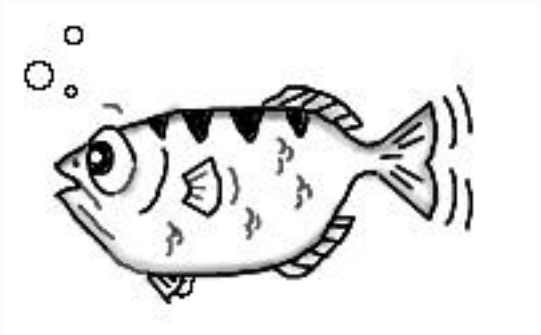
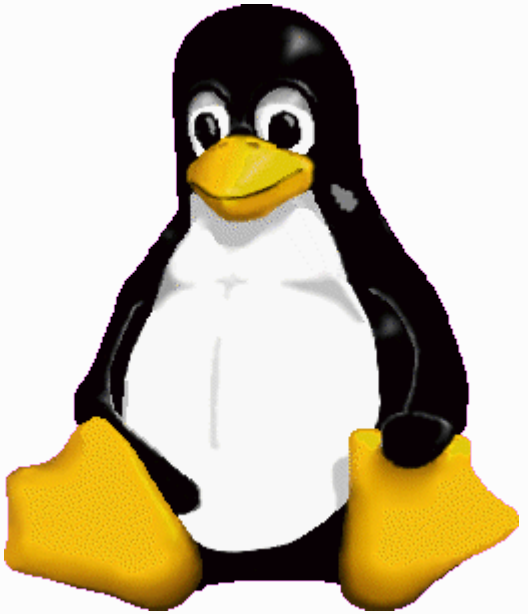
KGTP, Linux 内核中的 GDB 快刀





kgtp.googlecode.com

朱辉 (teawater@gmail.com)

什么是 KGTP ？

- KGTP 是 Linux kernel GDB tracepoint module。
- KGTP 是一个 灵活 轻量级 实时 Linux 调试器 和 跟踪器 。

- 处理在线服务器上的问题。
- 处理嵌入式系统中的问题。
- 被一些公司使用，最主要的是在去年 1 月合入了 alibaba(taobao) 的内核 tree。

轻量级 代码轻量级

- 主要是开发者利用业余时间维护，所以是一个轻量级的项目。
- 因为分析数据主要使用 GDB 所以不需要很多数据分析代码。

  另：因为对 GDB 的依赖，所以 http://code.google.com/p/gdbt/ 提供 GDB for KGTP 支持。 x86_64 和 i386 的 static GDB 下载，还提供 Ubuntu PPA， Opensuse 源支持， Ubuntu, fedora, debian 等软件都可支持直接安装。

- KGTP 20130915:

  gtp.c 大小 277K 13045 行

轻量级 实现轻量级

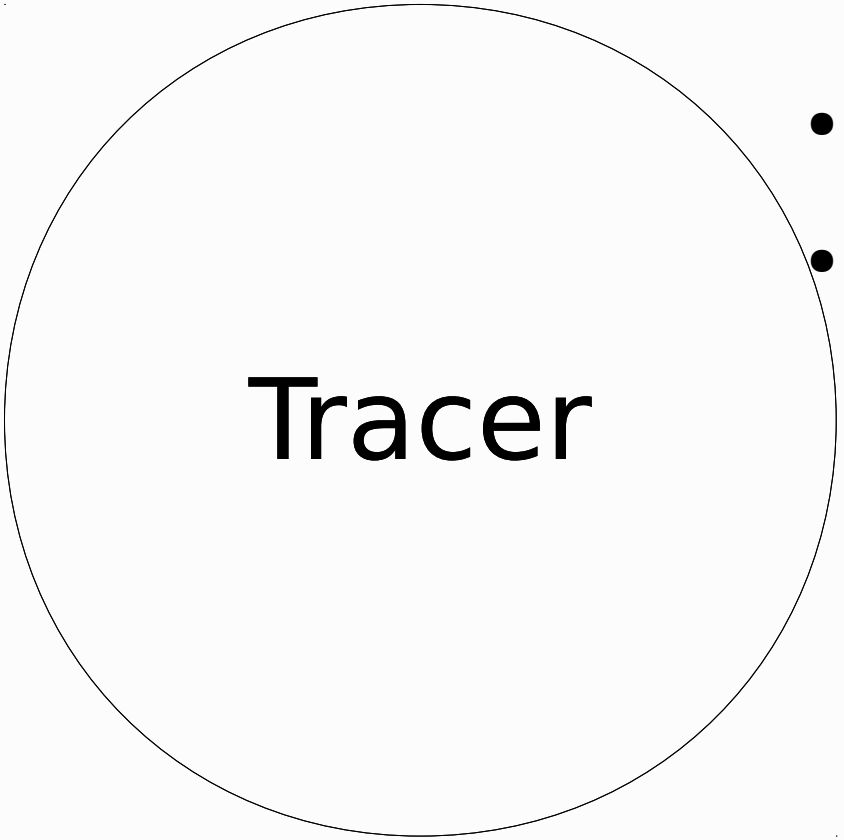
- 大部分 trace 功能基于 Kprobe 。

  Kprobes-optimization 还可提高 kprobe 的速度。

- 所以大部分时间，使用 KGTP 不需要重新编译内核。只需要编译 KGTP 模块和 insmod 。

- 当不需要 KGTP 的时候，只需要 rmmod 。跟什么也没发生一样。

实时

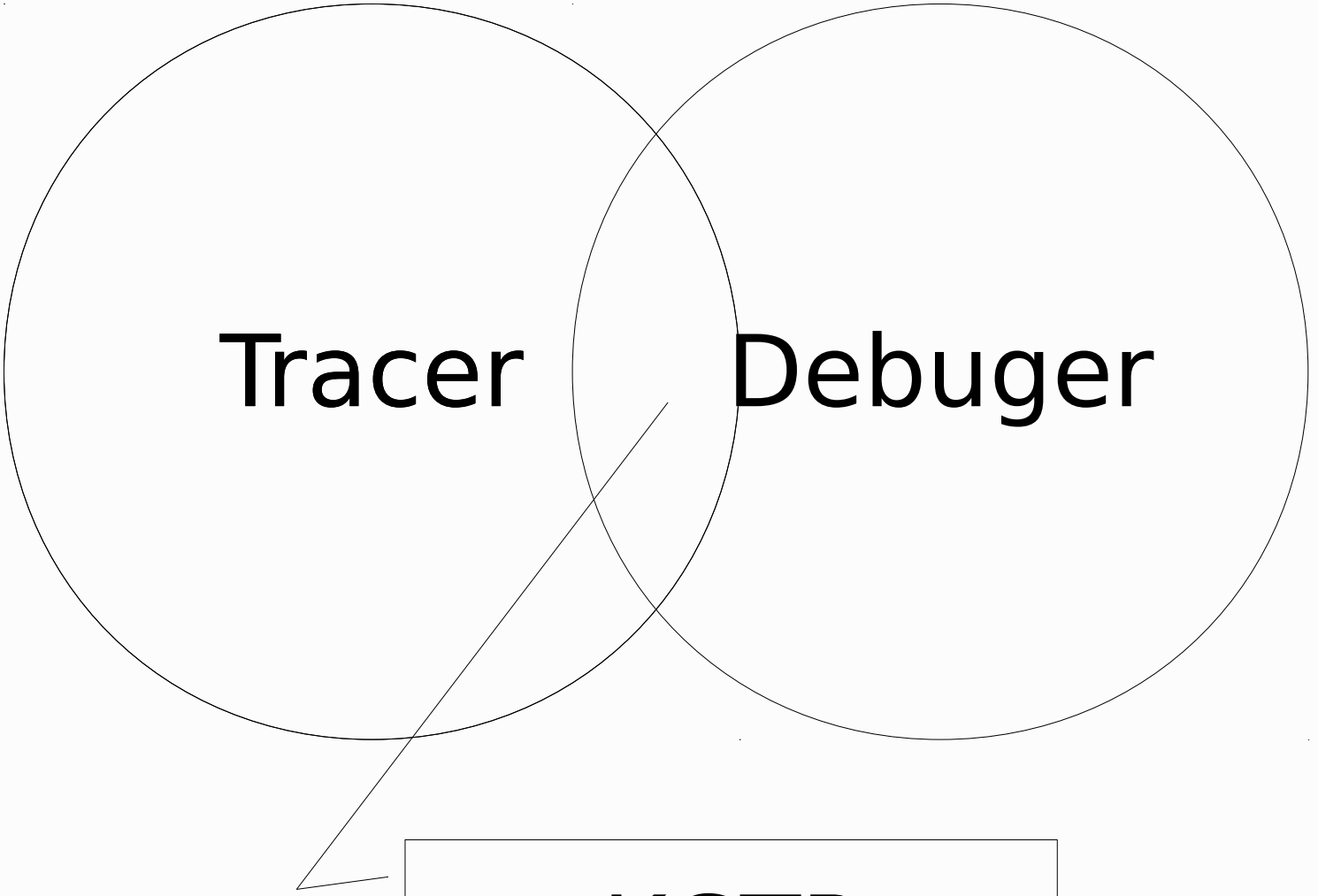


- 不停止 Linux 内核
- 不能被 GDB 控制

实时

- 将停止 Linux 内核
- 可以被 GDB 控制

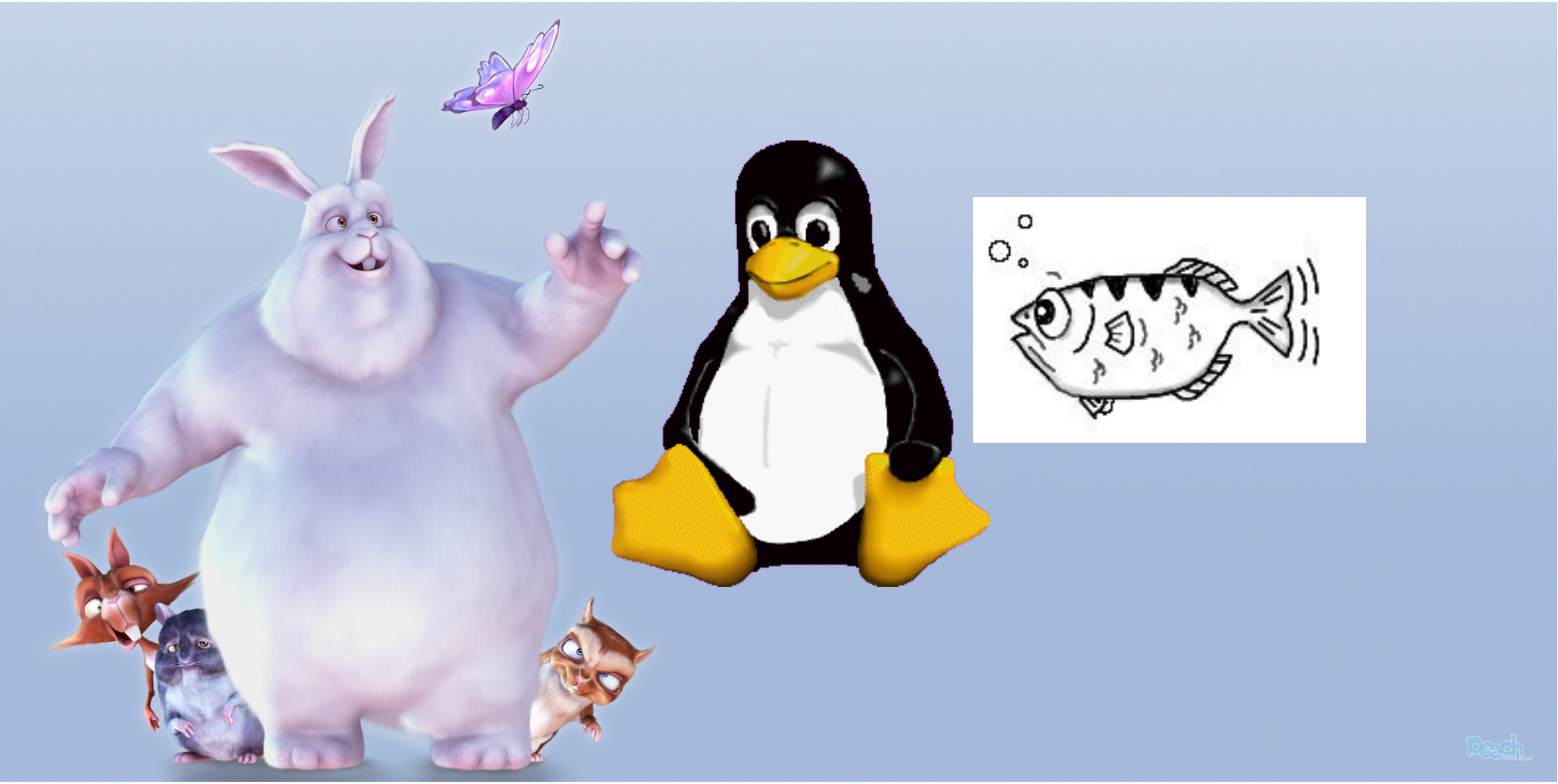
实时



KGTP

不停止 Linux 内核
可以被本地或者远程的 **GDB** 控制

演示助手



实时 演示

- 演示 1： 直接访问 Linux 内核中的变量
- 演示 2： trace 内核
- 演示 3： 直接访问用户程序的内存
- 演示 4： trace 用户程序
- 演示 5： trace 用户程序的系统调用

灵活 对内核的支持灵活

- 灵活算是轻量级引入的带来的优点。

- 大部分时候，使用 KGTP **不需要** 在 Linux 内核上打 PATCH 或者重新编译，只要编译 KGTP 模块并 insmod 就可以。
- 支持 **X86-32**， **X86-64**， **MIPS** 和 **ARM** 。
- 支持 **Linux** 内核 **2.6.18** 到 **upstream** 。
- 支持 **Android**：

  演示 6： 连接 Android 中的 KGTP。

灵活 连接方式灵活

- 从本机 GDB 连接 KGTP 。
- 从远程主机 GDB 连接 KGTP 。

- GDB 不连接 KGTP - 离线调试。

  演示 7：在 Android 上离线调试。

灵活 灵活的设置 tracepoint

- 灵活的在不同的地址设置 tracepoint 包括 inline 函数中。
- 用 tracepoint actions 可以设置 tracepoint 被触发后的行为。
- 用 tracepoint condition 可以设置 tracepoint 被触发的条件。

灵活 灵活的 trace 状态变量 (TSV)

- Trace 状态变量 (TSV) 是 GDB tracepoint 的子功能。
- TSV 可以被用户或者调试目标 (KGTP) 定义。
- TSV 的值从 KGTP 中取得。
- GDB 可以在任何时候读 TSV 的值。

  tracepoint actions 可以访问 TSV 的值。

KGTP 中的 TSV

- 用户可以定义普通 TSV.
- 用户可以定义普通 per-CPU TSV.
- KGTP 定义了很多特殊 TSV 支持一些特殊功能，例如 $bt ， $current_task_pid 。
- 用户可以用插件增加 TSV 到 KGTP 中，这样 KGTP 可以调用内核中的任何函数并且取得返回值。

  演示 8: 演示 KGTP 中的插件。

灵活

- 演示 8

  这个例子记录了每个 CPU 上关闭 IRQ 时间最长的函数的 stack dump 。

灵活 灵活的数据处理方式 python

- GDB 支持 PYTHON 脚本。
- 演示 9： 在演示 8 取得的 trace frame 中用 python 脚本找出每个 CPU 最慢的 frame id 。

灵活 直接将数据输出到系统日志

- KGTP 可以通过特殊格式的 action 命令输出数据到系统日志。
- 其结合离线调试功能让调试更加方便。

Watch tracepoint

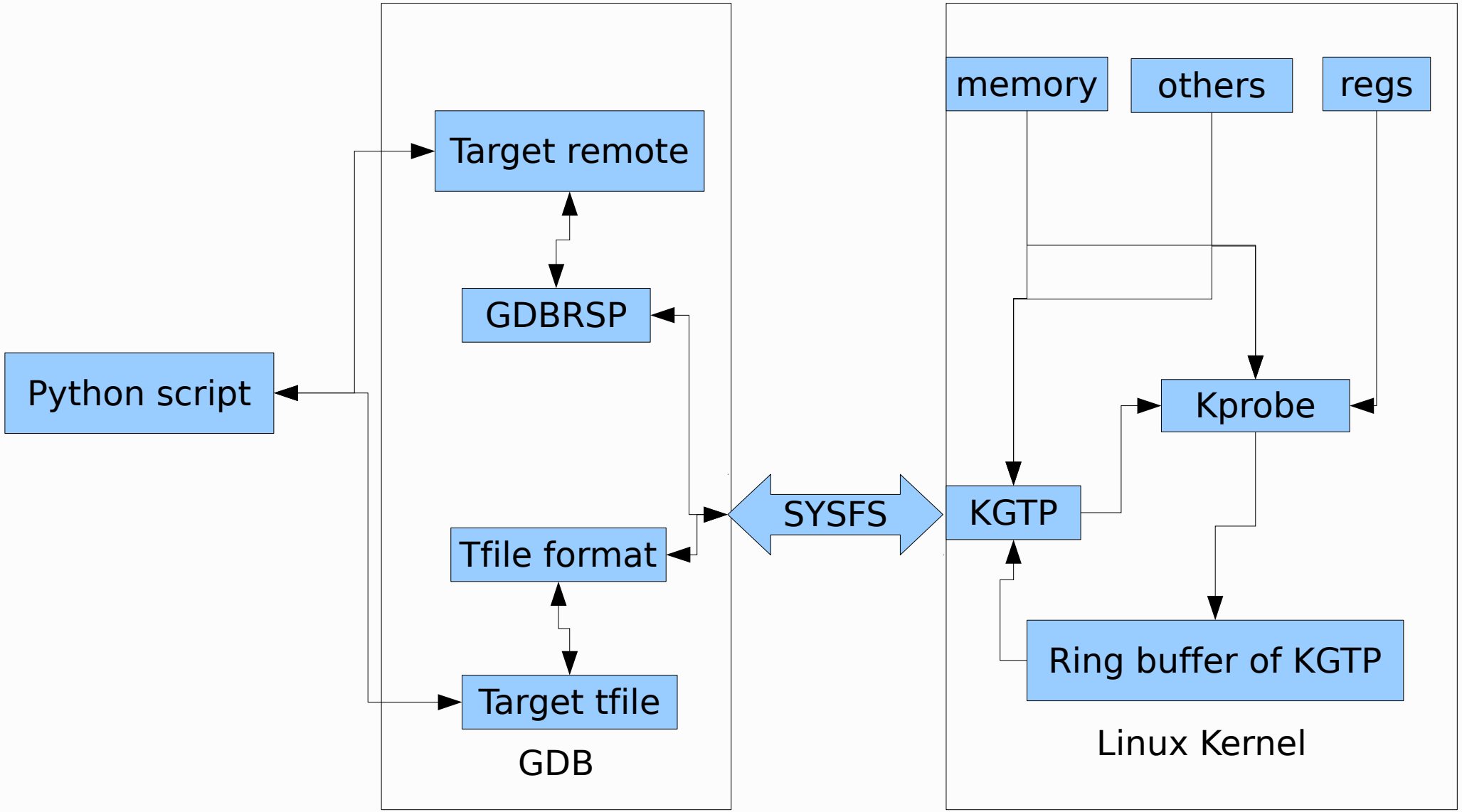
- Watch tracepoint 可以控制硬件断点记录 Linux kernel 的内存访问。 X86 支持这个功能。
- 演示 10: 静态 watch tracepoint.
- 演示 11: 动态 watch tracepoint.

单步和反向执行调试

- 使用 while-stepping 让 Linux 内核做单步。并用 GDB 执行命令（包括反向调试命令）回放 traceframe。

  演示 12: 做单步和回放 traceframe。

KGTP 的结构


Target remote
GDBRSP
Python script
Tfile format
Target tfile
GDB
SYSFS
memory
others
regs
Kprobe
KGTP
Ring buffer of KGTP
Linux Kernel

URL

- 主页 http://kgtp.googlecode.com
- 中文 Howto
  http://code.google.com/p/kgtp/wiki/HOWTOCN

  https://raw.github.com/teawater/kgtp/master/kgtpcn.pdf
- 邮件列表 http://www.freelists.org/list/kgtp
- 我的信箱 teawater@gmail.com
- Irc.freenode.net #hellogcc teawater

谢谢！

问题？