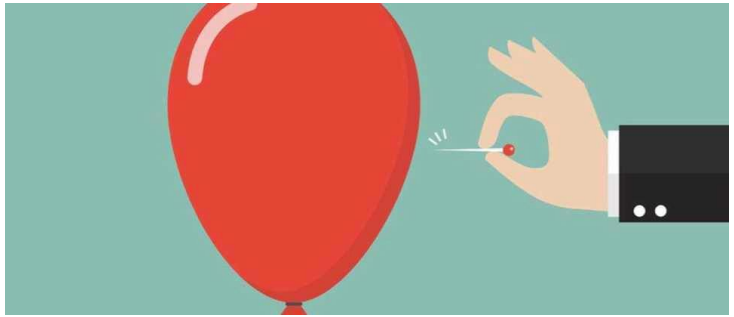
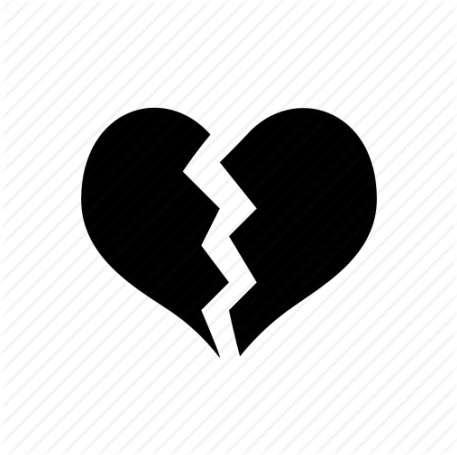
It won’t make a sound

… when it breaks



Would this?

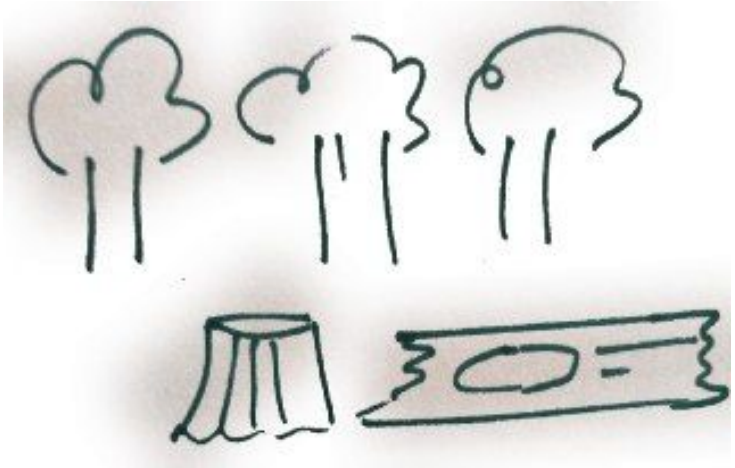


How about this?

How about this?

How about this?

How about this?



How about this?

You've got Page
Now what?

Flavors of Failure :

- ~~Software~~
- Human
- Network
- Process
- Culture

Outage # 1

Customer Service reports /login is down

Check

Datadog,
Papertrail,
NewRelic,
Cloudwatch

Tracing ✔️
Servers ✔️
Load ✔️
Errors ✔️

Meanwhile on Twitter...

What was it?

DevOps manually alter security groups,

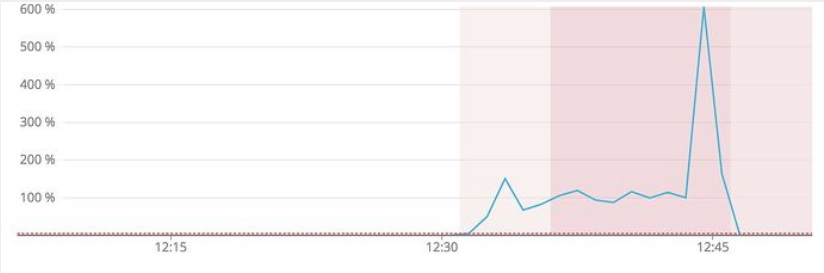
Accidentally deletes 443 rule

What’s the real Root Cause?

Outage # 2

~7:30 AM, 25 hours before a country launch, pagerDuty goes off

… ElasticSearch shows 500s are on the loose



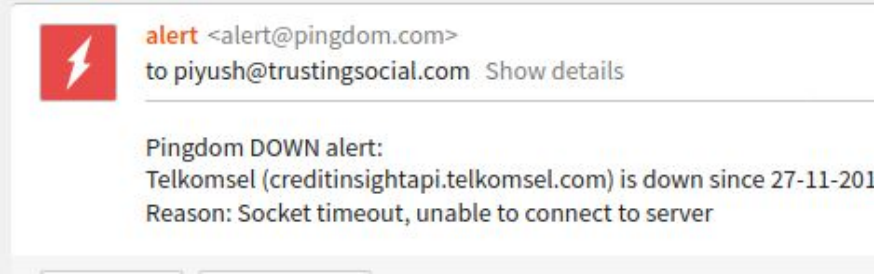
Logs come in at 1 mbps, No Correlation ID to isolate

We start copying the logs

Few minutes later...

500s stop, Pagerduty is autoresolved.

5 minutes into it, Pagerduty goes off again and public-API is unreachable



All hands on the Deck!

Check rundeck, was there a new deployment? ✘

Is Firewall down? ✘

Check Grafana
Check Stackdriver

Servers ✅
Load ✅
Docker ✅
APM ✅

Build custom script to search logs

... DB error on ‘some’ requests

- Elasticsearch
- Stackdriver
- Sentry
- Prometheus
- SREs

20 hours later;
we find...

`mount` command
hadn't run on a
db shard,
rebooted,
data wiped!

What’s the real Root Cause?

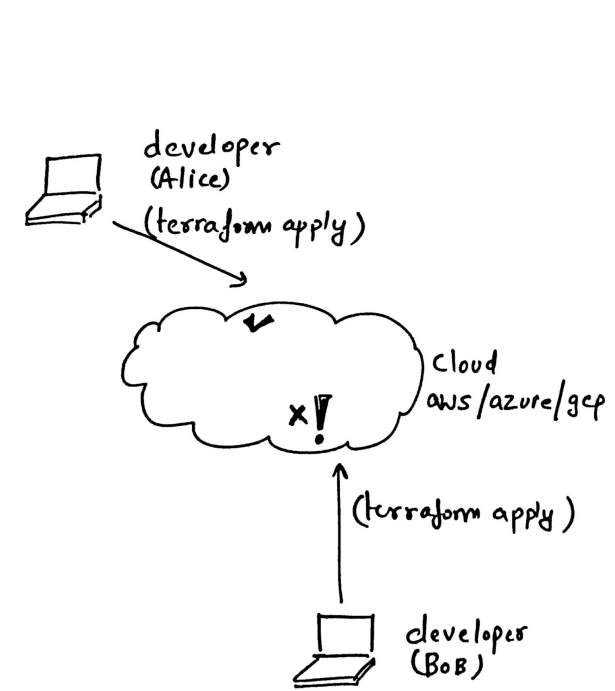
How was it working so far?

What else is breaking?

Outage #3

Multiple Jobs Run despite a distributed Lock

Lock failure doesn’t look like this



But more like this

Ideal Behaviour

```
curl http://127.0.0.1:2379/v2/keys/foo
-XPUT -d value=one

curl
http://127.0.0.1:2379/v2/keys/foo?prevVal
ue=one -XPUT -d value=two
{

     "action": "compareAndSwap",
     "node": {
          "createdIndex": 8,
          "key": "/foo",
          "modifiedIndex": 9,
          "value": "two"
     }
}
```

```
curl
http://127.0.0.1:2379/v2/keys/foo?prevExi
st=false -XPUT -d value=three
{
     "cause": "/foo",
     "errorCode": 105,
     "index": 39776,
     "message": "Key already exists"
}

curl
http://127.0.0.1:2379/v2/keys/foo?prevVal
ue=two -XPUT -d value=three
{
     "cause": "[two != one]",
     "errorCode": 101,
     "index": 8,
     "message": "Compare failed"
}
```

Default start with a key=stopped

Both process try and set key=started with preValue=stopped;

Only one should win

Run A, acquires a CAS lock on a key with TTL;

succeeds

A tries to update status; key not found.

B tries to update status, key not found

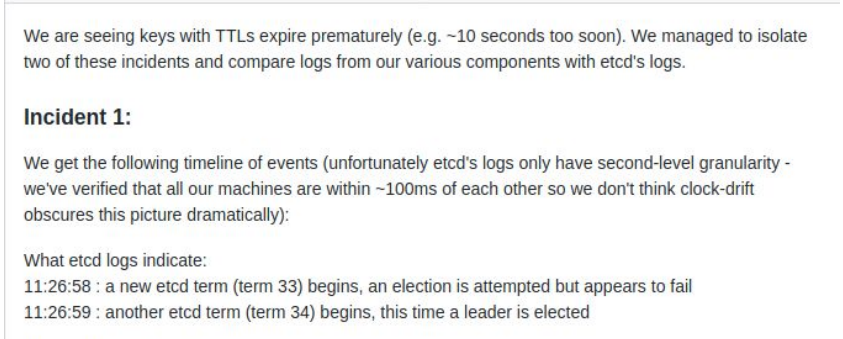
Run A, acquires a CAS lock;

Key not found!

Run B, acquires a CAS lock;

Key not found!

Our Diagnosis & Solution



We are seeing keys with TTLs expire prematurely (e.g. ~10 seconds too soon). We managed to isolate two of these incidents and compare logs from our various components with etcd's logs.

Incident 1:

We get the following timeline of events (unfortunately etcd's logs only have second-level granularity - we've verified that all our machines are within ~100ms of each other so we don't think clock-drift obscures this picture dramatically):

What etcd logs indicate:
11:26:58 : a new etcd term (term 33) begins, an election is attempted but appears to fail
11:26:59 : another etcd term (term 34) begins, this time a leader is elected

TTLs expires soon - related to etcd elections

Replace etcD with Consul

Real Reason?

The clock on the machines are not perfectly(within in a second) synced.

The TTL keys are managed by the leader.

If the clock is not synced, the new leader might remove the key "early" in the view of clients.

What’s the real Root Cause?

Outage #4

What will be the real Root Cause?

“The only real mistake is the one from which we learn nothing”

- Henry Ford

Thank You

- Piyush Verma
http://last9.io