Abstract

BACKGROUND: The number of users of web services is increasing annually and, therefore, all developers aim to develop a high reliable software. However, no system can withstand any load, so site reliability engineers prepare for potential service failures by implementing various reliability practices to minimize associated risks. Feedback control system is a system that can give feedback about its current status so that engineers can react to it in advance the risks of a fall.

OBJECTIVE: However, for feedback control systems developers face a problem that each reliability practice is configured in its own way and combinations of these settings can give different results in the reliability of the system. This paper outlines a web-service for reproducible load testing, visualization, and configuration of feedback control systems

METHODS: The web-services was developed by using Kotlin and Spring Boot framework, and Kafka broker message was used for communication. Yandex tank was chosen as a high load generator for the ability to use several cores for load generation and a convenient API. Additionally, by using Angular framework I developed a UI interface for a more convenient and visual use of the service.

RESULTS: The proposed service can handled services for configuration, executed load testing scenarios and provided real-time results as graphs.

CONTRIBUTION AND APPLICABILITY: This service can be used in testing systems under various load scenarios with different configuration variants. These practices help to find the most effective combination of parameters that ensures optimal system performance.

Chapter 1

Literature Review

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

This is the second paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should

match the language.

I Another Section

And after the second paragraph follows the third paragraph. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

THIS IS ASS

TABLE I
Simulation Parameters

A	B
Parameter	Value
Number of vehicles	$\mid \mathcal{V} \mid$
Number of RSUs	$\mid \mathcal{U} \mid$
RSU coverage radius	150 m
V2V communication radius	30 m
Smart vehicle antenna height	1.5 m
RSU antenna height	25 m
Smart vehicle maximum speed	v_{max} m/s
Smart vehicle minimum speed	v_{min} m/s
Common smart vehicle cache capacities	$[50, 100, 150, 200, 250]$ mb
Common RSU cache capacities	$[5000, 1000, 1500, 2000, 2500]$ mb
Common backhaul rates	$[75, 100, 150]$ mb/s

Fig. 1. One kernel at x_s (dotted kernel) or two kernels at x_i and x_j (left and right) lead to the same summed estimate at x_s. This shows a figure consisting of different types of lines. Elements of the figure described in the caption should be set in italics, in parentheses, as shown in this sample caption.

This description implies several essential properties of the task at hand:

1. Watermark must contain all necessary information, but still, be placeable and recognizable even on smaller images. The produced watermark must be compact but have the possibility to store enough information.

2. To prevent easy tampering, the watermark must be invisible to the naked eye (and, preferably, to basic image parsing tools). If malefactor does not know about the existence of watermark, they might not even try to remove it and disable it.