

Beyond the Backend: Automated End-to-End Performance Testing for Any Platform

Bugs a'loud 2025

About me

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Lukas Makaravičius

- Senior QA Engineer
- 6+ years in QA
- Mainly focused around native clients

 makaravicius@protonmail.com

 linkedin.com/in/lukas-makaravicius/



What to expect

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

- Explain why this project was implemented
- Provide real world examples on what issues it solved
- Explain how to implement automated E2E performance tests
- Explain how to setup process for using performance results

Context

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

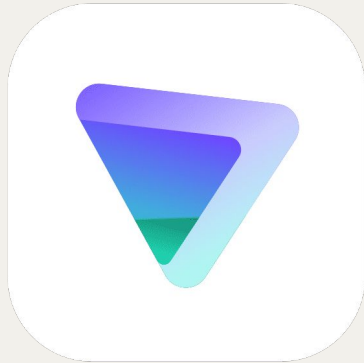
11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



ProtonVPN

Context

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

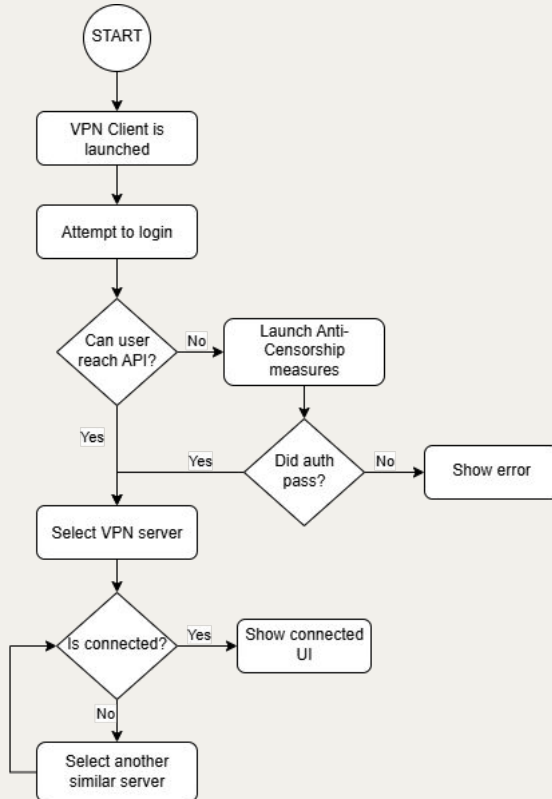
11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Main components:

- VPN Server
- Guest Session VPN server
- Backend
- Alternative Backend
- VPN Client

Context

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

E2E performance tests are important:

- Validates full E2E paths
- Detects issues across the entire stack
- Helps to prevent performance issues early

Analytics are important:

- Monitors how application behaves in real world
- Dataset is huge

Limitations

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

When E2E performance tests are executed manually it is:

- Costly
- Performed infrequently
- Hard to see trends

When using analytics:

- Artifacts are limited
- Context is limited
- Issues are caught only after the release

Idea

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Let's measure performance of specific flows in E2E tests

Concept

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

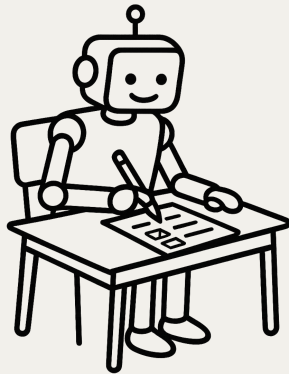
11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Automated tests



Store
performance
results



Visualize it

Setup

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Automated tests



Metrics/Logs
Storage



Visualization

Examples

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

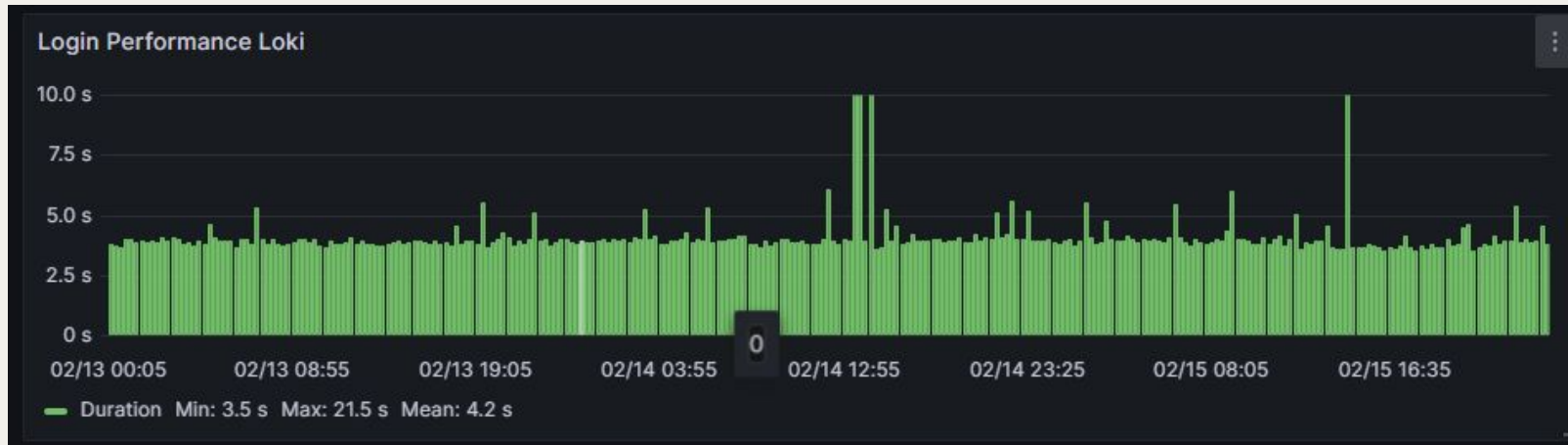
11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Examples

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

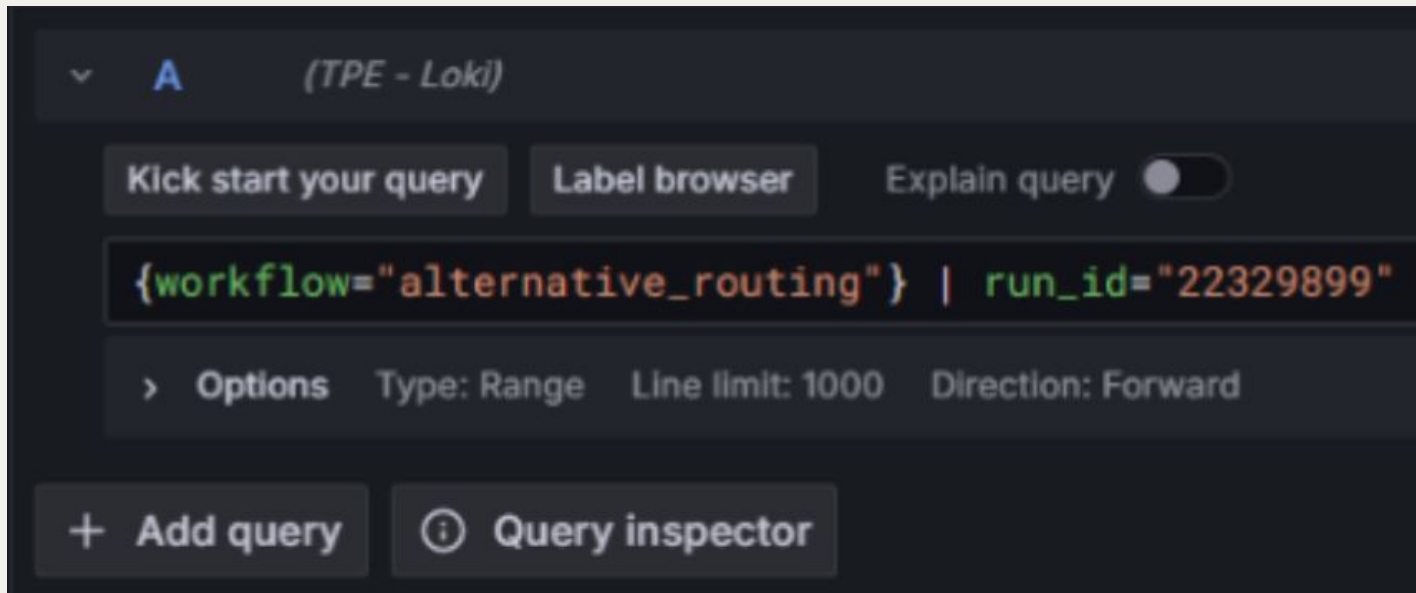
11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Examples

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

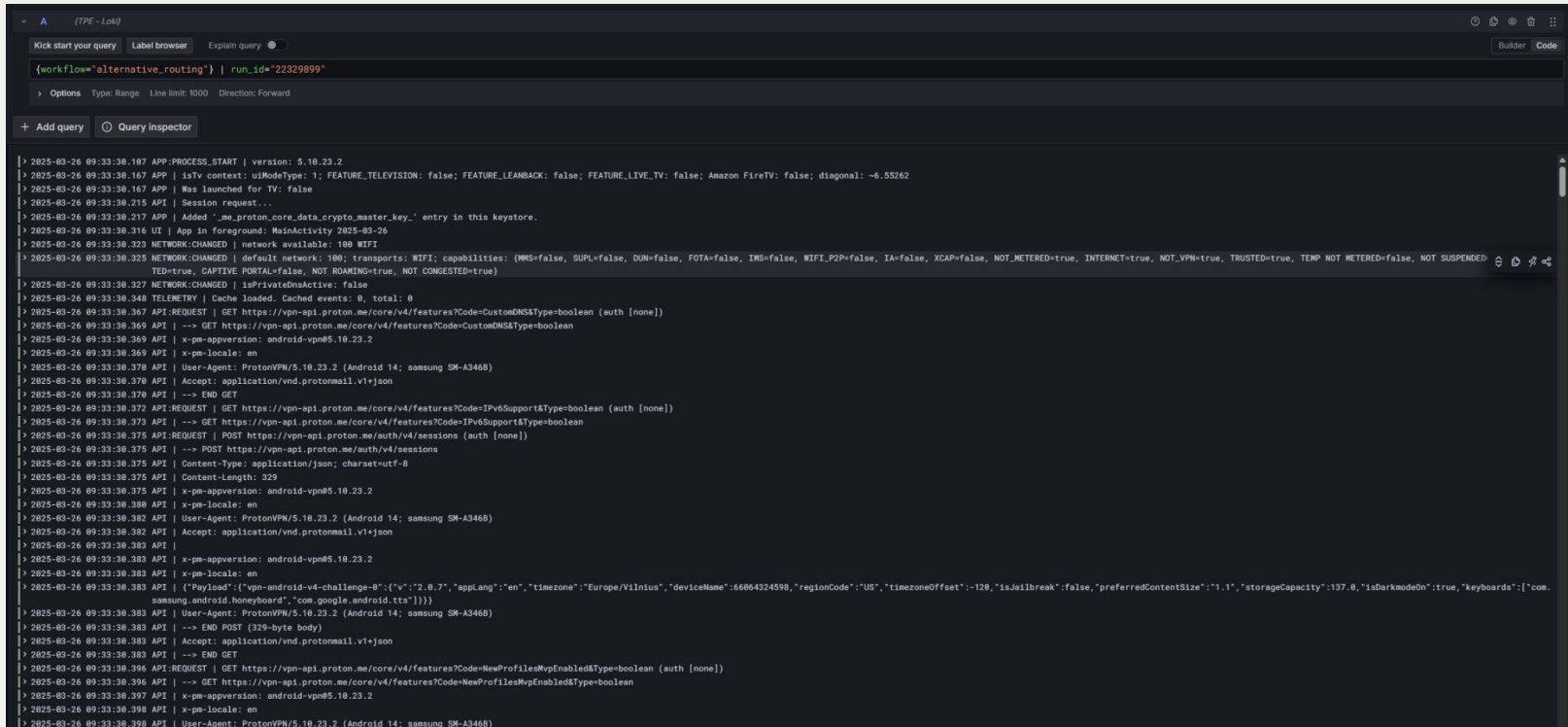
11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



The screenshot shows the TPE - Loki web interface. At the top, there's a navigation bar with 'TPE - Loki' and a search icon. Below it, a query input field contains the query: `{workflow="alternative_routing"} | run_id="22329899"`. To the right of the query input are buttons for 'Builder' and 'Code'. Below the query input, there's a section for 'Options' with 'Type: Range', 'Line limit: 1000', and 'Direction: Forward'. The main content area is divided into two tabs: '+ Add query' and 'Query inspector'. The 'Query inspector' tab is active, showing a list of log entries. The log entries are in a table format with columns for time, host, and message. The messages are log messages from an Android device, showing various system events and network activity. The log entries are sorted by time, with the most recent at the top.

```
> 2025-03-26 09:33:30.187 APP:PROCESS_START | version: 5.10.23.2
> 2025-03-26 09:33:30.167 APP | isTv context: uiModeType: 1; FEATURE_TELEVISION: false; FEATURE_LEANBACK: false; FEATURE_LIVE_TV: false; Amazon FireTV: false; diagonal: ~6.55262
> 2025-03-26 09:33:30.167 APP | Was launched for TV: false
> 2025-03-26 09:33:30.215 API | Session request...
> 2025-03-26 09:33:30.217 APP | Added "_me_proton_core_data_crypto_master_key_" entry in this keystore.
> 2025-03-26 09:33:30.316 UI | App in foreground: MainActivity 2025-03-26
> 2025-03-26 09:33:30.323 NETWORK:CHANGED | network available: 100 WIFI
> 2025-03-26 09:33:30.325 NETWORK:CHANGED | default network: 100; transports: WIFI; capabilities: (MMS=false, SUPL=false, DUN=false, FOTA=false, IMS=false, WIFI_P2P=false, IA=false, XCAP=false, NOT_METERED=true, INTERNET=true, NOT_VPN=true, TRUSTED=true, TEMP_NOT_METERED=false, NOT_SUSPENDED
  TED=true, CAPTIVE_PORTAL=false, NOT_ROAMING=true, NOT_CONGESTED=true)
> 2025-03-26 09:33:30.327 NETWORK:CHANGED | isPrivateDnsActive: false
> 2025-03-26 09:33:30.348 TELEMETRY | Cache loaded. Cached events: 0, total: 0
> 2025-03-26 09:33:30.367 API:REQUEST | GET https://vpn-api.proton.me/core/v4/features?Code=CustomDNS&Type=boolean (auth [none])
> 2025-03-26 09:33:30.369 API | --> GET https://vpn-api.proton.me/core/v4/features?Code=CustomDNS&Type=boolean
> 2025-03-26 09:33:30.369 API | x-pm-appversion: android-vpn5.10.23.2
> 2025-03-26 09:33:30.369 API | x-pm-locale: en
> 2025-03-26 09:33:30.370 API | User-Agent: ProtonVPN/5.10.23.2 (Android 14; samsung SM-A346B)
> 2025-03-26 09:33:30.370 API | Accept: application/vnd.protonmail.v1+json
> 2025-03-26 09:33:30.370 API | --> END GET
> 2025-03-26 09:33:30.372 API:REQUEST | GET https://vpn-api.proton.me/core/v4/features?Code=IPv6Support&Type=boolean (auth [none])
> 2025-03-26 09:33:30.373 API | --> GET https://vpn-api.proton.me/core/v4/features?Code=IPv6Support&Type=boolean
> 2025-03-26 09:33:30.375 API:REQUEST | POST https://vpn-api.proton.me/auth/v4/sessions (auth [none])
> 2025-03-26 09:33:30.375 API | --> POST https://vpn-api.proton.me/auth/v4/sessions
> 2025-03-26 09:33:30.375 API | Content-Type: application/json; charset=utf-8
> 2025-03-26 09:33:30.375 API | Content-Length: 329
> 2025-03-26 09:33:30.375 API | x-pm-appversion: android-vpn5.10.23.2
> 2025-03-26 09:33:30.380 API | x-pm-locale: en
> 2025-03-26 09:33:30.382 API | User-Agent: ProtonVPN/5.10.23.2 (Android 14; samsung SM-A346B)
> 2025-03-26 09:33:30.382 API | Accept: application/vnd.protonmail.v1+json
> 2025-03-26 09:33:30.383 API |
> 2025-03-26 09:33:30.383 API | x-pm-appversion: android-vpn5.10.23.2
> 2025-03-26 09:33:30.383 API | x-pm-locale: en
> 2025-03-26 09:33:30.383 API | ("Payload":{"vpn-android-v4-challenge-0":{"v":"2.0.7","appLang":"en","timezone":"Europe/Vilnius","deviceName":"66864324598","regionCode":"US","timezoneOffset":-120,"isJailbreak":false,"preferredContentSize":"1.1","storageCapacity":137.0,"isDarkmodeOn":true,"keyboards":["com.
  samsung.android.honeyboard","com.google.android.tts"]}})
> 2025-03-26 09:33:30.383 API | User-Agent: ProtonVPN/5.10.23.2 (Android 14; samsung SM-A346B)
> 2025-03-26 09:33:30.383 API | --> END POST (320-byte body)
> 2025-03-26 09:33:30.383 API | Accept: application/vnd.protonmail.v1+json
> 2025-03-26 09:33:30.383 API | --> END GET
> 2025-03-26 09:33:30.396 API:REQUEST | GET https://vpn-api.proton.me/core/v4/features?Code=NewProfilesMvpEnabled&Type=boolean (auth [none])
> 2025-03-26 09:33:30.396 API | --> GET https://vpn-api.proton.me/core/v4/features?Code=NewProfilesMvpEnabled&Type=boolean
> 2025-03-26 09:33:30.397 API | x-pm-appversion: android-vpn5.10.23.2
> 2025-03-26 09:33:30.398 API | x-pm-locale: en
> 2025-03-26 09:33:30.398 API | User-Agent: ProtonVPN/5.10.23.2 (Android 14; samsung SM-A346B)
```

Examples

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

DoH Login Success rate [> 98%]

99.6%

Impact

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



Impact

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions



It also helped...

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

- Continuously test new features
- Provide concrete evidence that some flows performance should be improved
- Involve whole team into performance testing process

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

How was this achieved?

Automated tests

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

```
[Test]
```

```
🚩 | 0 references
```

```
public void QuickConnectPerformance()
```

```
{
```

```
    LoginRobot.Login(TestUserData.PlusUser);
```

```
    NavigationRobot.Verify.IsOnMainPage();
```

```
    HomeRobot
```

```
        .ConnectViaConnectionCard()
```

```
        .Verify.IsConnected();
```

```
    HomeRobot.Disconnect();
```

```
}
```

Automated tests

- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI
- 10 Usage
- 11 SLI
- 12 SLO
- 13 Error budget
- 14 SLA
- 15 Conclusions

```
[Test]
[Duration, TestStatus]
[Sli("quick_connect")]
0 | 0 references
public void QuickConnectPerformance()
{
    LoginRobot.Login(TestUserData.PlusUser);
    NavigationRobot.Verify.IsOnMainPage();

    HomeRobot.ConnectViaConnectionCard();

    SliHelper.MeasureTime(() =>
    {
        HomeRobot.Verify.IsConnected();
    });

    HomeRobot.Disconnect();
}
```

Automated tests

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Pros:

- Any automation framework can be used
- Already existing tests can be reused
- Real actions are being performed

Cons:

- Results can be polluted by flaky tests
- The volume of generated metrics remains limited (~48 events in a day).

Metrics/Logs storage

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Why I recommend using **Grafana Loki**:

- It can store both metrics **and logs**
- Can handle large volume of logs
- It can be Integrated with Grafana easily
- LogQL is easy to use and is similar to PromQL



Metadata

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Labels:

- Measurement name
- Build type
- Platform
- Device model
- Product

Structured metadata:

- Run ID
- App version
- Build ID
- OS Version

Visualization

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Success rate



Visualization

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Time series line chart



CI Setup

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Setup has to be:
Predictable

To make it predictable:

- On-premise CI runners
- Real devices
- Release build
- Scheduled job

Build Storage

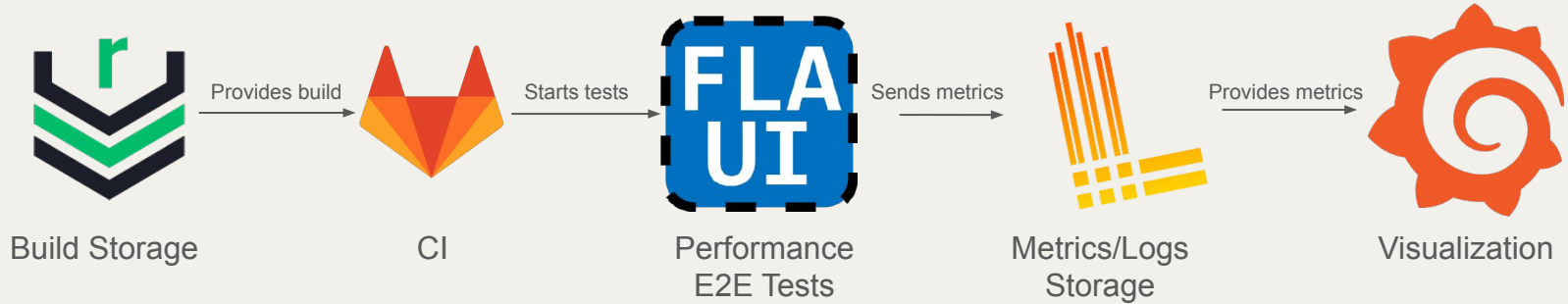
- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI**
- 10 Usage
- 11 SLI
- 12 SLO
- 13 Error budget
- 14 SLA
- 15 Conclusions



sonatype
nexus repository

What we have so far...

- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI**
- 10 Usage
- 11 SLI
- 12 SLO
- 13 Error budget
- 14 SLA
- 15 Conclusions



01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

How to use collected results?

Service Level Management

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

It consists of:

- Agreeing on expectations
- Monitoring the performance
- Taking action

Service Level Indicator (SLI)

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

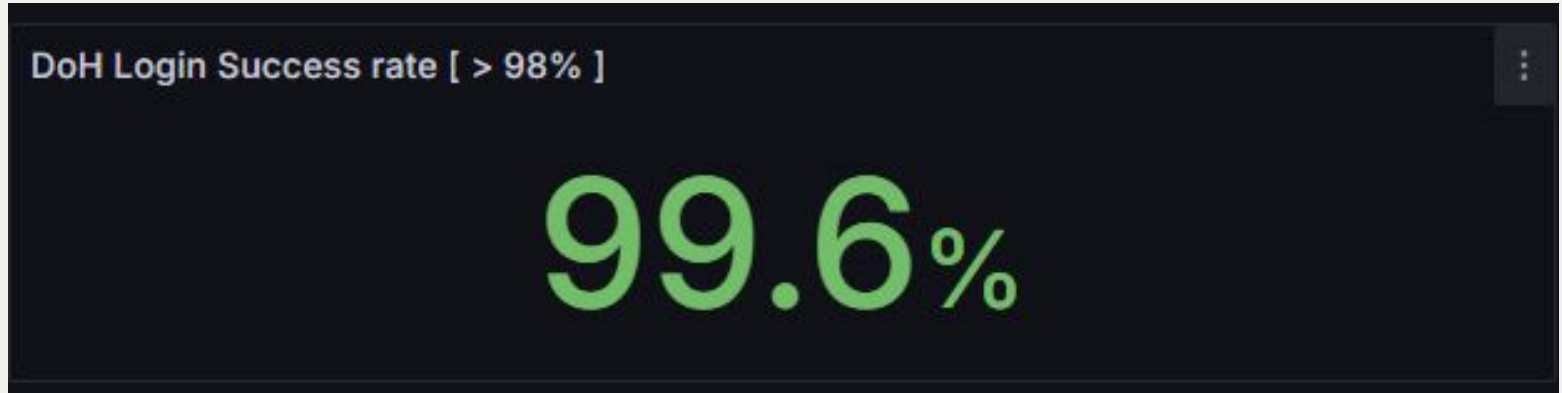
13 Error budget

14 SLA

15 Conclusions

SLI:

Any metric that can be summarized into one number



Service Level Indicator (SLI)

- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI
- 10 Usage
- 11 SLI**
- 12 SLO
- 13 Error budget
- 14 SLA
- 15 Conclusions

SLI:
Any metric that can be summarized into one number

Success rate	Average Values	Median
$SLI = \left(\frac{\text{good events}}{\text{bad events}} \right) \times 100$	$SLI = \frac{SLI_1 + \dots + SLI_n}{n}$	$SLI = [4, 8, \mathbf{10}, 15, 20]$

Service Level Objective (SLO)

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

SLO:

A goal that should not be breached



Service Level Management

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

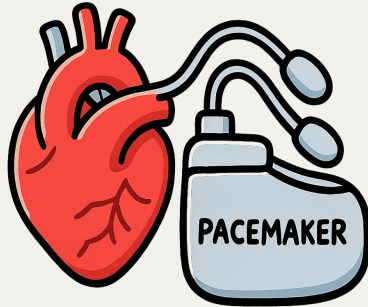
14 SLA

15 Conclusions

Keep in mind

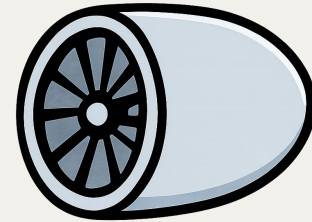
It's impossible to be 100% reliable

HEART PACEMAKER



1% fail in 5 years

JET ENGINE



**1 failure in 300,000
hours**

Service Level Objective (SLO)

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

How to setup meaningful objectives?

- It's impractical/impossible to achieve 100% stability
- Every additional 9 might seem like a small difference, but in reality it's costly
- The less data you have the more relaxed goals should be used
- Be flexible

SLO	Yearly Error Budget
99%	3.65 days
99.9%	8.76 hours
99.99%	52.56 minutes
99.999%	5.2 minutes
99.9999%	31.5 seconds

Error Budget

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

Error budget:

An amount of failure a service is allowed to have

Calculation:

Error Budget = 100% - SLO Target

Error Budget(1Y) = 100% - 99% = 1% = 3.65 days

Error budget

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

When used, it will allow you to:

- Make informed decisions about release
- Understand where to invest development effort

How to visualize error budget

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

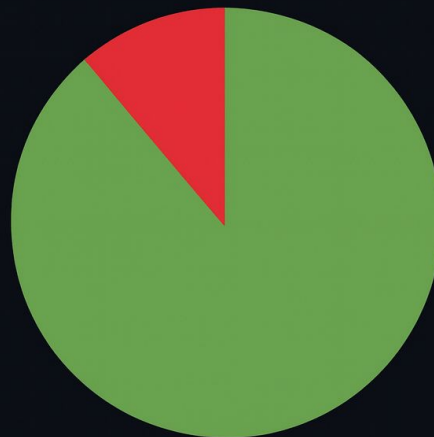
12 SLO

13 Error budget

14 SLA

15 Conclusions

Windows Alternative routing error budget



RemainingErrorBudget

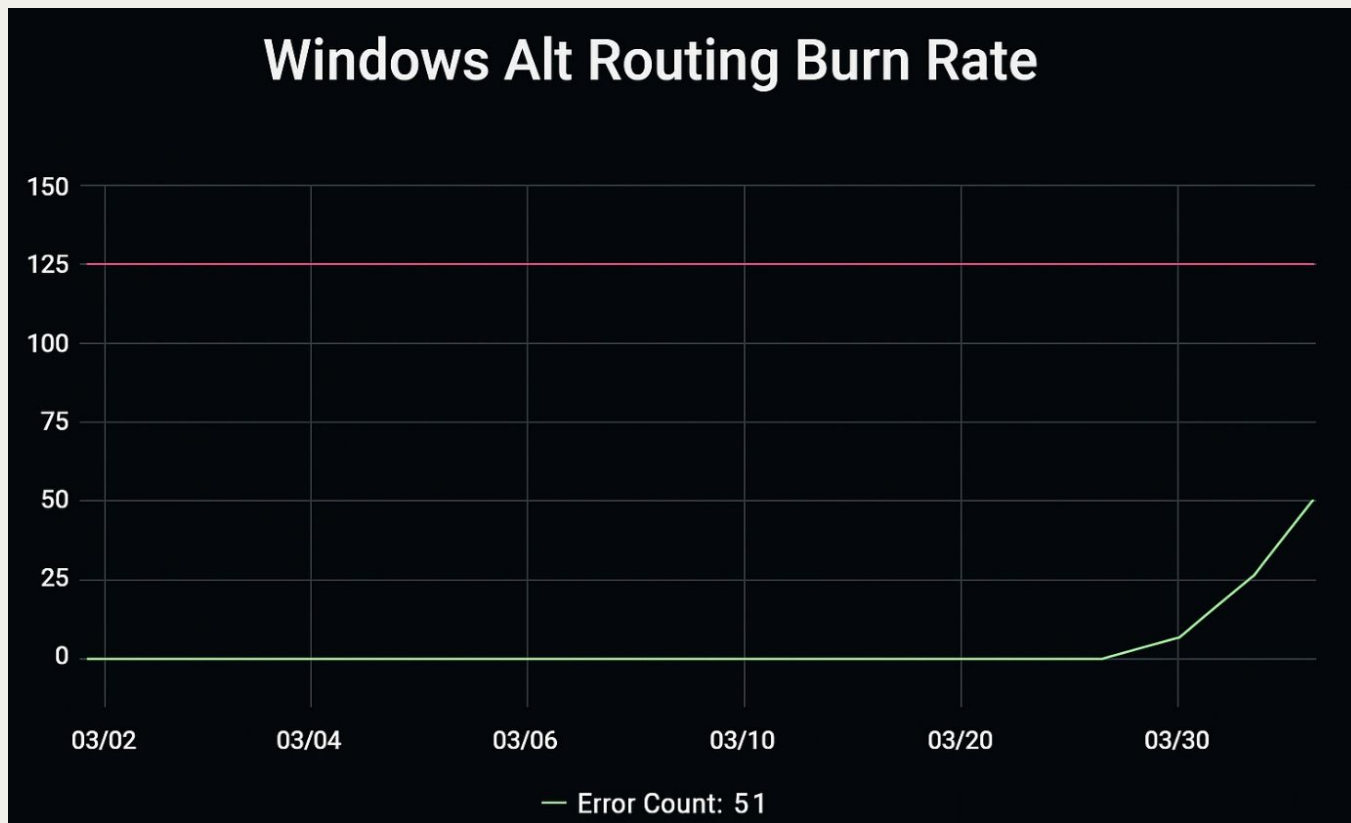
25.8

WindowsAltRoutingFailedLogins

3

How to visualize error budget

- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI
- 10 Usage
- 11 SLI
- 12 SLO
- 13 Error budget**
- 14 SLA
- 15 Conclusions



Service Level Agreement (SLA)

- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI
- 10 Usage
- 11 SLI
- 12 SLO
- 13 Error budget
- 14 SLA**
- 15 Conclusions

SLA

A contract between a service provider and customer



Tips

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

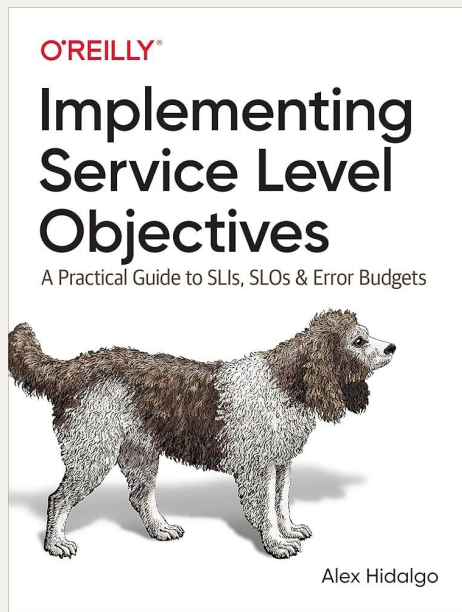
14 SLA

15 Conclusions

1. Service level management is all about the people
2. For starters own the process
3. Don't forget to log issues that this setup has caught
4. It takes patience for this process to fully kick in
5. Measure only critical business flows
6. Set review periods

Literature

- 01 Goals
- 02 Context
- 03 Concept
- 04 Examples
- 05 Impact
- 06 Tests
- 07 Storage
- 08 Visualization
- 09 CI
- 10 Usage
- 11 SLI
- 12 SLO
- 13 Error budget
- 14 SLA**
- 15 Conclusions



[Implementing Service Level Objectives: A Practical Guide to SLIs, SLOs, and Error Budgets](#)

Conclusions

01 Goals

02 Context

03 Concept

04 Examples

05 Impact

06 Tests

07 Storage

08 Visualization

09 CI

10 Usage

11 SLI

12 SLO

13 Error budget

14 SLA

15 Conclusions

When E2E automated performance tests are created you will:

- Expand your monitoring stack
- Easily debug anomalies inside the whole stack. Especially helps to catch race conditions
- Prevent performance issues early
- Continuously test specific product features

When Service Level Management process is introduced:

- Encourage transparency and discussions inside your team
- Understand what “Good enough” means for your customer
- Improve velocity of releases
- Increase trust of the customer

Thank you!

Agenda

01 Speaker

02 Goals

03 Topic 1

04 Topic 1

05 Topic 1

06 Topic 1

07 Topic 1

08 Topic 1

09 Topic 1



makaravicius@protonmail.com



[linkedin.com/in/lukas-makaravicius/](https://www.linkedin.com/in/lukas-makaravicius/)

