

SVN UUID不匹配和SVN資料同步

目錄

[簡介](#)

[SVN UUID不匹配](#)

[驗證跨PCRFCClient VM的SVN同步](#)

[修復SVN儲存庫同步](#)

[重新初始化SVN同步](#)

[指令碼](#)

[check svn_sync.sh](#)

[svn uuid_mismatch.sh](#)

簡介

本文檔介紹如何在PCRFCClient01和PCRFCClient02之間同步Subversion控制元件(SVN)儲存庫。

SVN UUID不匹配

策略生成器(PB)中的所有策略配置均儲存為SVN儲存庫。因此，為了具有高可用性(HA)，您需要兩個PCRFCClient虛擬機器(VM)具有相同的SVN通用唯一識別符號(UUID)。但是，思科策略套件(CPS)版本7.X在PCRFCClient VM之間沒有相同的UUID。也就是說，較高的CPS版本具有相同的UUID。已輸入增強功能錯誤，思科錯誤ID [CSCuu85319](#)。如果兩個PCRFCClient0X VM的修訂版號相同，則可能會達到HA。

驗證跨PCRFCClient VM的SVN同步

由於兩個PCRFCClient VM的UUID不同，請輸入以下命令：

```
svn info http://pcrfclient01/repos/configuration
svn info http://pcrfclient02/repos/configuration
```

附註：使用這些命令時，必須提供PCRFCClient01和PCRFCClient02虛擬機器的確切主機名。您還可以使用指令碼[check svn_sync.sh](#)檢查SVN儲存庫是否在PCRFCClient VM之間同步。此指令碼在本文結尾提供。

您可能會注意到這些Quantum Network suite(QNS)錯誤日誌。可以使用指令碼[svn uuid_mismatch.sh](#)檢查這些錯誤消息。此指令碼在本文結尾提供。

```
2015-06-10 00:51:15,058 [pool-4-thread-1] WARN
c.b.r.impl.ReferenceDataManager - SVN directory failed in updating
2015-06-10 01:51:44,050 [pool-4-thread-1] WARN
c.b.r.impl.ReferenceDataManager - SVN directory failed in updating
2015-06-10 03:52:41,061 [pool-4-thread-1] WARN
c.b.r.impl.ReferenceDataManager - SVN directory failed in updating
```

==Service-qns-1.log==

```
svn: Repository UUID '70e98d90-b9c7-4c5b-a3db-681dd68c62a6'
doesn't match expected UUID 'ed2f2f76-f588-443e-9bb8-3fb05abd903b'
svn: Repository UUID '70e98d90-b9c7-4c5b-a3db-681dd68c62a6'
doesn't match expected UUID 'ed2f2f76-f588-443e-9bb8-3fb05abd903b'
svn: Repository UUID '70e98d90-b9c7-4c5b-a3db-681dd68c62a6'
doesn't match expected UUID 'ed2f2f76-f588-443e-9bb8-3fb05abd903b'
```

如下圖所示，您可以觀察到各個VM的UUID不同，但版本必須相同。如果兩個PCRFCClient VM中的SVN版本相同，則存在SVN儲存庫同步，隨後您將具有高可用性。

===PCRFCClient01===

```
[root@pcrfclient01 ~]# svn info http://pcrfclient01/repos/configuration
Path: configuration
URL: http://pcrfclient01/repos/configuration
Repository Root: http://pcrfclient01/repos
Repository UUID: ed2f2f76-f588-443e-9bb8-3fb05abd903b
Revision: 392 <<<< revision number
Node Kind: directory
Last Changed Author: broadhop
Last Changed Rev: 392
Last Changed Date: 2015-06-01 15:52:12 -0600 (Mon, 01 Jun 2015)
```

===PCRFCClient02===

```
[root@pcrfclient01 ~]# svn info http://pcrfclient02/repos/configuration
Path: configuration
URL: http://pcrfclient02/repos/configuration
Repository Root: http://pcrfclient02/repos
Repository UUID: 70e98d90-b9c7-4c5b-a3db-681dd68c62a6
Revision: 392 <<<< revision number
Node Kind: directory
Last Changed Author: broadhop
Last Changed Rev: 392
Last Changed Date: 2015-06-01 15:52:12 -0600 (Mon, 01 Jun 2015)
```

修復SVN儲存庫同步

上一節說明如何檢查SVN儲存庫同步。本節討論如何修復SVN同步。假定PCRFCClient01為主要，PCRFCClient02為輔助，並且PCRFCClient02 SVN儲存庫不同步。

完成以下步驟，使PCRFCClient02 SVN儲存庫與PCRFCClient01同步：

1. 在PCRFCClient01上，禁用SVN同步作業：

```
crontab -e
```

```
Comment this cron job entry:
```

```
/usr/bin/svnsync sync http:// pcrfclient02/repos-proxy-sync
```

附註：此處，PCRFCClient02是PCRFCClient02 VM的客戶主機名。

2. 登入到PCRFCClient02並編輯「/etc/httpd/conf.d/subversion.conf」，然後使用以下粗體文本對

其進行更新：

```
<Location /repos-proxy-sync>
    DAV svn
    SVNPath /var/www/svn/repos
    Order deny,allow
    Deny from all
    Allow from pcrfclient01 <<< # customer host name of PCRFCClient01 VM
</Location>
```

3. 在PCRFCClient02 VM上，輸入以下命令以重新啟動HTTPD服務。

```
service httpd restart
```

4. 在PCRFCClient01上，輸入以下命令以同步SVN資料。SVN儲存庫同步可能需要時間，這取決於修訂版本的數量和大小。

```
/usr/bin/svnsync sync http://pcrfclient02/repos-proxy-sync
```

附註：此處，PCRFCClient02是PCRFCClient02 VM的客戶主機名。如果報告鎖定錯誤，請輸入以下命令，然後重試svnsync命令：

```
/usr/bin/svn propdel svn:sync-lock --revprop -r 0 http:// pcrfclient02/repos-proxy-sync
```

如果仍然出現錯誤，請繼續[重新初始化SVN同步](#)。

5. 要驗證SVN儲存庫的修訂版號在PCRFCClient01和PCRFCClient02中是否相同，請使用[check_svn_sync.sh](#)指令碼或[Verify SVN Synchronization Across PCRFCClient VM](#)中提到的CLI命令。

6. 在PCRFCClient01上，啟用SVN同步作業：

```
crontab -e
```

```
/usr/bin/svnsync sync http://pcrfclient02/repos-proxy-sync
```

重新初始化SVN同步

只有在[修復SVN儲存庫同步](#)一節中的步驟4中遇到錯誤時，才執行這些步驟。如果在上一節中沒有遇到任何錯誤，則可以忽略這些步驟。

附註：執行這些步驟之前，請先聯絡思科技術支援團隊。

1. 重新初始化SVN同步之前，請先在haproxy配置中從svn_proxy禁用PCRFCClient02: 轉到standby lb (本例中為lb02) 並修改「/etc/haproxy/haproxy.cfg」，然後註釋掉此處顯示的粗體行：

```
listen svn_proxy lbvip02:80
mode http
balance roundrobin
option httpchk
option httpclose
option abortonclose
server pcrfclient01 pcrfclient01:80 check
#server pcrfclient02 pcrfclient02:80 check backup
```

前往作用中lb (此範例中為lb01) 並輸入以下命令以進行HA容錯移轉：

```
/etc/init.d/heartbeat restart
```

驗證「haproxy/memcached/VIP」是否移至其他lb (在此範例中為lb02)。轉到standby lb (現在lb01為standby) 並修改「/etc/haproxy/haproxy.cfg」並註釋掉粗體行：

```
listen svn_proxy lbvip02:80
mode http
balance roundrobin
option httpchk
option httpclose
option abortonclose
server pcrfclient01 pcrfclient01:80 check
```

```
#server pcrfclient02 pcrfclient02:80 check backup
```

轉到active lb (現在lb02為active) 並輸入以下命令以執行HA容錯移轉：

```
/etc/init.d/heartbeat restart
```

驗證「haproxy/memcached/VIP」是否移至其他lb (在本例中為lb01)。

2. 在PCRFCliient02上，輸入以下命令以重新初始化SVN儲存庫目錄：

```
tar -czf /var/tmp/repos.tar.gz /var/www/svn/repos
```

```
mkdir -p /var/www/svn/repos
```

```
rmdir /var/www/svn/repos
```

```
/usr/bin/svnadmin create /var/www/svn/repos
```

```
chown -R apache:apache /var/www/svn/repos
```

3. 在PCRFCliient02上，使用以下資料在路徑「/var/www/svn/repos/hooks/」中建立名為「pre-revprop-change」的檔案：

```
#!/bin/sh
```

```
#/var/www/svn/repos/hooks/pre-revprop-change
```

```
exit 0
```

4. 在PCRFCliient02上，輸入以下命令以更改以前建立的檔案的許可權，並執行HTTPD服務重新啟動：

```
chmod 700 /var/www/svn/repos/hooks/pre-revprop-change
```

```
chown apache:apache /var/www/svn/repos/hooks/pre-revprop-change
```

```
service httpd restart
```

5. 在PCRFCliient01上，輸入以下命令以執行SVN資料同步：

```
/usr/bin/svnsync init http://pcrfclient02/repos-proxy-sync
```

```
http:// pcrfclient01/repos-proxy-sync
```

```
/usr/bin/svnsync sync http:// pcrfclient02/repos-proxy-sync
```

附註：根據您的設定，提供PCRFCliient01和PCRFCliient02的相應主機名，以代替PCRFCliient01和PCRFCliient02。

6. 要驗證SVN儲存庫是否已同步，請完成[驗證SVN跨PCRFCliient VM同步](#)中的步驟。

7. 在haproxy配置中從SVN代理啟用PCRFCliient02。基本上，取消註釋您在步驟1中註釋掉的行。請轉到standby lb (本例中為lb02) 並修改「/etc/haproxy/haproxy.cfg」並取消對加粗行的註釋：

```
listen svn_proxy lbvip02:80
```

```
mode http
```

```
balance roundrobin
```

```
option httpchk
```

```
option httpclose
```

```
option abortonclose
```

```
server pcrfclient01 pcrfclient01:80 check
```

```
server pcrfclient02 pcrfclient02:80 check backup
```

驗證「haproxy/memcached/VIP」是否移至其他lb (在本例中為lb02)。

```
/etc/init.d/heartbeat restart
```

驗證「haproxy/memcached/VIP」是否移至其他lb (在本例中為lb02)。轉到standby lb (現在lb01為standby) 並修改「/etc/haproxy/haproxy.cfg」並取消註釋此處顯示的粗體行：

```
listen svn_proxy lbvip02:80
```

```
mode http
```

```
balance roundrobin
```

```
option httpchk
```

```
option httpclose
```

```
option abortonclose
```

```
server pcrfclient01 pcrfclient01:80 check
```

```
server pcrfclient02 pcrfclient02:80 check backup
```

轉到active lb (nowlb02為active) 並輸入以下命令以完成HA容錯移轉：

```
/etc/init.d/heartbeat restart
```

驗證「haproxy/memcached/VIP」是否移至其他lb (在此範例中為lb01)。

8. 在PCRFCliient01上，啟用SVN同步作業：

```
crontab -e/usr/bin/svnsync sync http://pcrfclient02/repos-proxy-sync
```

指令碼

check_svn_sync.sh

請將此指令碼放在CPS系統中，以便檢查PCRFClient VM之間的SVN同步。

```
#!/bin/bash
a=$(svn info http://pcrfclient01/repos/configuration | grep -i Revision | grep -o '[0-9]*')
b=$(svn info http://pcrfclient02/repos/configuration | grep -i Revision | grep -o '[0-9]*')
if [ "$a" == "$b" ]; then
echo -e "SVN repository Synchronization:\e[0;32m PASS \e[0m"
echo -e "PCRFClient01 SVN repository revision number is : \e[1;33m $a \e[0m"
echo -e "PCRFClient02 SVN repository revision number is : \e[1;33m $b \e[0m"
else
echo -e "SVN repository Synchronization : \e[0;31m FAIL \e[0m"
echo -e "PCRFClient01 SVN repository revision number is : \e[1;33m $a \e[0m"
echo -e "PCRFClient02 SVN repository revision number is : \e[1;33m $b \e[0m"
fi
```

svn_uuid_mismatch.sh

請將此指令碼放在CPS系統中，以便監控來自各個QNS虛擬機器的與SVN UUID相關的錯誤消息。

```
#!/bin/bash
for HN in `hosts.sh |grep qns`;
do echo -e "\e[1;32m $HN\e[0m";
echo -e "\e[1;33m===qns-1.log===\e[0m"
ssh $HN grep -wh "SVN directory failed" 2>/dev/null /var/log/broadhop/qns-1.log | tail -10;
echo -e "\e[1;33m===service-qns-1.log===\e[0m"
ssh $HN grep -wh "match expected UUID" 2>/dev/null /var/log/broadhop/service-qns-1.log | tail -10;
done
```