

DISMANTLING COGNITIVE BIASES

in performance testing

Nicole van der Hoeven
Tricentis Flood
@n_vanderhoeven



What I if told you

You read the first line wrong?

1

What cognitive biases are

2

**What they have to do with
performance testing**

2

How to avoid them

Cognitive bias

A systematic pattern of deviation from norm or rationality in judgment

Intuition

Impressions

Involuntary

Effortless

Fast



Fast Thinking

$$2+2$$



Slow Thinking



Patient

Concentration

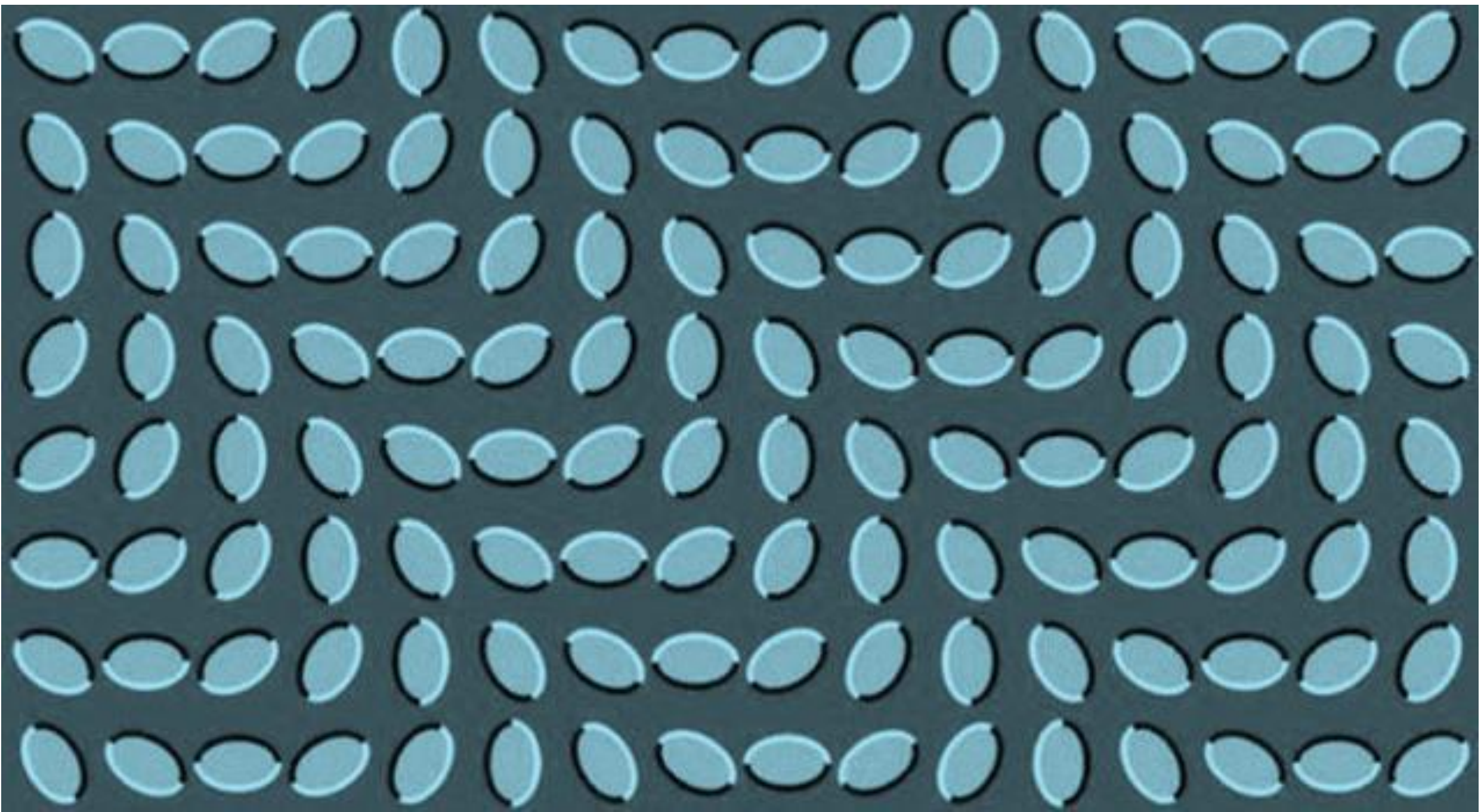
Reason

Effort

Slow



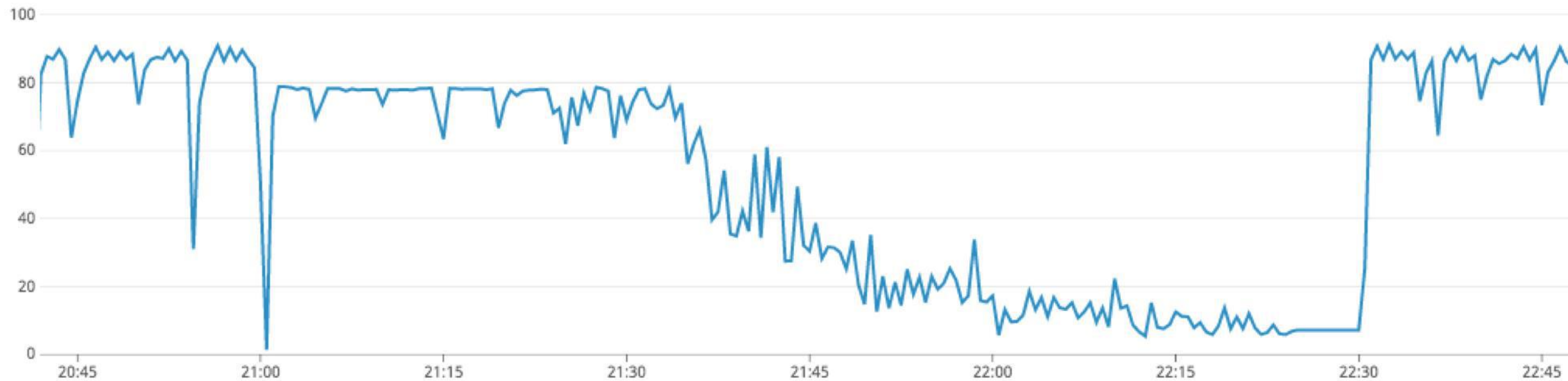
8x256





**What does this have to do with
performance testing?**

system.cpu.idle



89.93 Avg: 53.58 system.cpu.idle {*}

Anchoring effect

to rely too heavily on an initial piece of information offered (known as the "anchor") when making decisions



The diagram below shows the key interactions during user logon and the response time budgets allocated to each component in order to achieve the response time non functional requirements.

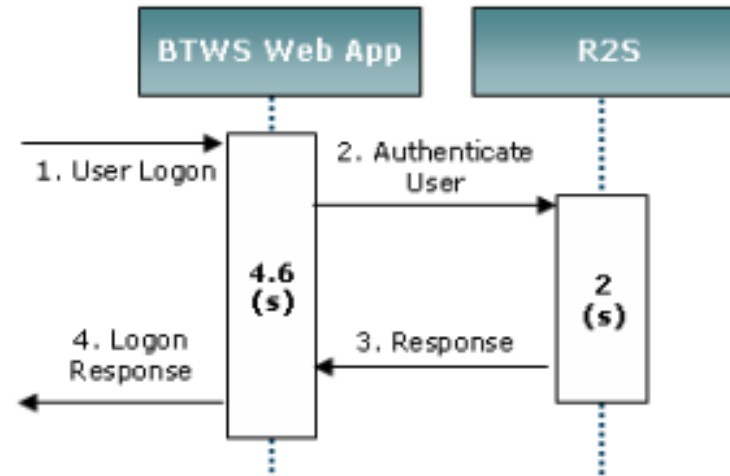


Figure 4 - User Logon Interaction Diagram

High Level Overview

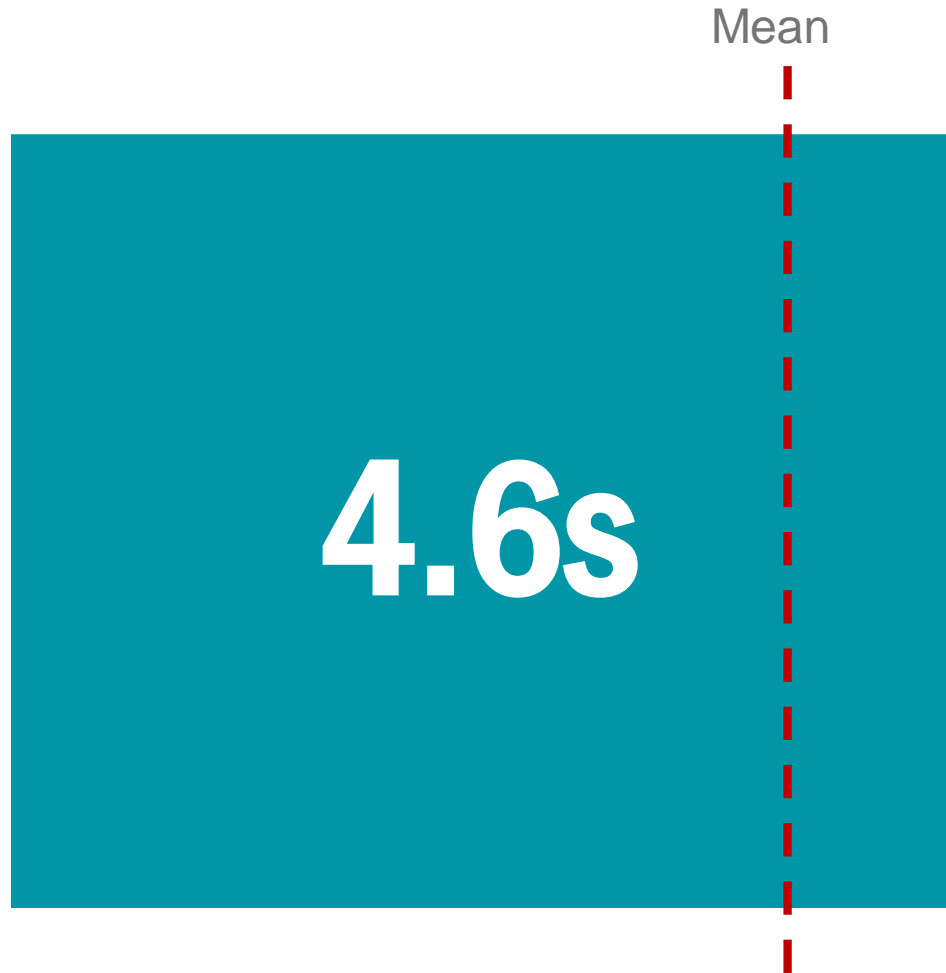
- User specifies credentials and logon is executed via BTWS Web App
- Smart Card authentication is performed (if required) and response provided by R2S infrastructure
- BTWS Web App completes logon and provides user with appropriate response

Expected End to End Response Time < 4.6 seconds 95th Percentile

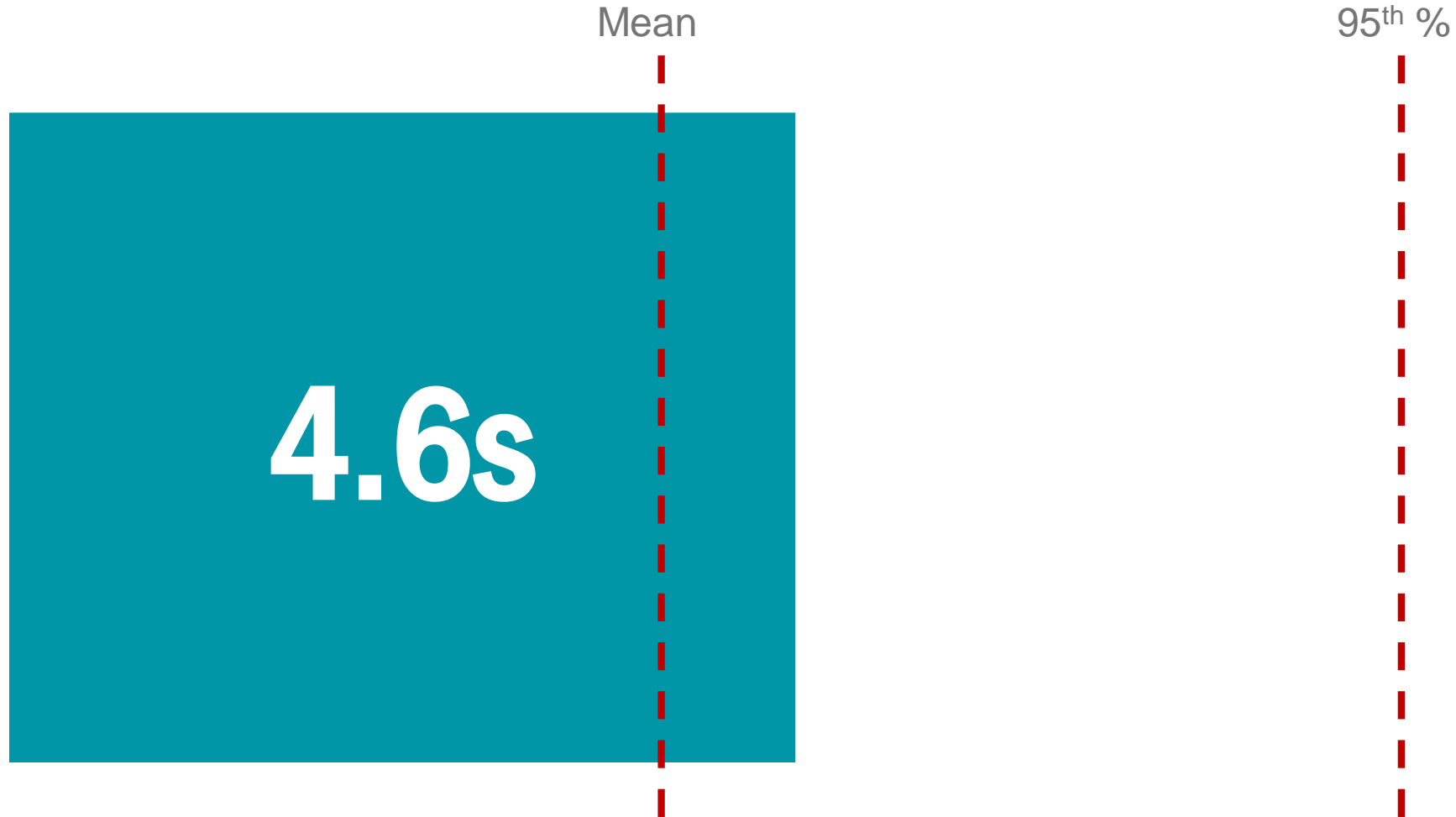
Assumptions:

- R2S infrastructure has a 2 second budget for smartcard authentication as part of user logon (production and PVT statistics unavailable)
- Time Budget for Password and Smart Card authentication is the same

It's pretty average



Moving to percentiles



Chasing symptoms

- "CPU is high"
- "Response time is high"
- "Throughput is low"



Cloud Based distributed Load Testing

for everyone



Gatling

Flood #1275 > Timeline

Timeline

Logs

All Regions

6 grids — 425 nodes



2m 15s

7m 15s

12m 15s

17m 15s

22m 15s

27m 15s

32m 15s

37m 15s

42m 15s

47m 15s

52m 15s

57m 15s

1h 2m 15s

1h 7m 15s

Aggregate of all transactions

6,841,010 users

Concurrent Users

34 s

Response Time

5,166,726 rpm

Transaction Rate

—

Latency

3,966,177 rpm

Passed Transactions

1,200,615 rpm

Failed Transactions



Label	↓ Timing	↓ Response Time	Requests per Minute	Error Rate
○ UMS - Concurrency - Check	<div style="width: 20%; background-color: green;"></div>	53 s	578,859 rpm	<div style="width: 37.74%; background-color: red;"></div> 37.74%
○ UMS - UM - Login	<div style="width: 10%; background-color: green;"></div>	35 s	30,019 rpm	<div style="width: 12.42%; background-color: red;"></div> 12.42%
○ UMS - Payment Init	<div style="width: 10%; background-color: green;"></div>	35 s	148,216 rpm	<div style="width: 97.75%; background-color: red;"></div> 97.75%
○ UMS - UM - Signup	<div style="width: 10%; background-color: green;"></div>	34 s	15,025 rpm	<div style="width: 12.29%; background-color: red;"></div> 12.29%
○ UMS - UM - Facebook	<div style="width: 10%; background-color: green;"></div>	34 s	15,009 rpm	<div style="width: 98.94%; background-color: red;"></div> 98.94%



2m 15s

7m 15s

12m 15s

17m 15s

22m 15s

27m 15s

32m 15s

37m 15s

42m 15s

47m 15s

52m 15s

57m 15s

1h 2m 15s

1h 7m 15s

Aggregate of all transactions

6,841,010 users
Concurrent Users

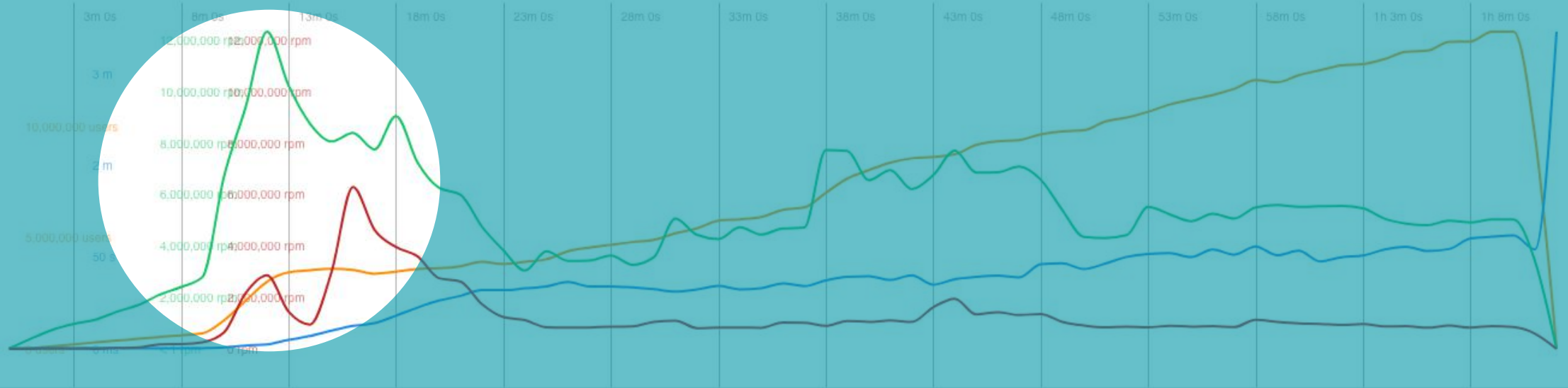
34 s
Response Time

5,166,726 rpm
Transaction Rate

—
Latency

3,966,177 rpm
Passed Transactions

1,200,615 rpm
Failed Transactions



Label	Timing	Response Time	Requests per Minute	Error Rate
UMS - Concurrency - Check	<div style="width: 20%;"></div>	53 s	578,859 rpm	37.74 %
UMS - UM - Login	<div style="width: 20%;"></div>	35 s	30,019 rpm	12.42 %
UMS - Payment Init	<div style="width: 20%;"></div>	35 s	148,216 rpm	97.75 %
UMS - UM - Signup	<div style="width: 20%;"></div>	34 s	15,025 rpm	12.29 %
UMS - UM - Facebook	<div style="width: 20%;"></div>	34 s	15,009 rpm	98.94 %



Nicole
support



Wil
ops



Tim
escalation



Hi, I'm running a test, but the dashboard is not coming up with any results.



Looks like the queue depth is increasing again. Got two warnings from VictorOps as well. Customers are reporting issues too. Looks like what happened the last time.

I haven't encountered this issue yet, so I'm not sure exactly how to diagnose.



You say this happened before? It wouldn't have been around the 3rd of this month, would it?



Just checked - actually, yes!



Availability bias

is a mental shortcut that relies on immediate examples that come to a given person's mind when evaluating a specific topic, concept, method or decision



Looks like what happened **the last time**.



I can understand the urgency! Our team is looking into this right now and we're hopeful this will be resolved soon.

How long is this going to take? We have very important tests to run.





Tim had to restart drain and pipe to fix it last time.
>>> Tim said: [a month ago]
“A simple restart of pipe and drain got the services working again. A quick code deploy could do the same thing for future reference.”



Can you do that now?

I'll take a look



Authority bias

the tendency to attribute greater accuracy to the opinion of an authority figure



Tim had to restart drain and pipe to fix it last time.
>>> Tim said: [a month ago]
“A simple restart of pipe and drain got the services working again. A quick code deploy could do the same thing for future reference.”



Can you do that now?

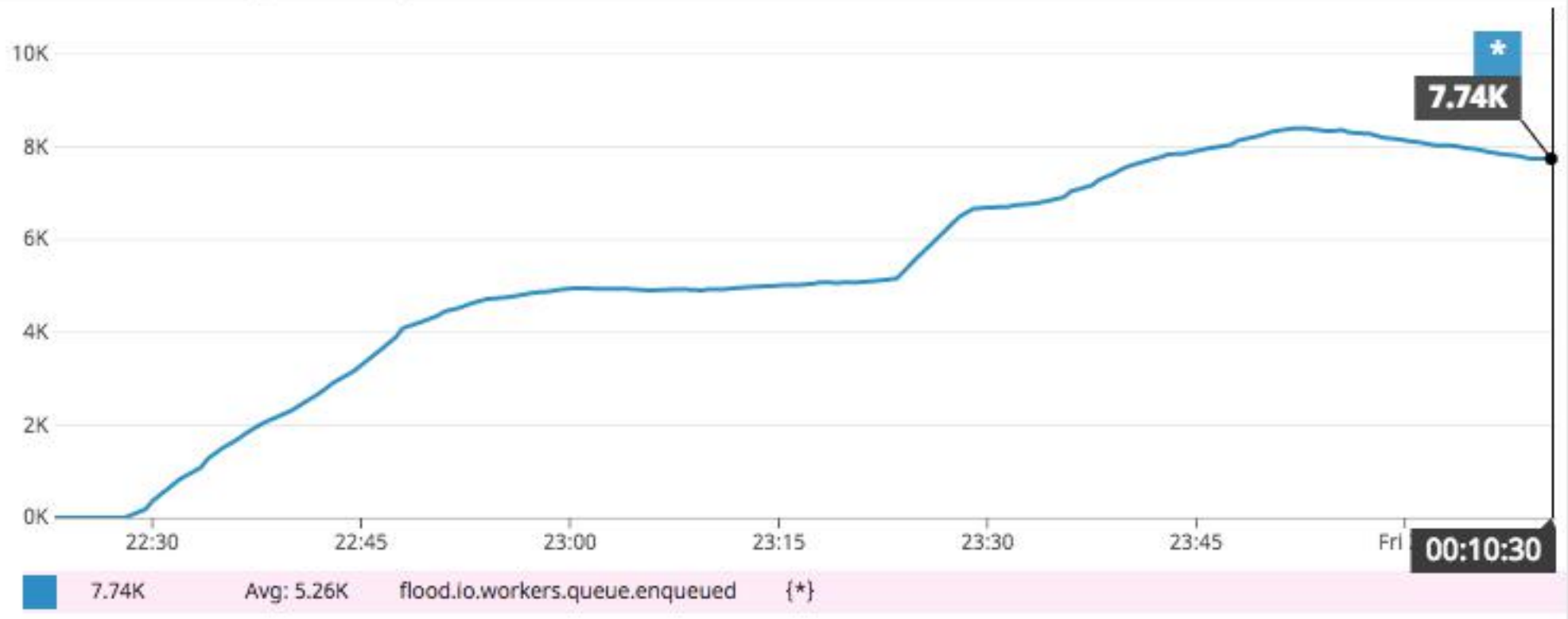
I'll take a look





Okay, looks like the queue depth is starting to decrease

flood.io.workers.queue.enqueue



[pipe] is spitting out a lot of these errors, which is not being picked up by datadog



```
Aug 23 14:11:20 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
2018/08/23 14:11:20 notifying bugsnap: label_normaliser: label
cardinality limit was exceeded
Aug 23 14:11:20 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:11:20Z" level=error msg="unable to process line"
error="label_normaliser: label cardinality limit was exceeded"
label=cardinality-too-high reason="cardinality too high"
Aug 23 14:11:20 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
2018/08/23 14:11:20 notifying bugsnap: label_normaliser: label
cardinality limit was exceeded
Aug 23 14:11:20 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:11:20Z" level=error msg="unable to process line"
error="label_normaliser: label cardinality limit was exceeded"
label=cardinality-too-high reason="cardinality too high"
Aug 23 14:11:20 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
2018/08/23 14:11:20 notifying bugsnap: label_normaliser: label
cardinality limit was exceeded
```

That's expected.





What's some recent log output without that message from pipe

It's almost exclusively that message with a few of these sprinkled in every few 100 lines



```
ug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:13:54Z" level=info msg=system goroutines=168
heap="49 MB" instance-id="pipe-20180427:1" os-mem="104 MB"
Aug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:13:54Z" level=info msg=go-workers floods-
inprogress=0 floods-queued=0 floods_status-inprogress=0 floods_status-
queued=0 grids-inprogress=0 grids-queued=0 nodes_status-inprogress=0
nodes_status-queued=0 proxy_write-inprogress=0 proxy_write-queued=7669
retry-depth=0
Aug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:13:54Z" level=info msg=worker db_avail_conns=0
db_curr_conns=150 db_max_conns=150 points-rejected=1 points-written=108
```



It's processing points, but slowly because a test might have slammed it with too many unique points which will be getting rejected

```
j -f | grep points-written  
Aug 23 14:47:14 ip-10-0-1-183.us-west-2.compute.internal docker[28373]:  
time="2018-08-23T14:47:14Z" level=info msg=smoothworker error-batches=0  
in-progress=0 influx=default points-errored=0 points-written=129  
queued=0
```

Confirmation bias

is the tendency to search for, interpret, favour, and recall information in a way that confirms one's pre-existing beliefs or hypotheses

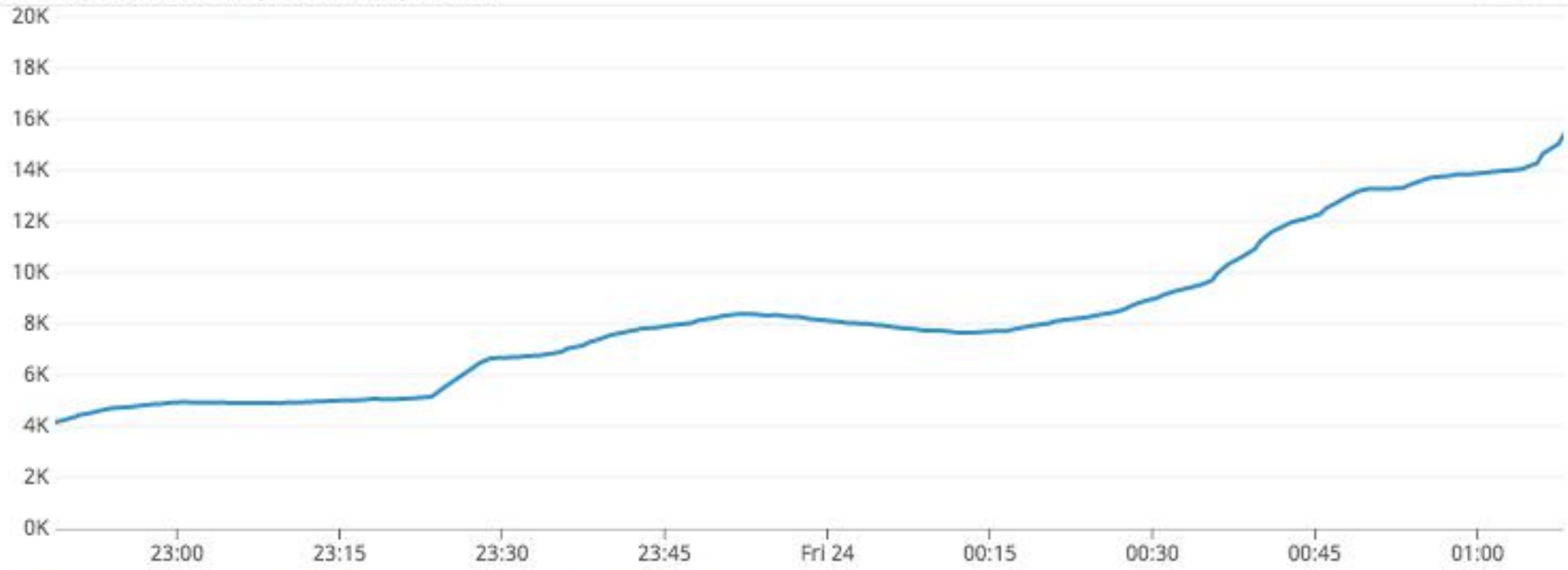


It's processing points, but slowly because a test might have slammed it with too many unique points which will be getting rejected

Queue depth is starting to increase again



flood.io.workers.queue.enqueued



15.06K Avg: 8.27K flood.io.workers.queue.enqueued {*}



We're still working on the issue, but we've made some progress. I'll keep you updated when it's resolved!

Can you give me an update





I think the bottleneck here though is DB connections, it's using all of them to check each label's ID before it's getting to reject them

m4.xlarge should have plenty to spare, I think we're sitting at about 570 connections



I'm creating another ASG for pipe, theory is I can deploy to new one with double capacity and run them side by side for a bit



It's pretty bad right now, I can't get PG connections back under 1000, going to need to clean up by hand



Oh crap



Rebooting PG now, interesting that connections aren't released (by pipe maybe) and I have to hard reboot PG. You would think destroying containers would get rid of connections but doesn't seem to be in tonight's case

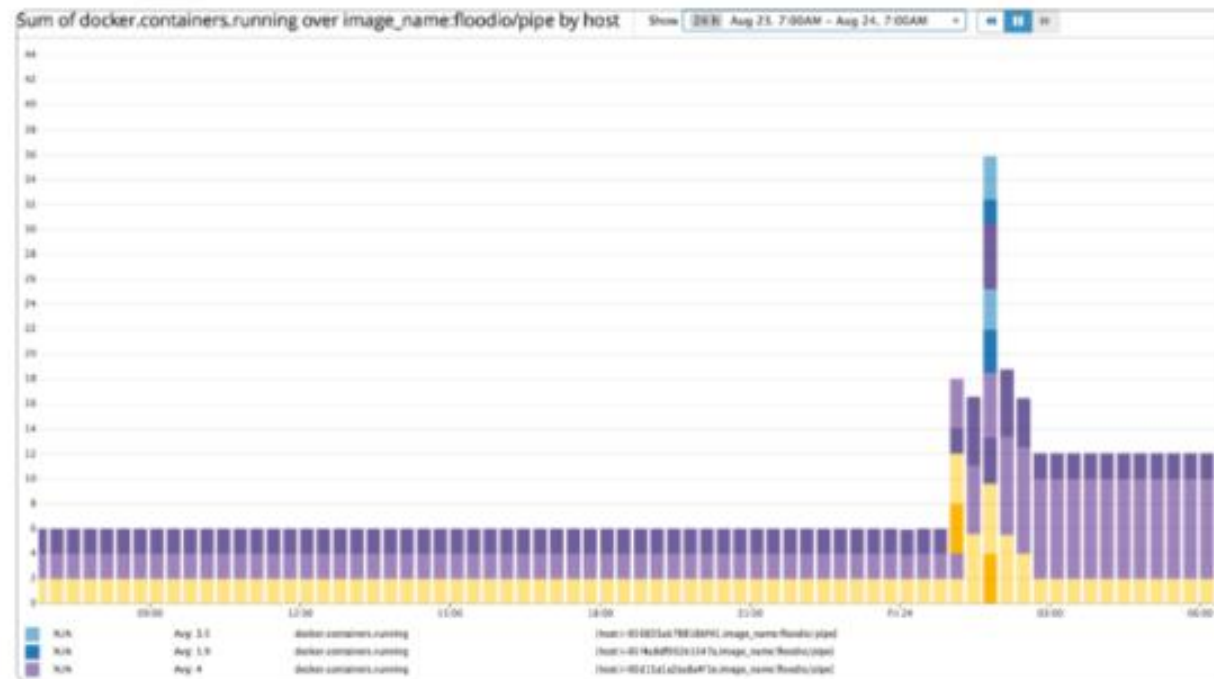
Anything I can do? Nuke idle connections?





OMG there were 8 containers running on one of the hosts, $8 * 150 =$ out of DB connections most likely

Interesting, I didn't think to check that, I assumed they would be managed by fleet





What's some recent log output without that message from pipe

It's almost exclusively that message with a few of these sprinkled in every few 100 lines



```
ug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:13:54Z" level=info msg=system goroutines=168
heap="49 MB" instance-id="pipe-20180427:1" os-mem="104 MB"
Aug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:13:54Z" level=info msg=go-workers floods-
inprogress=0 floods-queued=0 floods_status-inprogress=0 floods_status-
queued=0 grids-inprogress=0 grids-queued=0 nodes_status-inprogress=0
nodes_status-queued=0 proxy_write-inprogress=0 proxy_write-queued=7669
retry-depth=0
Aug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:
time="2018-08-23T14:13:54Z" level=info msg=worker db_avail_conns=0
db_curr_conns=150 db_max_conns=150 points-rejected=1 points-written=108
```

Inattention blindness

when an individual fails to perceive an unexpected stimulus that is in plain sight

```
ug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:  
time="2018-08-23T14:13:54Z" level=info msg=system goroutines=168  
heap="49 MB" instance-id="pipe-20180427:1" os-mem="104 MB"  
Aug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:  
time="2018-08-23T14:13:54Z" level=info msg=go-workers floods-  
inprogress=0 floods-queued=0 floods_status-inprogress=0 floods_status-  
queued=0 grids-inprogress=0 grids-queued=0 nodes_status-inprogress=0  
nodes_status-queued=0 proxy_write-inprogress=0 proxy_write-queued=7669  
retry-depth=0  
Aug 23 14:13:54 ip-10-0-1-183.us-west-2.compute.internal docker[27213]:  
time="2018-08-23T14:13:54Z" level=info msg=worker db_avail_conns=0  
db_curr_conns=150 db_max_conns=150 points-rejected=1 points-written=108
```

+5 hours of outage

Availability

Confirmation

Authority

Inattention blindness

Avoiding Biases

What to do about biases

~~Avoiding Biases~~

Availability bias

is a mental shortcut that relies on immediate examples that come to a given person's mind when evaluating a specific topic, concept, method or decision



Looks like what happened **the last time**.



Tim had to restart drain and pipe to fix it last time.
>>> Tim said: [a month ago]
“A simple restart of pipe and drain got the services working again. A quick code deploy could do the same thing for future reference.”



Can you do that now?

Authority bias

the tendency to attribute greater accuracy to the opinion of an authority figure



Tim had to restart drain and pipe to fix it last time.
>>> Tim said: [a month ago]
“A simple restart of pipe and drain got the services working again. A quick code deploy could do the same thing for future reference.”



Can you do that now?

I'll take a look



Identify which System of thought is appropriate for the situation.

- **What is the deadline?**
- **What are the potential effects of indecision?**
- **What are the potential effects of inaccuracy?**

-- Can we afford to be wrong?



Use both systems of thought in tandem.

- **Have I seen something like this before, and what did I learn from that? (Fast Thinking)**
- **Could there be something else causing the same symptom? (Slow Thinking)**

Intuition

Impressions

Involuntary

Effortless

Fast Thinking



Patient

Concentration

Reason

Effort

Slow Thinking



Assume you are biased.

Availability

Confirmation

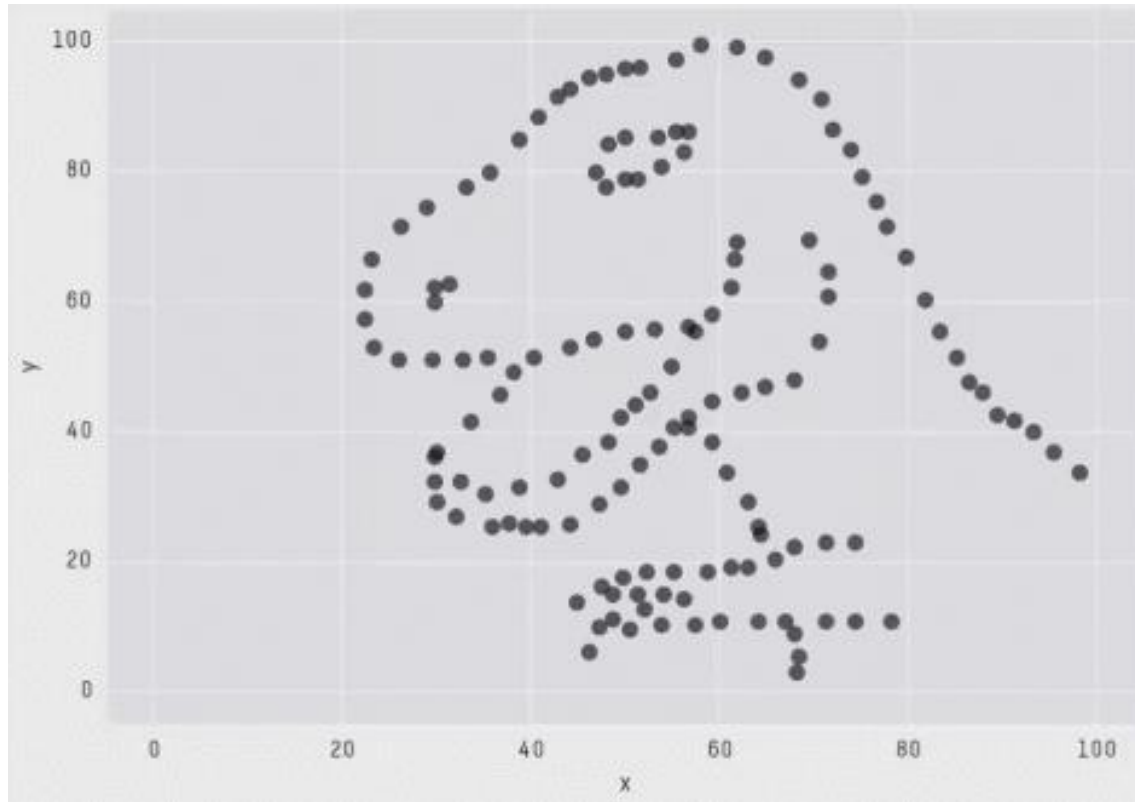
Authority

Inattentional blindness

Anchoring bias

to rely too heavily on an initial piece of information offered (known as the "anchor") when making decisions

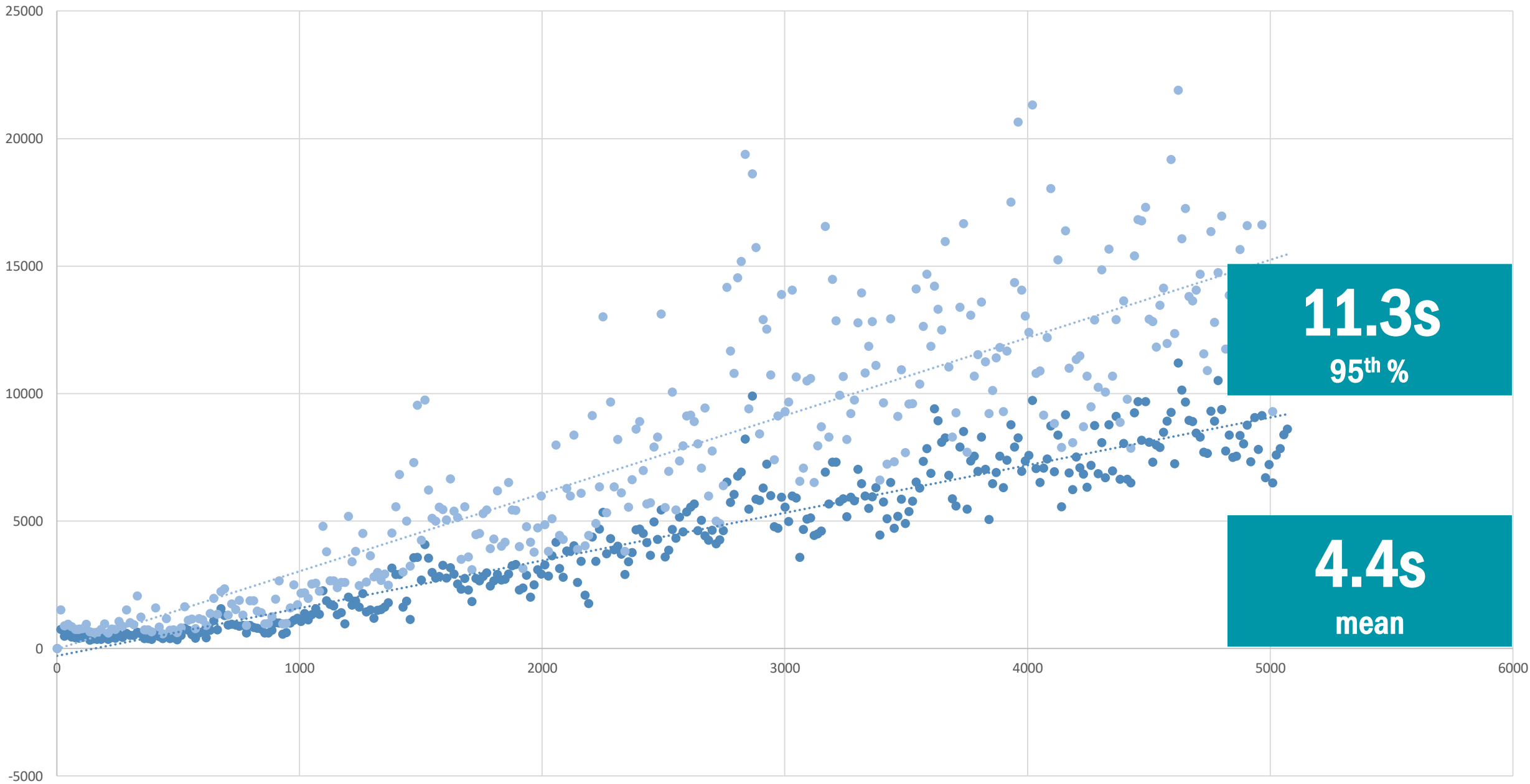
Visualize not summarize



```
X Mean: 54.2659224  
Y Mean: 47.8313999  
X SD : 16.7649829  
Y SD : 26.9342120  
Corr. : -0.0642526
```


“All measurements are wrong”

due to the presence of error, measured results can only ever be an approximation

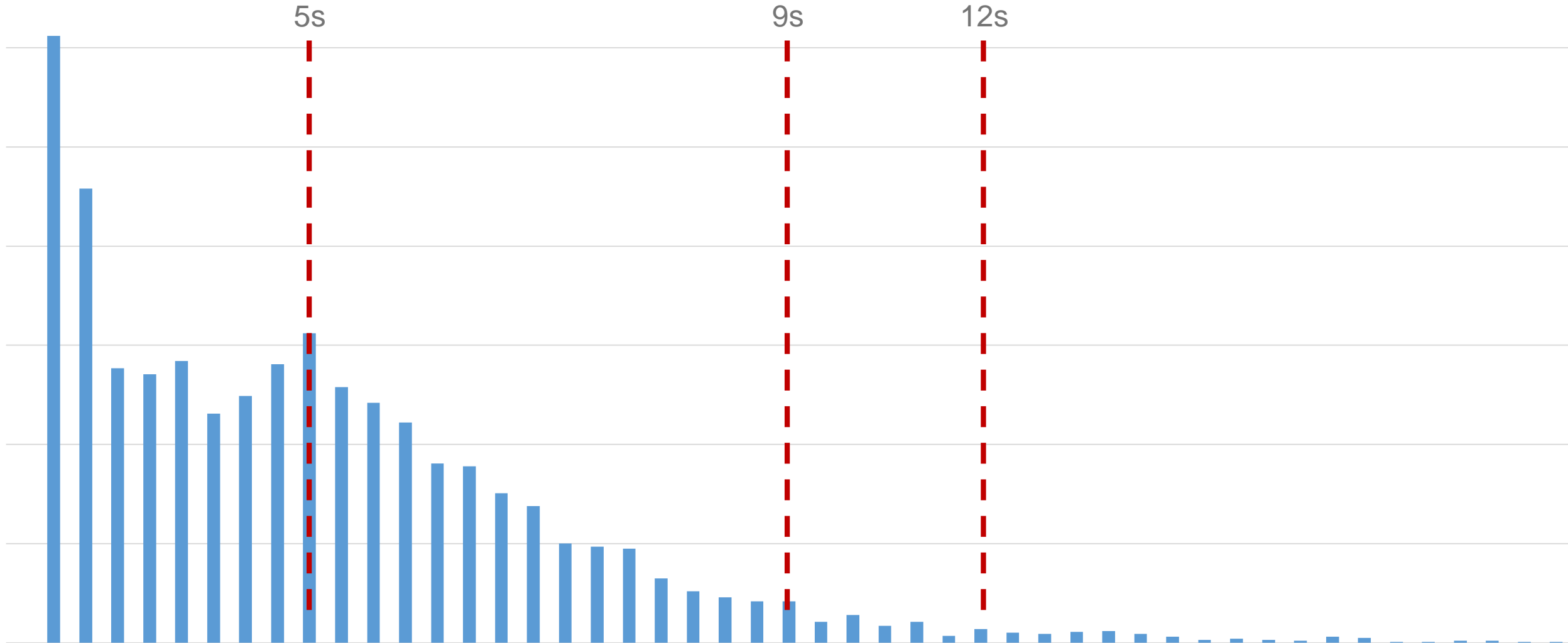


11.3s
95th %

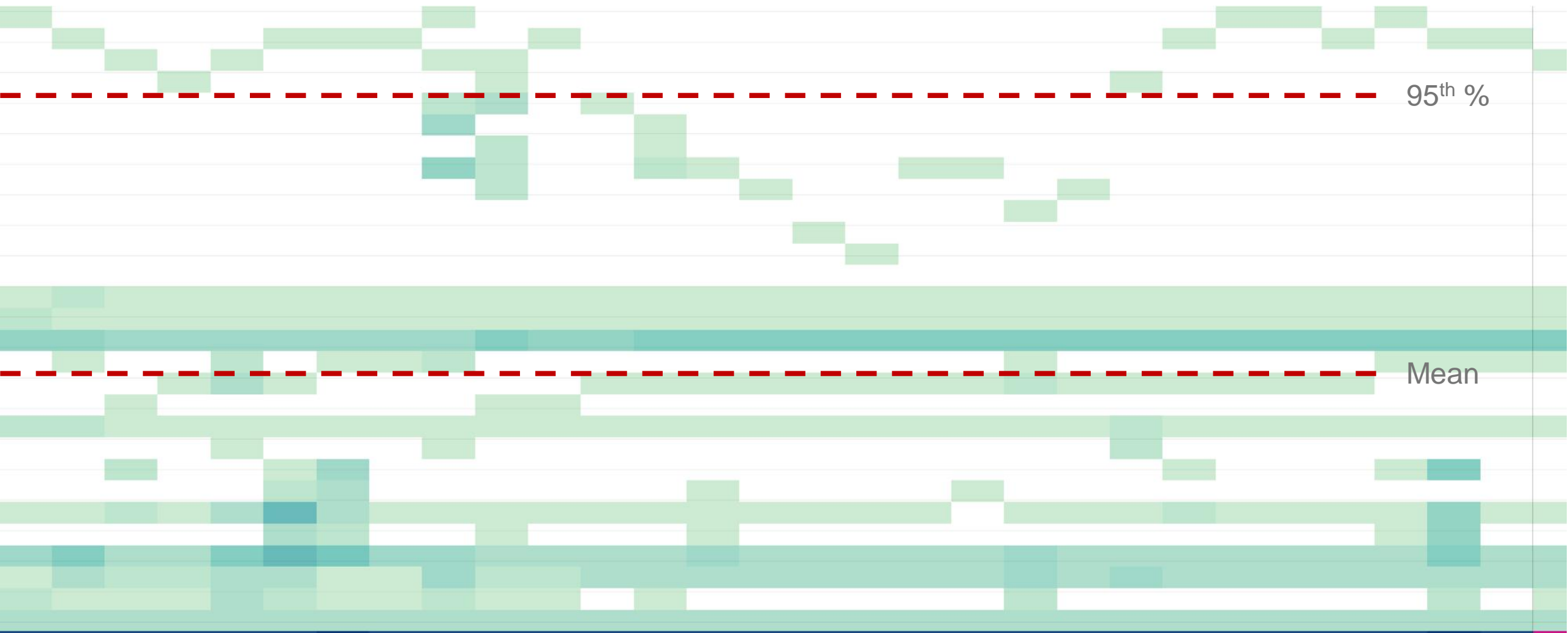
4.4s
mean

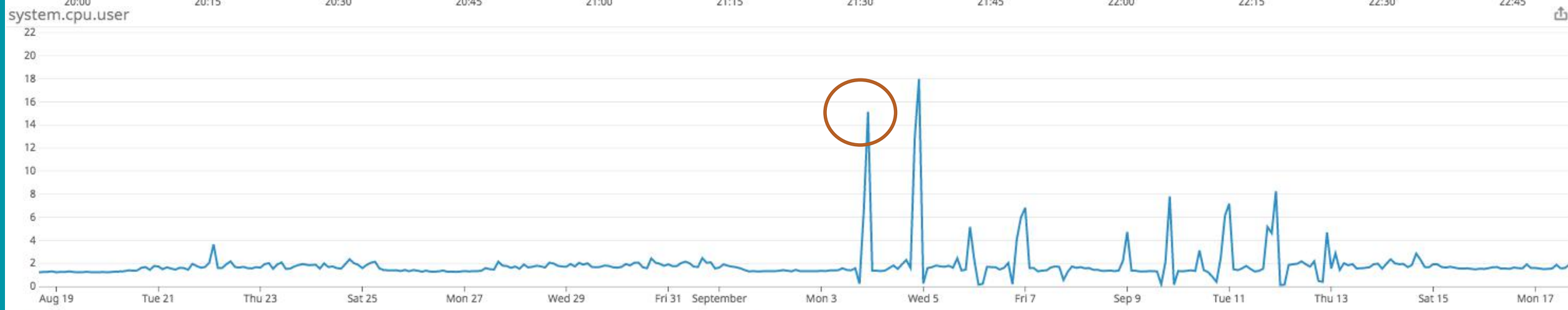
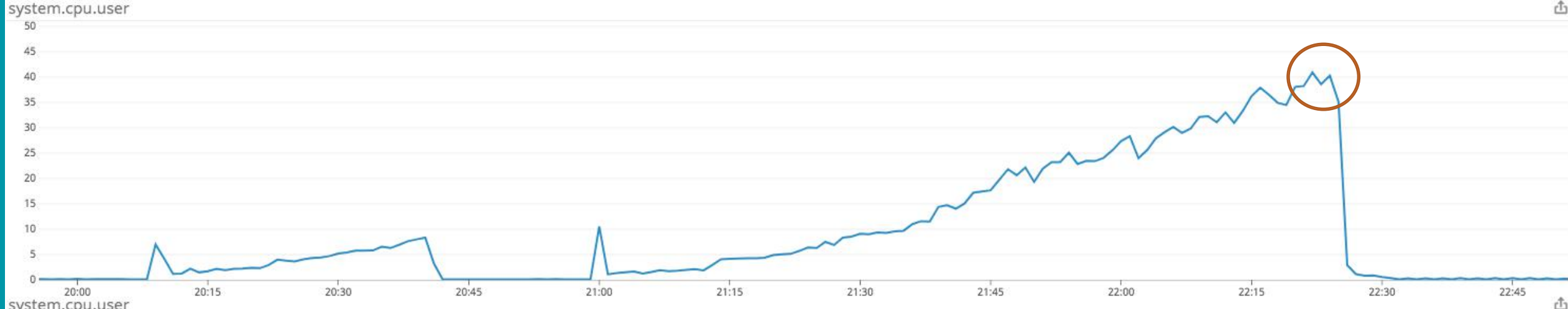
● mean ● percentile Linear (mean) Linear (percentile)

Frequency Distributions



Heatmaps

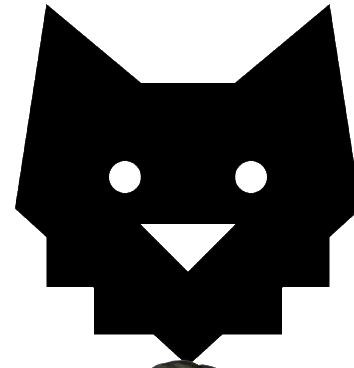
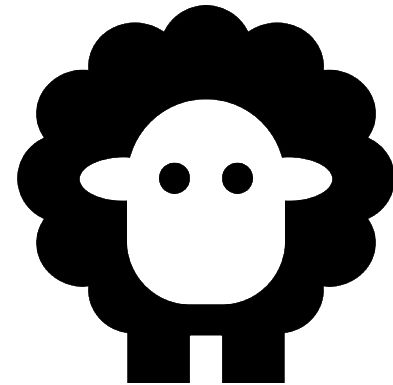




Authority bias

Avoid being a sheep

Try being the wolf



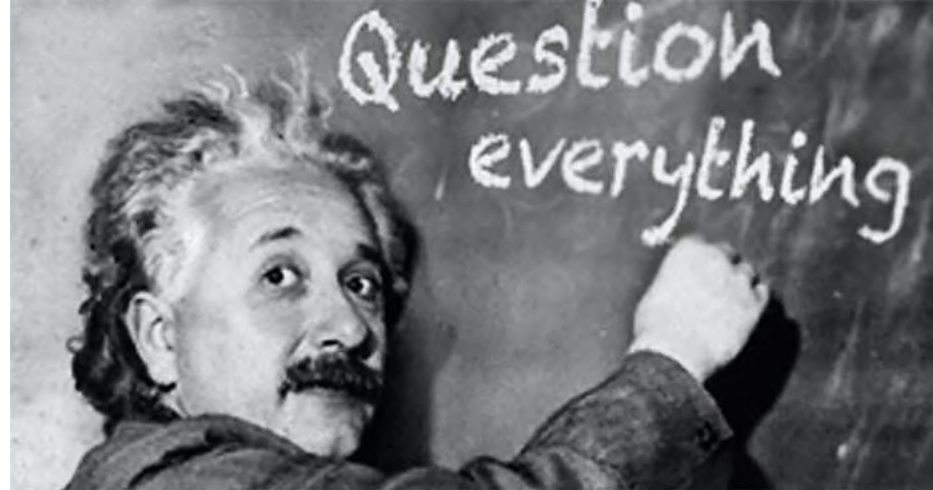
Availability bias

“Every time this happens, it almost always ends up being this problem. What else could be going wrong though?”

Confirmation bias

Exploratory instead of just confirmatory thought

Focus less on being right

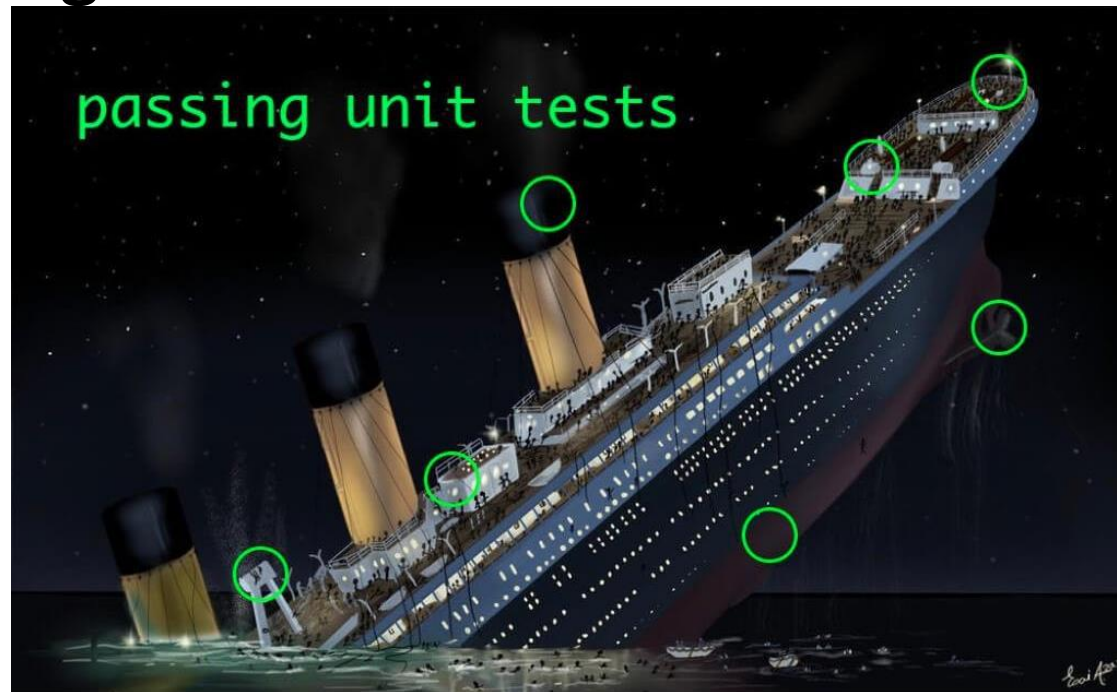


Inattentional bias

Sleep on it

Don't jump to conclusions

Is there something I've overlooked?



Develop expert intuition.



Nicole

support



Wil

ops



Tim

escalation

Seek independent sources.



Tim had to restart drain and pipe to fix it last time.
>>> Tim said: [a month ago]
“A simple restart of pipe and drain got the services working again. A quick code deploy could do the same thing for future reference.”



Can you do that now?

What to do about cognitive biases

- 1 Identify whether Fast or Slow Thinking is appropriate.**
- 2 Use BOTH systems.**
- 3 Assume you are biased.**
- 4 Develop expert intuition.**
- 5 Seek independent sources.**

Questions?

References

Kahneman, D. (2011). *Thinking, fast and slow*. Farrar, Straus and Giroux.

<https://blogs.scientificamerican.com/illusion-chasers/the-2017-best-illusion-of-the-year-contest/>

http://www.perfdynamics.com/Manifesto/gcaprules.html#tth_sEc2.25

https://www.reddit.com/r/coolguides/comments/8fz45a/flow_chart_guide_of_cognitive_biases/

<https://en.wikipedia.org/wiki/Anchoring>

https://en.wikipedia.org/wiki/Authority_bias

https://en.wikipedia.org/wiki/Confirmation_bias

https://en.wikipedia.org/wiki/Congruence_bias

<https://en.wikipedia.org/wiki/Causality>

https://en.wikipedia.org/wiki/Inattentional_blindness

<http://www.thefunctionalart.com/2016/08/download-datasaurus-never-trust-summary.html>

<https://www.saffo.com/02008/07/26/strong-opinions-weakly-held/>

@n_vanderhoeven