Systemd Hardening

From NixOS Wiki
Jump to: navigation, search

Systemd's service options are quite lax by default, and so it is often desirable to look at ways to harden systemd services.

A good way to get started on a given service is to look at the output of the command `systemd-analyze security myService` . From there, you can look at the documentation for the options you see in the output, often in `man systemd.exec` or `man systemd.resource-control` , and set the appropriate options for your service.

Accessing the network with a different RootDirectory

To be able to access the network while having a RootDirectory specified, you need to give access to `/etc/ssl` , `/etc/static/ssl` and `/etc/resolv.conf` . The simplest way of doing this is by simply putting `/etc` in the `BindReadOnlyPaths` option.

A more granular way, would be to put these 3 paths into `BindReadOnlyPaths` , and wait for the creation of `/etc/resolv.conf` through a `systemd.path` unit.

Dropping a shell inside a systemd service

While hardening a service, it often happens that you want a shell inside a hardened systemd unit, for example to check access to files, or check the network connectivity. One way to do this is to use tmux to create a session inside the service, and attaching to it outside of the service.

Simple example:

```
{ pkgs, ... }:
{
  systemd.services.myService = {
    serviceConfig = {
      ExecStart = "${pkgs.tmux}/bin/tmux -S /tmp/tmux.socket new-session -s my-session -d";
      ExecStop = "${pkgs.tmux}/bin/tmux -S /tmp/tmux.socket kill-session -t my-session";
      Type = "forking";

      # ...
    };
  };
}
```

Example with a `RootDirectory` specified:

```
{ pkgs }:
{
  systemd.services.myService = {
    serviceConfig = {
      ExecStart = "${pkgs.tmux}/bin/tmux -S /run/myService/tmux.socket new-session -s my-session -d";
      ExecStop = "${pkgs.tmux}/bin/tmux -S /run/myService/tmux.socket kill-session -t my-session";
      Type = "forking";

      # Used as root directory
      RuntimeDirectory = "myService";
      RootDirectory = "/run/myService";

      BindReadOnlyPaths = [
        "/nix/store"

        # So tmux uses /bin/sh as shell
        "/bin"
      ];

      # This sets up a private /dev/tty
      # The tmux server would crash without this
      # since there would be nothing in /dev
      PrivateDevices = true;
    };
  };
}
```

To attach to the shell, simply execute `tmux -S /path/to/tmux.socket attach` .

Hardening examples

This list contains proposed hardening options that are not yet upstreamed. Please use with caution, and please notify the author of the change if something breaks:

- Chrony: https://github.com/NixOS/nixpkgs/pull/104944/files (https://github.com/NixOS/nixpkgs/pull/104944/files)
- Isso: https://github.com/NixOS/nixpkgs/pull/140840/files (https://github.com/NixOS/nixpkgs/pull/140840/files)
- Mautrix-based bridge: https://github.com/mautrix/docs/pull/18/files (https://github.com/mautrix/docs/pull/18/files)
- Postfix: https://github.com/NixOS/nixpkgs/pull/93305/files (https://github.com/NixOS/nixpkgs/pull/93305/files)
- TheLounge: https://github.com/thelounge/thelounge-deb/pull/78 (https://github.com/thelounge/thelounge-deb/pull/78)

Retrieved from "https://nixos.wiki/index.php?title=Systemd_Hardening&oldid=12965 (https://nixos.wiki/index.php?title=Systemd_Hardening&oldid=12965)"

Categories (/wiki/Special:Categories): NixOS (/wiki/Category:NixOS) | Cookbook (/wiki/Category:Cookbook) | Security (/index.php?title=Category:Security&action=edit&redlink=1)