RobustIRC
or: IRC without Netsplits

Chaos Communication Congress, 2015-12-28

Michael Stapelberg
<michael@robustirc.net>

Motivation

- IRC widely used among FOSS, hackers
- no convincing alternative available
- our biggest problem: lack of stability
    - TCP disconnects split up an IRC network
    - hence software updates, reboots, … cause splits

Idea

- use a tunnel protocol to gloss over disconnects
- highly available databases exist, so let’s build an IRC network as a distributed system using Raft

Overview

- n RobustIRC servers make up 1 virtual IRC server
- minority of servers ($\leq$ floor($^n/2$)) can fail
  - 3 servers: 1 can fail. 5 servers: 2 can fail
- RobustSession protocol between servers/clients
- “bridge” tunnels IRC over RobustSession

How does it work?

- persist incoming IRC commands using Raft
- servers are state machines, generate same state

  → clients get same results when failing over

  → same state on server after reboot + reprocess

Fine print

- IRC latency = median latency of all servers
- robust networks require $\geq$ 3 failure domains
- throughput of 1000+ messages/s not yet high enough for biggest IRC networks

Connecting

- setting up a bridge (requires Go):
  - `export GOPATH=~/gocode`
  - `go get -u github.com/robustirc/bridge/robustirc-bridge`
  - `$GOPATH/bin/robustirc-bridge -network=robustirc.net`
  - connect your IRC client to `localhost:6667`
- …or use legacy-irc.robustirc.net (not as good)

The end

- http://robustirc.net/

  → docs/adminguide.html if you want to set it up

  → 40 minute tech talk if you want to learn more
- please talk to me if you have questions!