DTrace/SystemTap SDT Probes in OpenAFS

Andrew Deason

June 2019

OpenAFS Workshop 2019

Userspace DTrace Background

- Specify triggers on events (function calls)

- No special support needed

- Complex functionality builtin
  - threading issues
  - efficient aggregation

- No need to setup logging, auditing, etc

Generic Examples

Prints every time a specific function is called:

```
$ cat example.d
#!/sbin/dtrace -qs
pid$1::some_function:entry {
    printf("some_function(%d) called\n", arg0);
}

$ ./example.d `pgrep program`
```

Generic Examples

Prints the sum instead:

```
$ cat example.d
#!/sbin/dtrace -qs
pid$1::some_function:entry {
    @total = sum(arg0);
}

$ ./example.d `pgrep program`
```

Primitive OpenAFS Examples

```
$ cat example.d
pid$1::SVL_GetEntryByID*:entry {
    printf("Somebody looked up volume ID %d\n", arg1);
}

$ ./example.d `pgrep vlserver`
```

Primitive OpenAFS Examples

```
$ cat example.d
pid$1::SVL_GetEntryByName*:entry {
    printf("Somebody looked up volume name %s\n",
           copyinstr(arg1));
}

$ ./example.d `pgrep vlserver`
```

~~Gross~~ Complex OpenAFS Example

```
$ cat example.d
pid$1::SVL_GetEntryByName*:entry {
    this->conn = copyin(arg0 + 0x60,
                        sizeof(uintptr_t));
    this->peer = copyin(this->conn + 0x8,
                        sizeof(uintptr_t));
    this->host = copyin(this->peer + 0x8,
                        sizeof(int));
    printf("Host %s looked up volume name %s\n",
           inet_ntoa(this->host), copyinstr(arg1));
}

$ ./example.d `pgrep vlserver`
```

Complex OpenAFS Example

- Low-level
  - Must know function names
  - Manual structure unpacking
  - Hard-coded memory offsets

- Automatic debug type inference

- Need a higher-level interface

Better OpenAFS Example

```
$ cat example.d
pid$1:::GetEntry-start {
    printf("IP %s looked up volname %s\n",
           args[1]->ci_remote, args[3]->volname);
}
pid$1:::GetEntry-done {
    printf("Request from %s, status: %s\n",
           args[1]->ci_remote, afs_errorstr[args[0]]);
}

$ ./example.d `pgrep vlserver`
```

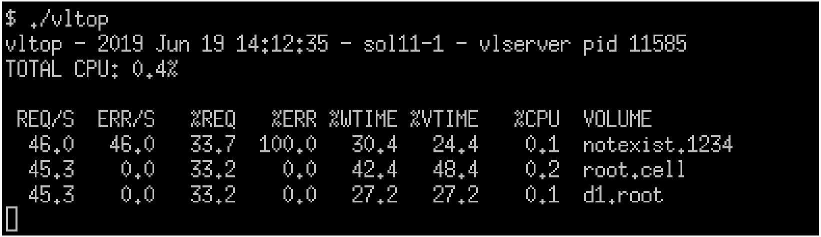
SDT Probes

- Requires OpenAFS code changes

- `/usr/lib/dtrace/openafs.d` translates into symbolic info:

```
translator conninfo_t < struct rxcall_info_int *rx > {
    ci_remote = *COPYIN1(rx->ip_raddr_n,
                         uint32_t) == 0 ? "<unknown>" :
                inet_ntoa(COPYIN1(rx->ip_raddr_n,
                                  ipaddr_t));
};
```

vltop



```
$ ./vltop
vltop - 2019 Jun 19 14:12:35 - sol11-1 - vlserver pid 11585
TOTAL CPU: 0.4%

 REQ/S  ERR/S   %REQ   %ERR %WTIME %VTIME   %CPU  VOLUME
  46.0   46.0   33.7  100.0   30.4   24.4    0.1  notexist.1234
  45.3    0.0   33.2    0.0   42.4   48.4    0.2  root.cell
  45.3    0.0   33.2    0.0   27.2   27.2    0.1  d1.root
```

Future

- More subsystems (Rx, fileserver)

- SystemTap:

```
probe process("vlserver").provider("openafs")
.mark("GetEntry-start") {
    printf("IP %s looked up volname %s\n",
            $arg1, $arg2);
}
```

Code

Top Commit

https://gerrit.openafs.org/13386

All Commits

https://gerrit.openafs.org/#/q/topic:dtrace-usdt-probes

Slides

http://dson.org/talks

?