facebook

facebook

Using tracing at Facebook scale

- Yannick Brosseau
- Production Engineer – Kernel team
- scientist@fb.com
- 2014-10-13

Agenda

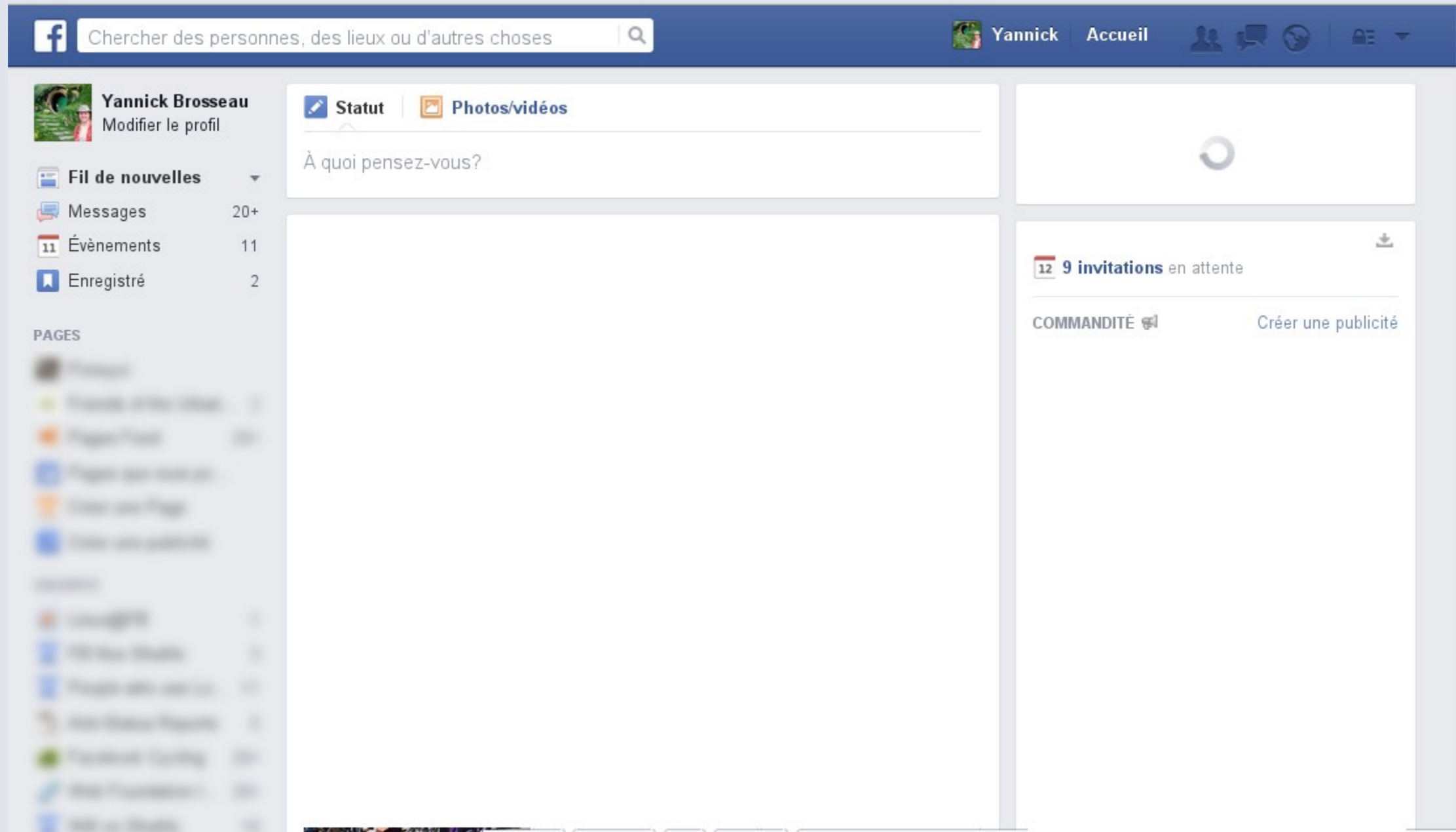|	
1	Current use
2	Challenges
3	Future

# A generic morning...

Kernel Development @FB

- An upstream first approach

- Running more recent kernels with less patches

- Sending our patches upstream so we don't have to forward port them on new kernels roll

- Don't have to fix problems already fixed

- Tracing: be able to run newer tools

What are we looking for ?

- Too long latencies

- General slowness

- Wrong behaviors
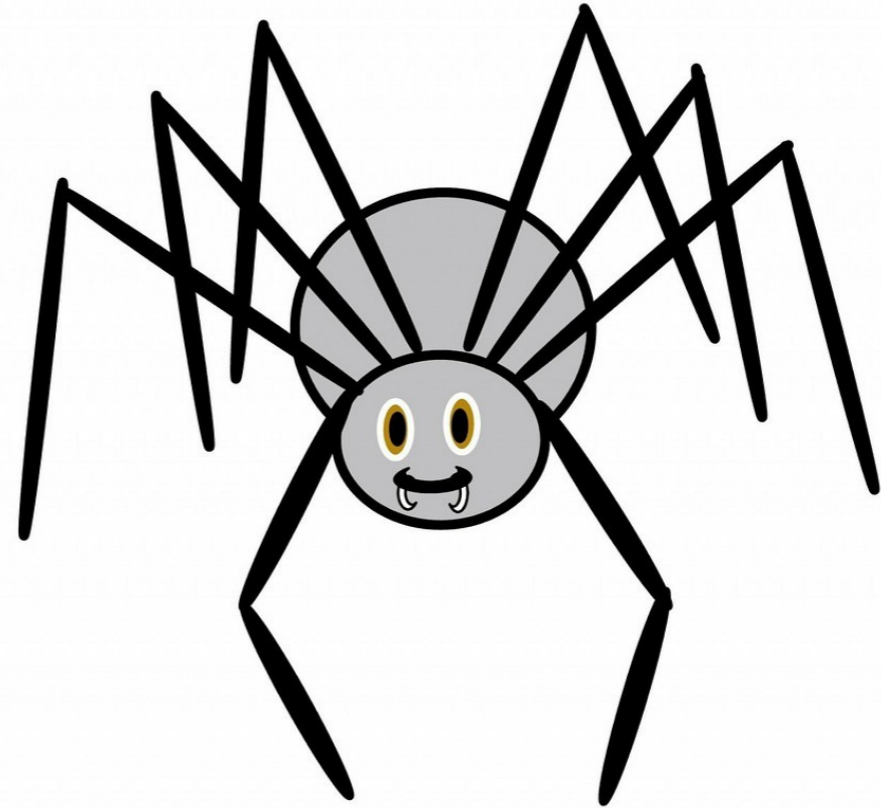
Current use of tracing
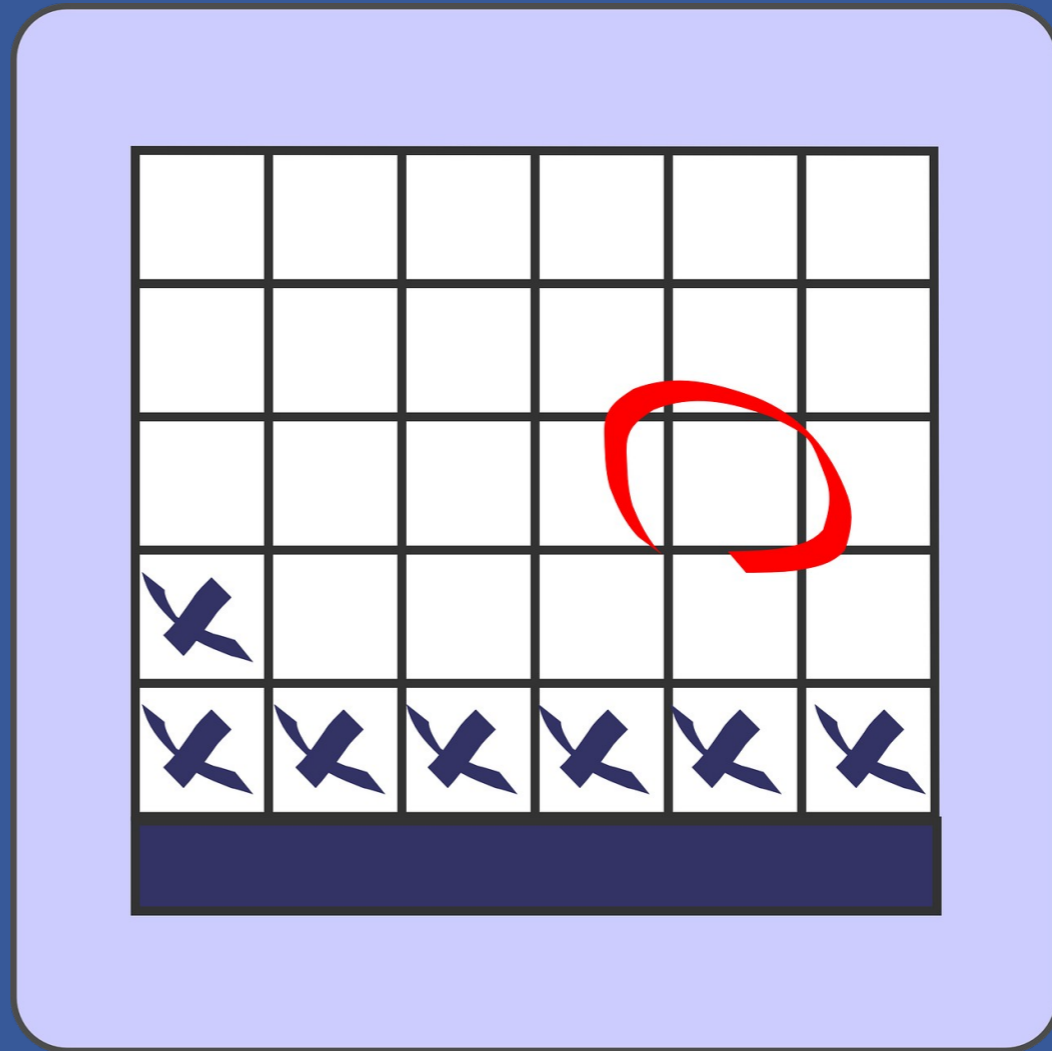
Userspace tracing

- Perflab

- Fbtrace

- etc

Kernel tracing – Debugging
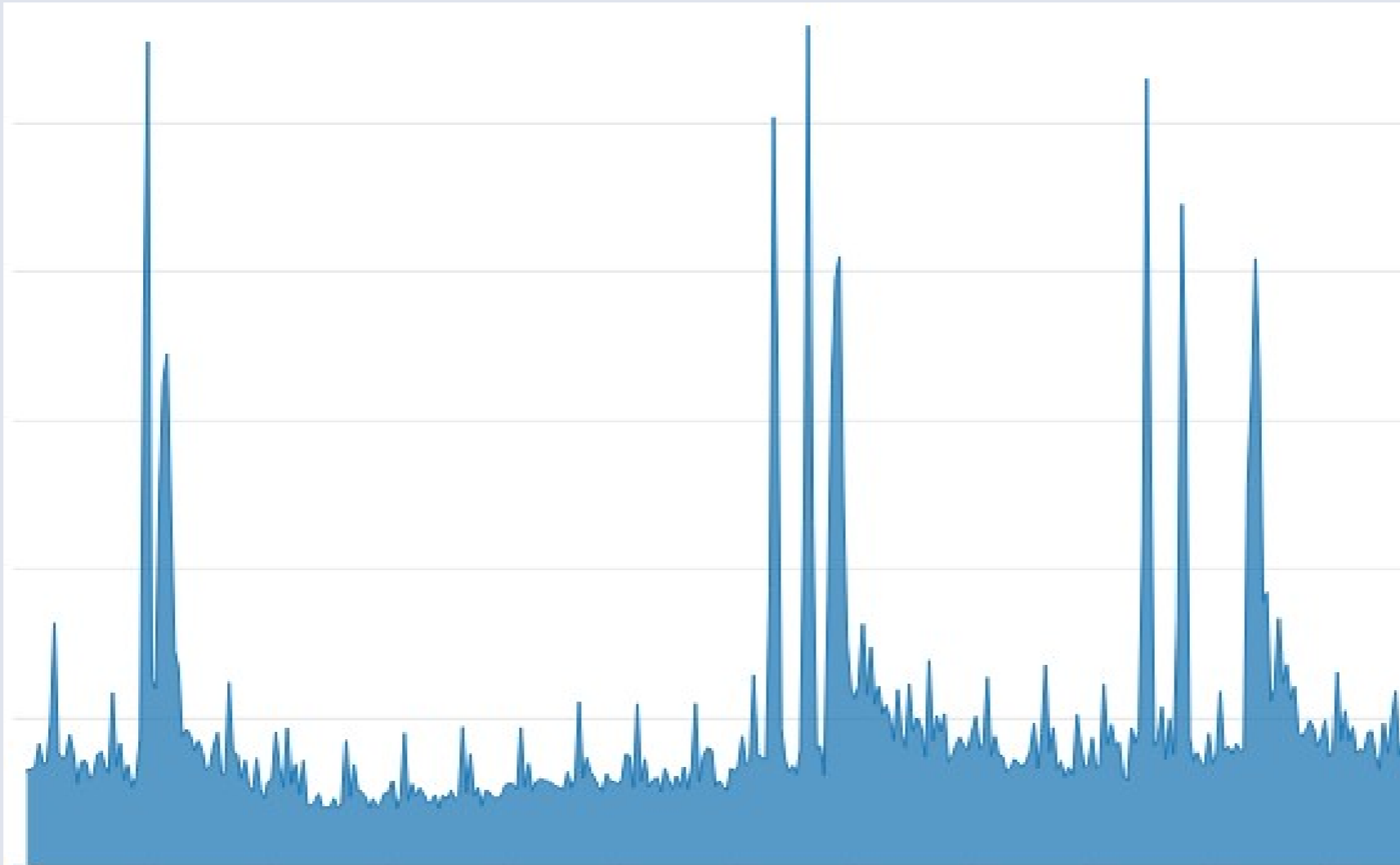
- SystemTap

- Perf

- Ftrace

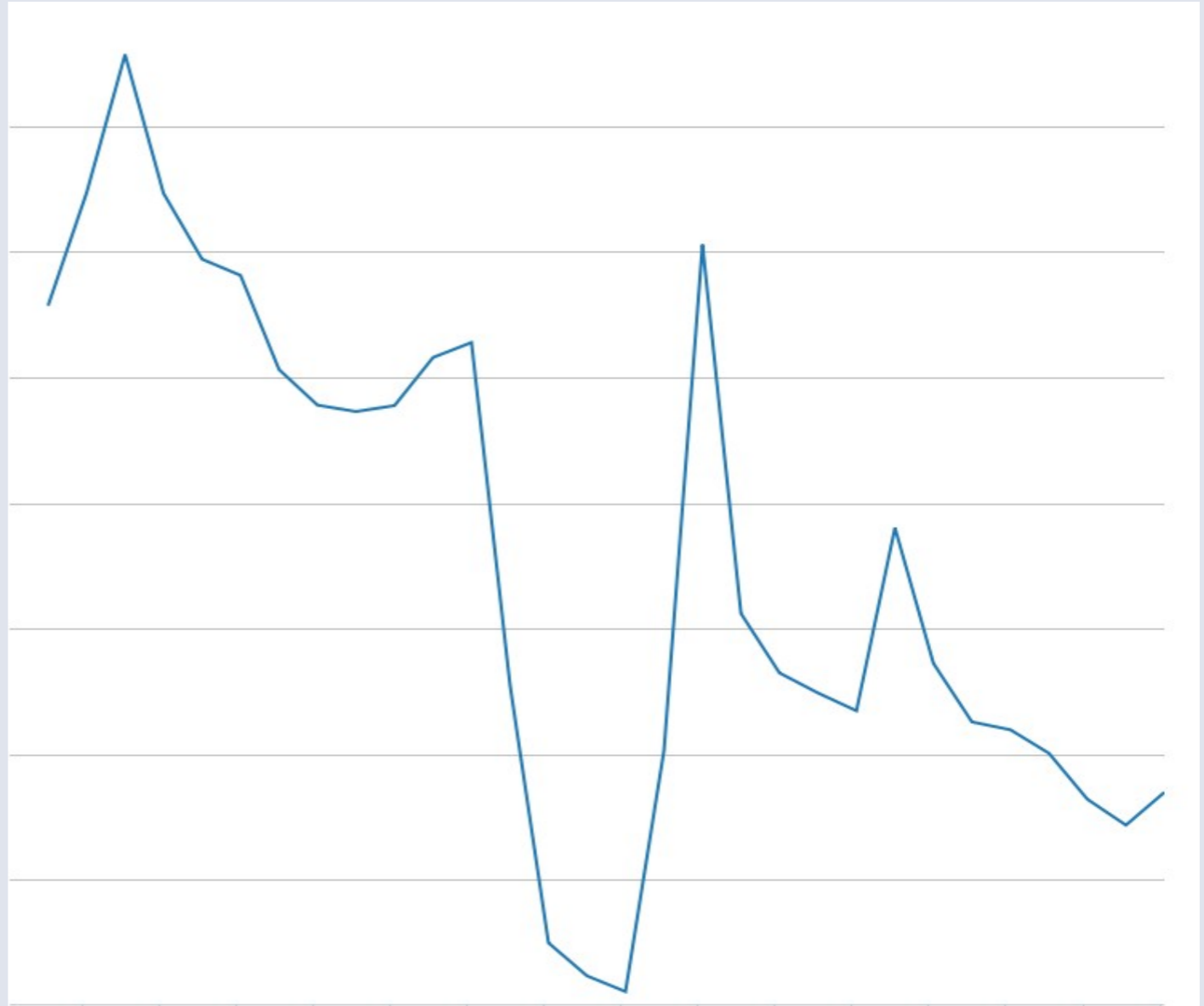- BlkTrace

Kernel tracing – monitoring

Kernel iostats

- Collect a sample of io tracepoints latency using ftrace.

- Log data into ODS

TCP estat

- Collect tcp statistics.

- Use ftrace buffer

Strobelight

- Take a perf snapshot

- Log into scribe and hive and displayed with Scuba

Other kernel issues monitoring

- Kdump crash collection

- Netconsole

- Kernel log collection

Current challenges

- Versionning
- Packaging
- Stability
- Performance
- Amount of data

Challenges – Versionning

- Kernel version mismatch

- Compiler version mismatch

- Running newer version on old distribution

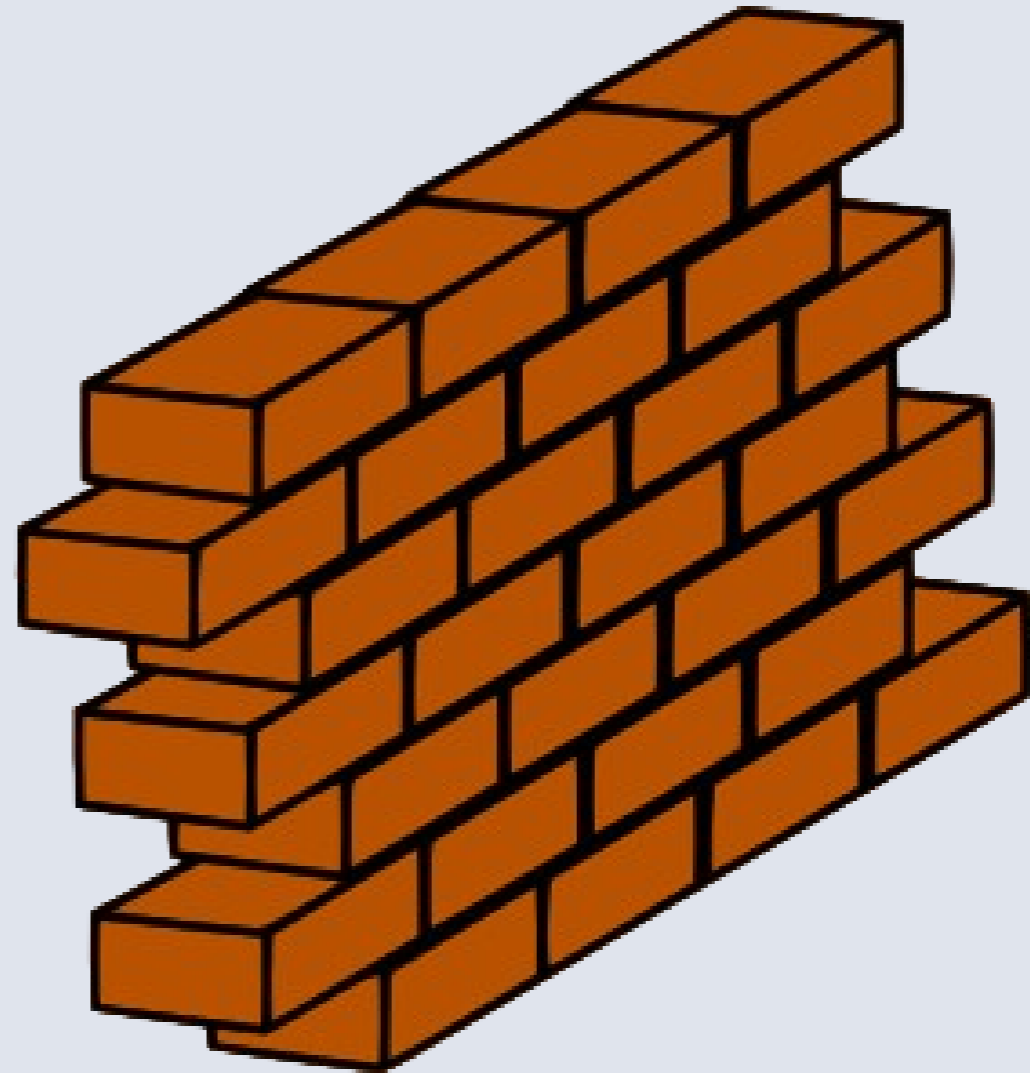- Or running old versions on newer kernels

Challenges – Packaging and distributing

- Perf source/packages

  - No standalone source

- Trace-cmd packages

  - No recent version in distro

- LTTng kernel modules
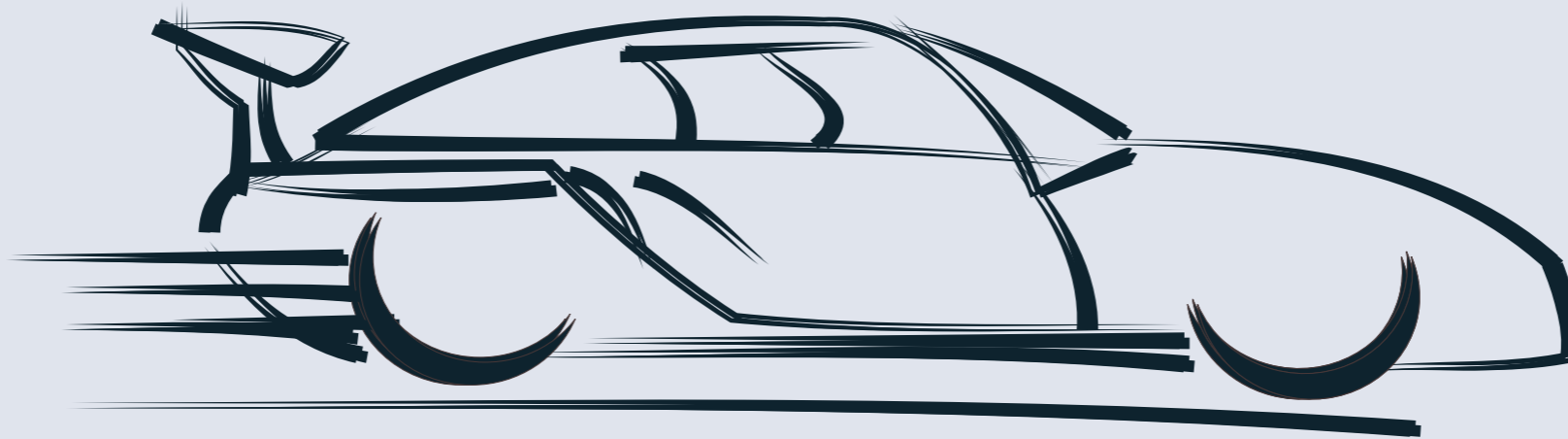
  - Out of tree modules
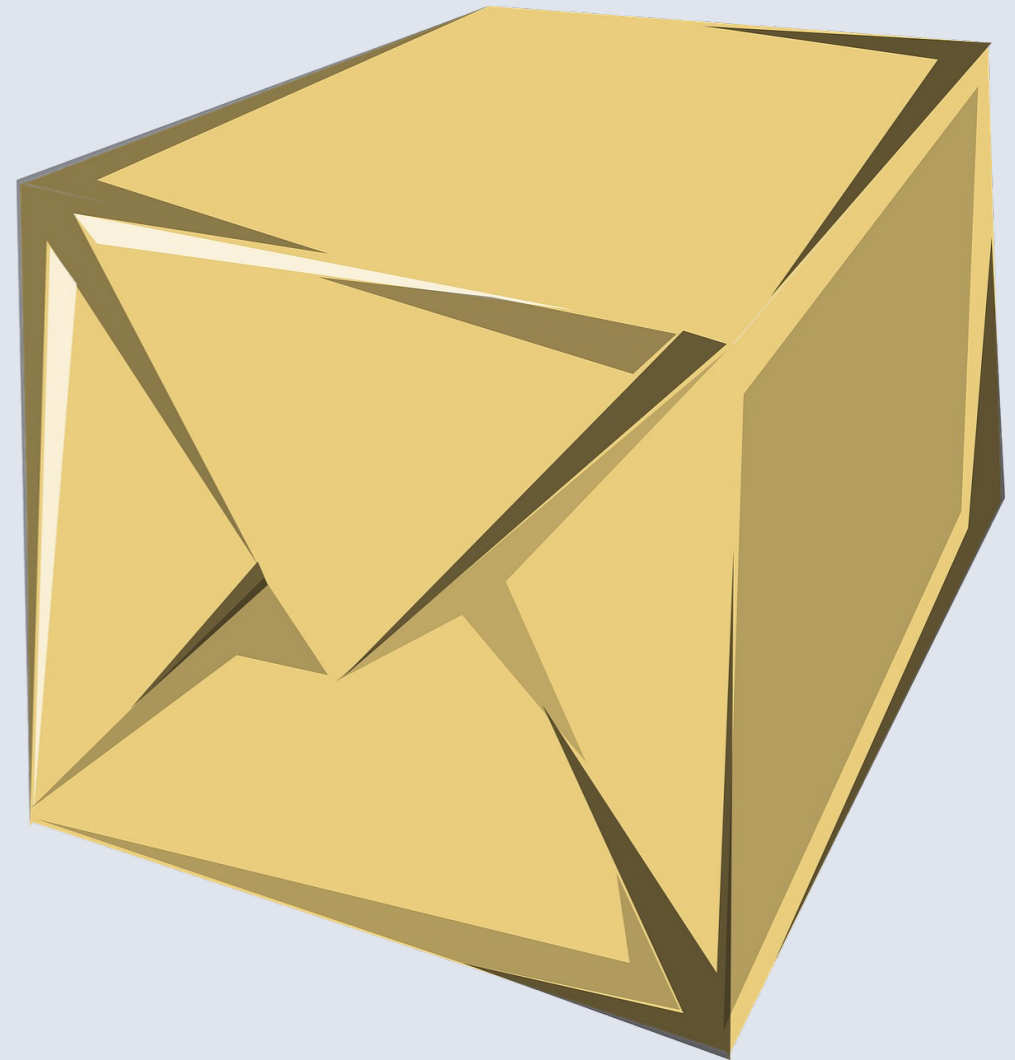
Challenges – Stability

- Crashes

- Lockup

- Memory leaks

Challenges – Performance

- Seem ok

Challenges - Amount of data

- Too much data to be useful?

- Sampling is good at our scale but can still miss errors

Future plans and wishlist

Dynamic tracing

More network tracepoints

Extract trace buffer from crash dump

Upstream tcpestat

Auto sampling

- Record 1 sec each minute

More analysis

Questions / Comments?

facebook