

NAME

`nsenter` - run program with namespaces of other processes

SYNOPSIS

nsenter [options] [*program* [*arguments*]]

DESCRIPTION

Enters the namespaces of one or more other processes and then executes the specified program.

Enterable namespaces are:

mount namespace

Mounting and unmounting filesystems will not affect the rest of the system (**CLONE_NEWNS** flag), except for filesystems which are explicitly marked as shared (with **mount --make-shared**; see */proc/self/mountinfo* for the **shared** flag).

UTS namespace

Setting hostname or domainname will not affect the rest of the system. (**CLONE_NEWUTS** flag)

IPC namespace

The process will have an independent namespace for System V message queues, semaphore sets and shared memory segments. (**CLONE_NEWIPC** flag)

network namespace

The process will have independent IPv4 and IPv6 stacks, IP routing tables, firewall rules, the */proc/net* and */sys/class/net* directory trees, sockets, etc. (**CLONE_NEWNET** flag)

PID namespace

Children will have a set of PID to process mappings separate from the **nsenter** process (**CLONE_NEWPID** flag). **nsenter** will fork by default if changing the PID namespace, so that the new program and its children share the same PID namespace and are visible to each other. If **--no-fork** is used, the new program will be exec'ed without forking.

user namespace

The process will have a distinct set of UIDs, GIDs and capabilities. (**CLONE_NEWUSER** flag)

See [clone\(2\)](#) for the exact semantics of the flags.

If *program* is not given, then “`/${SHELL}`” is run (default: `/bin/sh`).

OPTIONS**-t, --target *pid***

Specify a target process to get contexts from. The paths to the contexts specified by *pid* are:

<code>/proc/<i>pid</i>/ns/mnt</code>	the mount namespace
<code>/proc/<i>pid</i>/ns/uts</code>	the UTS namespace
<code>/proc/<i>pid</i>/ns/ipc</code>	the IPC namespace
<code>/proc/<i>pid</i>/ns/net</code>	the network namespace
<code>/proc/<i>pid</i>/ns/pid</code>	the PID namespace
<code>/proc/<i>pid</i>/ns/user</code>	the user namespace
<code>/proc/<i>pid</i>/root</code>	the root directory
<code>/proc/<i>pid</i>/cwd</code>	the working directory respectively

-m, --mount[=*file*]

Enter the mount namespace. If no file is specified, enter the mount namespace of the target process. If file is specified, enter the mount namespace specified by file.

- u, --uts[=*file*]**
Enter the UTS namespace. If no file is specified, enter the UTS namespace of the target process. If file is specified, enter the UTS namespace specified by file.
- i, --ipc[=*file*]**
Enter the IPC namespace. If no file is specified, enter the IPC namespace of the target process. If file is specified, enter the IPC namespace specified by file.
- n, --net[=*file*]**
Enter the network namespace. If no file is specified, enter the network namespace of the target process. If file is specified, enter the network namespace specified by file.
- p, --pid[=*file*]**
Enter the PID namespace. If no file is specified, enter the PID namespace of the target process. If file is specified, enter the PID namespace specified by file.
- U, --user[=*file*]**
Enter the user namespace. If no file is specified, enter the user namespace of the target process. If file is specified, enter the user namespace specified by file. See also the **--setuid** and **--setgid** options.
- G, --setgid *gid***
Set the group ID which will be used in the entered user namespace.
- S, --setuid *uid***
Set the user ID which will be used in the entered user namespace.
- r, --root[=*directory*]**
Set the root directory. If no directory is specified, set the root directory to the root directory of the target process. If directory is specified, set the root directory to the specified directory.
- w, --wd[=*directory*]**
Set the working directory. If no directory is specified, set the working directory to the working directory of the target process. If directory is specified, set the working directory to the specified directory.
- F, --no-fork**
Do not fork before exec'ing the specified program. By default, when entering a PID namespace, **nsenter** calls **fork** before calling **exec** so that any children will also be in the newly entered PID namespace.
- V, --version**
Display version information and exit.
- h, --help**
Display help text and exit.

SEE ALSO

[setns\(2\)](#), [clone\(2\)](#)

AUTHOR

[Eric Biederman](#)

AVAILABILITY

The nsenter command is part of the util-linux package and is available from [Linux Kernel Archive](#).