Capabilities

Capabilities (POSIX 1003.1e, **capabilities(7) (https://jlk.fjfi.cvut.cz/arch/manpages/man/capabilities.7)**) provide fine-grained control over superuser permissions, allowing use of the root user to be avoided. Software developers are encouraged to replace uses of the powerful **setuid** attribute in a system binary with a more minimal set of capabilities. Many packages make use of capabilities, such as `CAP_NET_RAW` being used for the `ping` binary provided by **iputils (https://www.archlinux.org/packages/?name=iputils)**. This enables e.g. `ping` to be run by a normal user (as with the **setuid** method), while at the same time limiting the security consequences of a potential vulnerability in `ping`.

Contents

- 1 Implementation
- 2 Administration and maintenance
- 3 Other programs that benefit from capabilities
  - 3.1 beep
  - 3.2 chvt
  - 3.3 iftop
  - 3.4 mii-tool

- 4 Useful commands
- 5 See also

Implementation

Capabilities are implemented on Linux using **extended attributes** (**xattr(7) (https://jlk.fjfi.cvut.cz/arch/manpages/man/xattr.7)**) in the *security* namespace. Extended attributes are supported by all major Linux **file systems**, including Ext2, Ext3, Ext4, Btrfs, JFS, XFS, and Reiserfs. The following example prints the capabilities of ping with `getcap`, and then prints the same data in its encoded form using `getfattr`:

```
$ getcap /usr/bin/ping
```

```
/usr/bin/ping = cap_net_raw+ep
```

```
$ getfattr -d -m "^security\\." /usr/bin/ping
```

```
# file: usr/bin/ping
security.capability=0sAQAAAgAgAAAAAAAAAAAAAAAAAAA=
```

Extended attributes are copied automatically by `cp -a`, but some other programs require a special flag: `rsync -X`.

Capabilities are set by package install scripts on Arch (e.g. `iputils.install`).

Administration and maintenance

It is considered a bug if a package has overly permissive capabilities, so these cases should be reported rather than listed here. A capability essentially equivalent to root access (`CAP_SYS_ADMIN`) or trivially allowing root access (`CAP_DAC_OVERRIDE`) does not count as a bug since Arch does not support any **MAC/RBAC** systems.

Warning: Many capabilities enable trivial privilege escalation. For examples and explanations see Brad Spengler's post **False Boundaries and Arbitrary Code Execution (http://forums.grsecurity.net/viewtopic.php?f=7&t=2522&sid=c6fbcf62fd5d3472562540a7e608ce4e#p10271)**.

Other programs that benefit from capabilities

The following packages do not have files with the setuid attribute but require root privileges to work. By enabling some capabilities, regular users can use the program without privilege elevation.

beep

```
# setcap cap_dac_override,cap_sys_tty_config+ep /usr/bin/beep
```

chvt

```
# setcap cap_dac_read_search,cap_sys_tty_config+ep /usr/bin/chvt
```

iftop

```
# setcap cap_net_raw+ep /usr/bin/iftop
```

mii-tool

```
# setcap cap_net_admin+ep /usr/bin/mii-tool
```

Useful commands

Find setuid-root files:

```
$ find /usr/bin /usr/lib -perm /4000 -user root
```

Find setgid-root files:

```
$ find /usr/bin /usr/lib -perm /2000 -group root
```

See also

- Man pages: `capabilities(7)` (https://jlk.fjfi.cvut.cz/arch/manpages/man/capabilities.7), `setcap(8)` (https://jlk.fjfi.cvut.cz/arch/manpages/man/setcap.8), `getcap(8)` (https://jlk.fjfi.cvut.cz/arch/manpages/man/getcap.8)
- **Grsecurity Appendix: Capability Names and Descriptions (https://en.wikibooks.org/wiki/Grsecurity/Appendix/Capability_Names_and_Descriptions)**
- **The Linux Kernel Archives: SECure COMPuting with filters (https://www.kernel.org/doc/Documentation/prctl/seccomp_filter.txt)**

Retrieved from "https://wiki.archlinux.org/index.php?title=Capabilities&oldid=492084"
