دليل تنفيذ وتصميم TokenCaching

المحتويات

المقدمة
المتطلبات الأساسية
المتطلبات
المكونات المستخدمة
الاصطلاحات
التكوين
الرسم التخطيطي للشبكة
التكوينات
تكوين إدخال اسم المستخدم وكلمة المرور
تكوين TokenCaching على CiscoSecure ACS Windows
تكوين TokenCaching في CiscoSecure ACS UNIX
التحقق من الصحة
استكشاف الأخطاء وإصلاحها
Debug TokenCaching على CiscoSecure ACS UNIX
معلومات ذات صلة

المقدمة

يجب أن يناقش نطاق هذا المستند إعداد TokenCaching واستكشاف أخطائه وإصلاحها. يتم عادة إنهاء جلسات عمل بروتوكول الاتصال من نقطة إلى نقطة (PPP) لمستخدمي مهايئ ISDN الطرفي (TA) في كمبيوتر المستخدم. وهذا يسمح للمستخدم بالتحكم في جلسة PPP بنفس الطريقة التي يتحكم بها اتصال هاتفي (مودم) غير متزامن، مما يعني اتصال جلسة العمل وفصلها حسب الحاجة. هذا يسمح المستعمل أن يستعمل كلمة مصادقة بروتوكول (PAP) in order to دخلت ال one-time كلمة (OTP) للنقل.

ومع ذلك، إذا تم تصميم قناة B الثانية بحيث تظهر تلقائيا، فيجب مطالبة المستخدم ببروتوكول OTP جديد للقناة B الثانية. لا يقوم برنامج PC PPP بتجميع بروتوكول OTP الثاني. بدلا من ذلك، يحاول البرنامج إستخدام نفس كلمة المرور المستخدمة للقناة B الأساسية. يرفض خادم بطاقة الرمز المميز إعادة إستخدام OTP حسب التصميم. يقوم CiscoSecure ACS ل UNIX (الإصدار 2.2 والإصدارات الأحدث) و CiscoSecure ACS ل Windows (2.1 والإصدارات الأحدث) بتنفيذ TokenCaching لدعم إستخدام نفس بروتوكول OTP على القناة الثانية B. يتطلب هذا الخيار خادم المصادقة والتفويض والمحاسبة (AAA) للحفاظ على معلومات الحالة حول اتصال مستخدم الرمز المميز.

راجع دعم كلمات المرور لمرة واحدة على ISDN للحصول على مزيد من المعلومات.

المتطلبات الأساسية

المتطلبات

يفترض هذا المستند أن هذه العناصر قد تم تكوينها بالفعل بشكل صحيح:

- مودم اتصال يعمل بشكل صحيح.
- تم تكوين خادم الوصول إلى الشبكة (NAS) بشكل صحيح، باستخدام AAA الذي يشير إلى CiscoSecure ACS UNIX أو ACS Windows.
- تم إعداد ACE/SDI بالفعل باستخدام CiscoSecure ACS UNIX أو ACS Windows، ويعمل بشكل صحيح.

المكونات المستخدمة

تستند المعلومات الواردة في هذا المستند إلى إصدارات البرامج والمكونات المادية التالية:

- Cisco Secure ACS UNIX 2.2 أو إصدار أحدث
- Cisco Secure ACS Windows 2.1 أو إصدار أحدث

تم إنشاء المعلومات الواردة في هذا المستند من الأجهزة الموجودة في بيئة معملية خاصة. بدأت جميع الأجهزة المستخدمة في هذا المستند بتكوين ممسوح (افتراضي). إذا كانت شبكتك مباشرة، فتأكد من فهمك للتأثير المحتمل لأي أمر.

الاصطلاحات

راجع اصطلاحات تلميحات Cisco التقنية للحصول على مزيد من المعلومات حول اصطلاحات المستندات.

التكوين

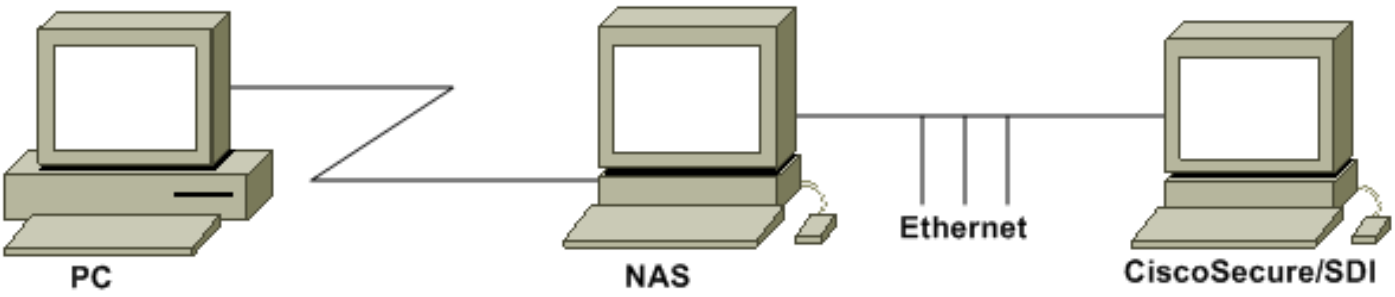
في هذا القسم، تُقدّم لك معلومات تكوين الميزات الموضحة في هذا المستند.

ملاحظة: أستخدم أداة بحث الأوامر (للعملاء المسجلين فقط) للحصول على مزيد من المعلومات حول الأوامر المستخدمة في هذا القسم.

الرسم التخطيطي للشبكة

يستخدم هذا المستند إعداد الشبكة التالي:

يتم تحويل جهاز الكمبيوتر إلى وحدة تخزين متصلة بالشبكة (NAS) ومودم ISDN، ويتم تكوينه لأمر **PPP متعدد الارتباطات**.



التكوينات

يستخدم هذا المستند التكوينات التالية:

- تكوين إدخال اسم المستخدم وكلمة المرور
- تكوين TokenCaching على CiscoSecure ACS Windows
- تكوين TokenCaching في CiscoSecure ACS UNIX

تكوين إدخال اسم المستخدم وكلمة المرور

في هذا المستند، يستخدم NAS بروتوكول مصادقة مصافحة الاستيبان (CHAP) لجلسة عمل PPP مع كلمة مرور SDI لمرة واحدة. إذا كنت تستخدم CHAP، فأدخل كلمة المرور في هذا النموذج:

- **اسم المستخدم**—fadi*pin+code (لاحظ * في اسم المستخدم)
- **كلمة السر**—كلمة السر

مثال على هذا: pin = 1234، chap password = cisco، username = fadi، والرمز الذي يظهر على الرمز المميز هو 987654. لذلك، يدخل المستخدم هذا:

- **اسم المستخدم** — فادي*1234987654
- **كلمة السر**—Cisco

ملاحظة: إذا تم تكوين CiscoSecure و NAS ل PAP، يمكن إدخال اسم المستخدم والرمز المميز بهذا الشكل:

- **username**—username*pin+code
- **كلمة المرور**—

أو:

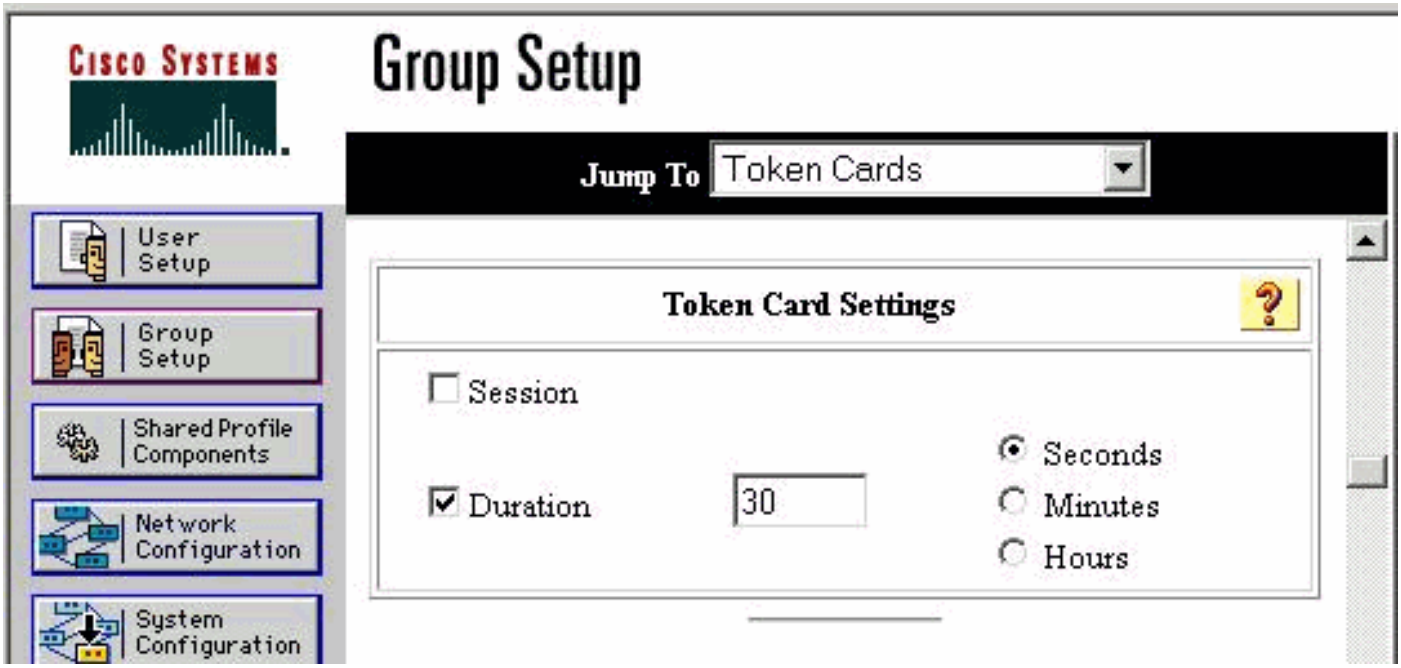
- **username**—username
- **كلمة السر**- سن+رمز

تكوين TokenCaching على CiscoSecure ACS Windows

يتم إعداد CiscoSecure ACS Windows user أو المجموعة كالمعتاد، مع تحديد PPP IP و PPP LCP إذا كنت تستخدم TACACS+. إذا كنت تستخدم RADIUS، فيجب تكوين التالي:

- السمة 6 = **Service_type = framed**
- السمة 7 = **framed_protocol = PPP**

بالإضافة إلى ذلك، يمكن التحقق من معلمات TokenCaching للمجموعة كما هو موضح في هذا المثال:



تكوين TokenCaching في CiscoSecure ACS UNIX

هناك أربع سمات TokenCaching. يتم تعيين السمة config_token_cache_absolute_timeout (بالثواني) في ملف install_directory/config/CSU.cfg$. يتم تعيين السمات الثلاث الأخرى (set server token-cache-timeout، and set server token-caching-timeout set server token-caching-expired-method) في ملفات تعريف المستخدم أو المجموعة. بالنسبة لهذا المستند، يتم تعيين السمة global config_token_cache_absolute_timeout على هذا في ملف install_directory/config/CSU.cfg$:

```
;NUMBER config_token_cache_absolute_timeout = 300
```

يتم تكوين ملفات تعريف سمة TokenCaching الخاصة بخادم المجموعة والمستخدم كما هو موضح في هذا المثال:

```
                                                                :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 ---!
```

التحقق من الصحة

لا يوجد حاليًا إجراء للتحقق من صحة هذا التكوين.

استكشاف الأخطاء وإصلاحها

يوفر هذا القسم معلومات يمكنك استخدامها لاستكشاف أخطاء التكوين وإصلاحها.

CiscoSecure ACS UNIX على Debug TokenCaching

يظهر سجل Cisco Secure UNIX هذا مصادقة ناجحة مع TokenCaching، عند حدوث المصادقة على قناتين 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
```

معلومات ذات صلة

- استجابات وتوجيهات أمان Cisco
- صفحة دعم منتج UNIX الآمن من Cisco
- مصدر المحتوى الإضافي الآمن من Cisco لصفحة دعم منتجات Windows
- الدعم التقني والمستندات - Cisco Systems

حول هذه الترجمة
ترجمت Cisco هذا المستند باستخدام مجموعة من التقنيات الآلية والبشرية لتقديم محتوى دعم للمستخدمين في جميع أنحاء العالم بلغتهم الخاصة. يُرجى ملاحظة أن أفضل ترجمة آلية لن تكون دقيقة كما هو الحال مع الترجمة الاحترافية التي يقدمها مترجم محترف. تخلي Cisco Systems مسئوليتها عن دقة هذه الترجمات وتُوصي بالرجوع دائمًا إلى المستند الإنجليزي الأصلي (الرابط متوفر).