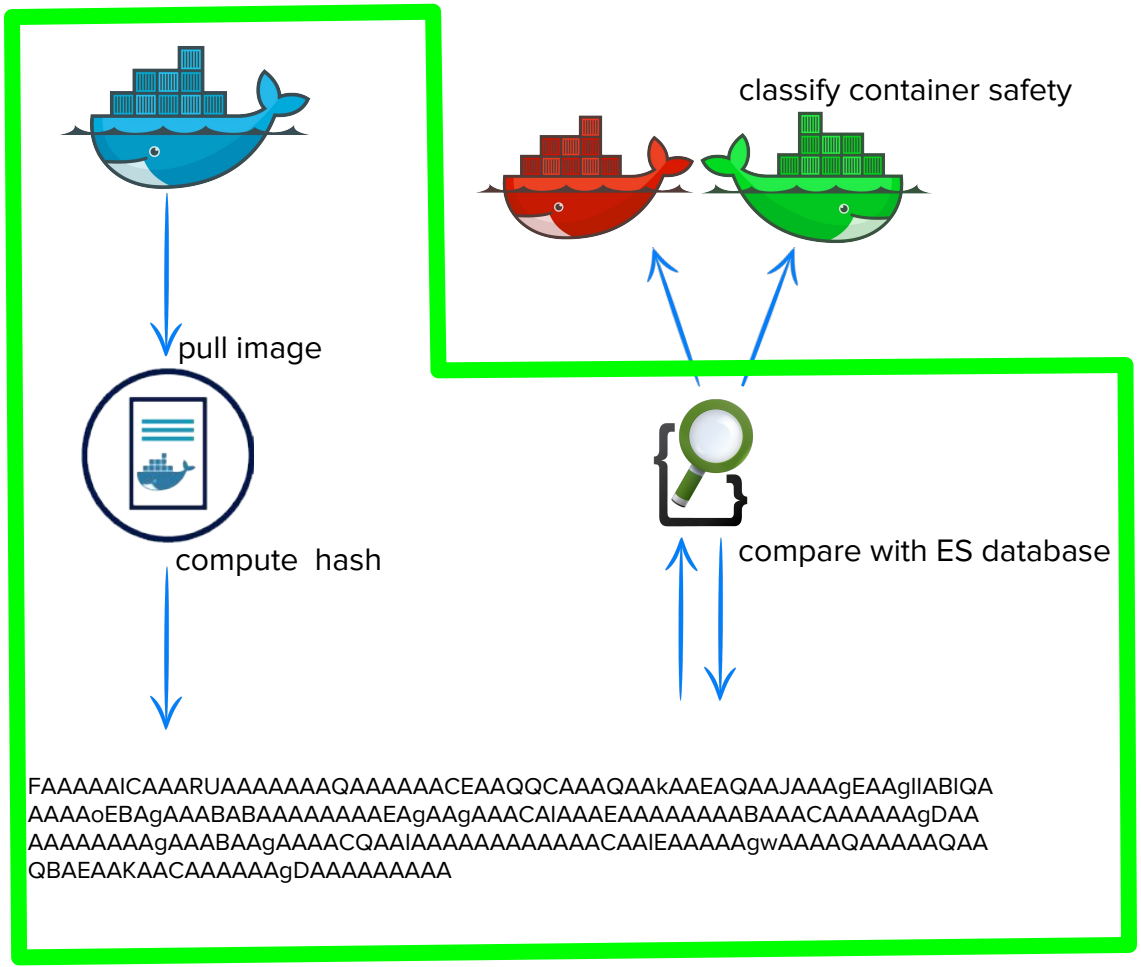
Container Code Classification

Gao, Jeremy, Kostas, Ozan, Rahul

Mentor: Sastry S Duri
(IBM Research)

Procedure:

1. Retrieve
2. Hash
3. Compare
4. Classify



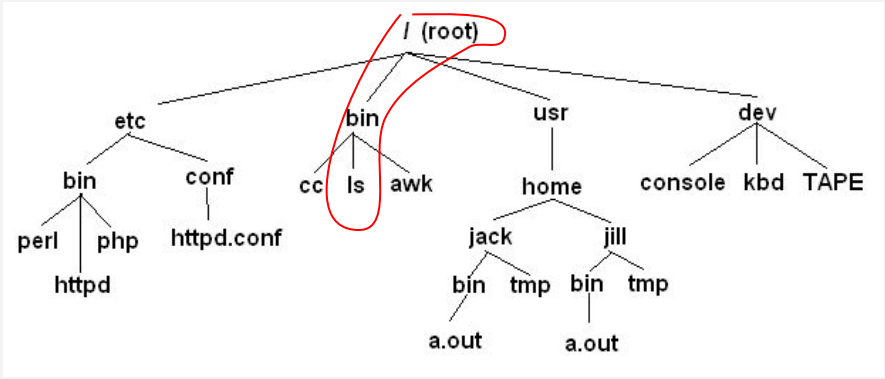
What we've done:

So Far (Sprint 2):

- configured endpoints on registry for notifications
- integrated image pull and extract on receipt of notification
- indexing of reference image to elasticsearch
- comparing custom image with reference image using elasticsearch and storing results

How are we detecting suspicious files?

- Check in Elasticsearch!
  - keys are paths
  - values are hashes



{ “/bin/ls” : “EIAAAQAQAAAEEgAAAAAQKABAIAAAAAAAAkAAAA
CAAAAACAAEAAAACAAAIgAEAgACAAAAYAAAAAA
AIAgQAAAAQCACQAAAIABAYCAAAIAAAAAAIAAAIAB “ }

What does Suspicious translate to?

- File not found (new file) → SUSPICIOUS
- Same file, different path → SUSPICIOUS
- Same file, Same path
  - sdHashes of the files NOT similar → SUSPICIOUS

```
sdbf:03:9:file1.txt:14030:sha1:256:5:7ff:160:2:77:CIEVAjQGU
EAHAASiSBZowLB3B4FAZDQAADKGoFQNHllAAlkDqmAA8AQOMJjIHQGSTAnA
cmBCtDQDtJIwCoGInUkSDybgowgdEwQHFllBiIBYMyjKFYspoDzsAlxrFEL
iiLogDUQAQOESKiQAIBAAyCgQGaDBKAIO+LgAYE5DA5mQmmZgwAgiBAEmA9
AAMiLBAU0g0QoAiGrIg7oSJtGOboBIQEEJAAwCQ4ll2ENeAUgmAOyU4ICzq
ECVhowoxVUwJ2CAAXBkDqNyiiAoQA/E8FlrYQF4QRAAQQi2lqDEtwSLKEAZ
ITuzD6gooJIr+KD2QSQIBzSFkasBAqQACz4AERqggAAAEIAlAUNoGCIAACT
AAAICCGWHIICwAQkgAkAEBBAQBAFEAgQAgAAAAARABEGAMgBYiAAYAGAQAK
IAKFAQCBCwIgAUCBLDICAGCgIAURAAEAkgAIAEBAAmVxMALBQBAAQQCIigF
CACAAICAAgQEhFREAgggIAAgwEghgABEAgQihSAAFAoLATUAADAASEAgABQ
BIUiABAiQAAEIAIDAAAAIggCJCARABgAJSApCaAwIAIBggwwAAAUUAINAIE
ABKEBkAIAMAECqEMIkNGpAoCFAkBCQAAAFBgAgAAwAB2BAAUEgA6AADIVAI
BEwMAIyEBQQAoAAYAAggeBGiA=
```

file1 hash

```
sdbf:03:9:file2.txt:13607:sha1:256:5:7ff:160:2:71:CIEVAjQGf
OEAGAASiSBZowLB2BYFAZDQAADKGwFQJHllAAlkDqmAA9AQOMJjIHwGSTAn
gcqBCtDQDtJIwCoGInUkSDzaioxgdAwQHFllBiIBYMyjKFYspoDzsAlxrFE
KiiLogDUQAQOECKiQAIBAAyCgQGaDBKAIK+LgCYG5DA5mQimZgwBggBAEuA
9AAMiDBAU0g0QoAiGrIg7pSBtGOaoBIUEEpAAxCA4ll2ENeAEgmAOzU4IKz
qECVhowoxVUwJ0CEAXBkDqNyiiAoQA/E8Flj4QF4QRBBQQg2liDEtwSLKEE
ZATuzD6gooJIp+KD2QSSIBzCFOasBAqSACz8AERoggAAAEIAlAUNoGCIAAC
TAAAICCGGHIICwAQkgAkAEBBAQBAFEAgQAgAAAAABABEGAEgBYiAAYAGAQA
KIAKFAQCBCgAAAECBLDICAGCgIAQRAAEAkgAIAEBAAmVRMALBQBAAQQCIig
FCACAAICAAgQEhFREAgggIAAgwEAhgAAEAgQihSAAFAoJATEAADAASEAgAB
QBAUiABAiAAAEIAIDAAAAAggCJCARABgAJSApCaAwIAIBAgwwAAAUUAINAI
EAAKEBkAIAMAECqEMIkNGpAgCFAkBCAAAAFBgAgAAwAB2AAAUEgAyAADIVA
IBEwMAIQEBQQAoAAYAAggaAGiA=
```

file2 hash

What do we do with suspicious files?

Possibilities:

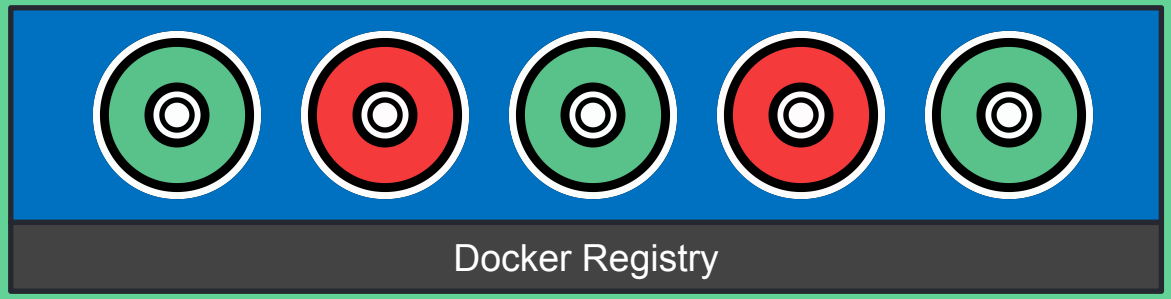
- Search in database of known “bad” files
  - find similar or identical hash

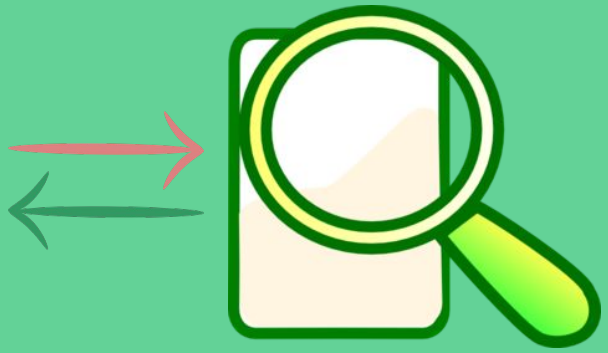
Is it dangerous?

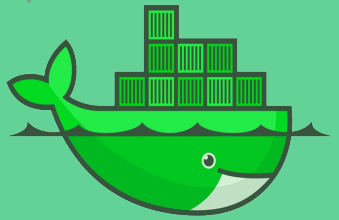
- Search in database of known “good” files
  - compare with hashes of safe files

Is it safe?

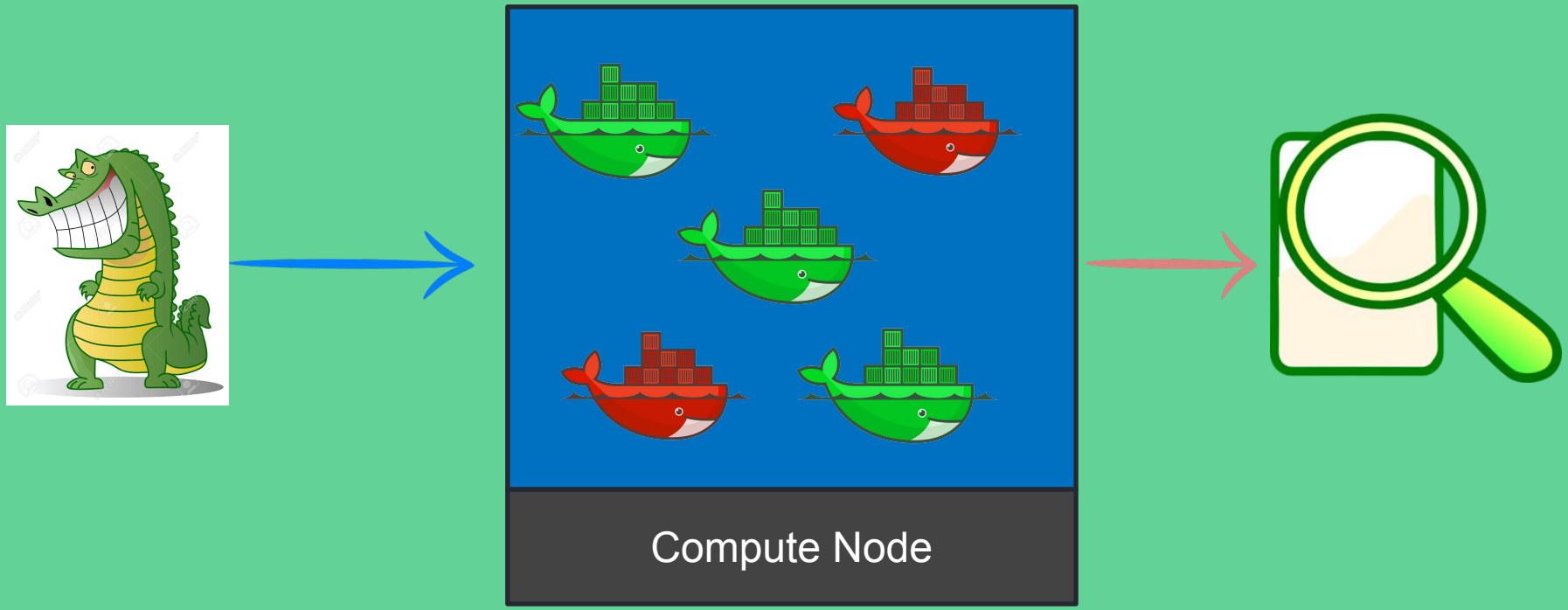
Usecase 1: Suspicious Images



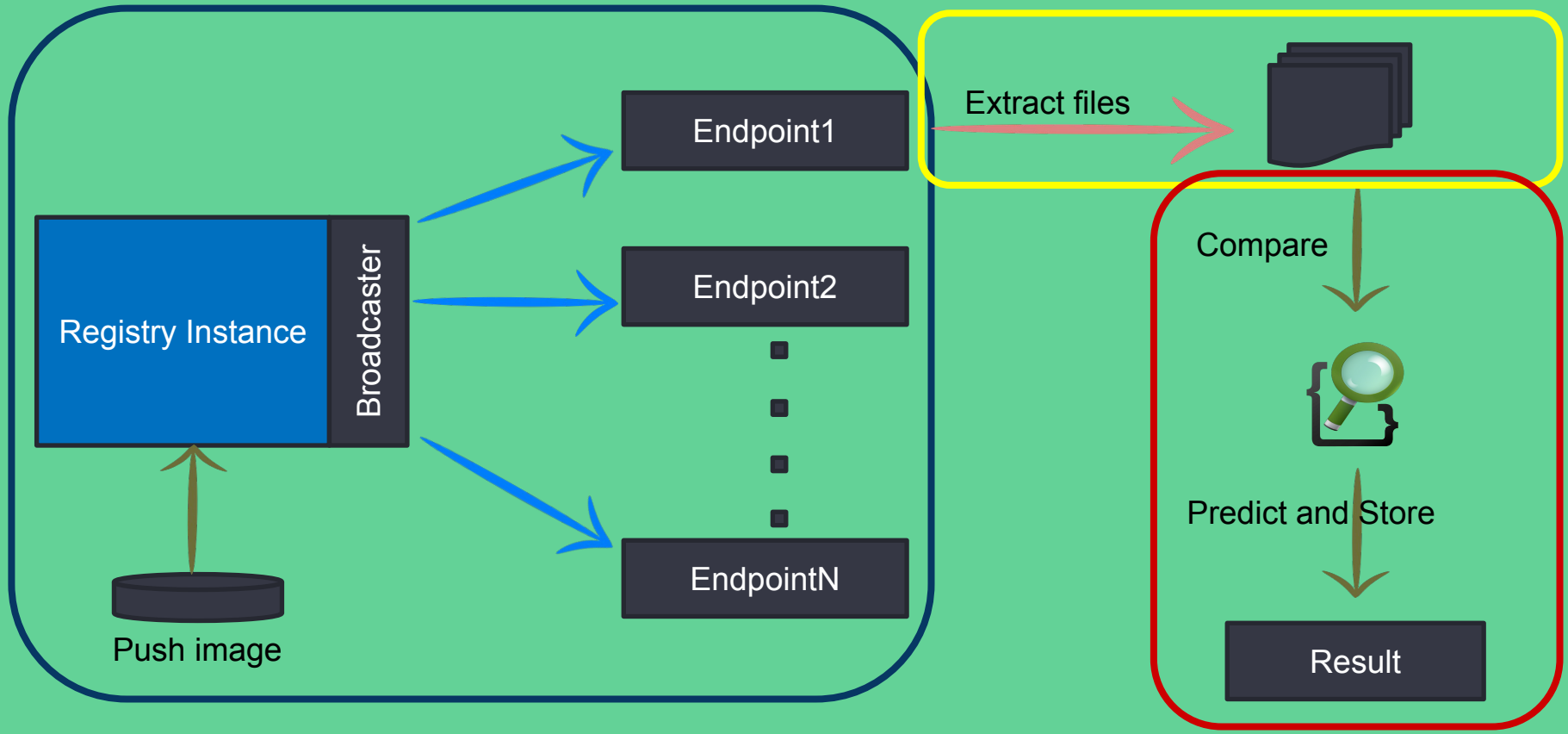




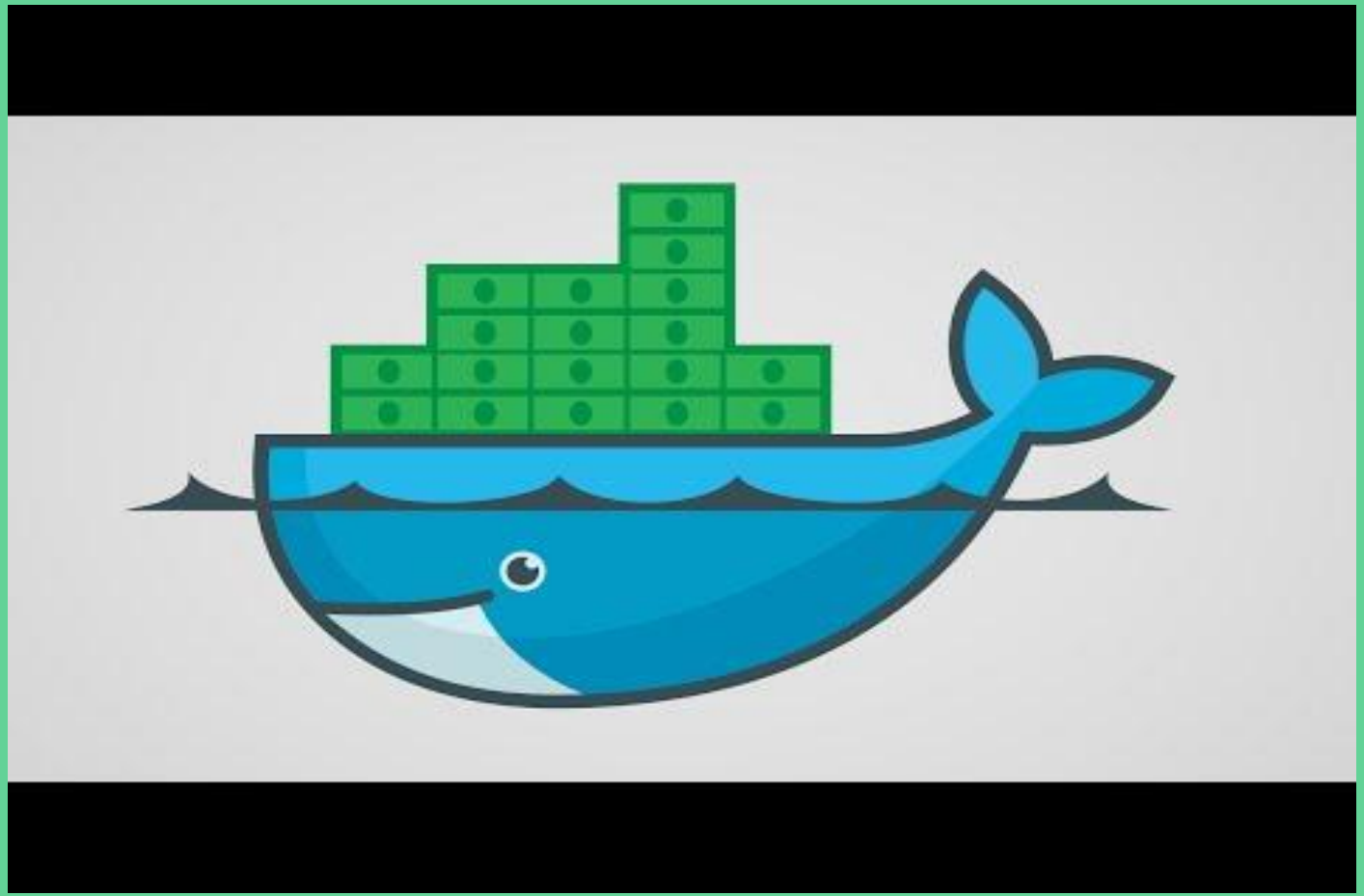
Usecase 2: Suspicious Containers



Notifications and processing



DEMO



Sprint 2 [NoBurn]

Setup environment on OpenStack

2/2

JM

Write python Script

1 3/3

JM



Tutorials of RabbitMQ

1

JM



Add a card…

In Progress Sprint 2

Add a card…

Sprint 2 - Demo Feb 23 Done

55 47

Configuring docker-registry for notifications and setting up REST API server to receive the notifications

1 13 13

Create python bindings for sdhash using swig

1 1/1 5 2

JM

Compute sdhash for ubuntu

1 8 8

JM

Capturing the events, extracting files, calculating sdhash and comparing

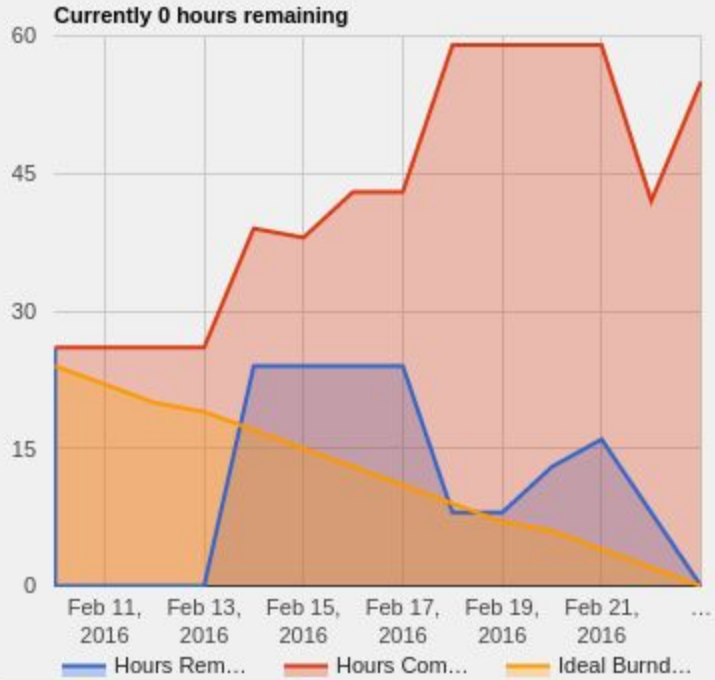
1 13 8

JM



Add a card…

Burndown Chart



Change Chart Type

(+)

Summary stats

Total Cards:	7
Remaining Cards:	0
Done Cards:	7
Percent of cards done:	100%
Hours at start:	26 (edit)
Hours est total:	47
Hours remaining:	0
Hours done:	55
Percent of hours done:	100%
----	----
Days Elapsed	13
Daily Burndown	4.23
Est. Days Left	0
Est. Completion Date	02/23/16

edit chart data | board settings | share

If this Trello Board represents a Sprint or a Product Backlog, adding a start and end date will allow the x-axis to be more accurate and a desired-velocity line to be drawn.

Date Range: 2016-02-10 to 2016-02-23 Update

Next sprint

- Come up with the sdhash threshold value ranges
- Come up with action plan for suspicious files
- Implement usecase1 and design for usecase2
- Fix recent issue faced where sdhash of same files comes out to be zero
- Integrate image push with hash-calculation

Thank you

&

Q/A