Second Key Generation

Version: 74
Last modification: Dec 04, 2012 14:51
Estimated time: 1 hour and 45 minutes

Roles
- KGA (Key Generation Administrator) facilitates key generation procedure and records data on their script copy
- SA (System Administrator) provides access to the signing box
- KSO (Keystore Security Officer) authorize keystore related operations, including backup and restoration
- DSO (Device Security Officer) authorize device related operations, including backup and restoration
- WI (Witness) attends the event as an observer.
- SAU (Security Auditor) reviews and audits the key generation procedure.

Abbreviations
- TEB: Tamper-Evident Bag
- MBC: Master Backup Copy
- OBC: Operative Backup Copy
- FD : Flash Drive

Materials

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop</td>
<td>1</td>
</tr>
<tr>
<td>CD with Live Linux Distribution</td>
<td>3</td>
</tr>
<tr>
<td>Projector</td>
<td>1</td>
</tr>
<tr>
<td>Printer</td>
<td>1</td>
</tr>
<tr>
<td>Photocopier</td>
<td>1</td>
</tr>
<tr>
<td>Flash Drives properly labeled and formatted</td>
<td>6</td>
</tr>
<tr>
<td>Spare formatted Flash Drives</td>
<td>2</td>
</tr>
<tr>
<td>Tamper-Evident bags</td>
<td>6</td>
</tr>
<tr>
<td>Pre-generated secure password for device backup</td>
<td>3</td>
</tr>
<tr>
<td>Sysadmin brings ssh key to access the signer</td>
<td>1</td>
</tr>
<tr>
<td>Hard copies of this script</td>
<td>9</td>
</tr>
<tr>
<td>Copy of previous Key Generation Procedure script</td>
<td>1</td>
</tr>
<tr>
<td>Copy of previous Hot-Standby Signer Initialization script</td>
<td>1</td>
</tr>
<tr>
<td>Participant sign-in sheet</td>
<td>1</td>
</tr>
</tbody>
</table>

Participants

<table>
<thead>
<tr>
<th>Title</th>
<th>Org</th>
<th>Printed Name</th>
<th>Signature</th>
<th>Date</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>KGA</td>
<td>NZRS</td>
<td>Sebastian Castro</td>
<td>[Signature]</td>
<td>05-12-2012</td>
<td>09:08</td>
</tr>
<tr>
<td>SA</td>
<td>Catalyst</td>
<td>James Dempsey</td>
<td>[Signature]</td>
<td>05-12-2012</td>
<td>09:12</td>
</tr>
<tr>
<td>DSO1</td>
<td>NZRS</td>
<td>Dave Baker</td>
<td>[Signature]</td>
<td>05-12-2012</td>
<td>09:10</td>
</tr>
<tr>
<td>DSO2</td>
<td>Knossos</td>
<td>John Rumsey</td>
<td>[Signature]</td>
<td>05-12-2012</td>
<td>09:08</td>
</tr>
</tbody>
</table>

[Signature]

New Zealand Registry Services
Safety Instructions
Estimated time: 5 min
Catalyst representative explains the safety procedures to follow in case of fire or earthquake, including Emergency Exits, Fire-fighting equipment and Assembly Point.

Internal Security Policy
Estimated time: 5 min
During the execution of this procedure, personal electronic devices may be used, as long as usage doesn't interfere with the normal course of the procedure. This includes mobile phones, laptops, etc. Mobile phones could be used to make phone calls in case of an emergency. One still camera may be present to take single images for archiving purposes. Video cameras and recording devices are not permitted.

Procedure

Initial preparation
Estimated time: 10 min
1. All the participants enter the room
2. KGA proceeds to validate the presence of all required participants
   3. Each participant will sign the KGA script copy. If the participant is not fulfilling a trusted role, it must provide a government-issued identification.
4. KGA retrieves:
   5. Laptop (includes power cable, video cable, power extension)
   6. CD,
   7. Flash Drives
   8. Tamper-Evident Bags

Laptop setup
Estimated time: 15 min
9. SA sets up the laptop for the key generation procedure
10. Connects power cable, network cable, and projector
11. Powers up laptop, hit ENTER to access boot menu
12. Boot-up laptop using a bootable CD
13. Enables display
14. Configures printer and print test page
15. Open terminal, and maximize for visibility
16. SA verifies the integrity of the Live CD by comparing the digest

**Second Key Generation Procedure**

```bash
openssl dgst -c -sha256 /dev/sr0
SHA256(/dev/sr0)=
f0:cl:51:a8:3a:4c:b3:ac:3d:26:16:f7:54:76:0e:78:
```

**TIME**

**9:27**

Matches record?  **YES**

17. SA verifies time and date on the laptop

```bash
root@laptop# date
```

**TIME**

18. KGA records date and time on their script copy

Date:  9:28:50  NZDT

Time:  5-12-2012

**Access to the signing box**

*Estimated time: 5 min*

19. KGA selects Flash Drive labeled **Key Gen Log**, records the serial number on their script copy and hands it out to SA

Flash Drive Serial #  0019E000F9AA95E87080D0389

20. SA plugs in the Flash Drive. By default the Flash Drive will be auto-mounted and its contents available at `/media/KEY_GEN_LOG`.

21. SA elevate privileges to access the Flash Drive

```bash
user@laptop$ sudo bash
root@laptop# 
```

**TIME**

**9:30**

22. SA verifies the FD serial number matches the serial number recorded on the script

```bash
lsusb -v -d 0x0951:0x1607 | grep -C 1 iProduct
iManufacturer 1 Kingston
iProduct 2 DataTraveler 2.0
iSerial 3 0019E000F9AA95E87080D0389
```

**TIME**

**9:31**

23. SA starts logging via **script**

```bash
root@laptop# cd /media/KEY_GEN_LOG
root@laptop# script script -`date +"%Y%m%d"` .log
Script started, file is script-20121205.log
```

**TIME**

**9:33**

24. SA accesses the standby signing box via SSH using their own account, providing their own SSH identity

```bash
ssh -i catalyst-sysadmin-ssh-key
sysadmin@sign2@internal.srs.net.nz
```

**TIME**

25. KGA checks the fingerprint for the server matches the records

**sign1 fingerprint**


**sign2 fingerprint**

Second Key Generation Procedure

The authenticity of host 'sign2.internal.srs.net.nz' can't be established.
Are you sure you want to continue connecting (yes/no)? yes

Matches record?  

26. SA enters the directory /var/lib/dnssec/keygen. Files generated during the key generation procedure will be stored here for later retrieval.

```bash
sysadmin@sign2: sudo -s
[sudo] password for sysadmin:
[/home/sysadmin]
root@sign2: cd /var/lib/dnssec/keygen
[/var/lib/dnssec/keygen]
root@sign2:
```

HSM Verification

Estimated time: 5 min

27. SA retrieves the HSM public key fingerprint

```bash
sysadmin@sign2: scadiag -f mca0
d34d-ba64-ac50-eb28-b785-5c09-ebee-201f-db7c-13ef
```

28. KGA verifies the HSM Fingerprint matches what's recorded in the previous script (step 36)

HSM Public Key Fingerprint

```plaintext
4f9d-9a88-f9e6-56a2-bc42-ad74
324c-9a64-f44f-2a76
```

Matches record?  

Key Purging

Estimated time: 5 min

Delete all the keys stored in the HSM that are no longer needed.

29. SA verifies the signer is the standby signer, output must indicate the standby_signer is LOCAL

```bash
sysadmin@sign2: get_active_signer
active_signer: 192.168.58.14|FULLY_AGREE|REMOTE
standby_signer: 192.168.62.14|FULLY_AGREE|LOCAL
```

30. SA lists the contents of the HSM. It must contain the same number of keys as seen after the previous Key Generation Procedure
Second Key Generation Procedure

```
ods-hsmutil list sca6000 | head -5
Listing keys in repository: sca6000
140 keys found.

Repository ID Type
----------- -- ------
sca6000 160d29b6d32b301356a22f545e1a5ddd RSA/2048
sca6000 33b6e77e122419a7e6893d2c5e2bcf6 RSA/2048
sca6000 9d893962239be58bfcd63fd45a6454a5 RSA/2048
sca6000 5ac0c4de0626543295d37bc850200f8e RSA/2048
sca6000 76394a2af741e324ad49646b4b59dd53 RSA/2048
```

31. Proceed to delete all unused keys in active policies
```
sudo -u opensnsec ods-purge-keys.sh
```

32. SA lists the contents of the HSM, to show a reduced number of keys
```
ods-hsmutil list sca6000 | head -5
Listing keys in repository: sca6000
115 keys found.
```

Key generation

*Estimated time: 15 min*

Create all the necessary keys for fourteen months of operation (one year plus two months extra for overlap).

33. SA executes the script to generate the keys for all active policies
```
sudo -u opensnsec ods-keygen.sh P14M
```

(Info) The key generation script will run a sanity check on the list of keys previous and after the generation step, to make sure only new keys are added and no existing keys are deleted.

34. SA prints the number of keys present in the HSM. Output would look as below:
```
ods-hsmutil list sca6000 | head -5
Listing keys in repository: sca6000
200 keys found.
```

Backup generation

*Estimated time: 10 min*

35. SA opens a second terminal and logs into the signing box using their own account.
```
ssh -i catalyst-sysadmin-ssh-key
sysadmin@sign2.internal.srs.net.nz
```

TIME
---
9:40
9:41
9:44
9:45
36. SA executes backup script in the first terminal. The backup files will be written to
/var/lib/dnssec/keygen/key-backup-YYYY-MM-DD.tar.gz

```
sudo -s
export-keydata nz-dnssec-keystore
```

Backups will be written to
/var/lib/dnssec/keygen/key-backup-YYYY-MM-DD.tar.gz
Exporting KASP database...
SQLite database set to: /var/opendnssec/kasp.db

Backing up keystore nz-dnssec-keystore...

You will be prompted for Keystore Security Officer (KSO) credentials. After entering them, the backup will pause while other Keystore Security Officers authorize the backup operation.

Press enter to continue.

37. KSO1 authorizes the backup using their password

```
Keystore = nz-dnssec-keystore.600121.{b129f5fa} (local)
Security Officer Login: nz-kso1
Security Officer Password:
```

NOTICE: Please wait while the other required 1 security officers authenticate this command. This command will time out in 5 minutes.

```
TIME
9:47
```

38. SA executes the HSM interface in the second window
```
scamgr -k nz-dnssec-keystore
```
Keystore = nz-dnssec-keystore.600121.{b129f5fa} (local)

```
TIME
9:49
```

39. A second KSO logs into the HSM using the second terminal to authorize the backup.

```
Security Officer Login: nz-kso2
Security Officer Password:
```

NOTICE: A Multi-Admin command is currently in progress. You are a member of the Multi-Admin role and may approve this command.
Command: backup
Initiating SO: nz-kso1

Authorization successful

```
| Time | 9:50 |
```

| i Any KSO pair combination can carry out this operation, using nz-kso1, and nz-kso2 is only relevant for the example |

40. SA closes the second HSM interface and window
```
scamgr> quit
```

```
TIME
```

41. The first terminal will show the backup command was authorized and will proceed. Output will look like the following example:
Second Key Generation Procedure

Update: Authenticated security officers: nz-kson
Update: Authenticated security officers: nz-kson nz-kso2

Done backing up keystore nz-dnssec-keystore. The sha256sum of this full keystore backup is

Backing up HSM Device Configuration...
You will be prompted for Device Security Officer (DSO) credentials and a Password to encrypt to the device backup.

Press enter to continue.

42. DSO1 authorizes the device backup with their password

Security Officer Login: nz-ds01
Security Officer Password:

43. SA enters the password to protect the backup, using a pre-generated password. Output should look as below:

Enter a password to protect the data:
Confirm password:
Backup to /tmp/tmp.cgHkVsl862/device-backup-YYYY-MM-DD successful.

Done backing up HSM device. The sha256sum of this device backup is
d:61:03:d8:9d:9b:c7:7e:38:0e:72:fe:02

Exported keystore Info:
Keystore: nz-dnssec-keystore
Serial #: 605403
Keystore ID: 5199201
All backups have been exported to
/var/lib/dnssec/keygen/key-backup-YYYY-MM-DD.tar.gz
Hash of key-backup-YYYY-MM-DD.tar.gz has been written to
key-backup-YYYY-MM-DD.tar.gz.sha256sum (sha256sum:

44. SA reads the digest from the screen, KGA records on its script copy

Keystore backup file digest
36:64:ad:
84:5b

45. SA closes the root session

root@sign2: exit

46. SA logs out from the signing box
Creating Master Backup Copy

Estimated time: 5 min

47. KGA takes the Flash Drive labeled as Master Copy to serve as Master Copy Container. KGA records the serial number on its script copy.

Flash Drive Serial #

48. KGA passes the Flash Drive to SA
49. SA plugs Flash Drive into the laptop
50. SA verifies the FD serial number matches the serial number recorded on the script.

51. SA copies the backup files from the signer to the Flash Drive

52. SA checks the backup file integrity

Creating Backup Operative Copies

Wellington Operative Backup Copy

Estimated time: 5 min

53. KGA picks Flash Drive labeled WELLINGTON, and records the serial number in its script copy.

54. KGA hands out the Flash Drive to SA
55. SA plugs the FD into the laptop
56. SA verifies the FD serial number matches the serial number recorded on the script. This command will show two serial numbers, one for the Master Backup and one for the Wellington Flash Drive.

57. SA copies the MBC FD contents into the Wellington OBC FD
58. SA checks the integrity of the backup

```bash
cd /media/REGENCY
sha256sum -c key-backup-YYYY-MM-DD.tar.gz.sha256sum
key-backup-YYYY-MM-DD.tar.gz: OK
```

59. SA unmounts and unplugs the OBC FD

```bash
umount /media/REGENCY
```

60. SA hands over the FD to the KGA

61. KGA labels a TEB as REGENCY, <DATE>, NZRS DNSSEC Key Backup

62. KGA records the TEB serial number in its script copy

```
TEB Serial #
32 34864
```

63. KGA places the REGENCY OBC FD in the TEB

64. KGA places copy of the Device Backup Password in the TEB

65. KGA seals the TEB

66. KGA tears off the TEB pre-perforated tab, and tapes it to its copy of the script

67. KGA hands out the TEB to Catalyst Representative

68. Catalyst Representative confirms the TEB serial matches the script log and signs in acknowledgement

```
Catalyst Representative signature
```

---

Albany Operative Backup Copy

*Estimated time: 5 min*

69. KGA picks the Flash Drive labeled ALBANY, and records the serial number in its script copy.

```
Flash Drive Serial #
00ACC0EC 34F1-FFB9067172675
```

70. KGA hands out the FD to the SA

71. SA plugs the FD into the laptop

72. SA verifies the FD serial number matches the serial number recorded on the script

```bash
lsusb -v -d 0x0951:0x1e65 | grep -C 1 iProduct
iManufacturer 1 Kingston
iProduct 2 DT 100 G2
iSerial 3 00ACC0EC34BF90671D25F1
```

73. SA copies the MCB FD contents into the Albany OBC FD

```
rsync -avW /media/MASTER_BACKUP/ /media/ALBANY/
```

74. SA checks the integrity of the backup

```bash
cd /media/ALBANY
sha256sum -c key-backup-YYYY-MM-DD.tar.gz.sha256sum
key-backup-YYYY-MM-DD.tar.gz: OK
```
75. SA unmounts and unplugs the OBC FD
   cd /
   umount /media/ALBANY

76. SA hands out the FD to the KGA
77. KGA labels a TEB as ALBANY, <DATE>, NZRS DNSSEC Key Backup
78. KGA records the TEB serial number in its script copy
   TEB Serial # 3234868

79. KGA places the ALBANY OBC FD in the TEB
80. KGA places copy of the Device Backup Password in the TEB
81. KGA seals the TEB
82. KGA tears off the TEB pre-perforated tab, and tapes it to its copy of the script

83. KGA hands out the TEB to Knossos Representative
84. Knossos Representative confirms the TEB serial matches the script log and signs in acknowledgement
   Knossos Representative signature

Auckland Operative Backup Copy
Estimated time: 5 min
85. KGA picks Flash Drive labeled AUCKLAND, and records the serial number in its script copy
   Flash Drive Serial # 001CC0EC32BC- FB 9067122608

86. KGA hands out the FD to the SA
87. SA plugs the FD into the laptop
88. SA verifies the FD serial number matches the serial number recorded on the script
   lsusb -v -d 0x0951:0x1653 | grep -C 1 iProduct
   iManufacturer 1 Kingston
   iProduct 2 DT 100 G2
   iSerial 3 001CC0EC34BEFB90671D25F1
   
   iManufacturer 1 Kingston
   iProduct 2 DT 100 G2
   iSerial 3 001CC0EC32BCFB9067122608

89. SA copies the MCB FD contents into the AUCKLAND OBC FD
   rsync -avW /media/MASTER_BACKUP/ /media/AUCKLAND

90. SA checks the integrity of the backup
   cd /media/AUCKLAND
   sha256sum -c key-backup-YYYY-MM-DD.tar.gz.sha256sum
   key-backup-YYYY-MM-DD.tar.gz: OK

91. SA unmounts and unplugs the OBC FD
   cd /
   umount /media/AUCKLAND
Second Key Generation Procedure

92. SA hands out the FD to the KGA
93. KGA labels a TEB as **AUSTRALIAN, <DATE>, NZRS DNSSEC Key Backup**
94. KGA records the TEB serial number in its script copy
   
   **TEB Serial #** 32 3 4 8 6 7

95. KGA places the AUSTRALIAN OBC FD in the TEB
96. KGA places copy of the Device Backup Password in the TEB
97. KGA seals the TEB
98. KGA tears off the TEB pre-perforated tab, and tapes it to its copy of the script

99. KGA hands out TEB to OSS Representative
100. OSS Representative confirms the TEB serial matches the script log and signs in acknowledgement
   
   **OSS Representative signature**

Finishing steps

*Estimated time: 3 min*

101. SA unmounts and unplugs the MBC FD
    
    ```
    cd / 
    umount /media/MASTER_BACKUP
    ```
    
    **TIME** 10:24

102. SA hands out the MBC FD to the KGA
103. KGA labels a TEB as **Master Copy, <DATE>, NZRS DNSSEC Key Backup**
104. KGA records the TEB serial number in its script copy
   
   **TEB Serial #** 3 2 3 4 8 6 5

105. KGA places the MBC FD in the TEB
106. KGA places copy of the Device Backup Password in the TEB
107. KGA seals the TEB
108. KGA tears off the TEB pre-perforated tab, and tapes it to its copy of the script

109. KGA hands out TEB to KSO1
110. KSO1 confirms the TEB serial matches the script log and signs in acknowledgement
    
    **KSO1 signature**

Closing steps

*Estimated time: 12 min*

111. SA finishes script logging
    
    ```
    root@laptop> exit
    ```
    
    **TIME** 11:26

112. KGA selects Flash Drive labeled **Key Gen Copy** and hands it out to SA
113. SA plugs in the Flash Drive
114. SA copies **Key Gen Log** Flash Drive contents into **Key Gen Copy** Flash Drive
Second Key Generation Procedure

```bash
rsync -avW /media/KEY_GEN_LOG/ /media/KEYGEN_COPY
```

115. SA generates a printable copy of the script

```
cd /media/KEYGEN_COPY
enscript -G -U 2 -o script-`date +%Y%m%d`.ps script-`date +%Y%m%d`.log
```

116. SA generates sha256 digest for the printable copy of the script. Output should look like this:

```bash
openssl dgst -c -sha256 script-`date +%Y%m%d`.ps
b0:47:25:d3:1b:e4
```

117. KGA records the sha256 digest into the script copy

```
sha256 digest
:35:70:eb:ce:de:9c:99:
```

118. SA prints the script

```
 lpr script-`date +%Y%m%d`.ps
```

119. SA copies the printable copy to the Key Gen Log Flash Drive

```
 cp script-`date +%Y%m%d`.ps /media/KEY_GEN_LOG
```

120. SA unmounts KEY_GEN_LOG FD

```
 cd /
 umount /media/KEY_GEN_LOG
```

121. SA unplugs Flash Drive and hands it out to KGA
122. KGA takes a TEB and records the serial number in its script copy

**TEB Serial #**

```
3234266
```

123. KGA places KeyGen_Log FD in the TEB and seals it
124. KGA tears off the TEB perforated tab. and tapes it to its copy of the script

```
NO. 3234866
```

125. SA unmounts KEYGEN_COPY FD and hands it out to KGA

```
 cd /
 umount /media/KEYGEN_COPY
```

126. SA unmounts and unplugs the Flash Drive carrying his key
127. SA shuts down laptop

```
 shutdown -h now
```

128. SA disconnects cables from laptop
129. Unplug laptop cables
130. KSO1 takes TEB containing Key Generation Log FD, TEB containing Master Backup Copy and copies of the script log for secure storage
131. KGA signs off the key generation procedure

Signature

Date/Time

132. KGA makes at least 3 photocopies of its copy of the script: one for onsite storage, offsite storage, one for KGA. Additional copies can be made by participants request.