Appendix Data-driven Mutation Testing:

# GomSpace:

# LibGCSP Fault Model

## Target function

The probes will be inserted in the function *csp\_service\_handler*, contained in the file *libgscsp/lib/libcsp/src/csp\_service\_handler.c*

This function handles an incoming connection and depending on the destination port, handles the packet to the corresponding service handler. The function takes as a parameter a connection structure of type *csp\_conn\_t* and a CSP packet of type *csp\_packet\_t*.

## Target data structure:

In this section, we describe the data structures targeted for the data-driven mutation process.

### csp\_conn\_t

|  |  |
| --- | --- |
| **Member** | **Type** |
| type | csp\_conn\_type\_t |
| state | csp\_conn\_state\_t |
| idin | csp\_id\_t (union) |
| idout | csp\_id\_t (union) |
| rx\_event | csp\_queue\_handle\_t |
| rx\_queue | csp\_queue\_handle\_t |
| socket | csp\_queue\_handle\_t |
| timestamp | uint32\_t |
| opts | uint32\_t |
| rdp | csp\_rdp\_t |

We are interested in mutating *idin* field, which contain the header of the CSP packet on the incoming connection.

### csp\_id\_t

The packet header is represented by the data structure *csp\_id\_t* and contains the following fields.

|  |  |  |
| --- | --- | --- |
| **Member** | **Type** | **Description** |
| ext | uint32\_t |  |
| pri | unsigned int | Priority |
| src | unsigned int | Source |
| dst | unsigned int | Destination |
| dport | unsigned int | Destination port |
| sport | unsigned int | Source port |
| flags | unsigned int | Flags |

## Identifier Fault Model

The probe will be inserted before the switch statement, before the invocation of the service handlers.

|  |  |  |
| --- | --- | --- |
| **Member** | **Type** | **Fault Classes** |
| ext | uint32\_t | None |
| pri | unsigned int | INV(Min=0; Max=3; D=0), VAT(Threshold=3; Delta=1),  BF(Min=0; Max=0; State=1), BF(Min=1; Max=1; State=1), |
| src | unsigned int | INV(Min=0; Max=31; D=0), VAT(Threshold=31;D=1) |
| dst | unsigned int | INV(Min=0; Max=31; D=0), VAT(Threshold=31;D=1) |
| dport | unsigned int | INV(Min=8; Max=30; D=0), VOR(Min=8; Max=30;D=1), SS(Delta=1), SS(Delta=-1) |
| sport | unsigned int | INV(Min=8; Max=30; D=0), VOR(Min=8; Max=30;D=1), SS(Delta=1), SS(Delta=-1) |
| flags | unsigned int | BF(Min=0; Max=0; State=1), BF(Min=1; Max=1; State=1), BF(Min=2; Max=2; State=1), BF(Min=3; Max=3; State=1) |

One of the objectives of the fault model is to mutate the *dport* field from the identifier; a different value of this field changes the service required. In the following, a table with the full list of services is presented.

|  |  |  |
| --- | --- | --- |
| **Service** | **Code** | **Description** |
| GS\_CSP\_CMP | 0 | CSP Management Protocol |
| GS\_CSP\_PING | 1 | CSP Ping |
| GS\_CSP\_PS | 2 | Process status |
| GS\_CSP\_MEM\_FREE | 3 | Show memory free |
| GS\_CSP\_REBOOT | 4 | Reboot/reset request |
| GS\_CSP\_BUF\_FREE | 5 | Number of free CSP buffers |
| GS\_CSP\_UPTIME | 6 | Show uptime |
| GS\_CSP\_PORT\_RPARAM | 7 | Parameter service (libparam) |
| GS\_CSP\_PORT\_FTP | 9 | File Transfer Service (libftp) |
| GS\_CSP\_PORT\_RLOG | 11 | Remote log service (liblog) |
| GS\_CSP\_PORT\_RGOSH | 12 | Remote GOSH service (librgosh) |
| GS\_CSP\_PORT\_AIS | 13 | AIS command port (libais) |
| GS\_CSP\_PORT\_ADSB | 14 | ADS-B command port (libadsb) |
| GS\_CSP\_PORT\_GSSB | 16 | GomSpace Sensor Bus (libgssb) |
| GS\_CSP\_PORT\_FP | 18 | Flight Planner (libfp) |
| GS\_CSP\_PORT\_ADCS | 20 | ADCS (libadcs) |
| GS\_CSP\_PORT\_HK | 21 | House Keeping (libhk) |
| GS\_CSP\_PORT\_GSCRIPT | 22 | G(omSpace) script service (libgosh) |
| GS\_CSP\_PORT\_MONITOR | 26 | Gomspace Monitor application command port |
| GS\_CSP\_PORT\_REMOTE\_SHELL | 27 | Remote shell (libgosh) |
| GS\_CSP\_PORT\_HK\_BEACON | 30 | Housekeeping beacon port (libhk) |