Do the Time Warp – the Rocky Horror PTP Show

Verification of Network Time Synchronization in the Real World

Johannes Zink – j.zink@pengutronix.de
$whoami

- Johannes Zink, a.k.a. dickenhobelix
- Former Systems Engineer, mostly worked on Realtime Networking Systems for ~10 years
- Senior Kernel Developer at Pengutronix since 2022

- Embedded Linux consulting & support since 2001
- ~ 8700 patches in Linux kernel

✉️ j.zink@pengutronix.de
Disclaimer

- This talk contains references to the Rocky Horror Show and therefore some profanity. Viewers discretion is advised.
- Rocky Horror is about Interaction, so let’s get interactive!
- This talk represents my personal experience
- YMMV, especially on other hardware
- This talk and its examples were prepared with linuxptp v3.1, in the meantime v4 has been released, stability has improved a lot
Agenda

- A brief Introduction to the Precision Time Protocol
- Linux Kernel Components
- Userspace Components
- Measurement Methods
- Examples
- Common Pitfalls
- Best Practices
- Recap and Q&A
Science Fiction – Double Feature

- Synchronize Multiple Clocks over a Network
- Automagically select the best reference clock possible
- Compensate for Network Path delays
  - Unknown path delay
  - Queuing
A brief Introduction to PTP – 2step sync

Leader

T1

Sync

Follow-Up (T1)

Delay Request

T4

Delay Response (T4)

Follower

T2

T3

Timepoints known to Follower

T2
T1, T2
T1, T2, T3
T1, T2, T3, T4

delay = ((T2 – T1) + (T4 – T3)) / 2

offset = (T2 -T1) - delay
A brief Introduction to PTP – BMCA

- Every Clock Listens, may announce its Capabilities, if it thinks its better than what is already announced in the network
- Decision Tree
  - User configurable Fields (priority1, priority2)
  - Clock Quality (clock class, clock accuracy, clock variance)
  - Tie braker (clock identity, port number)
- TL;DR: Autoselect the best possible clock, unless user configures otherwise
Clock Types in Bridges
- Boundary Clock: Run Leader/Follower per Port
- Transparent Clock: Adjust Messages for Queuing Delay
- Ordinary Clock: Just pass Messages without Correction

Sync Type (1step vs 2step)

Transport Layer

Delay Measurement (P2P vs E2E)

Protocol Extensions (Redundancy)
PTP in Linux - Kernel

- PTP Hardware Clock
  - See Documentation/driver-api/ptp.rst

- Packet Timestamping Support
  - Software, or – better – Hardware offloaded
  - Software Timestamping: generic in network stack
  - HW Offloading usually done in the MAC/NIC drivers, sometimes in the PHY drivers
  - See Documentation/networking/timestamping.rst
PTP in Linux - Userspace

- **linuxptp**
  - well-established and maintained codebase
  - supports many different profiles
  - tricky to configure
  - only seldom releases new versions – use master quarterly release schedule as of version 4.0

- some other projects exist
  - often only cover smaller subsets of profiles
  - often „industry code quality“
PTP in Linux – Measurement Methods

- compare 2 Systems’ outputs directly against each other
  - Scope, Time Domain Analyzer
- Reverse-Sync Method
  - Send additional Sync Messages from Follower (DUT) to Leader
- Ingress Measurement Method
  - Follower (DUT) reports incoming sync errors against own reference
- Egress Measurement Method
  - Follower provides timestamps to its messages sent to Leader
Dammit, Janet

- Murphy’s law is very strong when setting up PTP
- Settings offer lots of possible permutations
- The list of possible misconfig given in this talk is not exhaustive

- Verify your measurement Setup
- Add plausability checks
- Always check your assumptions!
Demo Setup

GNSS Receiver

PPS Capture ➔ PHC ➔ PTP Leader ➔ PTP Follower ➔ PHC ➔ PPS out

Scope

Reference

DUT
Reference – Good Measurement

- Synchronized
- Stable over >3h
  - Persistency → ∞
- Excursions normally distributed
Touch-A-Touch-A-Touch-A-Touchy... (1)

- Unsynchronized
  - Reference Signal Capture failed
  - Link Issue
  - Incompatible Settings

...
Touch-A-Touch-A-Touch-A-Touchy... (2)

- Large time offset
- Only momentary snapshot
- cannot determine Jitter and Wander
- TL;DR: unknown state
Touch-A-Touch-A-Touch-A-Touchy... (3)

- Trailing edge
- Syntonomous, but not synchronous
  - Initial offset at PPS generation
  - Timescale
  - Delay over-compensated
  - broken PHC
- EEE enabled
- Influences
- Timestamping
- Asymmetric distribution of error

- temporary Sync loss
- Here: leader missed tx ts interrupt
- Permanent Loss of leader
- Thermally stable system
- Too short measurements (15 min)
Reference – Startup

ptp4l[36.485]: delay filtered        378   raw        374
ptp4l[36.819]: port 1: UNCALIBRATED to SLAVE on MASTER_CLOCK_SELECTED
ptp4l[37.445]: rms 17871811494171372 max 35743622988344432 freq -23588 +/- 8915 delay 378 +/- 0
ptp4l[37.485]: port 1: delay timeout
ptp4l[37.485]: delay filtered        377   raw        374
ptp4l[38.446]: rms 68 max    78 freq -26851 +/- 27 delay 377 +/- 0
ptp4l[38.485]: negative delay     -87918
ptp4l[38.485]: delay = (t2 - t3) * rr + (t4 - t1)
ptp4l[38.485]: t2 - t3 =         +0
ptp4l[38.485]: t4 - t1 =    -175836
ptp4l[38.485]: rr = 0.00000000056
ptp4l[38.485]: delay filtered        375   raw     -87918
ptp4l[39.447]: rms 58 max    79 freq -26797 +/- 11 delay 375 +/- 0
ptp4l[39.485]: port 1: delay timeout
ptp4l[39.485]: negative delay     -79398
ptp4l[39.485]: delay = (t2 - t3) * rr + (t4 - t1)
ptp4l[39.485]: t2 - t3 =         +0
ptp4l[39.486]: t4 - t1 =    -158796
ptp4l[39.486]: rr = 0.00000000056
ptp4l[39.486]: delay filtered        374   raw     -79398
ptp4l[40.448]: rms 22 max    42 freq -26819 +/- 21 delay 374 +/- 0
ptp4l[40.485]: port 1: delay timeout
ptp4l[40.486]: delay filtered        374   raw        370
ptp4l[41.312]: clearing fault on port 1
ptp4l[41.449]: rms 22 max    28 freq -26874 +/- 12 delay 374 +/- 0
...

note: Output looks slightly different for L3 Transport
Reference – stable running Follower

...
Follower looses Sync from Leader

...  
ptp4l[2216.058]: delay filtered 16065 raw 16060
ptp4l[2216.106]: port 1: master sync timeout
ptp4l[2216.231]: port 1: master sync timeout
ptp4l[2216.356]: port 1: master sync timeout
ptp4l[2216.474]: port 1: master sync timeout
ptp4l[2216.481]: port 1: master tx announce timeout
ptp4l[2216.606]: port 1: master sync timeout
ptp4l[2216.731]: port 1: master sync timeout
ptp4l[2216.856]: port 1: master sync timeout
ptp4l[2216.981]: port 1: master sync timeout
ptp4l[2217.058]: port 1: delay timeout
ptp4l[2217.058]: delay filtered 16065 raw 16080
ptp4l[2217.106]: port 1: master sync timeout
ptp4l[2217.231]: port 1: master sync timeout
ptp4l[2217.356]: port 1: master sync timeout
ptp4l[2217.474]: port 1: master tx announce timeout
ptp4l[2217.481]: port 1: master sync timeout
ptp4l[2217.606]: port 1: master sync timeout
ptp4l[2217.732]: port 1: master sync timeout
ptp4l[2217.857]: port 1: master sync timeout
ptp4l[2217.982]: port 1: master sync timeout
...

- One of the intermediate links falls back to half-duplex
...  

ptp4l[2840.920]: port 0: INITIALIZING to LISTENING on INIT_COMPLETE  
ptp4l[2840.920]: port 1: received link status notification  
ptp4l[2840.920]: interface index 3 is down  
ptp4l[2840.920]: port 1: link down  
ptp4l[2840.920]: port 1: LISTENING to FAULTY on FAULT_DETECTED (FT_UNSPECIFIED)  
ptp4l[2840.952]: waiting 2^{4} seconds to clear fault on port 1  
ptp4l[2840.952]: selected local clock 00049f.fffe.079c42 as best master  
ptp4l[2840.952]: port 1: assuming the grand master role  
ptp4l[2856.952]: clearing fault on port 1  

...
Touch-A-Touch-A-Touch-A-Touchy... (10)

- Hardware Bug, incomplete Driver
- Too Large Peer Delay → Loss of Sync Domain Membership
Leader missing a TX Timestamp
Causes increasing Fault Backoff
Do the time warp - common Pitfalls

- Multiple Time sources in the system (NTP, ...)
  - system time jump
  - NTP also tunes the PHC if using HW timestamping
- PTP profiles
- Missing, incomplete or defective timestamping and clock support in hardware or drivers
- Timestamping in MAC vs in PHY
- Hardware often only supports subset of profiles (1step/2step, Layer2/Layer3, P2P/E2E, ...)

```
Do the time warp - common Pitfalls

- Multiple Timescales – Offsets, Leap Seconds, etc
- False positive Debug output
- Daemon stability
- Measurement Method – PPS frequency: resolution vs precision, rollovers
- Sporadic dropouts
  - delayed TX timestamps (mitigate with --tx_timestamp_timeout)
- Check selected leader clock
- Never rely solely on datasheets! Measure yourself!
Best Practices

- Choose the correct profile. Often this is dictated by the application anyway
- Do not rely on example commands in the internet™
- Check Hardware Clock Availability, Stability and Configuration
- Analyze Logs in Endpoints and in bridges
- Read the fine Manpage – subtle differences in parameters
- Thoroughly test over $aLongerPeriodOfTime™
- Always check your assumptions!
Recap

- PTP can offer great performance when done right
- PTP has lots of parameters to tweak
- Performance and stability depends strongly on optimizing your setup properly
- Some effects only sporadically show up
- There’s a light in everybody’s measurement setup...
- Always check your assumptions!
By the way...

... we are hiring

jobs.pengutronix.de
Thank you for your Attention!

Any questions?

✉️ j.zink@pengutronix.de