

Guide to localize crashes and to report about them

Introduction

For the debugging and system testing of the OpenSCADA it is given considerable time of developers, but in view of limited resources and even a practical impossibility to cover all the configuration options and performance of the OpenSCADA system, errors can occur either as a failure of some functions, incorrectness of their implementation and even crash of the program. Understanding of this from the user side is very important, and with his hand it is required goodwill to cost some time to prepare a report on the problem with a view of its subsequent removal by developers.

Before preparing the report of the problem it is recommend to review the list of errors and comments in the corresponding part [Errors and comments to the OpenSCADA system](#).

Requirements for the report of an error

To avoid unnecessary suggestive questions and to accelerate the localization of the problem it is recommended to follow the following requirements for the report.

- To specify the execution environment of the OpenSCADA, namely the distribution and version of operating system.
- To specify the version of the OpenSCADA system.
- To specify the particular features of the configuration and execution. It is desirable to achieve the demonstration of the problem in standard configurations and on the demonstration database. If this can not be achieved, it is necessary to describe the particular configuration and to attach to the report the database, in which the problem takes place.
- Describe the actions causing the error.
- To attach the report of OpenSCADA messages for the session with the error.
- In the case of the crash it is necessary to form a file of pre-mortem dump of the program and get out from him the spread of the stack of execution at the time of the crash.

Variants of the report

To report about an error in the OpenSCADA system it is possible in several ways:

- by e-mail address oscada@oscada.org or [developers addresses](#);
- in the appropriate forum section of the official site: <http://oscada.org/en/forum> ;
- in the part [Errors and comments to the OpenSCADA system](#) in the knowledge base of the project;
- in the [Bug Tracking System](#) of the distribution kit ALTLinux of the OS Linux for the package openscada.

Getting the pre-mortem dump file and its processing

In the Linux operating system while crashing of programs the OS kernel can generate the pre-mortem memory dump of the program. With this dump it is often possible to identify a place in the program that caused the emergency stop. To enable the generation of pre-mortem memory dump of the program by the kernel it is necessary to execute the following commands:

```
# Check the possibility of generation of memory dumps
# Specifies the "core" if it is enabled
$ cat /proc/sys/kernel/core_pattern
```

```
# Enabling the generation of memory dumps
$ echo "core" > /proc/sys/kernel/core_pattern
```

After this you need before calling OpenSCADA to specify the command "ulimit-c unlimited". For example, it can be done in the calling scripts of the special configuration: /etc/openscada_demo_ru and /etc/openscada_start:

```
#!/bin/sh
ulimit -c unlimited
openscada --Config=/etc/oscada_demo_ru.xml $@
```

Then you can just make sure that the user which running OpenSCADA, is entitled to write in the working directory OpenSCADA, parameter "Workdir" in the configuration file of the program.

Then the OpenSCADA system is running and the crashe is reproduced, the result of which the file is created in a working directory of the OpenSCADA **core.N**, where N - number of crashed process.

Next, you need to process the resulting memory dump file in order to obtain reversal of the execution stack at the time of the accident. For this purpose you will need the debugger gdb, in which the following commands are executed for the global configuration:

```
Transfer to the working directory of OpenSCADA
(gdb) cd /var/spool/openscada
# Specifying an executable file, not scripts!
(gdb) file /usr/bin/openscada
# Specifying the dump file of the program
(gdb) core-file /var/spool/openscada/core.26658
# Getting the reversal of the stack execution
(gdb) bt
#0  0xb7d104c0 in pthread_cancel () from /lib/librt.so.1
#1  0xb7d1edaa in start_thread () from /lib/libpthread.so.0
#2  0xb7dfcf5e in clone () from /lib/libc.so.6
```