(doc to move to nodejs/ctc repository after meeting)

Node Foundation CTC Meeting 2017-03-22

Links

* **Audio recording**: <>
* **GitHub Issue**: [CTC#82](https://github.com/nodejs/CTC/issues/82)
* **Minutes Google Doc**:
<https://docs.google.com/document/d/1nIXl2UMQs1n-yPencn53MYlfu3RnUltX3MfuaStPzx8>
* _Previous Minutes Google Doc:
<https://docs.google.com/document/d/1ZbzN6qO9AqmGjOMR-wfWTmdeE_69rr1hcfhk2oTUGpA>_

Present

* Anna Henningsen @addaleax (CTC)
* Ben Noordhuis @bnoordhuis (CTC)
* Сковорода Никита Андреевич @ChALkeR (CTC)
* James M Snell @jasnell (CTC)
* Josh Gavant @joshgav (observer/TSC)
* Brian White @mscdex (CTC)
* Myles Borins @MylesBorins (CTC)
* Rod Vagg @rvagg (CTC)
* Sakthipriyan Vairamani @thefourtheye (CTC)
* Trevor Norris @trevnorris (CTC)
* Rich Trott @Trott (CTC)
* Arunesh Chandra (observer/Microsoft)
* Hitesh Kanwathirtha (observer/Microsoft)

Standup

* Anna Henningsen @addaleax (CTC)
  * not much besides Issue/PR review
* Ben Noordhuis @bnoordhuis (CTC)
  * the usual

* Сковорода Никита Андреевич @ChALkeR (CTC)
  * nothing significant
* James M Snell @jasnell (CTC)
  * Mostly nearForm onboarding
  * PR landing
  * Buffer PR splitting
* Josh Gavant @joshgav (observer/TSC)
  * diag meeting Thursday
  * typical issue triage
* Brian White @mscdex (CTC)
  * Continued work on various optimizations in core
  * Reviewed PRs, commented on issues
* Myles Borins @MylesBorins (CTC)
  * Released v4.8.1 && v6.10.1
  * 100th release of citgm
  * attending TC39 this week
* Rod Vagg @rvagg (CTC)
  * Minor infra maintenance
* Sakthipriyan Vairamani @thefourtheye (CTC)
  * Not much
* Trevor Norris @trevnorris (CTC)
  * Submitted AsyncHooks PR
* Rich Trott @Trott (CTC)
  * Fix some flaky tests
  * Increase test coverage
  * Helped some first-time contributors via NodeTodo

Agenda

* buffer: discuss future direction of Buffer constructor API
[node#9531](https://github.com/nodejs/node/issues/9531)
* module: add support for abi stable module API
[node#11975](https://github.com/nodejs/node/issues/11975)

Previous Meeting Review

* Symbol key props visible in inspection by default
[node#9726](https://github.com/nodejs/node/pull/9726)
  * Landed!
* Updating the Copyright
[TSC#174](https://github.com/nodejs/TSC/issues/174)
  * Moved to ctc-review, no new updates.

* TC39 + Node.js Summit / Dinner
[CTC#67](https://github.com/nodejs/CTC/issues/67)
    * Awareness.

Minutes

buffer: discuss future direction of Buffer constructor API
[node#9531](https://github.com/nodejs/node/issues/9531)

* should we ever schedule a runtime deprecation and when for `new Buffer()`.
* should we precede that with a `--pending-deprecation` flag
* use zero-filling or random-filling. Why random-fill? So that people wouldn't realize buffers were being zero-filled and rely on that behavior. @trevnorris: but random filling doesn't fit pattern of UInt8Array.

@rvagg: Can we deprecate just one signature of the constructor, i.e. the one that takes an int?

@trevnorris: Once we bump a minor version, we don't support previous minor versions, so why are we concerned with those using older minors?

@bnoordhuis: Those who depend on releases from Linux distros don't get minor updates for a while.

@mylesborins: That's true of any security patches, if you use unpatched earlier versions you don't have the patch…

@nikita: Later modules may come to rely on zero-filled buffers, and those modules may be used with out-of-date earlier versions.

@rvagg: This is a massive ship to be turning, we should opt for greater patience, encourage people through docs and other channels.

@jasnell: This is one of the reasons for --pending-deprecation. Also other tools and linters are warning more.

@nikita: I can support optin deprecate and plan for later actual
deprecation.

Two PRs from @jasnell:
* Instead of zero-filling buffers, fill with random byte.
* Introduce --pending-deprecation runtime flag and use for buffer
deprecation.

@jasnell: If we will land with filling it should be decided before
8.0 RCs (4/4).

@rvagg: May not be semver-major, especially if random-fill.

Next steps

* Discuss within issues in GitHub.

---

module: add support for abi stable module API
[node#11975](https://github.com/nodejs/node/issues/11975)

Export an API and ABI stable C and C++ interface for native module
developers.

@arunesh: Experimental, guarded by runtime flag. Goal is more
feedback from community and module authors. V8 APIs used in 5+ of top
30 native modules are covered by a N-API equivalent.

@jasnell: Goal is to get into 8.0.0 as experimental.

@trevnorris: Converting a lot to void*, how does it ensure the safety
of what's being passed?

@digitialinfinity: Type checking is done within implementation. (?)

@jasnell: Does anyone object to this going into 8.0.0?

@bnoordhuis: Pretty big PR, will be hard to review by 4/4.

@jasnell: set a target a week before final 8.0.0 release? It's not
semver-major.

@rvagg: Can NAN be backed by this API?
@digitalinfinity: C++ header-only wrapper similar to NAN. Some
discussion of considering that NAN 3.0. Haven't considered backing
NAN, worth considering. Hope to provide tools to aid migration after
some adoption and feedback.

---

async_hooks initial implementation
[#11883](https://github.com/nodejs/node/pull/11883)

@trevnorris: rewrote more cleanly, would like more feedback. Tried to
be as least invasive as possible. Comes with a whole bunch of tests.

There are async_hooks in the async_hooks tests to check for stuck
previous tasks.

Touches many places in Node, hence want a lot of review.

@mylesborins: thoughts about using PromiseHooks from V8? @trevnorris:
won't be in V8 in 8.0.0, but as soon as it is I'll integrate.

@rvagg: not a semver-major, and anyway still experimental, but still
would be nice to have in 8.

@trevnorris: need to add docs, move around some tests.

Q&A, Other

Upcoming Meetings

* **Node.js Foundation Calendar**:
https://calendar.google.com/calendar/embed?src=kap.co_i17i575te0aes6k
aanfjr2e4hs%40group.calendar.google.com

Click `+GoogleCalendar` at the bottom right to add to your own Google calendar.

* **Collaborators Summit**: May 4-5,
  [summit#39](https://github.com/nodejs/summit/issues/39)
* **TC39 Meetup**: May 22,
  [CTC#67](https://github.com/nodejs/CTC/issues/67)

* **CTC**: March 29, 1600 UTC
* **TSC**: March 23, 2000 UTC
* **Diag**: March 23, 2100 UTC