Implementing active resource management

Nishal Kulkarni

Mentors

Benjamin Berg | Florian Müllner

Leveraging cgroups for desktop apps
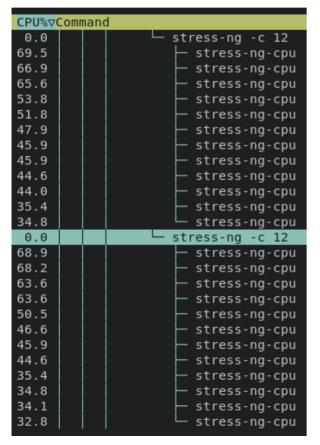
control groups & systemd

- **Each application runs in a different cgroup**
- **systemd provides an interface to limit resources on a cgroup**
- **Dynamically allocate more resources to an active application**
- **Monitoring and making policy decisions using various pieces of information.**

What difference does it make?

Before

```
CPU%▽Command
   0.0                  └ stress-ng -c 12
  69.5                    ├ stress-ng-cpu
  66.9                    ├ stress-ng-cpu
  65.6                    ├ stress-ng-cpu
  53.8                    ├ stress-ng-cpu
  51.8                    ├ stress-ng-cpu
  47.9                    ├ stress-ng-cpu
  45.9                    ├ stress-ng-cpu
  45.9                    ├ stress-ng-cpu
  44.6                    ├ stress-ng-cpu
  44.0                    ├ stress-ng-cpu
  35.4                    ├ stress-ng-cpu
  34.8                    ├ stress-ng-cpu
   0.0                  └ stress-ng -c 12
  68.9                    ├ stress-ng-cpu
  68.2                    ├ stress-ng-cpu
  63.6                    ├ stress-ng-cpu
  63.6                    ├ stress-ng-cpu
  50.5                    ├ stress-ng-cpu
  46.6                    ├ stress-ng-cpu
  45.9                    ├ stress-ng-cpu
  44.6                    ├ stress-ng-cpu
  35.4                    ├ stress-ng-cpu
  34.8                    ├ stress-ng-cpu
  34.1                    ├ stress-ng-cpu
  32.8                    ├ stress-ng-cpu
```

After

```
CPU%▽Command
   0.0                  └ stress-ng -c 12
  94.7                    ├ stress-ng-cpu
  94.7                    ├ stress-ng-cpu
  94.1                    ├ stress-ng-cpu
  92.8                    ├ stress-ng-cpu
  91.5                    ├ stress-ng-cpu
  90.8                    ├ stress-ng-cpu
  90.1                    ├ stress-ng-cpu
  90.1                    ├ stress-ng-cpu
  89.5                    ├ stress-ng-cpu
  88.8                    ├ stress-ng-cpu
  83.0                    ├ stress-ng-cpu
  83.0                    ├ stress-ng-cpu
   0.0                  └ stress-ng -c 12
  13.1                    ├ stress-ng-cpu
  11.8                    ├ stress-ng-cpu
   9.8                    ├ stress-ng-cpu
   9.1                    ├ stress-ng-cpu
   9.1                    ├ stress-ng-cpu
   9.1                    ├ stress-ng-cpu
   9.1                    ├ stress-ng-cpu
   9.1                    ├ stress-ng-cpu
   9.1                    ├ stress-ng-cpu
   8.5                    ├ stress-ng-cpu
   7.8                    ├ stress-ng-cpu
   7.8                    ├ stress-ng-cpu
```

2 Terminals running stress-ng

Equal CPU utilization when not in use.

Focusing on one terminal gives it more CPU weight.

Use-Case: Smooth web browsing experience.

What's more?

Currently working on...

- **Pipewire integration (audio information)**
- **CGroups through XDG portal**

In the near future!

- **Using hints from .desktop file (Game mode)**
- **Detect battery draining background apps (CPU Pressure)**
- **Integration with other tools - open to suggestions!**

Thank you!

Contact

nishalkulkarni@gmail.com

@nishalkulkarni (everywhere)

Repo Links

- gitlab.freedesktop.org/nishalkulkarni/uresourced/-/tree/app_monitor
- gitlab.gnome.org/nishalkulkarni/window-tracker-extension