

llnode Status Update

@joyeecheung



Overview of llnode

- Node.js (or V8, more precisely) plugin of lldb
- Inspect JavaScript stacks and objects in a V8 process
 - running or terminated (core dump)
 - Users can debug C++ and JavaScript at the same time
- Scan the V8 heap for JavaScript types, instances and their references, report stats about them
- Reverse-engineering the V8 object model with post-mortem metadata: symbols prefixed with `v8dbg_`

Recent work

- Link with lldb reported by [llvm-config](#) on MacOS: no need to use the XCode one, can work with custom builds or lldb installed with Homebrew
- Loading symbols with shared libraries: can work with libv8 + d8 built with post-mortem support

Recent work

- Fixed tests and Travis integration
 - Flake due to neutered array buffers:
<https://github.com/nodejs/llnode/issues/150>
- `LLNODE_DEBUG=true` to print details of failures during symbol loading, turned on in Travis

nodejs / llnode



build **passing**

Current

Branches

Build History

Pull Requests

More options



✓ **Pull Request #167** test: move core saving and loading into common.js

🔗 #364 passed

🔗 Commit cb321ae

🕒 Ran for 1 min 13 sec

🔗 #167: test: move core saving and loading into common.js

🕒 Total time 4 min 14 sec

🔗 Branch master

📅 a day ago

👤 Joyee Cheung authored and committed

Build Jobs

✓ # 364.1



</> Node.js: 4



no environment variables set



1 min 8 sec

✓ # 364.2



</> Node.js: 6



no environment variables set



1 min 8 sec

✓ # 364.3



</> Node.js: 8



no environment variables set



59 sec

✓ # 364.4



</> Node.js: 9



no environment variables set



59 sec

JS API (C++ addon installed with npm)

- Original work by @rnchamberlain, rewrote with NAN and some API changes, e.g. use iterators for instance lists to save memory.
- Working now, has tests, instance representations are strings at the moment

```
require('llnode')
```

```
.fromCoredump('path/to/core', 'path/to/executable')
```

```
.getHeapTypes()[0].instances // iterator to list of instances
```

JS API (C++ addon installed with npm)

- Looking for feedback on the API design:
<https://github.com/joyeecheung/llnode/blob/js-api/JSAPI.md>
- Need reviews: <https://github.com/nodejs/llnode/pull/147>

Update to work with V8 6.4

- Metadata changes, probably some layout changes too:
<https://github.com/nodejs/llnode/issues/158>
- Backtrace & Map fixes:
<https://github.com/nodejs/llnode/pull/168>
- Object inspection is broken ATM due to metadata & descriptor changes, fixes in progress
- Currently we only have tests with Node.js v4, v6, v8 and v9 using Travis, need build integration with Node.js master/PRs

Node.js Event Loop Inspection

- Upstream done by @mmarchini
<https://github.com/nodejs/node/pull/14901>
 - adding `nodedbg_*` metadata for tools to restore internal structures of Node.js like the `v8dbg_` metadata from V8
- llnode PR: <https://github.com/nodejs/llnode/pull/122>
- Documentation:
<https://github.com/nodejs/node/blob/master/doc/guides/node-postmortem-support.md>

Other Plans

- Work with more types: Errors, Symbols, etc.
- Better support for TypedArray inspection
- Support printing UTF8 strings in the lldb prompt