



System Architecture Support Libraries (SASL)

Copyright © 1997-2024 Ericsson AB. All Rights Reserved.
System Architecture Support Libraries (SASL) 4.2
December 5, 2024

Copyright © 1997-2024 Ericsson AB. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0> Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. Ericsson AB. All Rights Reserved..

December 5, 2024


```

7> rb:show(4).

PROGRESS REPORT <0.20.0>                                1996-10-16 16:16:36
=====
supervisor                                           {local,sasl_sup}
started
[{pid,<0.24.0>},
 {name,release_handler},
 {mfa,{release_handler,start_link,[]}},
 {restart_type,permanent},
 {shutdown,2000},
 {child_type,worker}]

ok
8> rb:show(9).

CRASH REPORT <0.24.0>                                    1996-10-16 16:16:21
=====
Crashing process
pid                                                    <0.24.0>
registered_name                                       release_handler
error_info                                           {undef,{release_handler,mbj_func,[]}}
initial_call
{gen,init_it,
 [gen_server,
 <0.20.0>,
 <0.20.0>,
 {erlang,register},
 release_handler,
 release_handler,
 [],
 []]}
ancestors                                             [sasl_sup,<0.18.0>]
messages                                             []
links                                                 [<0.23.0>,<0.20.0>]
dictionary                                           []
trap_exit                                             false
status                                                running
heap_size                                             610
stack_size                                           142
reductions                                           54

ok

```

Search Reports

All reports containing a common pattern can be shown. Suppose a process crashes because it tries to call a non-existing function `release_handler:mbj_func/1`. The reports can then be shown as follows:


```
start_function      {release_handler,start_link,[]}  
restart_type        permanent  
shutdown            2000  
child_type          worker  
  
ok
```

Stop Server

Use function `rb:stop()` to stop the `rb_server`:

```
13> rb:stop().  
ok
```

2 Reference Manual

The SASL application provides support for alarm handling, release handling, and related functions.


```
error_logger_mf_maxbytes = integer()
```

Specifies the maximum size of each individual file written by `log_mf_h`. If this parameter is undefined, the `log_mf_h` handler is not installed.

```
error_logger_mf_maxfiles = 0<integer()<256
```

Specifies the number of files used by `log_mf_h`. If this parameter is undefined, the `log_mf_h` handler is not installed.

The new `logger_disk_log_h` might be an alternative to `log_mf_h` if log rotation is desired. This does, however, write the log events in clear text and not as binaries.

See Also

```
alarm_handler(3),error_logger(3),logger(3),log_mf_h(3),rb(3),release_handler(3),  
systools(3)
```


See Also

`error_logger(3)`, `gen_event(3)`

stop()

Stops `rb_server`.

stop_log()

Closes the log file. The output from the RB tool is directed to `standard_io`.

Note:

The list of applications must contain the Kernel and STDLIB applications.

See Also

`application(3)`, `relup(4)`, `systools(3)`

