MAELSTROM

Mitigating Datacenter-level Disasters by Draining Independent Traffic Safely and Effectively

Kaushik Veeraraghavan, Justin Meza, Scott Michelson, Sankaralingam Paneerselvam, Alex Gyori, David Chou, Sonia Margulis, Daniel Obenshain, Shruti Padmanabha, Ashish Shah, Yee Jiun Song, Tianyin Xu*

Facebook, Inc. *UIUC
But...
Leased Data Center
Forest City Data Center
Can we run Facebook without one of our data centers?
Fault tolerance without compromising correctness

Must track dependencies between micro services

Testing to verify dependencies and configuration
Draining
Runbooks

1. Status of Network
2. Start migration services
   - Unicorn
   - Dragon
3. Search (Bandwidth permitting)
4. War Room Testing
   - Set up VIP or VLB servers
5. Employee Testing
   - Set up latest
6. User Traffic - 11
7. Regional Load Test (Monday?)

Other services
- Nima
- Expect to lose one
- If significant bandwidth, coordinate in #eurotech
Maelstrom

- Drains traffic from data centers
- Minimal operator interaction
- Used in tests and production events
Monolithic system
Micro services
Traffic types

Stateless

Stateful

Sticky

Replication
**Task** TrafficShift

```json
{
  vip_type: <virtualIP>,
  target: <datacenter>,
  ratio: <amount>
}
```
DEPENDENCE

AUTOMATION

VALIDATION
Dependence

Captures service dependence

Directed acyclic graph

Manually curated in runbooks
A U T O M A T I O N

Traffic shifting

Job manipulation

Configuration changes

Shard movement
VALIDATION

Metrics and thresholds

When to launch a step

When a step has completed

Service health checks
Dependence

Automation

Validation
<table>
<thead>
<tr>
<th>Task</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster Traffic</td>
<td>Ready</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
<tr>
<td>Cluster Traffic</td>
<td>Done</td>
</tr>
</tbody>
</table>
Drain tests

Performed weekly

Verify dependencies

Ensure tools work
Limitations

Must plan for shared resources

Reactive approach to mitigation

Need to avoid cascading failures

Not fastest for every disaster
Stateless traffic

Requests per second:
- 0E+00 (0)
- 1.4E+13
- 2.1E+13
- 2.8E+13
- 3.5E+13

Multiplier value:
- 0.0
- 0.2
- 0.4
- 0.6
- 0.8
- 1.0

Date and time:
- September 13, 8 AM PDT
- 9 AM PDT
Sticky traffic

Multiplier value

Active sessions

September 13, 8 AM PDT

9 AM PDT

Session eviction
Fault tolerance without compromising correctness

Must track dependencies between micro services

Tests verify dependencies and configuration