispositifs qui ne peuvent pas satisfaire leurs exigences minimales de sécurité, NE DEVRAIENT PAS entrer dans, ou poursuivre des échanges d'authentification qui ne peuvent pas satisfaire à leurs exigences minimales de sécurité, et DEVRAIENT vérifier que l'achèvement des échanges d'authentification résulte en des services de sécurité qui satisfont leurs exigences minimales de sécurité. Noter que chaque point d'extrémité doit vérifier indépendamment que ses exigences de sécurité sont satisfaites.

Afin de détecter les attaques en dégradation sur le mécanisme le moins sûr pris en charge, le client peut découvrir les mécanismes SASL que le serveur rend disponibles à la fois avant l'échange d'authentification SASL et après qu'a été installée la couche de sécurité SASL négociée (avec au moins la protection de l'intégrité des données) au moyen de la facilité de découverte des mécanismes du protocole. Si le client trouve que la liste protégée en intégrité (la liste obtenue après l'installation de la couche de sécurité) contient un mécanisme plus fort que ceux de la liste obtenue précédemment, le client devrait supposer que la liste obtenue précédemment a été modifiée par un attaquant et DEVRAIT clore la connexion de transport sous-jacente.

L'initiation par le client de l'échange SASL, incluant le choix d'un mécanisme SASL, est faite en clair et peut être modifiée par un attaquant actif. Il est important que tout nouveau mécanisme SASL soit conçu de façon telle qu'un attaquant actif ne puisse pas obtenir une authentification avec des propriétés de sécurité plus faibles en modifiant le nom du mécanisme SASL et/ou les défis et réponses.

La négociation à plusieurs niveaux de dispositifs de sécurité est encline à des attaques en dégradation. Les concepteurs de protocoles devraient éviter d'offrir des négociations de dispositifs de sécurité à des niveaux supérieurs dans les protocoles (par exemple, au dessus de la négociation de mécanisme SASL) et les concepteurs de mécanismes devraient éviter la négociation de dispositifs de sécurité à des niveaux inférieurs dans les mécanismes (par exemple, en dessous de la négociation de mécanisme SASL).

6.1.3. Attaques en répétition

Certains mécanismes peuvent être soumis à des attaques en répétition si ils ne sont pas protégés par des services externes de sécurité des données (par exemple, TLS).

6.1.4. Attaques en troncature

La plupart des couches de sécurité SASL existantes n'offrent pas elles-mêmes de protection contre l'attaque en troncature. Dans une attaque en troncature, l'attaquant actif cause la clôture de la session de protocole, causant une troncature du flux de données éventuellement protégé en intégrité ce qui conduit à un comportement d'un ou des deux homologues du protocole qui avantage de façon inappropriée l'attaquant. Les attaques en troncature sont assez faciles à déjouer dans les

protocoles de niveau application en mode connexion. Un protocole peut se défendre contre ces attaques en s'assurant que chaque échange d'informations a un résultat final clair et que chaque session de protocole a un mécanisme de clôture en douceur, et qu'ils sont protégés en intégrité.

6.1.5. Autres attaques actives

Lorsque l'utilisation d'une couche de sécurité est négociée par l'échange de protocole d'authentification, le receveur DEVRAIT fermer en douceur toute mémoire tampon de données protégées plus grande que la taille maximale définie/négociée. En particulier, il NE DOIT PAS allouer aveuglément la quantité de mémoire spécifiée dans le champ Taille de mémoire tampon, car ceci peut causer une condition de "plus de mémoire". Si le receveur détecte un grand bloc, il DEVRAIT clore la connexion.

6.2. Attaques passives

De nombreux mécanismes sont sujets à diverses attaques passives, incluant le simple espionnage d'informations d'accréditifs non protégés aussi bien que d'attaques de dictionnaire en ligne et hors ligne d'informations d'accréditifs protégées.

6.3. Changement de clés

Les durées de vie des couches de sécurité des mécanismes SASL sûrs ou permis administrativement sont déterminées. Les clés de chiffrement s'affaiblissent lorsque elles sont utilisées et avec le temps ; plus il y a de temps et/ou de texte chiffré qu'a un cryptanalyste après le premier usage d'une clé, et plus il est facile au cryptanalyste de monter des attaques contre la clé.

Les limites administratives à la durée de vie d'une couche de sécurité peuvent prendre la forme de limites de temps exprimées dans des certificats X.509, dans des tickets V Kerberos, ou des répertoires, et sont souvent souhaitées. En pratique, un effet probable de limites administratives de durée de vie est que les applications peuvent trouver que les couches de sécurité arrêtent de fonctionner au milieu des opérations de protocole de l'application, comme peut-être durant de grands transferts de données. Par suite de cela, la connexion sera close (voir le paragraphe 3.7) ce qui résulte en une expérience déplaisante pour l'utilisateur.

Le changement de clés (processus de renégociation de clés) est loin de régler les faiblesses des clés de chiffrement. Le cadre SASL ne fournit pas par lui même de mécanisme de changement de clés mais les mécanismes SASL peuvent le faire. Les concepteurs de futurs mécanismes SASL devraient envisager de fournir des services de changement de clés.

Les mises en œuvre qui souhaitent changer les clés des couches de sécurité SASL lorsque le mécanisme ne fournit pas de changement de clés DEVRAIENT réauthentifier les mêmes identifiants et remplacer les couches de sécurité arrivées à expiration ou qui sont sur le point de le faire. Cette approche exige la prise en charge de la réauthentification dans les protocoles d'application (voir le paragraphe 3.8).

6.4. Autres considérations

Les concepteurs de protocoles et de mises en œuvre devraient comprendre les considérations de sécurité des mécanismes afin qu'ils puissent choisir ceux qui sont applicables à leurs besoins.

Les mises en œuvre réparties de serveur doivent être prudentes quant à la façon d'accorder leur confiance aux autres parties. En particulier, les secrets d'authentification ns devraient être divulgués qu'aux autres parties qui sont de confiance pour gérer et utiliser ces secrets d'une manière acceptable à celui qui les divulgue. Les applications qui utilisent SASL supposent que les couches de sécurité SASL qui fournissent la confidentialité des données sont sûres même lorsque un attaquant choisit le texte à protéger par la couche de sécurité. De même, les applications supposent que la couche de sécurité SASL est sûre même si l'attaquant peut manipuler le résultat du texte chiffré de la couche de sécurité. De nouveaux mécanismes SASL sont attendus pour satisfaire ces hypothèses.

Les considérations de sécurité d'Unicode [UTR36] s'appliquent aux chaînes d'identité d'autorisation, ainsi que les considérations de sécurité de UTF-8 [RFC3629] lorsque UTF-8 est utilisé. Les considérations de sécurité de SASLprep [RFC4013] et de StringPrep [RFC3454] s'appliquent aussi lorsque ces profils sont utilisés.

7. Considérations relatives à l'IANA

7.1. Registre du mécanisme SASL

Le registre des mécanismes SASL est tenu par l'IANA. Le registre est actuellement disponible à <http://www.iana.org/assignments/sasl-mechanisms>.

L'objet de ce registre n'est pas seulement de s'assurer de l'unicité des valeurs utilisées pour désigner les mécanismes SASL, mais aussi de fournir une référence définitive aux spécifications techniques qui précisent chaque mécanisme SASL disponible sur l'Internet.

Il n'y a pas de convention de dénomination des mécanismes SASL ; tout nom qui se conforme à la syntaxe d'un nom de mécanisme SASL peut être enregistré.

La procédure décrite au paragraphe 7.1.1 doit être utilisée pour l'enregistrement d'une valeur désignant un mécanisme individuel spécifique.

La procédure détaillée au paragraphe 7.1.2 est à utiliser pour l'enregistrement d'une valeur désignant une famille de mécanismes en rapports.

Des commentaires peuvent être inclus dans le registre comme exposé au paragraphe 7.1.3 et peuvent être changés comme décrit au paragraphe 7.1.4.

Le registre des mécanismes SASL a été mis à jour pour refléter que le présent document fournit la spécification technique définitive pour SASL et que cette section fournit la procédure d'enregistrement pour ce registre.

7.1.1. Procédure d'enregistrement de nom de mécanisme

L'IANA enregistrera les noms des nouveaux mécanismes SASL sur la base du premier arrivé premier servi, comme défini dans le BCP 26 [RFC2434]. L'IANA a le droit de rejeter les demandes d'enregistrement visiblement erronées, mais n'exercera aucun contrôle sur les revendications faites dans les formulaires d'enregistrement.

L'enregistrement d'un mécanisme SASL est demandé en remplissant le formulaire suivant :

Objet : Enregistrement d'un mécanisme SASL X
Nom du mécanisme SASL (ou préfixe pour une famille) :
Considérations sur la sécurité :
Spécification publiée (recommandé) :
Adresse de la personne & courriel à contacter pour plus d'informations :
Utilisation prévue : (un parmi "commun", "usage limité", ou "obsolète")
Auteur/contrôleur des changements :
(Note : toutes les autres informations que l'auteur estime pertinentes peuvent être ajoutées ici.)

Il est à envoyer par messagerie électronique à l'IANA à < iana@iana.org >.

Bien que cette procédure d'enregistrement n'exige pas de révision par un expert, les auteurs de mécanismes SASL sont encouragés à rechercher la relecture et les commentaires de la communauté chaque fois que c'est faisable. Les auteurs peuvent inviter à la relecture par la communauté en envoyant la spécification de leur proposition de mécanisme comme projet Internet. Les mécanismes SASL destinés à une large utilisation devraient être normalisés par les processus normaux de l'IETF, lorsque c'est approprié.

7.1.2. Procédure d'enregistrement de nom de famille

Comme noté précédemment, il n'y a pas de convention générale de désignation pour les mécanismes SASL. Cependant, les spécifications peuvent réserver une portion de l'espace de noms du mécanisme SASL pour un ensemble de mécanismes SASL liés, une "famille" de mécanismes SASL. Chaque famille de mécanismes SASL est identifiée par un préfixe unique, tel que X-. L'enregistrement des noms de nouvelles familles de mécanismes SASL exige une revue par expert comme défini dans le BCP 26 [RFC2434].

L'enregistrement d'un nom de famille SASL est demandé en remplissant le formulaire suivant :

Objet : Enregistrement de la famille de mécanismes SASL X

Nom de famille SASL (ou préfixe pour la famille) :
Considérations de sécurité :
Spécification publiée (recommandé) :
Adresse de la personne & courriel à contacter pour plus d'informations :
Utilisation prévue : (un parmi "commun", "usage limité", ou "obsolète")
Auteur/contrôleur des changements :
(Note : toutes les autres informations que l'auteur estime pertinentes peuvent être ajoutées ici.)

Il est à envoyer via messagerie électronique à la liste de diffusion SASL de l'IETF à < ietf-sasl@imc.org > et en copie à l'IANA à < iana@iana.org >. Après deux semaines d'apports de la communauté sur la liste de diffusion SASL de l'IETF, l'expert va déterminer l'opportunité de la demande d'enregistrement et va approuver ou désapprouver la demande avec information au demandeur, à la liste de diffusion, et à l'IANA.

La revue devrait se concentrer sur l'opportunité du nom de famille demandé pour l'utilisation proposée et l'opportunité du plan de désignation et d'enregistrement proposé pour les noms de mécanismes existants et futurs dans la famille. La portée de cette revue de demande peut englober la considération des aspects pertinents de toute spécification technique fournie, comme dans leur section Considérations relatives à l'IANA. Cependant, cette revue est étroitement concentrée sur l'opportunité de l'enregistrement demandé et non sur l'adéquation globale de la spécification technique produite.

Les auteurs sont invités à rechercher la relecture communautaire en proposant la spécification technique comme projet Internet et à solliciter les commentaires en l'adressant aux listes de diffusion appropriée de l'IETF.

7.1.3 Commentaires sur l'enregistrement de mécanisme SASL

Les commentaires sur un mécanisme/famille SASL enregistré devraient d'abord être envoyés au "propriétaire" du mécanisme/famille et/ou à la liste de diffusion < ietf-sasl@imc.org >.

Les auteurs de commentaires peuvent, après avoir raisonnablement tenté de contacter le "propriétaire", demander à l'IANA de joindre leur commentaire à l'enregistrement de mécanisme SASL lui-même en envoyant un message à <iana@iana.org>. À la seule discrétion de l'IANA, le commentaire peut être joint à l'enregistrement de mécanisme SASL.

7.1.4. Contrôle des changements

Une fois qu'un enregistrement de mécanisme SASL a été publié par l'IANA, l'auteur peut demander un changement à sa définition. La demande de changement suit la même procédure que la demande d'enregistrement.

Le propriétaire d'un mécanisme SASL peut passer la responsabilité du mécanisme SASL à une autre personne ou agence en informant l'IANA ; ceci peut être fait sans discussion ni revue.

L'IESG peut réallouer la responsabilité d'un mécanisme SASL. Le cas le plus courant sera pour permettre que des changements soient faits aux mécanismes lorsque l'auteur de l'enregistrement est mort, a quitté son entreprise, ou est d'une autre façon dans l'incapacité de faire des changements qui sont importants pour la communauté.

Les enregistrements de mécanisme SASL ne peuvent pas être supprimés ; les mécanismes dont on estime que leur utilisation n'est plus appropriée peuvent être déclarés obsolètes par un changement à leur champ "Utilisation prévue" ; un tel mécanisme SASL sera clairement marqué dans les listes publiées par l'IANA.

L'IESG est considéré comme étant le propriétaire de tous les mécanismes SASL qui sont sur la voie de la normalisation par l'IETF.

7.2. Enregistrement des changements

L'IANA a mis à jour le registre des mécanismes SASL comme suit :

1) Changé le "Usage de destination" des enregistrement de mécanisme KERBEROS_V4 et SKEY en OBSOLETE.

2) Changé la "Spécification publiée" du mécanisme EXTERNAL du présent document comme indiqué ci-dessous :

Sujet : Enregistrement du mécanisme SASL EXTERNAL
Famille des mécanismes SASL : NON
Nom de mécanisme SASL : EXTERNAL
Considérations pour la sécurité : Voir le paragraphe A.3 de la RFC 4422

Spécification publiée (facultative, recommandée) : RFC 4422
Personne et adresse de messagerie à contacter pour des informations complémentaires :
Alexey Melnikov <Alexey.Melnikov@isode.com>
Usage de destination : COMMUN
Propriétaire/contrôleur des changements : IESG iesg@ietf.org

Note : Met à jour l'entrée existante pour EXTERNAL

8. Références

8.1. Références normatives

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.

[RFC2244] C. Newman, J. G. Myers,"ACAP – Protocole d'accès à la configuration d'application", novembre 1997. *(P.S.)*

[RFC2434] T. Narten et H. Alvestrand, "Lignes directrices pour la rédaction d'une section Considérations relatives à l'IANA dans les RFC", BCP 26, octobre, 1998. *(Rendue obsolète par la* RFC5226)

[RFC2743] J. Linn, "Interface générique de programme d'application de service de sécurité, version 2, mise à jour 1", janvier 2000. (*MàJ par* RFC5554)

[RFC3454] P. Hoffman et M. Blanchet, "Préparation de chaînes internationalisées ("stringprep")", décembre 2002. *(P.S.)*

[RFC3629] F. Yergeau, "UTF-8, un format de transformation de la norme ISO 10646", STD 63, novembre 2003.

[RFC4013] K. Zeilenga, "SASLprep : Profil Stringprep pour les noms d'utilisateur et mots de passe", février 2005.

[RFC4234] D. Crocker et P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", octobre 2005. *(Remplace* RFC2234, remplacée par RFC5234)

[ASCII] "Coded Character Set--7-bit American Standard Code for Information Interchange", ANSI X3.4-1986.

[Unicode] The Unicode Consortium, "Norme Unicode, version 3.2.0" défini par "The Unicode Standard, Version 3.0" (Reading, MA, Addison-Wesley, 2000. ISBN 0-201- 61633-5), amendée par "Norme Unicode Annexe n° 27 : Unicode 3.1" (http://www.unicode.org/reports/tr27/) et par "Norme Unicode Annexe n° 28 : Unicode 3.2" (http://www.unicode.org/reports/tr28/).

[CharModel] K. Whistler et M. Davis, "Unicode Technical Report #17, Character Encoding Model", UTR17, <http://www.unicode.org/unicode/reports/tr17/ >, August 2000.

[Glossary] The Unicode Consortium, "Glossaire Unicode", <http://www.unicode.org/glossary/ >.

8.2 Références pour information

[RFC2831] P. Leach et C. Newman, "Utilisation de l'authentification par résumé comme mécanisme SASL", mai 2000. (*Obsolète, voir* RFC6331)

[RFC3206] R. Gellens, "Codes de réponse POP SYS et AUTH", février 2002. *(P.S.)*

[RFC3548] S. Josefsson, "Codages de données Base16, Base32, et Base64", juillet 2003. *(Obsolète, voir* 4648) (Info)

[RFC4301] S. Kent et K. Seo, "Architecture de sécurité pour le protocole Internet", décembre 2005. *(P.S.) (Remplace la* RFC2401)

[RFC4346] T. Dierks et E. Rescorla, "Protocole de sécurité de la couche Transport (TLS) version 1.1", avril 2006. *(Remplace* RFC2246 *; Remplacée par* RFC5246 *; MàJ par* RFC4366, RFC4680, RFC4681, RFC5746,

RFC6176, RFC7465, RFC7507, RFC7919)

[RFC4752] A. Melnikov, éd., "Méthode d'utilisation de l'interface de programme d'application de service générique de sécurité (GSS-API) Kerberos v5 dans le mécanisme d'authentification simple et couche de sécurité (SASL)", novembre 2006.

[UTR36] Davis, M., "(Draft) Unicode Technical Report #36, Character Encoding Model", UTR17, <http://www.unicode.org/unicode/reports/tr36/>, février 2005.

[CRAM-MD5] L. Nerenberg, "The CRAM-MD5 SASL Mechanism", Travail en cours.

9. Remerciements

Le présent document est une révision de la RFC 2222 écrite par John Myers.

Cette révision a été produite par le groupe de travail Authentification simple et couche de sécurité (SASL) de l'IETF.

Les individus suivants ont contribué de façon significative à cette révision : Abhijit Menon-Sen, Hallvard Furuseth, Jeffrey Hutzelman, John Myers, Luke Howard, Magnus Nystrom, Nicolas Williams, Peter Saint-Andre, RL 'Bob' Morgan, Rob Siemborski, Sam Hartman, Simon Josefsson, Tim Alsop, et Tony Hansen.

Appendice A. Le mécanisme SASL EXTERNAL

Le présent appendice est normatif.

Le mécanisme EXTERNAL permet à un client de demander au serveur d'utiliser les accréditifs établis par des moyens externes au mécanisme pour authentifier le client. Le moyen externe peut être, par exemple, le service de sécurité IP [RFC4301] ou TLS [RFC4346]. En l'absence d'un accord préalable entre le client et le serveur, le client ne peut faire aucune hypothèse sur le moyen externe utilisé par le serveur pour obtenir les accréditifs du client, ni faire d'hypothèse sur la forme des accréditifs. Par exemple, le client ne peut pas supposer que le serveur va utiliser les accréditifs que le client a établis via TLS.

A.1. Spécification technique de EXTERNAL

Le nom de ce mécanisme est "EXTERNAL".

Le mécanisme ne fournit pas une couche de sécurité.

Le mécanisme est capable de transférer une chaîne d'identité d'autorisation. Si elle est vide, le client demande à agir comme l'identité que le serveur a associée aux accréditifs du client. Si elle n'est pas vide, le client demande à agir comme l'identité représentée par la chaîne.

Le client est supposé envoyer d'abord les données dans l'échange d'authentification. Lorsque le client ne fournit pas les données de réponse initiale dans sa demande pour initier l'échange d'authentification, le serveur doit répondre à la demande par un défi initial vide et alors le client doit fournir sa réponse initiale.

Le client envoie la réponse initiale contenant le codage UTF-8 [RFC3629] de la chaîne d'identité d'autorisation demandée. Cette réponse n'est pas vide lorsque le client demande à agir comme l'identité représentée par la chaîne (non vide). Cette réponse est vide lorsque le client demande à agir comme l'identité que le serveur a associée à ses accréditifs d'authentification.

La syntaxe de la réponse initiale est spécifiée comme une valeur du produit <extern-initial-resp> précisé ci-dessous en utilisant la notation en forme Backus-Naur augmenté (ABNF) [RFC4234].

```
external-initial-resp = authz-id-string
authz-id-string       = *( UTF8-char-no-nul )
UTF8-char-no-nul      = UTF8-1-no-nul / UTF8-2 / UTF8-3 / UTF8-4UTF8-1-no-nul         = %x01-7F
```

où les produits <UTF8-2>, <UTF8-3>, et <UTF8-4> sont ceux définis dans la [RFC3629].

Il n'y a pas de défi et réponse supplémentaires.

Et donc, le serveur va retourner le résultat de l'échange d'authentification.

L'échange échoue si
- le client n'a pas établi ses accréditifs via un moyen externe,
- les accréditifs du client sont inadéquats,
- le client a fourni une chaîne d'identité d'autorisation vide et le serveur ne veut pas ou n'est pas capable d'associer une identité d'autorisation aux accréditifs du client,
- le client a fourni une chaîne d'identité d'autorisation non vide qui est invalide selon les exigences de syntaxe de la spécification applicable de protocole d'application,
- le client a fourni une chaîne d'identité d'autorisation non vide représentant une identité que le client n'est pas autorisé à emprunter, ou
- le serveur ne veut pas ou n'est pas capable de fournir le service au client pour une raison autre, quelconque.

Autrement, l'échange est réussi. Lors de l'indication d'un bon résultat, les données supplémentaires ne sont pas fournies.

A.2. Exemples de EXTERNAL SASL

Cette section fournit des exemples des échanges d'authentification EXTERNAL. Les exemples sont destinés à aider le lecteur à comprendre le texte précédent. Les exemples ne sont pas définitifs. Le protocole d'accès à la configuration d'application (ACAP) [RFC2244] est utilisé dans les exemples.

Le premier exemple montre l'utilisation de EXTERNAL avec une identité d'autorisation vide. Dans cet exemple, la réponse initiale n'est pas envoyée dans la demande du client pour initier l'échange d'authentification.

```
S: * ACAP (SASL "DIGEST-MD5")
C: a001 STARTTLS
S: a001 OK "Commencer la négociation TLS maintenant"
                        <négociation TLS, les commandes suivantes sont sous la couche TLS>
S: * ACAP (SASL "DIGEST-MD5" "EXTERNAL")
C: a002 AUTHENTICATE "EXTERNAL"
S: + ""C: + ""
S: a002 OK "Authentifié"
```

Le second exemple montre l'utilisation de EXTERNAL avec une identité d'autorisation de "fred@example.com". Dans cet exemple, la réponse initiale est envoyée avec la demande du client pour initier l'échange d'authentification. Cela économise un aller-retour.

```
S: * ACAP (SASL "DIGEST-MD5")
C: a001 STARTTLS
S: a001 OK "Commencer la négociation TLS maintenant"
                        <négociation TLS, les commandes suivantes sont sous la couche TLS>
S: * ACAP (SASL "DIGEST-MD5" "EXTERNAL")
C: a002 AUTHENTICATE "EXTERNAL" {16+}
C: fred@example.com
S: a002 NO "Ne peut pas assumer l'identité d'autorisation demandée"
```

A.3. Considérations pour la sécurité

Le mécanisme EXTERNAL ne fournit pas de protection de sécurité ; il est vulnérable à l'usurpation d'identité par un client ou serveur, aux attaques actives, à l'espionnage. Il ne devrait être utilisé que lorsque des services de sécurité adéquats ont été établis.

Appendice B. Changements par rapport à la RFC 2222

Le présent appendice n'est pas normatif.

Les matériaux de la RFC 2222 ont été réécrits de façon significative pour la production du présent document.

La RFC 2222, en n'établissant pas que la chaîne d'identité d'autorisation est une chaîne de caractères Unicode, laisse de côté les données de caractères, ce qui impliquait que la chaîne d'identité d'autorisation était une chaîne d'octets.

- La chaîne d'identité d'autorisation est maintenant définie comme une chaîne de caractères Unicode. Le caractère NUL (U+0000) est interdit. Alors que les spécifications de protocole sont chargées de la définition de la forme de l'identité d'autorisation, ainsi que de la syntaxe de la chaîne Unicode et de la sémantique qui s'y rapportent, les spécifications des mécanismes sont chargées de définir comment la chaîne Unicode est portée dans l'échange d'authentification.

- Suppression de "S'il en est ainsi, lorsque le client n'envoie pas les données d'abord, le défi initial DOIT être spécifié comme étant un défi vide."

Les changements techniques suivants ont été apportés au mécanisme EXTERNAL :

- La chaîne d'identité d'autorisation est à coder en UTF-8.

Noter que les exigences de la spécification de protocole et de mécanisme ont été significativement resserrées. Les spécifications de protocole et de mécanisme existantes devront être mises à jour pour satisfaire à ces exigences.

Adresse des éditeurs

Alexey Melnikov
Isode Limited
5 Castle Business Village
36 Station Road
Hampton, Middlesex,
TW12 2BX, United Kingdom
mèl : Alexey.Melnikov@isode.com
URI : http://www.melnikov.ca/

Kurt D. Zeilenga
OpenLDAP Foundation
mèl : Kurt@OpenLDAP.org

Déclaration complète de droits de reproduction

Propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourraient être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur répertoire en ligne des IPR de l'IETF à http://www.ietf.org/ipr.

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente

norme. Prière d'adresser les informations à l'IETF à ietf- ipr@ietf.org.

Remerciement

Le financement de la fonction d'édition des RFC est fourni par l'activité de soutien administratif (IASA) de l'IETF