Arvados - Bug #14911
[dispatch-cloud] Azure driver panic at startup
03/04/2019 08:33 PM - Tom Clegg

Status: Resolved
Priority: Normal
Assigned To: Tom Clegg
Category: Estimated time: 0.00 hour

Target version: 2019-03-13 Sprint

Description:

arvados-dispatch-cloud 1.3.1.20190301145524-1 seems unable to start on c97qk. Systemd unit is in a restart loop.

```
arvados-dispatch-cloud.service: Service hold-off time over, scheduling restart.
Stopped arvados-dispatch-cloud.
Starting arvados-dispatch-cloud...
panic: nil context
goroutine 1 [running]:
net/http.(*Request).WithContext(0xc4201ca820, 0x0, 0x0, 0x9e7440)
    /usr/local/go/src/net/http/request.go:331 +0x1ac
```
Subtasks:

Task #14913: Review 14911-azure-panic

Resolved

Associated revisions

Revision b2086247 - 03/05/2019 04:39 PM - Tom Clegg
Merge branch '14911-azure-panic'

fixes #14911

Arvados-DCO-1.1-Signed-off-by: Tom Clegg <tclegg@veritasgenetics.com>

History

#1 - 03/04/2019 08:58 PM - Tom Clegg

Looks like #14844 moved an Azure API call into the driver initialization func which, as a result, uses az.ctx before it's assigned.

With az.ctx initialization moved up, dispatcher starts up successfully on c97qk.

14911-azure-panic @ f1d38ded6991f7061e2e4d9f61db3501c9c90b2b

#2 - 03/04/2019 08:58 PM - Tom Clegg

- Status changed from New to In Progress

#3 - 03/05/2019 03:41 PM - Eric Biagiotti

Tom Clegg wrote:

Looks like #14844 moved an Azure API call into the driver initialization func which, as a result, uses az.ctx before it's assigned.

With az.ctx initialization moved up, dispatcher starts up successfully on c97qk.

14911-azure-panic @ f1d38ded6991f7061e2e4d9f61db3501c9c90b2b

LGTM, but while figuring out what context does, I came across the following on https://golang.org/pkg/context/

Do not store Contexts inside a struct type; instead, pass a Context explicitly to each function that needs it. The Context should be the first parameter, typically named ctx ...

Obviously there are exceptions to recommendations like this and I haven't figured out exactly how we use context, but I just wanted to point this out.

#4 - 03/05/2019 04:38 PM - Tom Clegg

Eric Biagiotti wrote:

    LGTM, but while figuring out what context does, I came across the following on https://golang.org/pkg/context/

    Do not store Contexts inside a struct type; instead, pass a Context explicitly to each function that needs it. The Context should be the first parameter, typically named ctx ...

    Obviously there are exceptions to recommendations like this and I haven't figured out exactly how we use context, but I just wanted to point this out.

Yes -- stdlib breaks this rule in http.Request, but adding a context to a struct is surely a flag that you might be doing it wrong. Here, arguably the driver should accept a context in List/Create/Destroy calls, leaving the caller responsible for cancelling when it wants to shut everything down.

#5 - 03/05/2019 07:01 PM - Tom Clegg

- Status changed from In Progress to Resolved
Applied in changeset arvadosjb20862479a05afec430f10067b8b417418b11eee.

#6 - 05/21/2019 10:27 PM - Tom Morris
- Release set to 15