Ensuring success during disaster

SRECON 2015
@dougbarth
Doug Barth
@dougbarth

3 yr old— Daddy, why do you have short hair?
Me— I look better with it.
5 yr old— Not really.
ENSURING SUCCESS DURING DISASTER

Dev

Signal Engage

Ops

Consider killing this
How is baby PagerDuty formed?
Reliability
Agenda
DR vs. HA
Data DR
Failover
Active/Active
Legacy Systems
Q&A
Disaster Recovery
A plan for surviving rare failure events that threaten our ability to continue operating
High availability
ENSURING SUCCESS DURING DISASTER

DR

Rare failures

Slow recovery

HA

Common failures

Fast recovery
Latency
DR needs to exist
DR needs to be tested
DR needs to be tested for correctness
DR needs to be tested for capacity
DR needs to be tested for execution
Data DR
You can never have too many copies
Lose a disk?
Lose a server?
Secondary
Data corruption?
Backups
Test restorations
DROP TABLE USERS;
DELETE FROM USERS;
Delayed secondary

Primary

Secondary (delayed 2 hrs)
Failover
Primary DC & Secondary DC
The flip
Testing == outage
So it goes untested
Tested infrequently
Breaking change

Failure window

Flip test

Flip test
Forgotten roles
Low capacity
Expensive
Active/Active
PagerDuty ❤
Active/Active
Multiple regions
Use all the AZs!
Constant validation
Failure Friday

HTTPS://WWW.PAGERDUTY.COM/BLOG/FAILURE-FRIDAY-AT-PAGERDUTY/
Trade latency for reliability
30ms RTT
Legacy Systems
Older systems still use failover
Correctness
Capacity
Execution
Correctness
Capacity
Execution
1% of requests go to DR site
Finds issues immediately
Requires DR uses production DB
30ms per DB call
Trade latency to test correctness
Correctness
Capacity
Execution
DR is a mirror of production
dr-compare.rb
Checks that DR footprint is the same as production
Surfaces missing roles
Correctness
Capacity
Execution
The flip
Production impact
Scale up staging
Practice flip there
Tested infrequently
PagerDuty ❤️  
Active/Active
Correctness
Capacity
Execution
ENSURING SUCCESS DURING DISASTER

pagerduty.com/jobs

DOUG@PAGERDUTY.COM
Questions?

doug@pagerduty.com