

A wide-angle photograph of the San Diego Supercomputer Center (SDSC) building at dusk or night. The building is a modern architectural structure with a light-colored, textured facade and large glass windows. A prominent feature is a cantilevered upper section supported by columns. The building is illuminated from within, with lights visible through the windows and along the exterior walkways. Small trees and streetlights are visible in the foreground.

SEA '18 Containers in HPC Symposium

Modifying the Singularity Installation on Comet

Trevor Cooper, SDSC

SDSC Data Enabled Scientific Computing

- **Visualization, High Performance Systems, User Services, Advanced Technology Lab, Scientific Computing Applications**
- **XSEDE Level 1 Service Provider (Comet)**
- **UCSD Campus Condo Cluster (TSCC)**
- **Private Cluster (Gordon, previously XSEDE)**

Background

Assumptions

- **Users are not trusted**
- **Container contents are not trusted**
- **Some form of privilege escalation is required**

Comet Facts

- Not everybody needs full virtualization ... *Comet VC*
- *Comet deployed OS/kernel does not support user namespaces*
- **Singularity deployment makes use of SUID binaries**
- **SDSC security team restricts SUID binary usage to minimize risk**

SDSC HPC Software Installation

- **Systems installed with Rocks® Cluster Distribution**
 - Appliance Types / Nodes defined in DB
 - Rocks® DAG selects RPMs and orders post-install configuration
- **Software is built into and managed with Rocks® Rolls**
 - **Basic**... configure, make, make install to produce RPMs
 - **Intermediate**... source patches, rpmbuild –rebuild, etc...
 - **Advanced**... bootstrap, roll dependencies, RPM.EXTRAS, etc...

singularity-roll

- Defines RPM destinations for various node types
- Pulls singularity source tarball from internal server
- Extracts singularity source and patches RPM spec file
- Builds RPMs for singularity using patched spec file
- Bundles RPMs and Rocks® node and graph definition files into ISO

sdscsec-roll

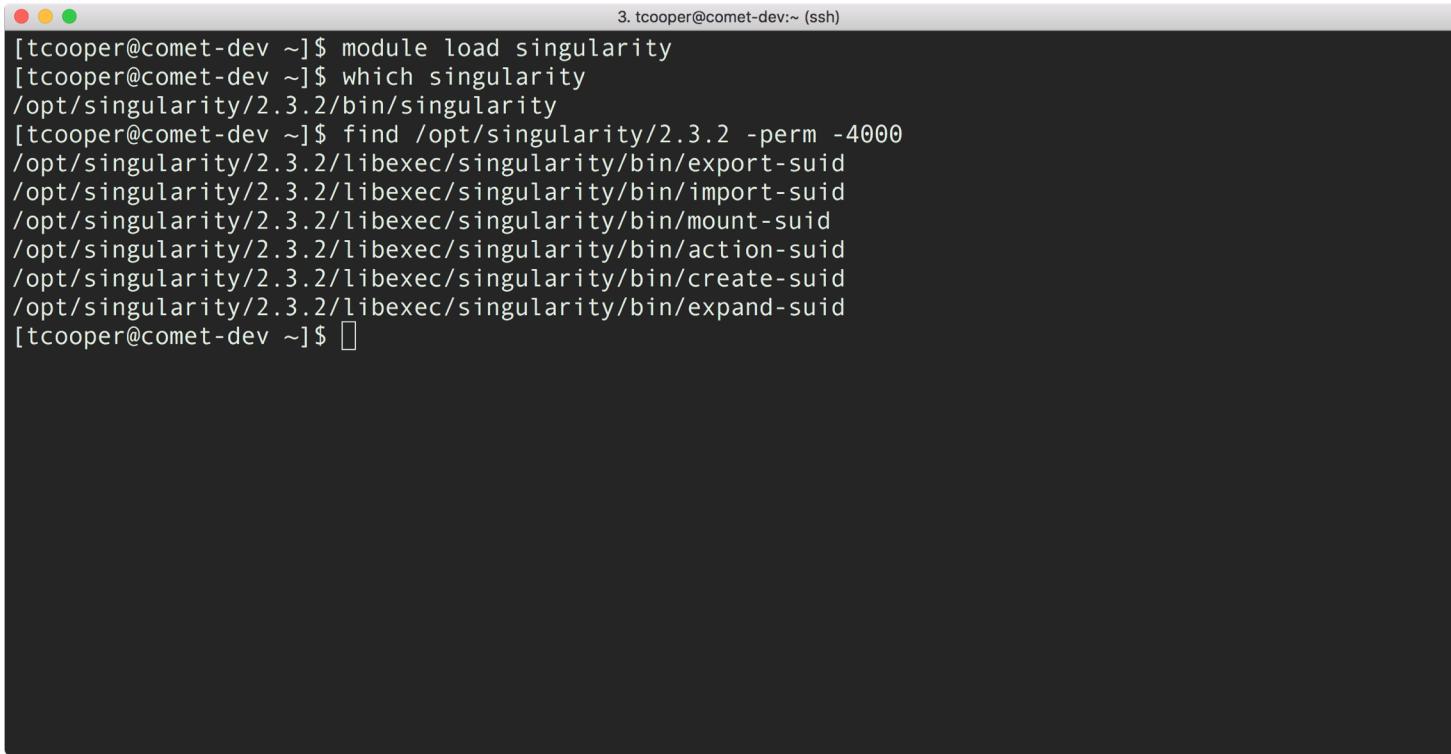
- **Replaces distro provided software with custom builds (openssh, sudo, etc...)**
- **Deploys security related configuration (rsyslog, ssh{d}, certificates, sudo, etc...)**
- **Removes unwanted permissions (SUID binaries) during post-install and/or first boot**

comet-config-roll

- **Deploys Singularity configuration file to frontend**
- **Singularity configuration file is (re)distributed by Rocks® 411 process**

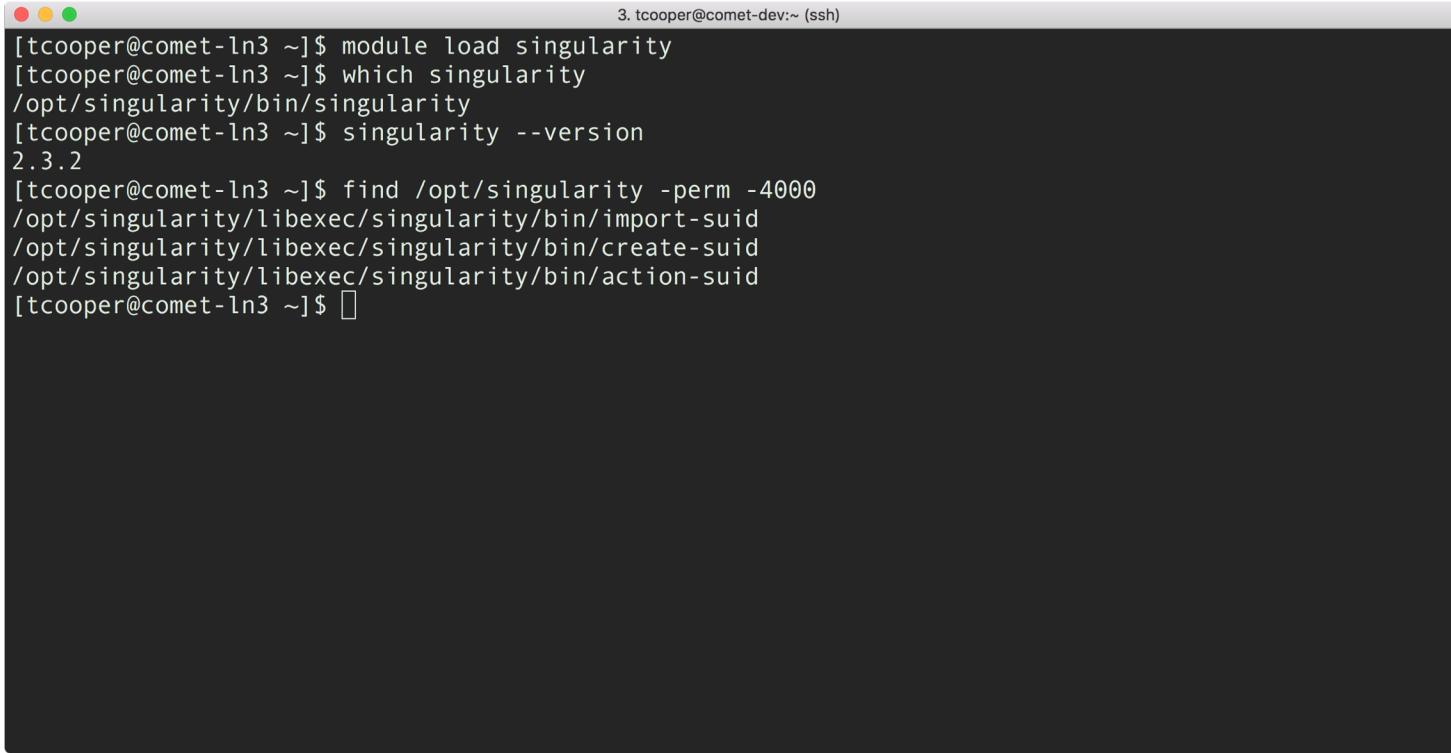
Details

Where are the SUID Components (dev)?

A terminal window titled "3. tcooper@comet-dev:~ (ssh)" showing the results of a command-line session. The session starts with loading the singularity module, then finding all files with the SUID bit set in the singularity directory. The output lists several files: singularity, export-suid, import-suid, mount-suid, action-suid, create-suid, and expand-suid.

```
[tcooper@comet-dev ~]$ module load singularity
[tcooper@comet-dev ~]$ which singularity
/opt/singularity/2.3.2/bin/singularity
[tcooper@comet-dev ~]$ find /opt/singularity/2.3.2 -perm -4000
/opt/singularity/2.3.2/libexec/singularity/bin/export-suid
/opt/singularity/2.3.2/libexec/singularity/bin/import-suid
/opt/singularity/2.3.2/libexec/singularity/bin/mount-suid
/opt/singularity/2.3.2/libexec/singularity/bin/action-suid
/opt/singularity/2.3.2/libexec/singularity/bin/create-suid
/opt/singularity/2.3.2/libexec/singularity/bin/expand-suid
[tcooper@comet-dev ~]$ 
```

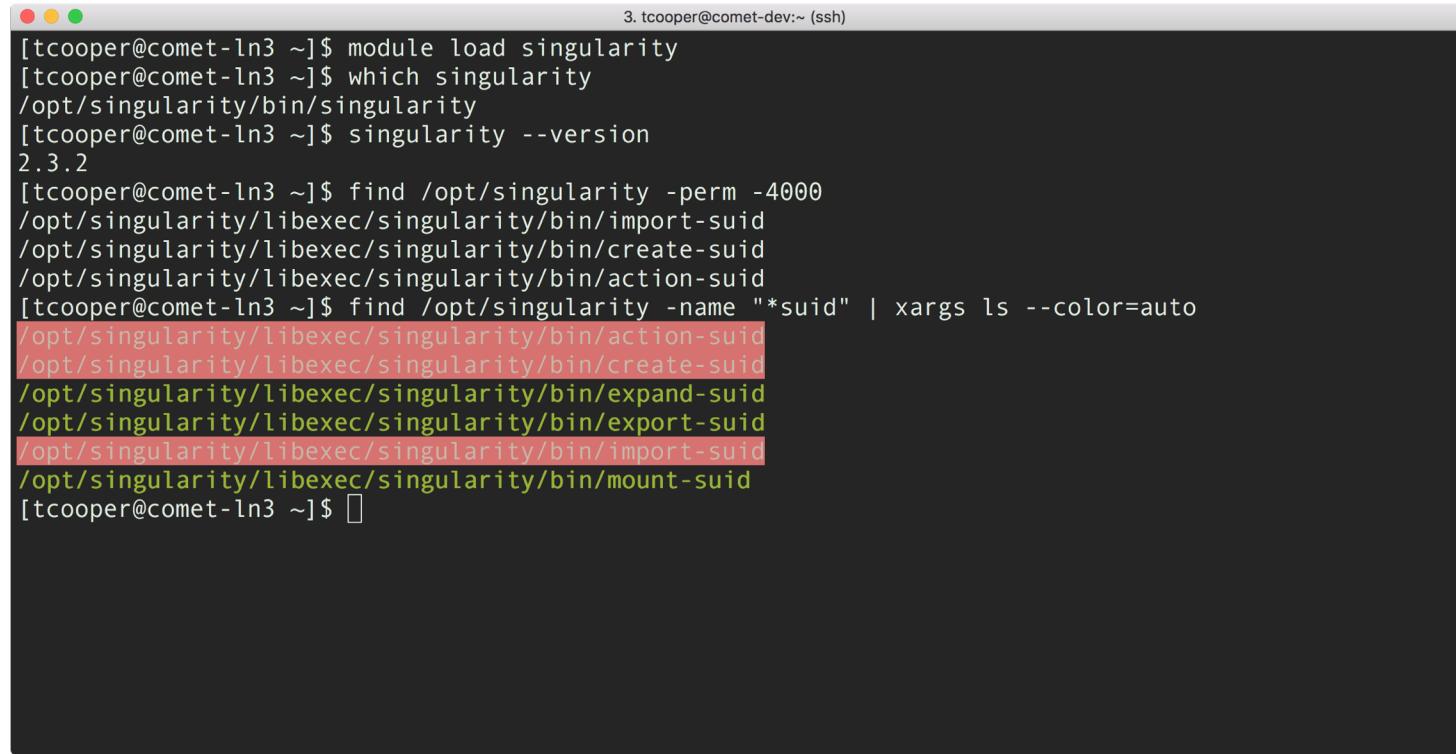
Where are the SUID Components (prod)?

A terminal window with a dark background and light-colored text. The title bar reads "3. tcooper@comet-dev:~ (ssh)". The terminal shows the following command-line session:

```
[tcooper@comet-ln3 ~]$ module load singularity
[tcooper@comet-ln3 ~]$ which singularity
/opt/singularity/bin/singularity
[tcooper@comet-ln3 ~]$ singularity --version
2.3.2
[tcooper@comet-ln3 ~]$ find /opt/singularity -perm -4000
/opt/singularity/libexec/singularity/bin/import-suid
/opt/singularity/libexec/singularity/bin/create-suid
/opt/singularity/libexec/singularity/bin/action-suid
[tcooper@comet-ln3 ~]$ █
```

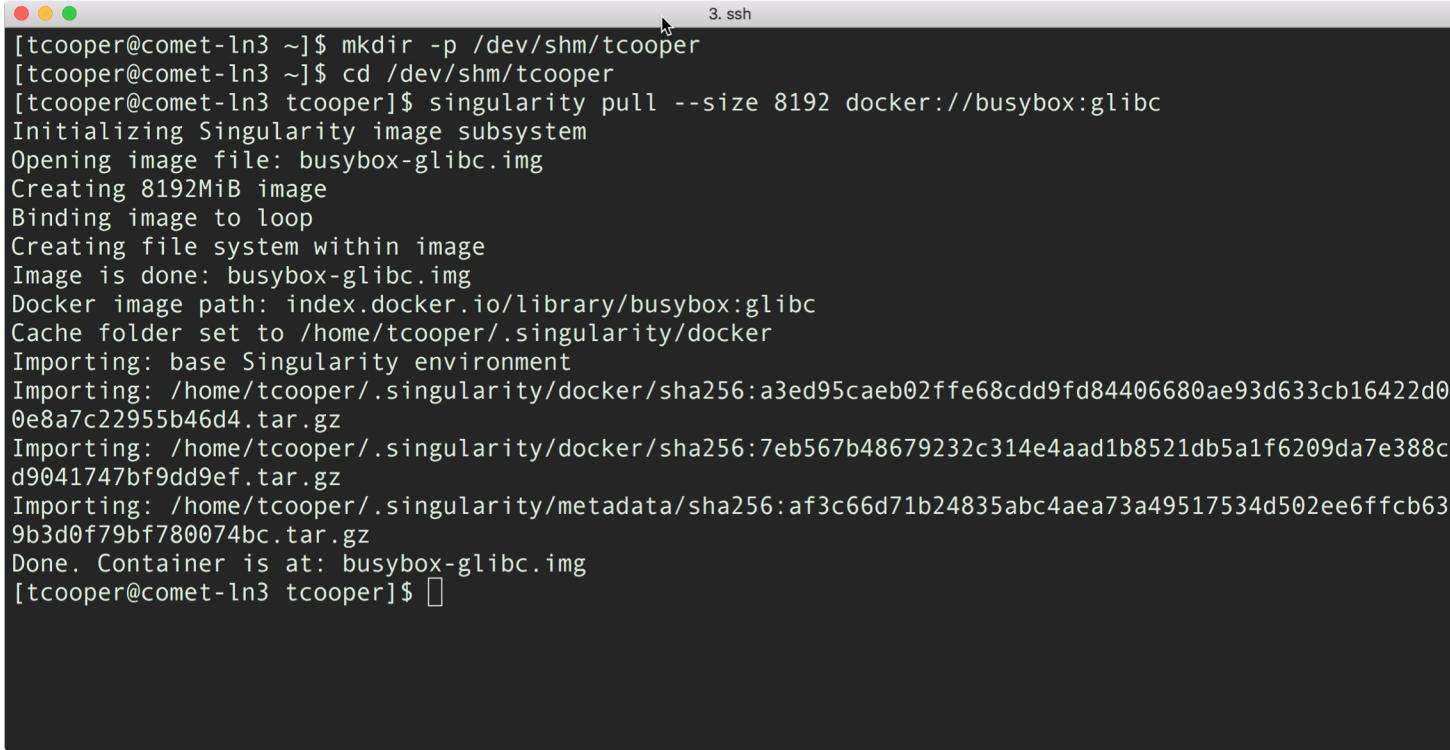
The window has a standard OS X-style title bar with red, yellow, and green buttons.

Where are the SUID Components (prod)?



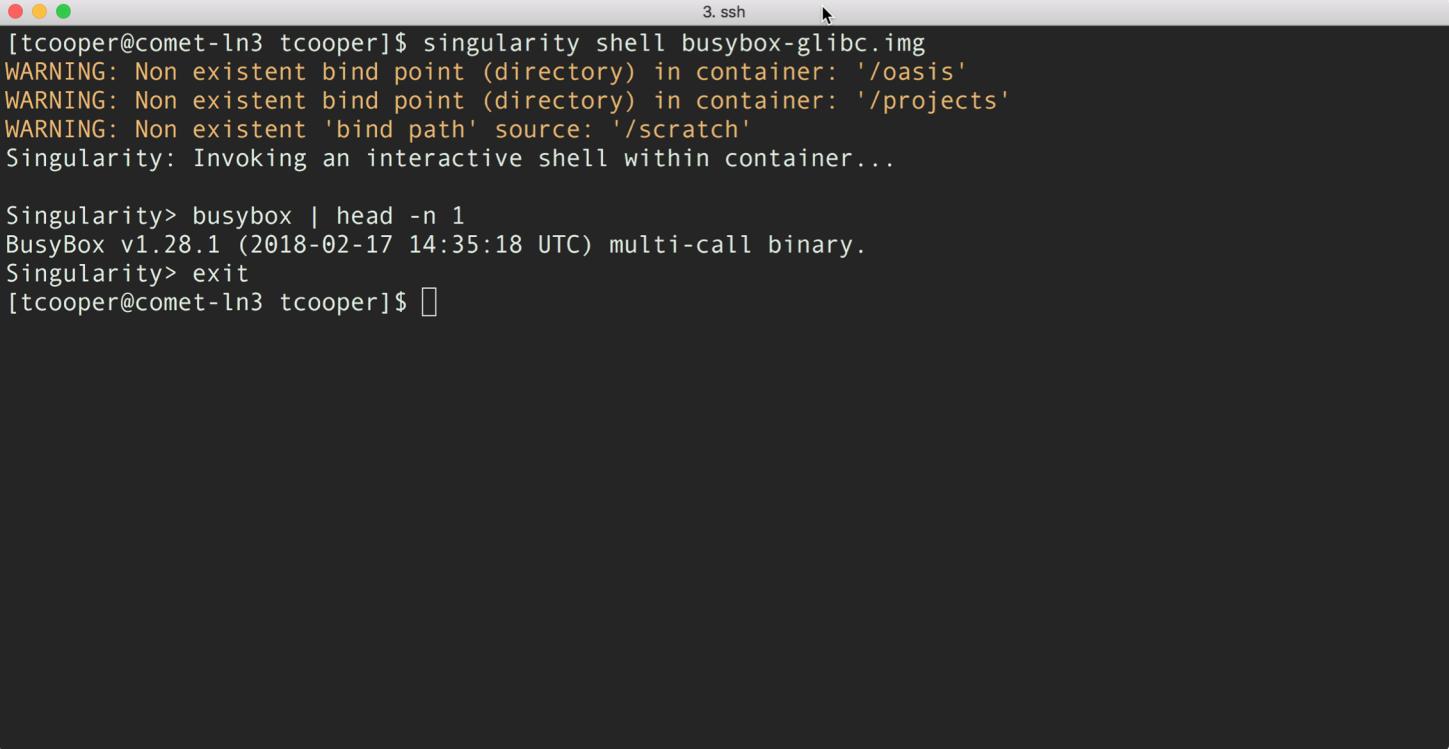
```
3. tcooper@comet-dev:~ (ssh)
[tcooper@comet-ln3 ~]$ module load singularity
[tcooper@comet-ln3 ~]$ which singularity
/opt/singularity/bin/singularity
[tcooper@comet-ln3 ~]$ singularity --version
2.3.2
[tcooper@comet-ln3 ~]$ find /opt/singularity -perm -4000
/opt/singularity/libexec/singularity/bin/import-suid
/opt/singularity/libexec/singularity/bin/create-suid
/opt/singularity/libexec/singularity/bin/action-suid
[tcooper@comet-ln3 ~]$ find /opt/singularity -name "*suid" | xargs ls --color=auto
/opt/singularity/libexec/singularity/bin/action-suid
/opt/singularity/libexec/singularity/bin/create-suid
/opt/singularity/libexec/singularity/bin/expand-suid
/opt/singularity/libexec/singularity/bin/export-suid
/opt/singularity/libexec/singularity/bin/import-suid
/opt/singularity/libexec/singularity/bin/mount-suid
[tcooper@comet-ln3 ~]$ 
```

create-suid + import-suid together enable container pull...



[tcooper@comet-ln3 ~]\$ mkdir -p /dev/shm/tcooper
[tcooper@comet-ln3 ~]\$ cd /dev/shm/tcooper
[tcooper@comet-ln3 tcooper]\$ singularity pull --size 8192 docker://busybox:glibc
Initializing Singularity image subsystem
Opening image file: busybox-glibc.img
Creating 8192MiB image
Binding image to loop
Creating file system within image
Image is done: busybox-glibc.img
Docker image path: index.docker.io/library/busybox:glibc
Cache folder set to /home/tcooper/.singularity/docker
Importing: base Singularity environment
Importing: /home/tcooper/.singularity/docker/sha256:a3ed95caeb02ffe68cdd9fd84406680ae93d633cb16422d0
0e8a7c22955b46d4.tar.gz
Importing: /home/tcooper/.singularity/docker/sha256:7eb567b48679232c314e4aad1b8521db5a1f6209da7e388c
d9041747bf9dd9ef.tar.gz
Importing: /home/tcooper/.singularity/metadata/sha256:af3c66d71b24835abc4aea73a49517534d502ee6ffcb63
9b3d0f79bf780074bc.tar.gz
Done. Container is at: busybox-glibc.img
[tcooper@comet-ln3 tcooper]\$

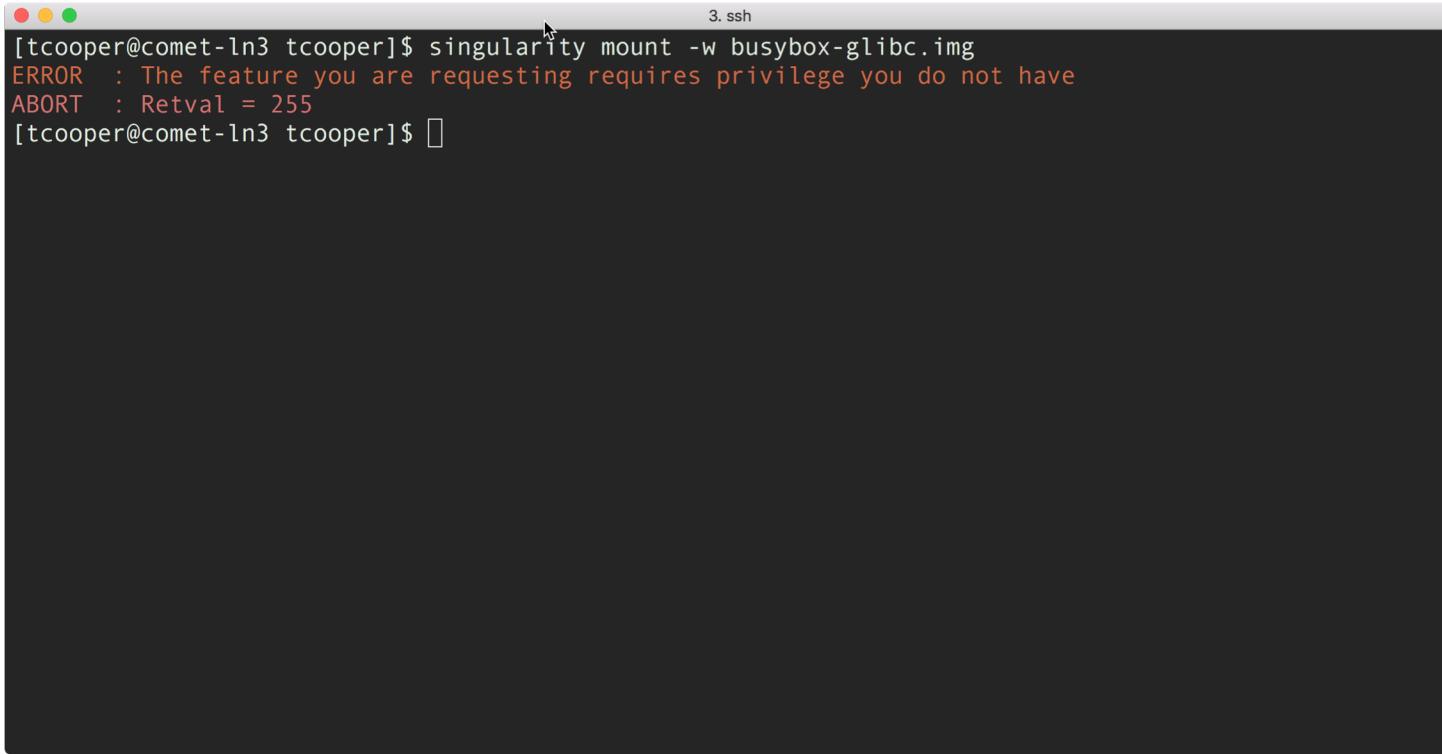
action-suid enables container exec|run|shell...



[tcooper@comet-ln3 tcooper]\$ singularity shell busybox-glibc.img
WARNING: Non existent bind point (directory) in container: '/oasis'
WARNING: Non existent bind point (directory) in container: '/projects'
WARNING: Non existent 'bind path' source: '/scratch'
Singularity: Invoking an interactive shell within container...

Singularity> busybox | head -n 1
BusyBox v1.28.1 (2018-02-17 14:35:18 UTC) multi-call binary.
Singularity> exit
[tcooper@comet-ln3 tcooper]\$

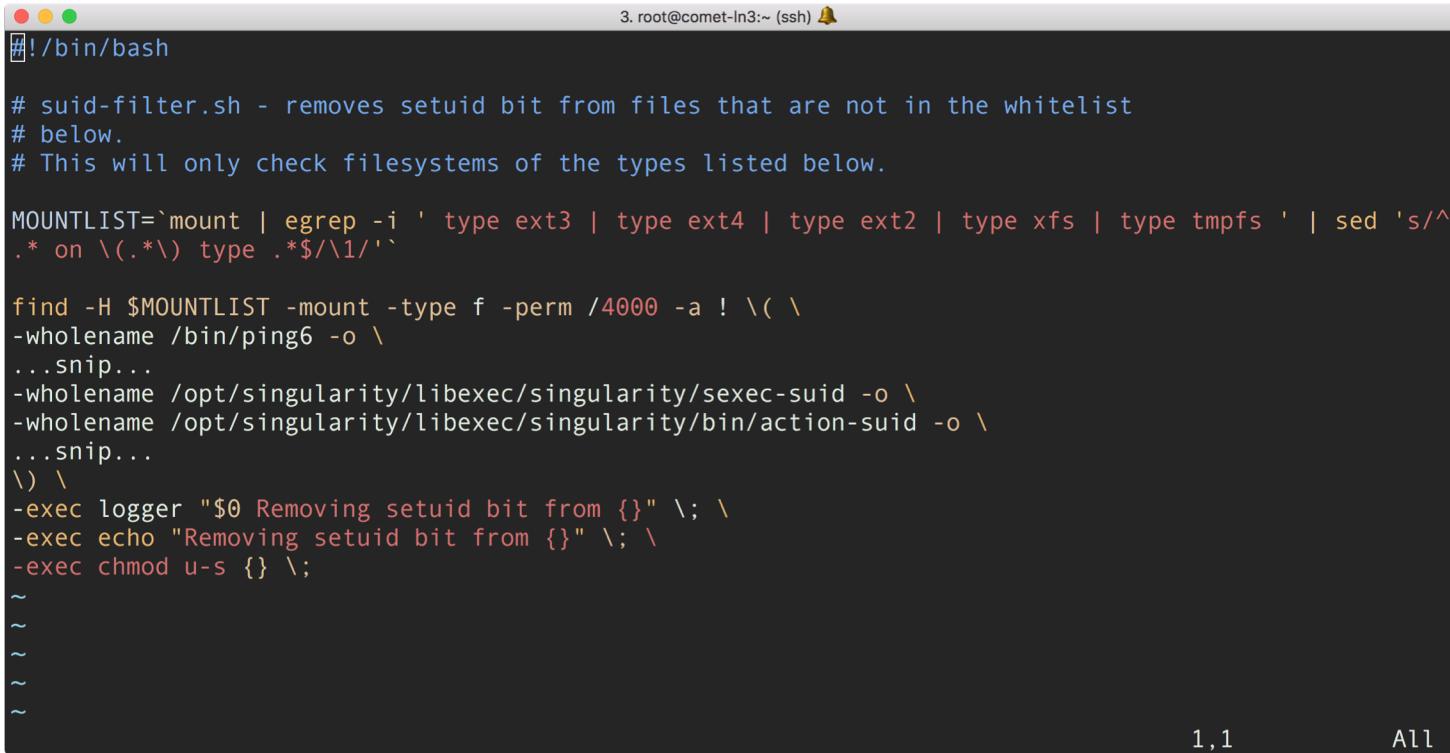
...but you cannot mount or modify a singularity image...



```
[tcooper@comet-ln3 tcooper]$ singularity mount -w busybox-glibc.img
ERROR : The feature you are requesting requires privilege you do not have
ABORT : Retval = 255
[tcooper@comet-ln3 tcooper]$ ]
```

Modifications

security-roll SUID whitelist (v.2.3)



A terminal window titled "root@comet-ln3:~ (ssh)" showing a shell script named "suid-filter.sh". The script removes the setuid bit from files not in the whitelist across various mounted filesystem types. It uses "find" to search for files with permission 4000 (setuid) and then removes the bit using "chmod u-s". The script also logs the changes with "logger" and "echo". The terminal shows several command-line continuations with backslashes and ends with several tilde (~) characters.

```
#!/bin/bash

# suid-filter.sh - removes setuid bit from files that are not in the whitelist
# below.
# This will only check filesystems of the types listed below.

MOUNTLIST=`mount | egrep -i ' type ext3 | type ext4 | type ext2 | type xfs | type tmpfs ' | sed 's/^.* on \(.*\) type .*$/\1/'` 

find -H $MOUNTLIST -mount -type f -perm /4000 -a ! \(\ \
-wholename /bin/ping6 -o \
...snip...
-wholename /opt/singularity/libexec/singularity/seexec-suid -o \
-wholename /opt/singularity/libexec/singularity/bin/action-suid -o \
...snip...
\) \
-exec logger "$0 Removing setuid bit from {}" \; \
-exec echo "Removing setuid bit from {}" \; \
-exec chmod u-s {} \;

~ 
~ 
~ 
~
```

1,1

All

Patching RPM spec during build (v.2.3)

```
[tcooper@comet-ln3 singularity-roll]$ cat src/singularity/patch-files/singularity.spec.patch
--- singularity.spec      2017-06-15 15:02:45.000000000 -0700
+++ patch-files/singularity.spec          2017-06-15 15:03:35.000000000 -0700
@@ -121,12 +121,12 @@
 
 #SUID programs
 %attr(4755, root, root) %{_libexecdir}/singularity/bin/action-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/create-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/copy-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/expand-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/export-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/import-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/mount-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/create-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/copy-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/expand-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/export-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/import-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/mount-suid
#
# Binaries
%{_libexecdir}/singularity/bin/action
[tcooper@comet-ln3 singularity-roll]$ ]
```

security-roll SUID whitelist (v.2.3.2)

```
#!/bin/bash

# uid-filter.sh - removes setuid bit from files that are not in the whitelist
# below.
# This will only check filesystems of the types listed below.

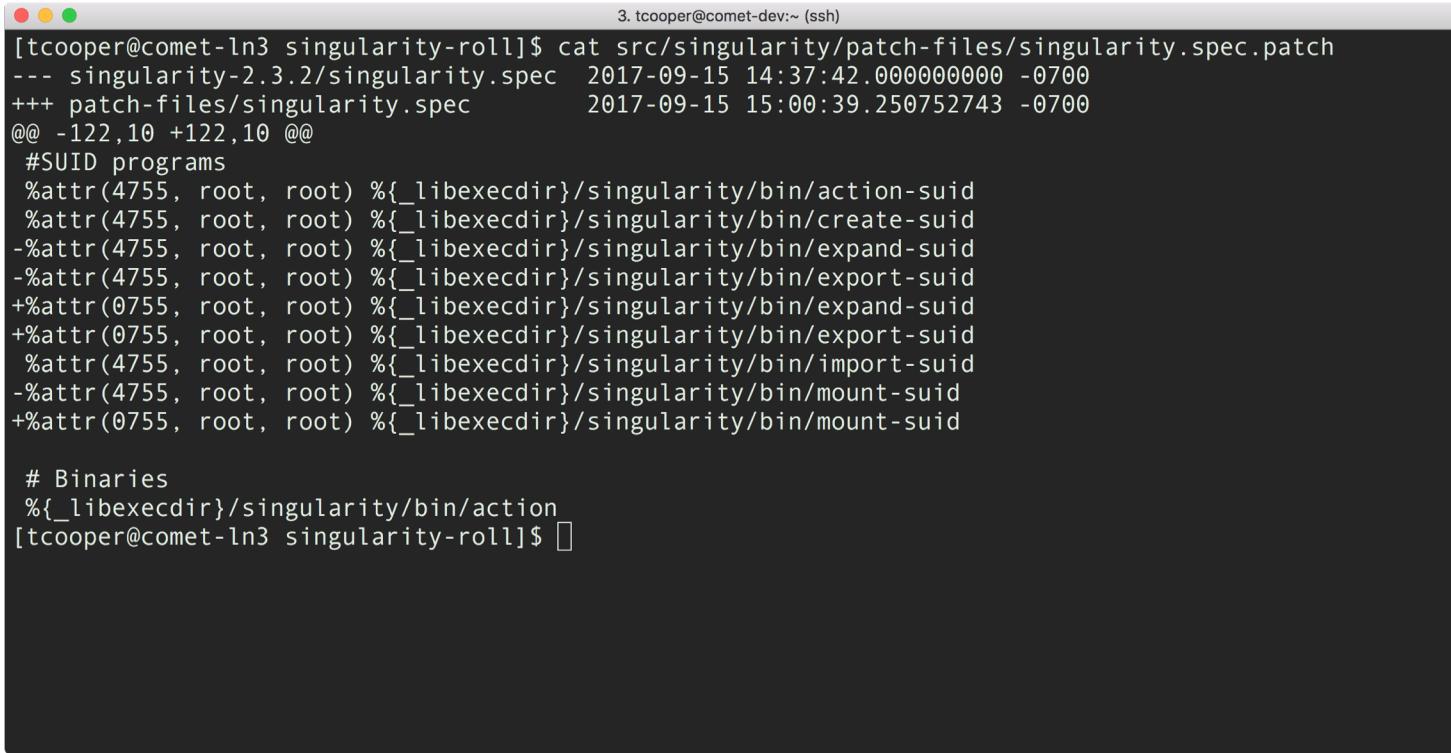
MOUNTLIST=`mount | egrep -i ' type ext3 | type ext4 | type ext2 | type xfs | type tmpfs ' | sed 's/^.* on \(.*\) type .*$/\1/'` 

find -H $MOUNTLIST -mount -type f -perm /4000 -a ! \(` \
-wholename /bin/ping6 -o ` \
...snip...
-wholename /opt/singularity/libexec/singularity/seexec-suid -o \
-wholename /opt/singularity/libexec/singularity/bin/action-suid -o \
-wholename /opt/singularity/libexec/singularity/bin/create-suid -o \
-wholename /opt/singularity/libexec/singularity/bin/import-suid -o \
...snip...
`\` \
-exec logger "$0 Removing setuid bit from {}" \; \
-exec echo "Removing setuid bit from {}" \; \
-exec chmod u-s {} \;
~
~
~
```

1,1

All

Patching RPM spec during build (v.2.3.2)



```
[tcooper@comet-ln3 singularity-roll]$ cat src/singularity/patch-files/singularity.spec.patch
3. tcooper@comet-dev:~ (ssh)
[tcooper@comet-ln3 singularity-roll]$ cat src/singularity/patch-files/singularity.spec.patch
--- singularity-2.3.2/singularity.spec 2017-09-15 14:37:42.000000000 -0700
+++ patch-files/singularity.spec      2017-09-15 15:00:39.250752743 -0700
@@ -122,10 +122,10 @@
 #SUID programs
 %attr(4755, root, root) %{_libexecdir}/singularity/bin/action-suid
 %attr(4755, root, root) %{_libexecdir}/singularity/bin/create-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/expand-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/export-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/expand-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/export-suid
 %attr(4755, root, root) %{_libexecdir}/singularity/bin/import-suid
-%attr(4755, root, root) %{_libexecdir}/singularity/bin/mount-suid
+%attr(0755, root, root) %{_libexecdir}/singularity/bin/mount-suid

# Binaries
%{_libexecdir}/singularity/bin/action
[tcooper@comet-ln3 singularity-roll]$ ]
```

Patching RPM spec during build (v.2.4)

```
[tcooper@comet-ln3 singularity-roll]$ cat src/singularity/patch-files/singularity.spec.patch
3. tcooper@comet-dev:~ (ssh)
[...]
[tcooper@comet-ln3 singularity-roll]$
```

Issues

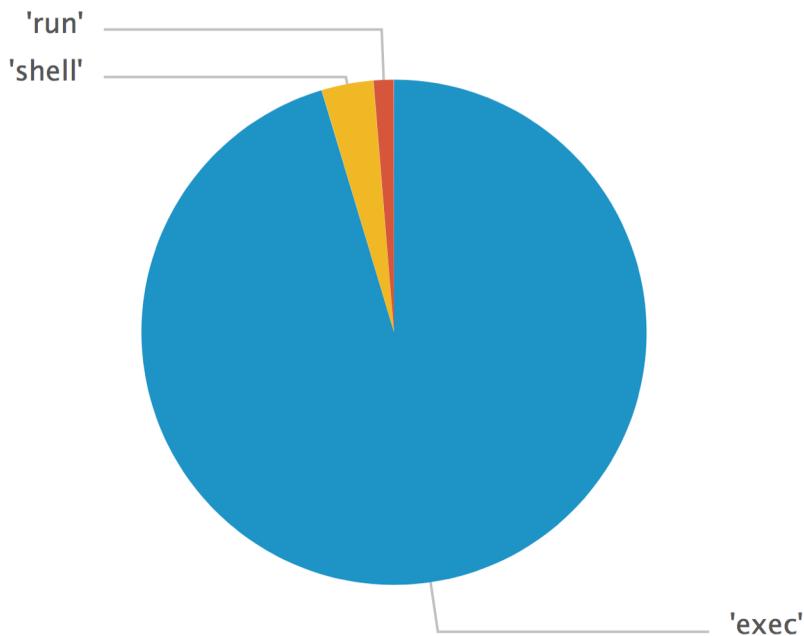
- **For each Singularity version we've needed to...**
 - Examine behavior of new SUID binaries
 - Update sdscsec-roll SUID whitelist in advance of rollout on multiple systems
 - Update singularity-roll build as SUID binaries have changed
- **Security updates...**
 - Early notification from Singularity Team
 - Double updates

Future

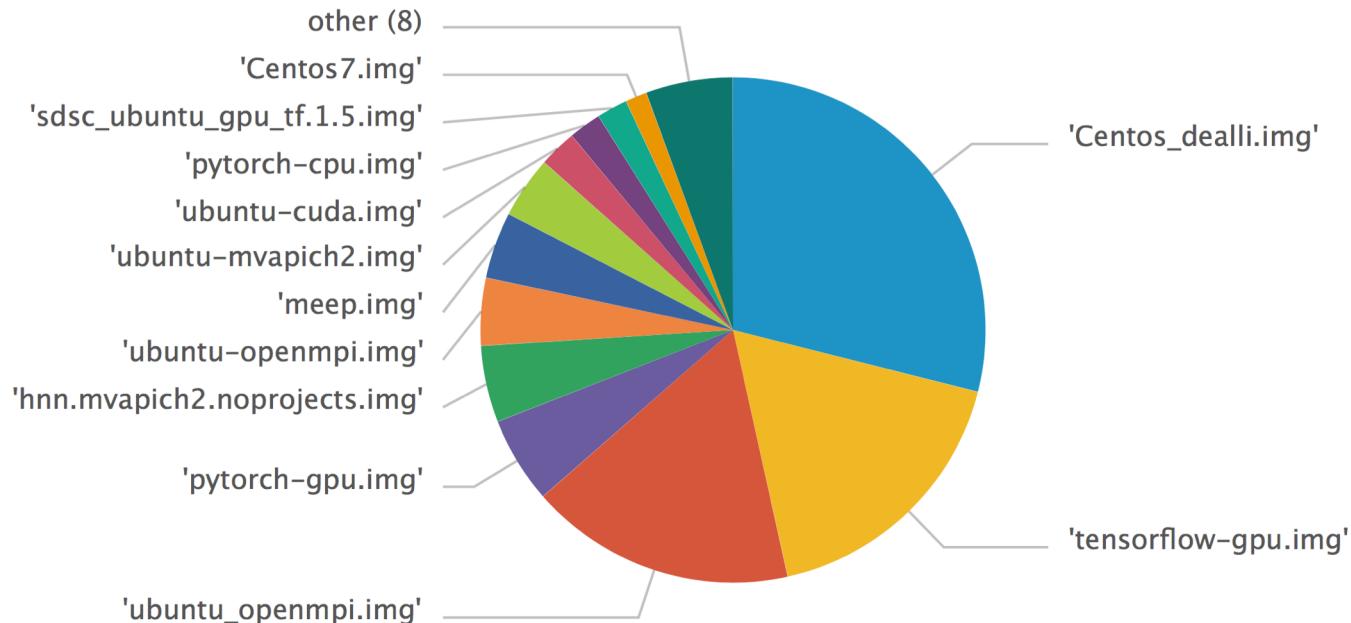
- **Singularity 3.x...**
 - Portions of Python and Bash code base will be replaced by Go
 - SUID binaries are C based and should remain
- **Comet upgrade to ROCKS7 / CentOS 7.4+**
 - User Namespace Support???

THANK YOU

Comet container usage (30 days)



Comet container usage (30 days)



Comet container usage (30 days)

