

# sysprims v0.1.14 for Go Teams

## What sysprims is

sysprims is a cross-platform process utilities library implemented in Rust with first-class Go bindings. It gives Go services and platform tooling typed process-control primitives without shelling out to `ps`, `lsof`, and `kill`.

## License posture

- Dual licensed: MIT OR Apache-2.0
- No GPL/LGPL/AGPL dependency requirements for core process-control path
- Suitable for embedded internal platform tooling where supply-chain policy matters

## Three shell-outs it replaces

1. `ps` parsing for process metadata via `ProcessGetWithOptions(pid, &ProcessOptions{IncludeEnv: true, IncludeThreads: true})`
2. `lsof` parsing for open files/sockets via `ListFds(pid, nil)`
3. Ad hoc kill loops for process trees via `KillDescendantsWithOptions(rootPID, &KillDescendantsOptions{...})`

## v0.1.14 additions

- `proc_ext` options exposed in Go: `ProcessOptions.IncludeEnv`, `ProcessOptions.IncludeThreads`
- CPU mode and sampling for descendants workflows: `CpuModeLifetime`, `CpuModeMonitor`, `SampleDuration`

## 5-line getting started snippet

```
cpu := 90.0
res, err := sysprims.KillDescendantsWithOptions(rootPID,
    &sysprims.KillDescendantsOptions{
        Signal: sysprims.SIGKILL, CpuMode: sysprims.CpuModeMonitor,
        SampleDuration: 3 * time.Second,
        Filter: &sysprims.ProcessFilter{CPUAbove: &cpu},
    })
```

## Where to file issues

- GitHub issues: <https://github.com/3leaps/sysprims/issues>
- Maintainer context: `MAINTAINERS.md`
- Release details: `docs/releases/v0.1.14.md`