Lessons Learned from Our First Worldwide Outage

Yoav Cohen, VP Engineering, Incapsula
@yoavcohen
What is Incapsula?
What is Incapsula?

Caching
DDoS Rules
WAF Rules
Client Classification
App Delivery Rules
Incapsula in High Level

• ~35 PoPs
  – Proxies (HTTP/S)
  – Behemoths (L3-4)

• Management Console
Customer Configuration

- Domain name, IP of origin server
- All servers need to know that
- With many updates happening at any given time
- 60 seconds SLA
Revlite

- Leader repository

- A client service that synchronizes changes to the file-system

- Applications read from the file-system
Our First Global Outage
IncapRules

- Custom WAF logic
- Tons of flexibility
  - Reading rules
  - Evaluating rules
IncapRules Safety Mechanism

```
curr_sites = [7]
```

- `read_rules(site_id=7)`
- `eval_rules(site_id=7)`
- `curr_sites.txt: [7]`
Initial Response
Post Mortem Conclusions

1. The safety mechanism did not work

2. The safety mechanism only works after a crash…
1. The safety mechanism did not work

• // Someone commented it out

• Why wasn’t it picked-up by testing?

• Because we didn’t have a test for it
Incapsula Reliability Framework

- P0 – Customer Traffic
- P1 – System Administration
- P2 – Customer Administration

Planning
System Design
2. The safety mechanism only works after a crash

• Simple (bad) solution – propagate slowly
  – How slow?
  – Will still impact production systems
  – Will degrade customer agility

• Break down the problem:
  – Crash when loading rules
  – Crash when evaluating rules → Safety Mechanism
Crash when loading rules

• Don’t publish configurations that will cause a crash
• But, crash happens when publishing

• Publish to a system that is not processing traffic
Revlite Sandbox

REPO SERVER
Sandbox Repo — Leader Repo

Management Console

SANDBOX PROXY
Local Repo — Application

PROXY
Local Repo — Application
Flying two mistakes high
Revert to a last known good
Wrapping it up

• Configuration changes can cause issues
  – While loading the change
  – After the fact, out of the blue

• Issues during a configuration change
  – Sandbox the change
  – Quarantine it if necessary

• Out of the blue
  – Detect "bad" objects and quarantine them
  – Roll-back to a last known good configuration
Questions?

@yoavcohen