Certmonger, SELinux and Keystores in random locations | Adam Young's Web Log

adam.younglogic.com (http://adam.younglogic.com/2018/02/certmonger-selinux-keystores/) · by Adam Young

In my last post, (http://adam.younglogic.com/2018/02/java-and-certmonger/) SELinux was reporting AVCs when certmonger tried to access an NSS Database in a non-standard location. To get rid of the AVC, and get SELinx to allow the operations, we need to deal with the underlying cause of the AVC.

Bottom Line Up Front:

Run these commands.

```
[root@sso standalone]# semanage fcontext -a -t cert_t $PWD/"keystore(/.*)?"
[root@sso standalone]# restorecon -R -v keystore
scontext=system_u:system_r:certmonger_t:s0 tcontext=unconfined_u:object_r:et
c_t:s0 tclass=file
```

[root@sso standalone]# semanage fcontext -a -t cert_t $PWD/"keystore(/.*)?"
[root@sso standalone]# restorecon -R -v keystore
scontext=system_u:system_r:certmonger_t:s0
tcontext=unconfined_u:object_r:etc_t:s0 tclass=file

Thanks to OZZ for that.
(https://twitter.com/jaosorior/status/964024716444028928)

Here's How I got there.

Debugging

The original error was:

```
type=AVC msg=audit(1518668324.903:6506): avc:  denied  { write } for  pid=15
310 comm="certmonger" name="cert9.db" dev="vda1" ino=17484324 scontext=syste
m_u:system_r:certmonger_t:s0 tcontext=unconfined_u:object_r:etc_t:s0 tclass=
file
```

type=AVC msg=audit(1518668324.903:6506): avc: denied { write } for pid=15310 comm="certmonger" name="cert9.db" dev="vda1" ino=17484324 scontext=system_u:system_r:certmonger_t:s0 tcontext=unconfined_u:object_r:etc_t:s0 tclass=file

Since I created the NSS database without a relabel or other operation, it is still in its default form. Looking at the whole subdirectory:

```
[root@sso standalone]# ls -Z keystore
-rw-------. root root unconfined_u:object_r:etc_t:s0   cert8.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   cert9.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   key3.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   key4.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   pkcs11.txt
-rw-------. root root unconfined_u:object_r:etc_t:s0   secmod.db
```

[root@sso standalone]# ls -Z keystore -rw-------. root root unconfined_u:object_r:etc_t:s0 cert8.db -rw-------. root root unconfined_u:object_r:etc_t:s0 cert9.db -rw-------. root root

unconfined_u:object_r:etc_t:s0 key3.db -rw-------. root root
unconfined_u:object_r:etc_t:s0 key4.db -rw-------. root root
unconfined_u:object_r:etc_t:s0 pkcs11.txt -rw-------. root root
unconfined_u:object_r:etc_t:s0 secmod.db

Compare with a properly configure system

Lets contrast this with an NSS Database that is properly labeled. For example, on my IPA server, where SELInux is enforcing, I can look at certmonger and see where it is tracking files.

```
$ ssh cloud-user@idm.ayoung.rdusalab
Last login: Wed Feb 14 22:53:20 2018 from 10.10.120.202
[cloud-user@idm ~]$ sudo -i
[root@idm ~]# getcert list
Number of certificates and requests being tracked: 9.
...
Request ID '20180212165505':
        status: MONITORING
        stuck: no
        key pair storage: type=NSSDB,location='/etc/httpd/alias',nickname='S
erver-Cert',token='NSS Certificate DB',pinfile='/etc/httpd/alias/pwdfile.txt
'
        certificate: type=NSSDB,location='/etc/httpd/alias',nickname='Server
-Cert',token='NSS Certificate DB'
...
```

$ ssh cloud-user@idm.ayoung.rdusalab Last login: Wed Feb 14 22:53:20 2018 from 10.10.120.202 [cloud-user@idm ~]$ sudo -i [root@idm ~]# getcert list Number of certificates and requests being tracked: 9. … Request ID '20180212165505': status: MONITORING stuck: no key pair storage: type=NSSDB,location='/etc/httpd/alias',nickname='Server-Cert',token='NSS Certificate DB',pinfile='/etc/httpd/alias/pwdfile.txt' certificate: type=NSSDB,location='/etc/httpd/alias',nickname='Server-Cert',token='NSS Certificate DB' …

So looking at

```
[root@idm ~]# ls -Z /etc/httpd/alias
-rw-r-----. root apache unconfined_u:object_r:cert_t:s0  cert8.db
-rw-r-----. root apache unconfined_u:object_r:cert_t:s0  cert8.db.orig
-rw-------. root root   unconfined_u:object_r:cert_t:s0  install.log
-rw-------. root root   system_u:object_r:ipa_cert_t:s0  ipasession.key
-rw-r-----. root apache unconfined_u:object_r:cert_t:s0  key3.db
-rw-r-----. root apache unconfined_u:object_r:cert_t:s0  key3.db.orig
lrwxrwxrwx. root root   system_u:object_r:cert_t:s0      libnssckbi.so -> /u
sr/lib64/libnssckbi.so
-rw-------. root apache unconfined_u:object_r:cert_t:s0  pwdfile.txt
-rw-r-----. root apache unconfined_u:object_r:cert_t:s0  secmod.db
-rw-r-----. root apache unconfined_u:object_r:cert_t:s0  secmod.db.orig
```

[root@idm ~]# ls -Z /etc/httpd/alias -rw-r-----. root apache unconfined_u:object_r:cert_t:s0 cert8.db -rw-r-----. root apache unconfined_u:object_r:cert_t:s0 cert8.db.orig -rw-------. root root unconfined_u:object_r:cert_t:s0 install.log -rw-------. root root system_u:object_r:ipa_cert_t:s0 ipasession.key -rw-r-----. root apache unconfined_u:object_r:cert_t:s0 key3.db -rw-r-----. root apache unconfined_u:object_r:cert_t:s0 key3.db.orig lrwxrwxrwx. root root system_u:object_r:cert_t:s0 libnssckbi.so -> /usr/lib64/libnssckbi.so -rw-------. root apache unconfined_u:object_r:cert_t:s0 pwdfile.txt -rw-r-----. root apache unconfined_u:object_r:cert_t:s0 secmod.db -rw-r-----. root apache unconfined_u:object_r:cert_t:s0 secmod.db.orig

The interesting value here is **cert_t**. From *man ls*

> Display security context so it fits on most displays. Displays only mode, user, group, security context and file name.

The Security context is unconfined_u:object_r:cert_t:s0 which is in user:role:type:level format. What we want to do, then, is change the type on our NSS Database files. We could use chcon to test out the change temporarily, and then semanage fcontext to make the change permanent.

Method

Lets get a method in place to make changes and confirm they happen. I use two terminals. In one I'll type command, but in the second, I'll use tail -f to see changes to the log.

```
[root@sso ~]# tail -f  /var/log/audit/audit.log | grep AVC
```

[root@sso ~]# tail -f /var/log/audit/audit.log | grep AVC

Once I request a cert, I will see a line like this added to the output

```
type=AVC msg=audit(1518708370.985:6639): avc:  denied  { write } for  pid=16
459 comm="certmonger" name="cert8.db" dev="vda1" ino=17484343 scontext=syste
m_u:system_r:certmonger_t:s0 tcontext=unconfined_u:object_r:etc_t:s0 tclass=
file
```

type=AVC msg=audit(1518708370.985:6639): avc: denied { write } for pid=16459 comm="certmonger" name="cert8.db" dev="vda1" ino=17484343 scontext=system_u:system_r:certmonger_t:s0 tcontext=unconfined_u:object_r:etc_t:s0 tclass=file

In the coding window, I can run commands like this to trigger output from the log;

```
[root@sso standalone]# ipa-getcert  request  -w -d dbm:$PWD/keystore -D $HOS
TNAME -K RHSSO/$HOSTNAME -n RHSSO
New signing request "20180215152610" added.
[root@sso standalone]# getcert stop-tracking  -i 20180215152610
Request "20180215152610" removed.
```

[root@sso standalone]# ipa-getcert request -w -d dbm:$PWD/keystore -D $HOSTNAME -K RHSSO/$HOSTNAME -n RHSSO New signing request "20180215152610" added. [root@sso standalone]# getcert stop-tracking -i 20180215152610 Request "20180215152610" removed.

chcon

Now that I have a baseline, I'm going to try chcon to ensure that I have the type correct.

```
[root@sso standalone]# sudo chcon -t cert_t keystore keystore/*
[root@sso standalone]# ls -Z keystore
-rw-------. root root unconfined_u:object_r:cert_t:s0  cert8.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  cert9.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  key3.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  key4.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  pkcs11.txt
-rw-------. root root unconfined_u:object_r:cert_t:s0  secmod.db
```

[root@sso standalone]# sudo chcon -t cert_t keystore keystore/* [root@sso standalone]# ls -Z keystore -rw-------. root root unconfined_u:object_r:cert_t:s0 cert8.db -rw-------. root root unconfined_u:object_r:cert_t:s0 cert9.db -rw-------. root root unconfined_u:object_r:cert_t:s0 key3.db -rw-------. root root unconfined_u:object_r:cert_t:s0 key4.db -rw-------. root root unconfined_u:object_r:cert_t:s0 pkcs11.txt -rw-------. root root unconfined_u:object_r:cert_t:s0 secmod.db

Lets run the test again:

Running:

```
# ipa-getcert  request  -w -d dbm:$PWD/keystore -D $HOSTNAME -K RHSSO/$HOSTN
AME -n RHSSO
New signing request "20180215153108" added.
```

ipa-getcert request -w -d dbm:$PWD/keystore -D $HOSTNAME -K RHSSO/$HOSTNAME -n RHSSO New signing request "20180215153108" added.

Produces no new output from our log. We also see that the cert is being tracked.

```
[root@sso standalone]# getcert list
Number of certificates and requests being tracked: 1.
Request ID '20180215153108':
        status: MONITORING
```

[root@sso standalone]# getcert list Number of certificates and requests being tracked: 1. Request ID '20180215153108': status: MONITORING

setenforce

Lets try this again but with SELinux enforcing. First cleanup from our last run

```
[root@sso standalone]# getcert stop-tracking  -i 20180215153108
Request "20180215153108" removed.
[root@sso standalone]# getcert list
Number of certificates and requests being tracked: 0.
```

[root@sso standalone]# getcert stop-tracking -i 20180215153108 Request "20180215153108" removed. [root@sso standalone]# getcert list Number of certificates and requests being tracked: 0.

And now:

```
[root@sso standalone]# getenforce
Permissive
[root@sso standalone]# setenforce 1
[root@sso standalone]# getenforce
Enforcing
[root@sso standalone]# ipa-getcert  request  -w -d dbm:$PWD/keystore -D $HOS
TNAME -K RHSSO/$HOSTNAME -n RHSSO
New signing request "20180215153334" added.
[root@sso standalone]# getcert list
Number of certificates and requests being tracked: 1.
Request ID '20180215153334':
        status: MONITORING
```

[root@sso standalone]# getenforce Permissive [root@sso standalone]# setenforce 1 [root@sso standalone]# getenforce Enforcing [root@sso standalone]# ipa-getcert request -w -d dbm:$PWD/keystore -D $HOSTNAME -K RHSSO/$HOSTNAME -n RHSSO New signing request "20180215153334" added. [root@sso standalone]# getcert list Number of certificates and requests being tracked: 1. Request ID '20180215153334': status: MONITORING

And the only thing we see in our log is a warning about switching enforcement.

```
type=USER_AVC msg=audit(1518708789.490:6646): pid=2501 uid=81 auid=429496729
5 ses=4294967295 subj=system_u:system_r:system_dbusd_t:s0-s0:c0.c1023 msg='a
vc:  received setenforce notice (enforcing=1)  exe="?" sauid=81 hostname=? a
ddr=? terminal=?'
```

type=USER_AVC msg=audit(1518708789.490:6646): pid=2501 uid=81 auid=4294967295 ses=4294967295 subj=system_u:system_r:system_dbusd_t:s0-s0:c0.c1023 msg='avc: received

setenforce notice (enforcing=1) exe="?" sauid=81 hostname=? addr=? terminal=?'

semanage

OK, so lets make this change permanent. First, restore it so we know we are having the desired effect.

```
[root@sso standalone]# restorecon -R -v keystore
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore context un
confined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/pkcs11.txt
 context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert9.db c
ontext unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key4.db co
ntext unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/secmod.db
context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert8.db c
ontext unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key3.db co
ntext unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0
[root@sso standalone]# ls -Z keystore
-rw-------. root root unconfined_u:object_r:etc_t:s0   cert8.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   cert9.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   key3.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   key4.db
-rw-------. root root unconfined_u:object_r:etc_t:s0   pkcs11.txt
-rw-------. root root unconfined_u:object_r:etc_t:s0   secmod.db
```

[root@sso standalone]# restorecon -R -v keystore restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/pkcs11.txt context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert9.db context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 restorecon reset

/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key4.db context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/secmod.db context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert8.db context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key3.db context unconfined_u:object_r:cert_t:s0->unconfined_u:object_r:etc_t:s0 [root@sso standalone]# ls -Z keystore -rw-------. root root unconfined_u:object_r:etc_t:s0 cert8.db -rw-------. root root unconfined_u:object_r:etc_t:s0 cert9.db -rw-------. root root unconfined_u:object_r:etc_t:s0 key3.db -rw-------. root root unconfined_u:object_r:etc_t:s0 key4.db -rw-------. root root unconfined_u:object_r:etc_t:s0 pkcs11.txt -rw-------. root root unconfined_u:object_r:etc_t:s0 secmod.db

Now use semanage to make the change persist:

```
[root@sso standalone]# semanage fcontext -a -t cert_t $PWD/"keystore(/.*)?"
[root@sso standalone]# restorecon -R -v keystore
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore context un
confined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/pkcs11.txt
 context unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert9.db c
ontext unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key4.db co
ntext unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/secmod.db
context unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert8.db c
ontext unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
restorecon reset /etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key3.db co
ntext unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0
```

[root@sso standalone]# semanage fcontext -a -t cert_t $PWD/"keystore(/.*)?"
[root@sso standalone]# restorecon -R -v keystore restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0 restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/pkcs11.txt context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0 restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert9.db context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0 restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key4.db context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0 restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/secmod.db context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0 restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/cert8.db context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0 restorecon reset
/etc/opt/rh/rh-sso7/keycloak/standalone/keystore/key3.db context
unconfined_u:object_r:etc_t:s0->unconfined_u:object_r:cert_t:s0

Do another list to check the current state of the file:

```
[root@sso standalone]# ls -Z keystore
-rw-------. root root unconfined_u:object_r:cert_t:s0  cert8.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  cert9.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  key3.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  key4.db
-rw-------. root root unconfined_u:object_r:cert_t:s0  pkcs11.txt
-rw-------. root root unconfined_u:object_r:cert_t:s0  secmod.db
```

[root@sso standalone]# ls -Z keystore -rw-------. root root
unconfined_u:object_r:cert_t:s0 cert8.db -rw-------. root root
unconfined_u:object_r:cert_t:s0 cert9.db -rw-------. root root
unconfined_u:object_r:cert_t:s0 key3.db -rw-------. root root

unconfined_u:object_r:cert_t:s0 key4.db -rw-------. root root
unconfined_u:object_r:cert_t:s0 pkcs11.txt -rw-------. root root
unconfined_u:object_r:cert_t:s0 secmod.db

One last time, stop tracking the existing cert, and request a new one:

```
[root@sso standalone]# getcert stop-tracking  -i 20180215153334
Request "20180215153334" removed.
[root@sso standalone]# ipa-getcert  request  -w -d dbm:$PWD/keystore -D $HOS
TNAME -K RHSSO/$HOSTNAME -n RHSSO
New signing request "20180215154055" added.
[root@sso standalone]# getcert list
Number of certificates and requests being tracked: 1.
Request ID '20180215154055':
        status: MONITORING
        stuck: no
        key pair storage: type=NSSDB,location='dbm:/etc/opt/rh/rh-sso7/keycl
oak/standalone/keystore',nickname='RHSSO',token='NSS Certificate DB'
        certificate: type=NSSDB,location='dbm:/etc/opt/rh/rh-sso7/keycloak/s
tandalone/keystore',nickname='RHSSO',token='NSS Certificate DB'
```

[root@sso standalone]# getcert stop-tracking -i 20180215153334 Request "20180215153334" removed. [root@sso standalone]# ipa-getcert request -w -d dbm:$PWD/keystore -D $HOSTNAME -K RHSSO/$HOSTNAME -n RHSSO New signing request "20180215154055" added. [root@sso standalone]# getcert list Number of certificates and requests being tracked: 1. Request ID '20180215154055': status: MONITORING stuck: no key pair storage: type=NSSDB,location='dbm:/etc/opt/rh/rh-sso7/keycloak/standalone/keystore',nickname='RHSSO',token='NSS Certificate DB' certificate: type=NSSDB,location='dbm:/etc/opt/rh/rh-sso7/keycloak/standalone/keystore',nickname='RHSSO',token='NSS Certificate DB'

adam.younglogic.com (http://adam.younglogic.com/2018/02/certmonger-selinux-keystores/) · by Adam Young