Arvados - Bug #14745

[crunch-dispatch-cloud] Azure cloud driver fixups

01/16/2019 04:12 PM - Tom Clegg

Status:	Resolved	**Start date:**	02/13/2019
Priority:	Normal	**Due date:**	
Assigned To:	Eric Biagiotti	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	2019-02-27 Sprint		

Description

Some items to address that could get more difficult if we wait:

- Change config keys to CamelCase to match other parts of Cluster configuration
- Move Azure code from lib/cloud to its own package, lib/cloud/azure
- Un-export identifiers -- the only public interface needed is var Driver = cloud.DriverFunc(newAzureInstanceSet)

Subtasks:

Task # 14752: Review 14745-azure-cloud-driver-fixups	**Resolved**

Related issues:

Related to Arvados - Story #13908: [Epic] Replace SLURM for cloud job schedul...	**Resolved**

Associated revisions

Revision a4fa0405 - 02/14/2019 07:56 PM - Eric Biagiotti

Merge branch '14745-azure-cloud-driver-fixups'

refs #14745

Arvados-DCO-1.1-Signed-off-by: Eric Biagiotti <eric.biagiotti@gmail.com>

History

#1 - 01/16/2019 04:27 PM - Tom Morris

- Target version changed from Arvados Future Sprints to 2019-01-30 Sprint

#2 - 01/16/2019 04:40 PM - Tom Morris

- Assigned To set to Eric Biagiotti

#3 - 01/28/2019 07:53 PM - Eric Biagiotti

- Status changed from New to In Progress

#4 - 01/30/2019 04:21 PM - Eric Biagiotti

- Target version changed from 2019-01-30 Sprint to 2019-02-13 Sprint

#5 - 01/30/2019 04:24 PM - Eric Biagiotti

- Story points set to 3.0

#6 - 02/08/2019 01:55 PM - Eric Biagiotti

We had discussed removing "image" from the config. Looking at the live example in the test (azure_test.go lines 113-121), it is expecting that value in the config, which is used in TestCreate to call AzureInstanceSet.Create. I imagine this code will change, just wondering if I should break the live test for now or if I should change anything in AzureInstanceSet.Create (i.e. the CreatOption parameter in azure.go line 383).

#7 - 02/08/2019 06:35 PM - Tom Clegg

It looks like that test case gets the image ID from the map[string]interface{} and won't be affected by removing the image field from the normal config struct.

Some options for making it less confusing:

- add comment in example azconfig.yml ("VM image to use in test suite")
- rename it to something like ImageIDForTestSuite

- rearrange azconfig.yml so the image ID is outside the driver config like it is in real life:

  ```
  ImageID: https://....
  DriverParameters:
    SubscriptionID: ...
    Region: ...
  ```

#8 - 02/12/2019 03:10 PM - Eric Biagiotti

A few more questions before review:

- Added a decode hook for arvados.Duration so that mapstructure can decode it appropriately. Right now it is in the duration.go file but not a part of the Duration struct, since it doesn't need a Duration object (it would be a static function in my c++ way of thinking). Not sure if this is the right place for it. As of right now it sits in the arvados package, but I could see it going in the config package or maybe some place else.
- Anything regarding packaging that needs to be updated? I'm assuming this isn't currently getting packaged, but don't know if we have something in the works that should be updated.
- Should I run the live tests? Where do I get the config info?

#9 - 02/12/2019 03:31 PM - Tom Clegg

Eric Biagiotti wrote:

> - Added a decode hook for arvados.Duration so that mapstructure can decode it appropriately. Right now it is in the duration.go file but not a part of the Duration struct, since it doesn't need a Duration object (it would be a static function in my c++ way of thinking). Not sure if this is the right place for it. As of right now it sits in the arvados package, but I could see it going in the config package or maybe some place else.

Darn, too bad we don't have the option of using something like the json.Unmarshaler interface.

Yes, I think arvados.WhateverDecodeHookFunc() is the right place. (arvados.StringOrNumberToDurationDecodeHookFunc()?) I think if we test int(1), float64(1.23), json.Number(1.23), and "1.23s", then it should work with Go literals like {"foo":1} as well as anything coming out of the JSON decoder.

> - Anything regarding packaging that needs to be updated? I'm assuming this isn't currently getting packaged, but don't know if we have something in the works that should be updated.

No packaging updates needed -- the dispatcher binary gets built from lib/dispatchcloud, and it looks like you've already updated that.

#10 - 02/12/2019 03:40 PM - Tom Clegg

- Related to Story #13908: [Epic] Replace SLURM for cloud job scheduling/dispatching added

#11 - 02/13/2019 01:26 PM - Eric Biagiotti

- Target version changed from 2019-02-13 Sprint to 2019-02-27 Sprint

#12 - 02/13/2019 08:21 PM - Eric Biagiotti

Ran the live tests successfully.

Latest: ff241c64
Tests: https://ci.curoverse.com/view/Developer/job/developer-run-tests/1066/

Also, you mentioned writing a helper function for dispatchcloud tests / writing tests for json decoding. Would you like this to be a part of this issue? Should I make a new story? Depending on your answer, this is ready for review. Thanks!

#13 - 02/13/2019 09:03 PM - Tom Clegg

Eric Biagiotti wrote:

> Also, you mentioned writing a helper function for dispatchcloud tests / writing tests for json decoding. Would you like this to be a part of this issue? Should I make a new story? Depending on your answer, this is ready for review. Thanks!

No, I was just thinking of test suites with driver parameters that now need to be JSON-encoded before passing to (cloud.Driver)InstanceSet(). It seems there aren't any, so there's nothing to do.

That said, we might as well preserve the "load params into struct" part of stub_driver.go, instead of ignoring params -- instead of this:

```
-        return &sis, mapstructure.Decode(params, &sis)
+        return &sis, nil
```

...maybe this:

```
-       return &sis, mapstructure.Decode(params, &sis)
+       return &sis, json.Unmarshal(params, &sis)
```

Am I missing something, or is this code in azure_test.go superfluous? It looks like azcfg never gets used, and if there's an Unmarshal() error, it would be caught by newAzureInstanceSet() on the following line.

```
                var azcfg azureInstanceSetConfig
                err = json.Unmarshal(exampleCfg.DriverParameters, &azcfg)
                if err != nil {
                        return nil, cloud.ImageID(exampleCfg.ImageIDForTestSuite), cluster, err
                }
```

Nit: This comment on Set() was obviously copied from Value, but Set is also part of the flag.Value interface, which might still be worth mentioning.

```
-// Value implements flag.Value
+// Set sets the current duration by parsing the string using time.ParseDuration
```

Doc comments should be full sentences (some are missing final periods).

Can delete lib/cloud/gocheck_test.go now that there are no tests here.

#14 - 02/13/2019 10:43 PM - Eric Biagiotti

Tom Clegg wrote:

> Eric Biagiotti wrote:
>
> > Also, you mentioned writing a helper function for dispatchcloud tests / writing tests for json decoding. Would you like this to be a part of this issue? Should I make a new story? Depending on your answer, this is ready for review. Thanks!
>
> No, I was just thinking of test suites with driver parameters that now need to be JSON-encoded before passing to (cloud.Driver)InstanceSet(). It seems there aren't any, so there's nothing to do.
>
> That said, we might as well preserve the "load params into struct" part of stub_driver.go, instead of ignoring params -- instead of this:
>
> [...]
>
> ...maybe this:
>
> [...]

Good call, fixed in: fba0cd4c

> Am I missing something, or is this code in azure_test.go superfluous? It looks like azcfg never gets used, and if there's an Unmarshal() error, it would be caught by newAzureInstanceSet() on the following line.
>
> [...]
>
> Nit: This comment on Set() was obviously copied from Value, but Set is also part of the flag.Value interface, which might still be worth mentioning.
> [...]
>
> Doc comments should be full sentences (some are missing final periods).
>
> Can delete lib/cloud/gocheck_test.go now that there are no tests here.

Haha, that was all debug code that I ended up forgetting about then fixing up as if it was needed :(This and remaining comments fixed in: 8be52d22

I also improved another comment in b7f7154a

Tests: https://ci.curoverse.com/view/Developer/job/developer-run-tests/1069/

#15 - 02/14/2019 03:05 PM - Tom Clegg

LGTM @ b7f7154a7, thanks

#16 - 02/14/2019 07:13 PM - Eric Biagiotti

Fixed dispatchcloud/worker/PoolSuite tests in c9127a21 and 1bd522fd. StubDriver.InstanceSet now allows for nil params. Seems like a better solution than forcing tests to supply a json.RawMessage that don't need to.

#17 - 02/14/2019 07:35 PM - Tom Clegg

Agreed. LGTM, thanks

#18 - 02/14/2019 08:26 PM - Eric Biagiotti

- Status changed from In Progress to Resolved

- Assigned To deleted (Eric Biagiotti)

#19 - 02/15/2019 03:27 PM - Eric Biagiotti

- Assigned To set to Eric Biagiotti

#20 - 03/01/2019 07:33 PM - Tom Morris

- Release set to 15