Guide de conception et d'implémentation de la mise en cache de jetons (TokenCaching)

Contenu

Introduction

La portée de ce document est de discuter l'installation et de la dépanner de TokenCaching. Des sessions de Protocole point à point (PPP) pour des utilisateurs d'adaptateur terminal RNIS (MERCI) sont typiquement terminées au PC utilisateur. Ceci permet à l'utilisateur pour contrôler la session PPP de la même manière qu'une connexion d'accès par réseau commuté async (de modem), qui signifie connecte et déconnecte la session comme nécessaire. Ceci permet à l'utilisateur pour employer le Password Authentication Protocol (PAP) afin d'entrer le mot de passe une fois (OTP) pour le transport.

Cependant, si le deuxième canal B est conçu pour être soulevé automatiquement, l'utilisateur doit être incité pour un nouvel OTP pour le deuxième canal B. Le logiciel de PPP PC ne collecte pas le deuxième OTP. Au lieu de cela, les essais de logiciel pour utiliser le même mot de passe utilisé pour le canal primaire B. Le serveur symbolique de carte refuse la réutilisation d'un OTP par conception. La CiscoSecure ACS pour UNIX (version 2.2 et ultérieures) et la CiscoSecure ACS pour Windows (2.1 et plus tard) exécutent TokenCaching afin de prendre en charge l'utilisation du même OTP sur le deuxième canal B. Cette option exige du serveur d'Authentification, autorisation et comptabilité (AAA) de mettre à jour des informations d'état sur la connexion de l'utilisateur symbolique.

Référez-vous à prendre en charge les mots de passe une fois sur le pour en savoir plus RNIS.

Conditions préalables

Conditions requises

Ce document suppose que vous faites déjà configurer ces derniers correctement :

- Un modem commuté qui fonctionne correctement.
- Le serveur d'accès à distance (NAS) configuré correctement, avec l'AAA qui indique la CiscoSecure ACS UNIX ou ACS Windows.
- ACE/SDI est déjà installé avec la CiscoSecure ACS UNIX ou ACS Windows, et fonctionne correctement.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- CiscoSecure ACS Unix 2.2 ou plus tard
- CiscoSecure ACS Windows 2.1 ou plus tard

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

Conventions

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à Conventions relatives aux conseils techniques Cisco.

Configurez

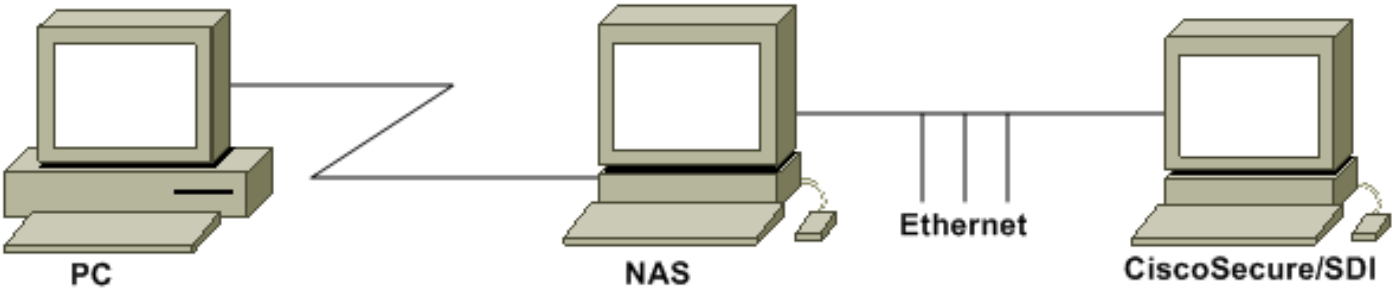
Cette section vous fournit des informations pour configurer les fonctionnalités décrites dans ce document.

Remarque: Utilisez l'outil Command Lookup Tool (clients enregistrés seulement) pour obtenir plus d'informations sur les commandes utilisées dans cette section.

Diagramme du réseau

Ce document utilise la configuration réseau suivante :

Un PC introduit dans le NAS et le modem RNIS, et est configuré pour la commande de **ppp multilink**.



Configurations

Ce document utilise les configurations suivantes :

- Configurez l'entrée de nom d'utilisateur et mot de passe
- Configurez TokenCaching sur la CiscoSecure ACS Windows
- Configurez TokenCaching dans la CiscoSecure ACS UNIX

Configurez l'entrée de nom d'utilisateur et mot de passe

Dans ce document, le NAS utilise le protocole d'authentification CHAP (Challenge Handshake Authentication Protocol) pour la session PPP avec le mot de passe une fois de SDI. Si vous utilisez le CHAP, entrez le mot de passe sous cette forme :

- **nom d'utilisateur** — fadi*pin+code (notez * dans le nom d'utilisateur)
- **mot de passe** — chappassword

Un exemple de ceci est : le nom d'utilisateur = le fadi, le mot de passe de CHAP = le Cisco, broche = 1234, et le code qui affiche sur le jeton est 987654. Par conséquent, l'utilisateur entre dans ceci :

- **nom d'utilisateur** — fadi*1234987654
- **mot de passe cisco**

Remarque: Si CiscoSecure et le NAS étaient configurés pour le PAP, le nom d'utilisateur et le jeton peuvent être écrits en tant que ceci :

- **nom d'utilisateur** — username*pin+code
- **mot de passe**—

Ou :

- **nom d'utilisateur nom d'utilisateur**
- **mot de passe** — pin+code

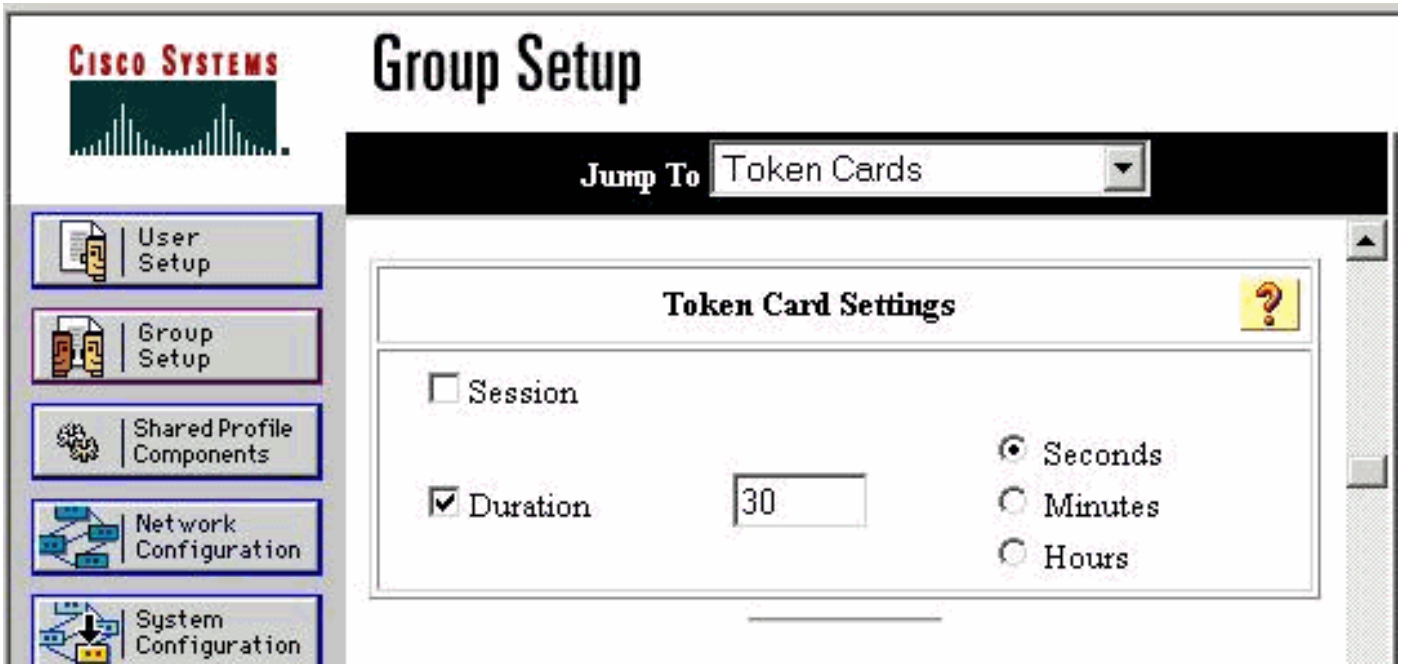
Configurez TokenCaching sur la CiscoSecure ACS Windows

L'utilisateur Windows ou le groupe de CiscoSecure ACS est installé comme d'habitude, avec IP et PPP LCP de PPP vérifiés si vous utilisez TACACS+. Si vous utilisez le RAYON, ceux-ci doivent être configurés :

- Attribut 6 = **Service_Type = encadré**
- Attribut 7 = **Framed_Protocol = PPP**

En outre, les paramètres de TokenCaching peuvent être vérifiés le groupe suivant les indications

de cet exemple :



Configurez TokenCaching dans la CiscoSecure ACS UNIX

Il y a quatre attributs de TokenCaching. L'attribut de config_token_cache_absolute_timeout (en quelques secondes) est placé dans le fichier $install_directory/config/CSU.cfg. Les trois autres attributs (placez la jeton-mise en cache de serveur, placez la jeton-mise en cache-expirer-méthode de serveur, et le jeton-mise en cache-délai d'attente de serveur de positionnement) sont placés dans les profils d'utilisateur ou de groupe. Pour ce document, le config_token_cache_absolute_timeout global d'attribut est placé à ceci dans le fichier $install_directory/config/CSU.cfg :

```
NUMBER config_token_cache_absolute_timeout = 300;
```

Les profils d'attribut de TokenCaching d'utilisateur et de serveur de groupe sont configurés suivant les indications de cet exemple :

```
Group Profile:

Group Profile Information
group = sdi{
profile_id = 42
profile_cycle = 5
default service=permit
set server token-caching=enable
set server token-caching-expire-method=timeout
set server token-caching-timeout=30
set server max-failed-login-count=1000

}

User Profile:

user = fadi{
profile_id = 20
set server current-failed-logins = 0
profile_cycle = 168
member = sdi
profile_status = enabled
```

```
password = chap "********"
password = sdi
password = pap "********"
password = clear "********"
default service=permit
set server max-failed-login-count=1000
 !--- The TACACS+ section of the profile. service=ppp { default protocol=permit protocol=ip {
set addr=1.1.1.1 } protocol=lcp { } !--- This allows the user to use the ppp multilink command.
protocol=multilink { } } service=shell { default attribute=permit } !--- The RADIUS section of
the profile. radius=Cisco12.05 { check_items= { 200=0 } } }
```

Vérifiez

Aucune procédure de vérification n'est disponible pour cette configuration.

Dépannez

Cette section fournit des informations que vous pouvez utiliser pour dépanner votre configuration.

Debug TokenCaching sur la CiscoSecure ACS UNIX

Ce log de CiscoSecure UNIX affiche une authentification réussie avec TokenCaching, quand l'authentification se produit sur deux canaux BRI :

```
Jun 14 13:44:29 cholera CiscoSecure: DEBUG - AUTHENTICATION START request
          (e7079cae)
!--- Detects the * in the username. Jun 14 13:44:29 cholera CiscoSecure: INFO - The character *
was found in username: username=fadi,passcode=3435598216 !--- Initializes ACE modules in
CiscoSecure. Jun 14 13:44:29 cholera CiscoSecure: DEBUG - sdi_challenge response timeout 5 Jun
14 13:44:29 cholera CiscoSecure: DEBUG - AceInit() Jun 14 13:44:29 cholera CiscoSecure: DEBUG -
AceInit(17477), ace rc=150, ed=1039800 Jun 14 13:44:29 cholera CiscoSecure: DEBUG -
acsWaitForSingleObject (17477) begin Jun 14 13:44:29 cholera CiscoSecure: DEBUG - aceCB(17477)
Jun 14 13:44:29 cholera CiscoSecure: DEBUG - aceCB(17477) AceGetUserData, ace rc=1, ed=1039800
Jun 14 13:44:29 cholera CiscoSecure: DEBUG - aceCB(17477): AceGetAuthenticationStatus, ace rc=1,
acm rc=0 Jun 14 13:44:29 cholera CiscoSecure: DEBUG - aceCB(17477): return Jun 14 13:44:29
cholera CiscoSecure: DEBUG - acsWaitForSingleObject(0) (17477) Jun 14 13:44:29 cholera
CiscoSecure: DEBUG - acsWaitForSingleObject (17477) end, rc=0 Jun 14 13:44:29 cholera
CiscoSecure: DEBUG - AceInit(17477), continue, acm rc=0 Jun 14 13:44:29 cholera CiscoSecure:
DEBUG - AceSetUsername(17477), username=fadi Jun 14 13:44:29 cholera CiscoSecure: DEBUG -
AceSetUsername(17477), ace rc=1 Jun 14 13:44:29 cholera CiscoSecure: INFO -
sdi_challenge(17477): rtn 1, state=GET_PASSCODE, user=fadi Jun 14 13:44:29 cholera CiscoSecure:
DEBUG - Token Caching. timeout_value is: 30 Jun 14 13:44:29 cholera CiscoSecure: DEBUG - Token
Caching. timeout enabled value: 30 Jun 14 13:44:29 cholera CiscoSecure: DEBUG -
profile_valid_tcaching TRUE ending. Jun 14 13:44:29 cholera CiscoSecure: DEBUG - Token Caching.
MISS. Jun 14 13:44:29 cholera CiscoSecure: DEBUG - AceSetPasscode(17477), passcode=3435598216
Jun 14 13:44:29 cholera CiscoSecure: DEBUG - AceSetPasscode(17477), ace rc=1 !--- Checks
credentials with ACE server. Jun 14 13:44:29 cholera CiscoSecure: DEBUG - AceCheck(17477) Jun 14
13:44:29 cholera CiscoSecure: DEBUG - AceCheck(17477), ace rc=150 Jun 14 13:44:29 cholera
CiscoSecure: DEBUG - acsWaitForSingleObject (17477) begin Jun 14 13:44:31 cholera CiscoSecure:
DEBUG - aceCB(17477) Jun 14 13:44:31 cholera CiscoSecure: DEBUG - aceCB(17477) AceGetUserData,
ace rc=1, ed=1039800 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - aceCB(17477):
AceGetAuthenticationStatus, ace rc=1, acm rc=0 Jun 14 13:44:31 cholera CiscoSecure: DEBUG -
aceCB(17477): return Jun 14 13:44:31 cholera CiscoSecure: DEBUG - acsWaitForSingleObject(0)
(17477) Jun 14 13:44:31 cholera CiscoSecure: DEBUG - acsWaitForSingleObject (17477) end, rc=0
Jun 14 13:44:31 cholera CiscoSecure: DEBUG - AceCheck(17477), continue, acm rc=0 Jun 14 13:44:31
cholera CiscoSecure: INFO - sdi_verify(17477): fadi authenticated by ACE Srvr Jun 14 13:44:31
cholera CiscoSecure: DEBUG - AceClose(17477) Jun 14 13:44:31 cholera CiscoSecure: INFO -
sdi(17477): fadi free external_data memory, state=GET_PASSCODE !--- The TokenCaching timeout is
set to 30 seconds. Jun 14 13:44:31 cholera CiscoSecure: DEBUG - Token Caching. timeout_value is:
```

```
30 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - Token Caching. timeout enabled value: 30 Jun 14
13:44:31 cholera CiscoSecure: DEBUG - profile_valid_tcaching TRUE ending. !--- The TokenCaching
takes place. Jun 14 13:44:31 cholera CiscoSecure: DEBUG - cache_insert (key<4>,
val<10><3435598216>, port_type<3>) Jun 14 13:44:31 cholera CiscoSecure: DEBUG - Cisco Cached
Tokens : 1 Jun 14 13:44:31 cholera CiscoSecure: INFO - sdi_verify(17477): rtn 1 Jun 14 13:44:31
cholera CiscoSecure: DEBUG - Authentication - LOGIN successful; [NAS=lynch.cisco.com,
Port=BRI0:1, User=fadi, Priv=1] !--- The authentication of the second BRI channel begins. Jun 14
13:44:31 cholera CiscoSecure: DEBUG - AUTHENTICATION START request (76f91a6c) Jun 14 13:44:31
cholera CiscoSecure: INFO - The character * was found in username:
username=fadi,passcode=3435598216 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - sdi_challenge
response timeout 5 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - AceInit() Jun 14 13:44:31
cholera CiscoSecure: DEBUG - AceInit(29111), ace rc=150, ed=1039984 Jun 14 13:44:31 cholera
CiscoSecure: DEBUG - acsWaitForSingleObject (29111) begin Jun 14 13:44:31 cholera CiscoSecure:
DEBUG - aceCB(29111) Jun 14 13:44:31 cholera CiscoSecure: DEBUG - aceCB(29111) AceGetUserData,
ace rc=1, ed=1039984 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - aceCB(29111):
AceGetAuthenticationStatus, ace rc=1, acm rc=0 Jun 14 13:44:31 cholera CiscoSecure: DEBUG -
aceCB(29111): return Jun 14 13:44:31 cholera CiscoSecure: DEBUG - acsWaitForSingleObject(0)
(29111) Jun 14 13:44:31 cholera CiscoSecure: DEBUG - acsWaitForSingleObject (29111) end, rc=0
Jun 14 13:44:31 cholera CiscoSecure: DEBUG - AceInit(29111), continue, acm rc=0 Jun 14 13:44:31
cholera CiscoSecure: DEBUG - AceSetUsername(29111), username=fadi Jun 14 13:44:31 cholera
CiscoSecure: DEBUG - AceSetUsername(29111), ace rc=1 Jun 14 13:44:31 cholera CiscoSecure: INFO -
sdi_challenge(29111): rtn 1, state=GET_PASSCODE, user=fadi Jun 14 13:44:31 cholera CiscoSecure:
DEBUG - Token Caching. timeout_value is: 30 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - Token
Caching. timeout enabled value: 30 Jun 14 13:44:31 cholera CiscoSecure: DEBUG -
profile_valid_tcaching TRUE ending. !--- Checks with the cached token for the user "fadi". Jun
14 13:44:31 cholera CiscoSecure: DEBUG - Token Caching. USER : fadi Jun 14 13:44:31 cholera
CiscoSecure: DEBUG - PASSWORD : 3435598216 len: 10 Jun 14 13:44:31 cholera CiscoSecure: DEBUG -
hashval_str: 3435598216 len: 10 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - port_type : BRI
len: 3 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - Token Caching. HIT. Jun 14 13:44:31 cholera
CiscoSecure: DEBUG - AceClose(29111) Jun 14 13:44:31 cholera CiscoSecure: INFO - sdi(29111):
fadi free external_data memory, state=GET_PASSCODE Jun 14 13:44:31 cholera CiscoSecure: INFO -
sdi_verify(29111): rtn 1 Jun 14 13:44:31 cholera CiscoSecure: DEBUG - Authentication - LOGIN
successful; [NAS=lynch.cisco.com, Port=BRI0:2, User=fadi, Priv=1] !--- After 30 seconds the
cached token expires. Jun 14 13:45:03 cholera CiscoSecure: DEBUG - Expiring Cisco Token Cache
Entry Jun 14 13:45:03 cholera CiscoSecure: DEBUG - Cisco Cached Tokens : 0
```

Informations connexes

- Avis de sécurité Cisco, réponses, et notices
- Page de support produit Unix de CiscoSecure
- CiscoSecure ACS pour la page de support produit de Windows
- Support et documentation techniques - Cisco Systems