<https://blog.csdn.net/yuzegao/article/details/39530043>

kamailio和opensips在代码核心架构上是一致的，下面以kamailio为例说明如何使用gdb进行调试。调试程序有两种方法：

方法1。直接用GDB运行kamailio：将kamailio.cfg配置为**fork=no， 这样kamailio会以单进程方式运行**，所以的逻辑都将在一个进程里执行，但单进程模式不支持TCP listen，所以要调试TCP逻辑，不能使用此方式。直接使用gdb调试kamailio主程序就行：

[root@xx ~]# gdb /usr/sbin/kamailio

GNU gdb (GDB) Red Hat Enterprise Linux (7.0.1-45.el5)

Copyright (C) 2009 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <<http://gnu.org/licenses/gpl.html>>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.  Type "show copying"

and "show warranty" for details.

This GDB was configured as "x86\_64-redhat-linux-gnu".

For bug reporting instructions, please see:

<<http://www.gnu.org/software/gdb/bugs/>>...

Reading symbols from /usr/sbin/kamailio...done.

(gdb) b main.c:10

Breakpoint 1 at 0x45725c: file main.c, line 10.

(gdb) run

方法2。将kamailio运行起来后，再用gdb attach调试：第一种方法的缺点非常明显，不能调试TCP进程，也不能将kamailio多进程的优势发挥出来。将kamailio.cfg配置为fork=yes将以多进程方式启动，与此相关的参数还有两个：

children=1            ;配置处理udp端口数据的进程数量

tcp\_children=1     ;配置处理tcp端口数据的进程数据，如果此参数未配置，默认使用children参数。

如果使用以上参数启动程序，如果listen参数有对应的端口，将会启动1个UDP处理进程和1个TCP处理进程，实例如下：

[root@xx sipserver]# kamctl ps

Process::  ID=0 PID=6651 Type=attendant

Process::  ID=1 PID=6653 Type=udp receiver child=0 sock=172.16.0.16:53

Process::  ID=5 PID=6661 Type=slow timer

Process::  ID=6 PID=6662 Type=timer

Process::  ID=7 PID=6665 Type=MI FIFO

Process::  ID=8 PID=6668 Type=ctl handler

Process::  ID=9 PID=6669 Type=SNMP AgentX

Process::  ID=10 PID=6672 Type=tcp receiver (generic) child=0

Process::  ID=14 PID=6679 Type=tcp main process

当kamailio收到TCP数据时，会由6672进程进行处理，我们可以使用gdb attach到指定进程进行调试，实例如下：

[root@xx ~]# gdb

GNU gdb (GDB) Red Hat Enterprise Linux (7.0.1-45.el5)

Copyright (C) 2009 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <<http://gnu.org/licenses/gpl.html>>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.  Type "show copying"

and "show warranty" for details.

This GDB was configured as "x86\_64-redhat-linux-gnu".

For bug reporting instructions, please see:

<<http://www.gnu.org/software/gdb/bugs/>>.

(gdb) attach 6672

Attaching to process 6672

......

(gdb) b redis\_base.c:100

Breakpoint 1 at 0x2b6c45e70082: file redis\_base.c, line 100.

(gdb) c

Continuing.

然后再设置断点，执行run即可，当服务器收到SIP消息并运行到指定代码即可单步调试，在调试过程中再结合日志输出，对定位问题非常有帮助。

关于如何使用gdb请参考文档：

<http://wenku.baidu.com/link?url=UNUB8A6f2OGt_KJxU4p7-df-j9QFw2z2CFN6nCj4smzijx_UiMIkUTFh7JlOE-byIylp2yGfaTbtKPpOGR3lhTd09p3NTQtZDPrWp9N5pmu>