Distributed Operating System

Filesystem actually

曹隽诚　李晋

2022 年 10 月 13 日

The view of the system is built upon three principles. First, resources are named and accessed like files in a hierarchical file system. Second, there is a standard protocol, called 9P, for accessing these resources. Third, the disjoint hierarchies provided by different services are joined together into a single private hierarchical file name space.

- files (files are just, files)
- inter-process communication (unix domain socket)
- process management (procfs)

Takeaway

Not everything is a file, everything is accessed as a file
A distributed operating system is just a distributed filesystem

Distributed Filesystem

Interface

VFS (virtual file system)

- allow processes to access local, remote or any filesystem transparently over an consistent *interface*
- VFS in Linux takes an inode based design

inode

- an inode can be a regular file, a directory, a FIFO, a device or any other beasts
- within a filesystem, every inode has a *unique* inode number
- inode numbers are allocated by the filesystem, and has no meaning other than an opaque identifier to the processes

Distributed Filesystem

Identifier allocation

> *All distrubuted systems, however designed, need means for syncronization, just some of them need less, by keeping resources local.*

- all nodes are allocated an unique node number on creation
- inode numbers are allocated locally, as a combination of node number and a sequence number
- local inodes are accessed as-is
- remote inodes are self-descriptive, the target node can be located without coordination

Distributed Filesystem

File handle

> *There are 2 hard problems in computer science: cache invalidation, naming things, and off-by-1 errors.*
>
> *- Leon Bambrick*

Inode themselves, are local states, but file handles, are not. Just google “nfs stale file handle”, and you will get 137,000 results. File handles, are just caches of the presence of specific inodes.
But there is a simple solution: like NFS, we just fail.

Distributed Filesystem

Remote procedure call

The whole VFS interface, served over the network

Message

We use protobuf, a language-neutral, platform-neutral, extensible mechanism for serializing structured data, to encode messages

Exchange

Messages are exchanged over TCP as a sequence of TLV, for simplicity of implementation, in a request and response manner

Distributed Operating System

zCore

- a zicron like microkernel, a natural fit for a distributed operating system
- has a e1000 network adaptor driver and a netstack based on smoltcp
- uses the same VFS abstraction as rCore, making the final implementation applicable to the mass
- more well-maintained compared to rCore

Distributed Operating System

Filesystem actually

曹隽诚　李晋

2022 年 10 月 13 日