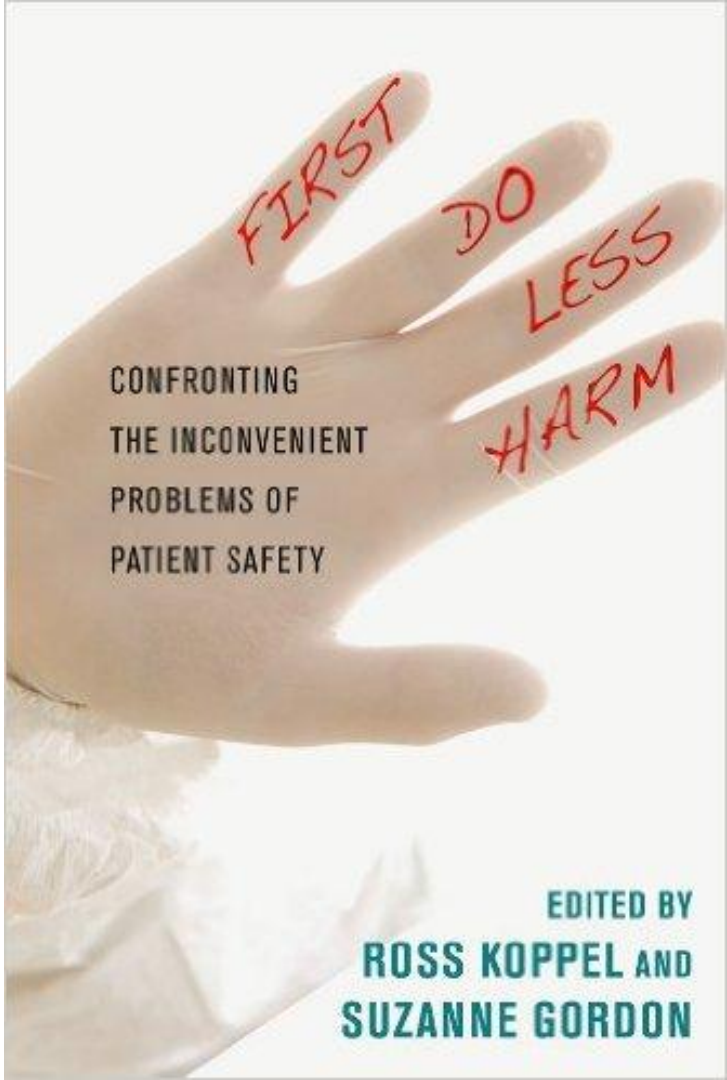
The “while we set up” non-slide.

Have international TCP connections?

You should investigate TCP-BBR (upstream in 4.9).

LWN: https://s442.net/h

Read this book → https://s442.net/i



The Sound of Silencing

Julien Goodwin
linux.conf.au 2017 Sysadmin Miniconf
jgoodwin@studio442.com.au @laptop006

Level 0 - No silencing

- Humans know to ignore “expected” alerts during maintenance events
  - Major failures can easily lead to alert overload
  - Easy for (usually) non-expert humans to miscorrelate an alert
  - What’s expected vs. unexpected?
- Still happens in major telcos
  - (One of these is the inspiration for this talk)

Level 1 - Turn all alerting off

- During planned work turn all alerts off
- Can work fine for smaller companies
- Stops being sensible once you need to keep service availability during instance-level events

Level 2 - Turn off a location while working on it

- What if something unrelated happens while you’re doing work?
  - ex: Router failure while upgrading a database
- May work for single-service deployments

Level 3 - Turn off only expected alerts

- Either manually, or tool assisted, disable expected alerts.
- How do you ensure you get it right?
- What if you forget to trigger the correct silences?

Level 4 - Change management integration

- Link the silence generator up to change management / automation systems.
  - Starting change triggers silence, marking complete removes.
- What about changes too small to track?
- What about changes too big for a simple silence?

Level 5 - Inhibiting Alerts

- Use service level indications (take a machine out of a load balancer pool) to avoid alerting on machines expected to fail.
- Fire “goes nowhere” alerts to achieve this in many monitoring systems

Level 6 - Global monitoring & preventing over-silencing

- Monitor on global live capacity, alert if too many sites down
- Use “always firing” alerts to trigger out of band alerts if they’re ever not firing

Getting there from here?

How do I recover from alert overload?

Incrementally.

Choose a bad alert, and do one change to make it better.

Regularly.

Eventually every alert should either be clearly legit, or fixed (removed or modified).

Questions?

Julien Goodwin
jgoodwin@studio442.com.au @laptop006