
ONE IDENTITY
by Quest

Sudo – watch and control your blind spots

Peter Czanik
Open Source Evangelist
One Identity
@PCzanik

ONE IDENTITY by Quest

About me

- Working at One Identity
- syslog-ng & sudo upstream
- Help in RPM and FreeBSD packaging
- Blogger and speaker at open source events

Overview

- What is sudo?
- A few lesser-known features
- Chroot, working directory
- JSON-formatted logging
- Relays
- Logging / intercepting sub-commands

What is sudo?

- Answers, depending on experience:
  - A tool to complicate life
  - A prefix for administrative commands
  - A way to control and log access

What is sudo?

- Sudo allows a system administrator to delegate authority by giving certain users the ability to run some commands as root or another user while providing an audit trail of the commands and their arguments. (https://www.sudo.ws/)

- A lot more than just a prefix

A basic /etc/sudoers

%wheel ALL=(ALL) ALL

- Who
- Where
- As which user
- Which command

Defaults

- Changing the default behavior:

```
Defaults secure_path="/usr/sbin:/usr/bin:/sbin:/bin"
Defaults env_keep = "LANG LC_ADDRESS LC_CTYPE"
Defaults !insults
```

- Making defaults user/host/etc specific

  Defaults:%wheel insults

Insults

- Fun, but not always politically correct :)

```
czanik@linux-mewy:~> sudo ls
[sudo] password for root:
Hold it up to the light --- not a brain in sight!
[sudo] password for root:
My pet ferret can type better than you!
[sudo] password for root:
sudo: 3 incorrect password attempts
czanik@linux-mewy:~>
```

Session recording

- Recording the terminal
- Playback
- Difficult to modify (not cleartext)
- Saved locally; therefore, easy to delete with unlimited access

- Sudo 1.9: central session recording using sudo_logsrvd

LDAP for central management

- Propagates in real-time
- Can’t be modified locally
- Many limitations (aliases, etc.)

Python support

- Extending sudo using Python
- Using the same APIs as C plugins

- API: https://www.sudo.ws/man/sudo_plugin.man.html
  - Python plugin documentation: https://www.sudo.ws/man/sudo_plugin_python.man.html

- No development environment or compilation is needed

IO logs API

- Accessing input and output from user sessions

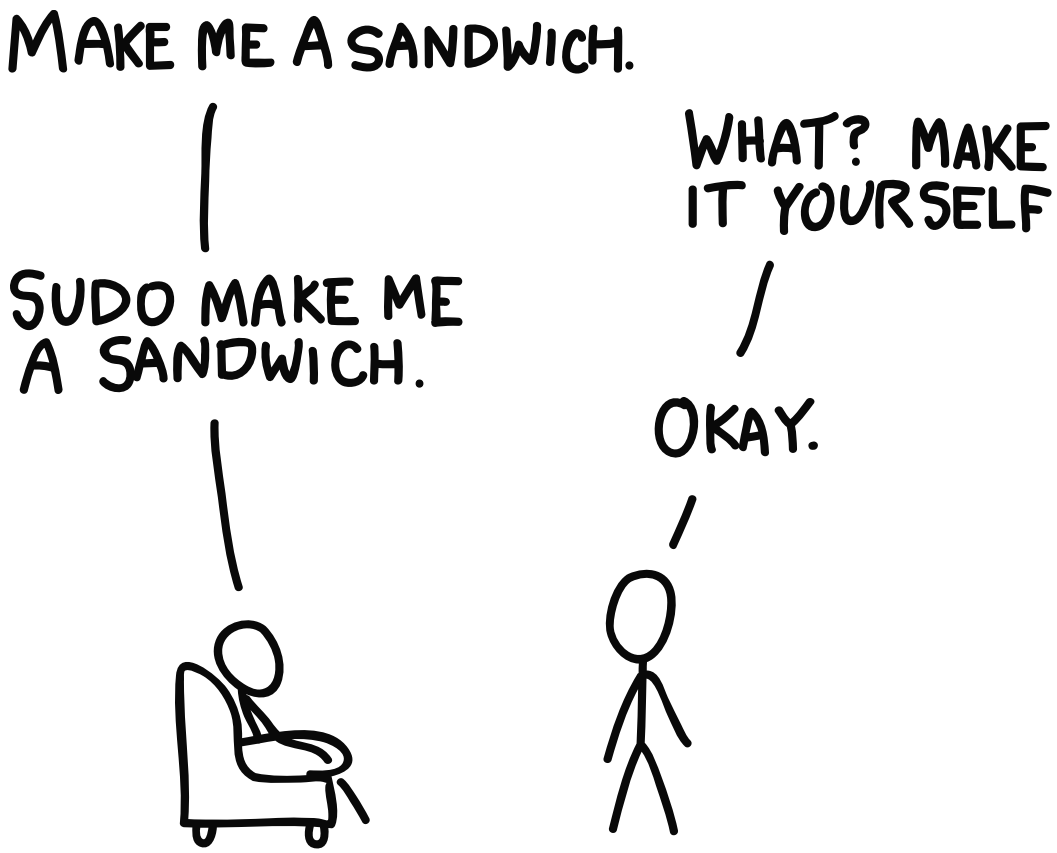
- Python examples:
  - Breaking session if a given text appears on screen
  - Breaking session if “rm -fr” is typed in the command line

IO logs API example: code

```
import sudo

class MyIOPlugin(sudo.Plugin):
    def log_ttyout(self, buf):
        if "MySecret" in buf:
            sudo.log_info("Don't look at my secret!")
            return sudo.RC_REJECT
```

It can get you a sandwich… (by XKCD)



Using chroot and cwd

- Previously full root shell access was needed
  - To use chroot
  - To start an application from a user inaccessible directory

- Starting with sudo 1.9.3 both can be configured from /etc/sudoers

Using cwd

- By default, the working directory is the current directory
- Problem, if an app expects /root/ or other closed directory

```
[czanik@centos7 ~]$ sudo --chdir /root pwd
/root
```

Configuring cwd

- It needs to be enabled explicitly in /etc/sudoers
- Defaults:%wheel runcwd=/var/lib/mock/epel-7-x86_64/root
- Defaults:%wheel runcwd=*

Using chroot

- The chroot command needs root privileges
- Using with sudo it is still possible to “sudo chroot /”
- Chroot support must be explicitly enabled in sudoers

Using chroot

- If directory is not restricted in sudoers:

Defaults:%wheel runchroot=*

- “sudo --chroot / -s” can do the same ☺
- But at least it is nicely logged:

Sep 24 15:58:55 centos7sudo sudo[8149]: czanik : TTY=pts/0 ; CHROOT=/ ; PWD=/home/czanik ; USER=root ; TSID=00001G ; COMMAND=/bin/bash

Using chroot

- Directory can be restricted in sudoers:

Defaults:%wheel runchroot=/var/lib/mock/epel7-x86_64/root

- If chroot or a given directory is not allowed, it is logged:

Sep 25 08:43:32 centos7sudo sudo[2640]: czanik : user not allowed to change root directory to /an/interesting/directory ; TTY=pts/0 ; CHROOT=/an/interesting/directory ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash

New options for logging

- JSON-formatted logs
- Forward logs to sudo_logsrvd
- Introduced in sudo 1.9.4

JSON-formatted logs

- Traditionally plain-text logs with minimal information
- Due to syslog constraints

- Nov 18 12:31:33 centos7sudo sudo[30666]: czanik : 3 incorrect password attempts ; TTY=pts/0 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash
- Nov 18 12:31:43 centos7sudo sudo[30670]: czanik : TTY=pts/0 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash
- Nov 18 12:31:49 centos7sudo sudo[30670]: czanik : command rejected by I/O plugin ; TTY=pts/0 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash

JSON-formatted logs

- JSON-formatted logs have more information in a structured format

Defaults log_format=json

Nov 18 12:40:30 centos7sudo sudo[30891]:
@cee:{"reject":{"reason":"command rejected by I/O
plugin","server_time":{"seconds":1605699630,"nanoseconds":9332939
11,"iso8601":"20201118114030Z","localtime":"Nov 18
11:40:30"},"submit_time":{"seconds":1605699620,"nanoseconds":130
500349,"iso8601":"20201118114020Z","localtime":"Nov 18
11:40:20"},"submituser":"czanik","command":"/bin/bash","runuser":"ro
ot","runcwd":"/home/czanik","ttyname":"/dev/pts/0","submithost":"cent
os7sudo.localdomain","submitcwd":"/home/czanik","runuid":0,"columns"
:118,"lines":60,"runargv":["/bin/bash"]}}

Logging to sudo_logsrvd

- Logging:
  - Syslog
  - Audit plugin API – reachable also from Python for custom logging

- Sudo 1.9.4 added logging to sudo_logsrvd

Defaults log_servers=172.16.167.150

Logging to sudo_logsrvd

- Sudo_logsrvd sends logs to syslog
- “HOST” field shows where logs are coming from

Nov 18 12:40:16 centos8splunk.localdomain sudo[21028]: czanik : 3 incorrect password attempts ; HOST=centos7sudo.localdomain ; TTY=pts/0 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash

Nov 18 12:40:23 centos8splunk.localdomain sudo[21028]: czanik : HOST=centos7sudo.localdomain ; TTY=pts/0 ; PWD=/home/czanik ; USER=root ; TSID=00000A ; COMMAND=/bin/bash

Nov 18 12:40:30 centos8splunk.localdomain sudo[21028]: czanik : command rejected by I/O plugin ; HOST=centos7sudo.localdomain ; TTY=pts/0 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash

- JSON formatting available

Using sudo_logsrvd in relay mode

- Sudo_logsrvd collects session recordings to a central location
- Originally all sudo clients sent recordings directly
- Sudo version 1.9.7 introduced relay mode
- You can have multiple levels of relays to structure your network

Why relay mode?

- Collect recordings even when central server is unavailable (maintenance or network problem)
- Have a single network connection through the firewall instead of granting each host access
- Run it on a gateway host to relay from networks without direct Internet access, like AWS private networks

Configuring relay mode

- Configuring the client or the central server is the same
- On the relay:
  - Where to forward
  - In case of unreliable networks: store first (default: false)

relay_host = 172.16.167.161

store_first = true

- TLS encryption available

Logging and intercepting sub-commands

- Before sudo 1.9.8 only session recording helped in case of shell or editor access
- Watching recordings is boring and time consuming
- 1.9.8 introduced:
  - Logging
  - Intercepting
- Works in most cases (does not work for built-in commands, etc.)

Logging sub-commands

- Enable with:

Defaults log_subcmds

- Turn on JSON formatting:

Defaults log_format=json

Logging sub-commands: editor screenshot

```
I    Unnamed (Modified)                                  Row 14    Col 1
czplaptop:/home/czanik # id
uid=0(root) gid=0(root) groups=0(root)
czplaptop:/home/czanik # ls /usr/share/syslog-ng/include/scl/
apache                  ewmm          logmatic  snmptrap
cee                     fortigate     mbox      solaris
checkpoint              graphite      netskope  sudo
cim                     graylog2      nodejs    sumologic
cisco                   iptables      osquery   syslogconf
collectd                junos         pacct     system
default-network-drivers  linux-audit  paloalto  telegram
discord                 loadbalancer  rewrite   websense
elasticsearch           loggly        slack     windowseventlog
czplaptop:/home/czanik # exit
```

Logging sub-commands

- Log without logging subcommands:

Aug 30 13:03:00 czplaptop sudo[10150]: Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/joe

Logging sub-commands

- Logs when logging subcommands:

```
Aug 30 13:13:14 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/joe
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/bin/sh –c /bin/bash
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/bin/bash
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/readlink /proc/10889/exe
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/dircolors –b /etc/DIR_COLORS
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/tput hs
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/tput –T dumb+sl hs
Aug 30 13:13:37 czplaptop sudo[10874]:   Czanik : TTY=pts/1 ; PWD=/home/Czanik ; USER=root ; COMMAND=/usr/bin/tput bold
Aug 30 13:13:37 czplaptop sudo[10874Aug 30 13:13:14 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/joe
Aug 30 13:13:37 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/sh -c /bin/bash
Aug 30 13:13:37 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/bin/bash
Aug 30 13:13:37 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/readlink /proc/10889/exe
Aug 30 13:13:37 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/dircolors -b /etc/DIR_COLORS
[…]
Aug 30 13:13:37 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/sed -r s@/*:|([^\\]):@\1\n@g;H;x;s@/\n@\n@
Aug 30 13:13:37 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/tty
Aug 30 13:13:42 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/id
Aug 30 13:13:56 czplaptop sudo[10874]:   czanik : TTY=pts/1 ; PWD=/home/czanik ; USER=root ; COMMAND=/usr/bin/ls -A -N --color=none -T 0 /usr/share/syslog-ng/include/scl/
```

Logging sub-commands

- Log with JSON formatting:

Aug 30 13:29:28 czplaptop sudo[11740]:
@cee:{"sudo":{"accept":{"uuid":"18f25b2438-0c44-ddaf-a264-
c70998d319","server_time":{"seconds":1630322968,"nanoseconds":12453428
3,"iso8601":"20210830112928Z","localtime":"Aug 30
11:29:28"},"submit_time":{"seconds":1630322965,"nanoseconds":357407987,
"iso8601":"20210830112925Z","localtime":"Aug 30
11:29:25"},"submituser":"czanik","command":"/usr/bin/joe","runuser":"root","r
uncwd":"/home/czanik","ttyname":"/dev/pts/1","submithost":"czplaptop","subm
itcwd":"/home/czanik","runuid":0,"columns":80,"lines":24,"runargv":["joe","/et
c/issue"],"runenv":["LANG=en_US.UTF-
8","COLORTERM=truecolor","TERM=xterm-
256color","MAIL=/var/mail/root","PATH=/usr/sbin:/usr/bin:/sbin:/bin:/usr/local
/bin:/usr/local/sbin","LOGNAME=root","USER=root","HOME=/root","SHELL=/bin
/bash","SUDO_COMMAND=/usr/bin/joe
/etc/issue","SUDO_USER=czanik","SUDO_UID=1000","SUDO_GID=100"]}}}

Intercepting sub-commands

- Can prevent applications from running
- Enabling is a two-step process in sudoers

Defaults intercept

- And the actual rule:

czanik ALL = (ALL) ALL, !/usr/bin/who

Intercepting sub-commands

- Even if running a shell with full root access:

```
czanik@czplaptop:~> sudo -s
czplaptop:/home/czanik # who
Sorry, user czanik is not allowed to execute
'/usr/bin/who' as root on czplaptop.
bash: /usr/bin/who: Permission denied
```

Intercepting sub-commands

- Shells can easily be disabled
- It has “side effects”

Defaults intercept
Cmnd_Alias SHELLS=/usr/bin/bash, /usr/bin/sh
czanik ALL = (ALL) ALL, !SHELLS

Intercepting sub-commands

- Starting a shell does not work anymore

```
czanik@czplaptop:~> sudo -s
Sorry, user czanik is not allowed to execute '/bin/bash'
as root on czplaptop.
```

Intercepting sub-commands

- Also prevents running external applications

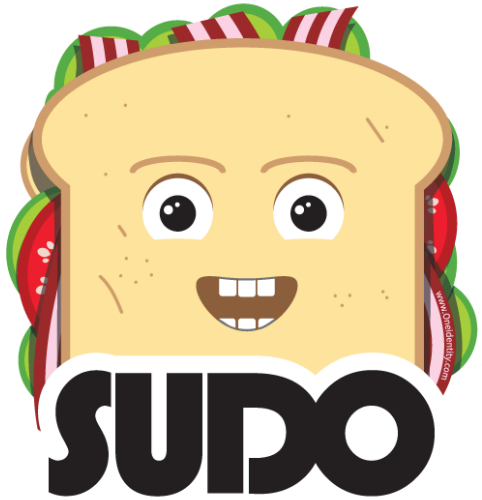
```
czanik@czplaptop:~> sudo vi /etc/issue
Sorry, user czanik is not allowed to execute '/bin/bash -c
/bin/ls' as root on czplaptop.
Cannot execute shell /bin/bash
Press ENTER or type command to continue
czanik@czplaptop:~>
```

Summary

- Recent versions of sudo let you see and control a lot more activities:
  - Less need for root shells
  - More detailed, easier to use log messages collected to a central location
  - Track and intercept sub-commands

Questions?

- Sudo website: https://www.sudo.ws/
- My email: peter.czanik@oneidentity.com
- Twitter: @Pczanik



ONE IDENTITY by Quest


ONE IDENTITY
by Quest