Arvados - Story #8833

[Keep] Azure SDK update to add metadata to ListBlobs call

03/30/2016 02:21 PM - Radhika Chippada

Status:	Resolved	**Start date:**	04/28/2016
Priority:	Normal	**Due date:**	
Assigned To:	Radhika Chippada	**% Done:**	100%
Category:	Keep	**Estimated time:**	0.00 hour
Target version:	2016-05-11 sprint		

Description

Subtasks:

Task # 9072: Review branch list-blob-metadata	**Resolved**

Related issues:

Blocked by Arvados - Story #8832: [Keep] Azure SDK requirements analysis - up...	**Resolved**	
Blocks Arvados - Feature #8556: [Keep] Implement trash/untrash behavior in az...	**Resolved**	**05/05/2016**

History

#1 - 04/27/2016 12:11 PM - Radhika Chippada

- Category set to Keep

- Assigned To set to Radhika Chippada

- Target version changed from Arvados Future Sprints to 2016-04-27 sprint

#2 - 04/27/2016 12:12 PM - Radhika Chippada

- Status changed from New to In Progress

#3 - 04/27/2016 04:46 PM - Brett Smith

- Target version changed from 2016-04-27 sprint to Arvados Future Sprints

#4 - 04/27/2016 07:12 PM - Tom Clegg

- Target version changed from Arvados Future Sprints to 2016-05-11 sprint

#5 - 04/27/2016 07:14 PM - Tom Clegg

- Subject changed from [Keep] Azure SDK update to add metadata to ListBlobs call to [Keep] Azure SDK update to add metadata to ListBlobs call

- Story points set to 0.5

#6 - 04/28/2016 02:21 PM - Tom Clegg

In queue.go, the metadata looks like this

```
type QueueMetadataResponse struct {
    ApproximateMessageCount int
    UserDefinedMetadata     map[string]string
}
```

Can we make the blob metadata work the same way, something like this...?

```
type Blob struct {
    Name       string            `xml:"Name"`
    Properties BlobProperties    `xml:"Properties"`
    Metadata   map[string]string `xml:"Metadata"`
}
```

This way callers could say resp.Blobs[0].Metadata["Foo"] instead of iterating over a slice and comparing keys.

#7 - 04/28/2016 06:46 PM - Tom Clegg

I couldn't find a way to make encoding/xml do this by itself, but if we make a BlobMetadata type that implements xml.Unmarshaler, we can convert it to a map while decoding, so blobs[i].Metadata["Foo"] works out of the box:

https://play.golang.org/p/P1xi66oeJw

#8 - 04/28/2016 09:25 PM - Radhika Chippada

Tom, thanks for the update. It's more appropriate implementation. I updated the test accordingly.

#9 - 05/02/2016 08:20 PM - Tom Clegg

This looks better, thanks.

IMO the test case is verbose, making it less obvious what it's doing. How about something like this:

```
var expectMeta map[string]BlobMetadata

for i := 0; i < 5; i++ {
    name := randString(20)
    c.Assert(cli.putSingleBlockBlob(cnt, name, []byte("Hello, world!")), chk.IsNil)
    c.Assert(cli.SetBlobMetadata(cnt, name, map[string]string{
        "foo": name,
        "bar_BAZ": "waz qux",
    }), chk.IsNil)
    expectMeta[name] = BlobMetadata{
        "Foo": name,
        "Bar_baz": "waz qux",
    }
}

// ...call ListBlobs and copy results into respBlobs...

for name, metadata := range expectMeta {
    c.Check(respBlobs[name].Metadata, chk.DeepEquals, expectMeta[name])
}
```

I'd also say skip the cli.GetBlobMetadata() call and the associated checks. That stuff is already tested by TestGetAndSetMetadata().

(sort.Strings(blobs) seems superfluous too)

Might be worth writing one blob without metadata, if only for the sake of exercising that case in the decoder.

```
c.Check(0, chk.Equals, len(respBlobs[nameWithoutMetadata].Metadata))
```

#10 - 05/02/2016 09:50 PM - Radhika Chippada

Tom:

- Updated the test as suggested and added a blob with no metadata
- While testing I realized that the metadata returned by ListBlobs should match the same behavior as GetBlobMetadata, where the keys are in lowercase. Made this update in blob.go as well.

Thanks.

#11 - 05/03/2016 03:38 PM - Tom Clegg

golint: blob_test.go:552:12: should omit 2nd value from range; this loop is equivalent to `for name := range ...`

With that fixed, LGTM @ commit:c922124. Should probably squash this into one commit before making a PR, to make it easier for upstream to review...

#12 - 05/04/2016 08:27 PM - Radhika Chippada

- Status changed from In Progress to Resolved