Tunables for the C Library

Siddhesh Poyarekar
Toolchain Engineer, Red Hat
7-Aug-2015

The Problems

- Constants in glibc
  - Decided years ago on the state of the art
  - May not be relevant anymore
  - May be heuristic
- Namespace Consistency
  - Enforce consistency in naming environment variables
  - Adding custom environment variables in distributions

The Goals

- Make a framework to identify and manage tunable variables
- Port current environment variables to the tunable framework
- Identify magic constants that could be tunables
- Add a configuration file (/etc/sysctl.user.conf?) for systemwide management
- Tweak some tunables at runtime

Progress

- 2013: Talked about it at the glibc BoF
- 2014: Talked about it at the glibc BoF
- 2015: Got a dedicated slot to talk about it!

The tunables framework (WIP!)

- Code on the `siddhesh/tunables` branch
- A framework to register tunables in a module
- A PoC with environment variables in malloc

Registering tunables

- TOP_NAMESPACE
  - `glibc` by default, but can be overridden for distributions
- TUNABLE_NAMESPACE
  - The module name. `malloc` in the example
- TUNABLES_NAMESPACE_BEGIN
  - Mark the beginning of the tunables list
- TUNABLE_REGISTER
  - Register a tunable
- TUNABLES_NAMESPACE_INIT
  - Finished registration, call initializers for all of them

TUNABLE_REGISTER

```
#define TUNABLE_REGISTER(id,alias,set) \
        ...
```

- ID: The tunable name
- ALIAS: The old environment variable name
- SET: The setter function

The tunables list

- Maintained in `tunables/tunables.list`
- Has a fixed format:

```
TOP_NAMESPACE {
  TUNABLE_NAMESPACE {
    TUNABLE
  }
}
```


What's next?

(You can probably tell that I'm just hand-waving now...)

- Figure out performance cost
- Clean up and get it included in master
- Port current (and relevant) environment variables
- System-wide configuration file
  - Bikeshedding on the name
  - Format:
    `top_namespace.tunable_namespace.tunable = val`
- Tweaking tunables at runtime
  - `echo foo > /proc/PID/top_namespace/tunable_namespace/tunable`

Thank you!