Extending the error budget model
SRECon2019 / NYC
Error budgets bring sanity to availability
Error budget mechanics

A life changing approach to achieving operating balance
SLI - Service Level Indicator
- a measurement of user value delivered

Error budget mechanics

A life changing approach to achieving operating balance
**Error budget mechanics**

A life changing approach to achieving operating balance

---

**SLO - Service Level Objective**

*a threshold that makes sense in your context*

- SLI - Service Level Indicator - a measurement of user **value** delivered

<table>
<thead>
<tr>
<th>Total downtime in last 30 days</th>
</tr>
</thead>
<tbody>
<tr>
<td>43.3 min / 99.9% target</td>
</tr>
</tbody>
</table>
Error budget mechanics

A life changing approach to achieving operating balance

Policy - take action when approaching threshold

- SLI - Service Level Indicator
  a measurement of user value delivered
- SLO - Service Level Objective
  a threshold that makes sense in your context

Total downtime in last 30 days: 43.3 min / 99.9% target

Focus on availability
Error budget mechanics

A life changing approach to achieving operating balance

- SLI - Service Level Indicator
  a measurement of user value delivered
- SLO - Service Level Objective
  a threshold that makes sense in your context
- Policy - take action
  when approaching threshold

- Total downtime in last 30 days:
  43.3 min / 99.9% target
- Focus on availability
The End
Security Issues

Uptime is great!
Uptime isn't success if you're infected

- Derrick Harris https://thenewstack.io/why-wells-fargo-repaves-its-entire-platform-every-day
Availability is important.

Security is also important.
The End
Availability is important. Security is important. Feature freshness is also important.
3 important domains
A model for regulating availability...
Error budget model

Availability ✓
Security ?
Feature Freshness ?
Security is important
Vulnerability budget

How long can we afford to be vulnerable to CVEs in our dependencies?
Vulnerability budget

How long can we afford to be vulnerable to CVEs in our dependencies?

SLI - days since running dependency patch was released

26 days since JVM v1.9.13 released
Vulnerability budget

How long can we afford to be vulnerable to CVEs in our dependencies?

SLO - max age threshold in days

- SLI - days since running dependency patch released

26 days since JVM v1.9.13 released

30 days target
Vulnerability budget

How long can we afford to be vulnerable to CVEs in our dependencies?

Policy - patch when near threshold

- SLI - days since running dependency patch released
- SLO - max age threshold in days

26 days since JVM v1.9.13 released

30 days target

Apply patches
Vulnerability budget

How long can we afford to be vulnerable to CVEs in our dependencies?

- **SLI** - days since running dependency patch released
- **SLO** - max age threshold in days
- **Policy** - patch when near threshold

26 days since JVM v1.9.13 released

30 days target

Apply patches
SLO threshold?
Hypothetical Story

Names have been changed....

- Large U.S. company
- Collects *a lot* of personal and financial information
- From almost everyone
- Even if they didn’t sign up
Equifax Breach

Pretty Infamous

- 143 Million people!
- *Really* valuable personal information
- Firings/Resignations, FBI Investigation, Congressional Hearings
Equifox Breach

Pretty Avoidable

- Caused by a known issue in Apache Struts
30 days SLO would have worked for Equifox
Real world example

A Pivotal teams experience over last 6 months

- SRE team running a platform for large SaaS tenant
- Patching Ubuntu OS
  - Rebuilding & repaving
- Max age threshold SLO of 30 days
26 days since base OS image released

30 days target

Apply patches
A Pivotal teams experience over last 6 months

Team Can Broadcast Value They’ve Delivered
Feature freshness is important
Legacy budget

New enough that it gets support; and not so old that no-one wants to use it

- Which version of k8s should one run?
- Given:
  - New release ~ every 90 days
  - N-2 support policy
**Legacy budget**

New enough that it gets support; and not so old that no-one wants to use it

---

**SLI - days since feature release**

143 days since k8s v1.11.x released
Legacy budget

New enough that it gets support; and not so old that no-one wants to use it

SLO - a range that describes how bleeding edge you want to be

- SLI - days since feature release

- Time

- Days since release

143 days since k8s v1.11.x released

- v1.11 supported

- v1.12

- v1.13

- v1.14 stable features
Legacy budget

New enough that it gets support; and not so old that no-one wants to use it

- SLI - days since feature release
- SLO - a range that describes how bleeding edge you want to be

Policy - upgrade every 90 days

Time

Days since release

v1.11 v1.12 v1.13 v1.14

143 days since k8s v1.11.x released

v1.12.12 - v1.13.6

Supported

Stable features
Legacy budget

New enough that it gets support; and not so old that no-one wants to use it

- SLI - days since feature release
- SLO - a range that describes how bleeding edge you want to be
- Policy - upgrade every 90 days

![Diagram showing days since release for different versions of k8s, with v1.11.x stable features released 143 days after v1.11.x and supported until v1.12.12 - v1.13.6.](image-url)
Recap:

Budget model:

Availability
Error Budgets ✓
Security
Vulnerability
Budget ✓
Feature Freshness
Legacy Budget ✓
The Model is the same for each

A life changing approach to achieving operating balance
SLI - Service Level Indicator
- a measurement of user value delivered

The Model is the same for each

A life changing approach to achieving operating balance
The Model is the same for each

A life changing approach to achieving operating balance

- SLI - Service Level Indicator - a measurement of user **value** delivered

**SLO - Service Level Objective**

a threshold that makes sense in your context
The Model is the same for each

A life changing approach to achieving operating balance

- SLI - Service Level Indicator
  a measurement of user value delivered
- SLO - Service Level Objective
  a threshold that makes sense in your context

Policy - take action when approaching threshold

Take action when near threshold
The Model is the same for each

A life changing approach to achieving operating balance

- **SLI** - Service Level Indicator
  - A measurement of user value delivered
- **SLO** - Service Level Objective
  - A threshold that makes sense in your context
- **Policy** - take action when approaching threshold
The End
The End
(of this talk at least)
Further conversation & Q&A
#srecon_americas