Métricas de qualidades e mitigação dos riscos

1 - Métricas de Qualidade

As métricas de qualidade são fundamentais para avaliar o desempenho e a confiabilidade de um sistema, garantindo que os serviços funcionem adequadamente. No SmartCity Hub, concentramos em avaliar o tempo de resposta, a taxa de falhas na ingestão de eventos, a disponibilidade do sistema e a precisão dos alertas.

1.1 - Tempo de Resposta

O requisito de latência máxima para o tráfego deve ser de 200 ms. Nos códigos desenvolvidos, essa métrica pode ser avaliada utilizando funções de temporização, como o chrono em C++, aplicadas no ciclo de atualização dos semáforos. Dessa forma, é possível identificar gargalos e validar a capacidade do sistema de responder de forma ágil em situações críticas.

1.2 - Taxa de Falhas na Ingestão de Eventos

A ingestão confiável demilhares de eventos IoT por minuto é um desafio técnico. A taxa de falhas representa a proporção de eventos perdidos ou processados incorretamente durante a coleta e transmissão de dados. Para simular cenários de alta demanda, foi utilizado o programa load_generator.cpp, que permite gerar grandes volumes de eventos de teste e mensurar a eficiência da infraestrutura de ingestão, identificando possíveis pontos de sobrecarga.

1.3 - Disponibilidade do Sistema

A disponibilidade de serviços críticos, como semáforos inteligentes e monitoramento de segurança, deve atingir níveis extremamente elevados, com meta mínima de 99,999%. Essa métrica envolve não apenas a estabilidade do software, mas também a redundância de componentes e a capacidade de detecção e recuperação de falhas.
O módulo de semáforos, documentado em README_semaforos.txt, exemplifica um subsistema que precisa operar de forma contínua. Nessa implementação, a adição de mecanismos de heartbeat e fallback local permite detectar falhas e manter operação mínima mesmo na ausência de conexão com o controlador central.

1.4 - Precisão dos Alertas

A precisão é essencial para evitar falsos positivos ou negativos em eventos que impactam diretamente a população. Um alerta incorreto pode gerar desperdício de recursos ou comprometer a confiança dos usuários. Nos módulos de coleta de resíduos, por exemplo, sensores simulados em lixeiras.txt indicam o estado das lixeiras inteligentes. Testes controlados com esses dados permitem verificar se os alertas emitidos correspondem corretamente às situações reais simuladas.

2 - Riscos Críticos

A implementação de uma plataforma urbana integrada envolve riscos significativos, tanto de natureza técnica quanto de segurança da informação. Entre os principais riscos identificados no SmartCity Hub, destacam-se o vazamento de dados sensíveis, falhas em semáforos inteligentes, ataques cibernéticos a dispositivos IoT e sobrecarga em eventos de grande porte.

2.1 - Vazamento de Dados Pessoais e de Geolocalização

A plataforma lida com dados de cidadãos e sensores distribuídos, o que inclui informações potencialmente sensíveis, como localização geográfica. O armazenamento inadequado desses dados pode configurar violação da Lei Geral de Proteção de Dados, portanto, exige grande segurança.
No módulo de reclamações (README_reclamacoes.md), por exemplo, a gravação de informações em arquivos de texto representaria um risco caso houvesse identificadores pessoais não anonimizados.

2.2 - Falhas em Semáforos Inteligentes

Falhas de sincronização ou perda de comunicação no controle de semáforos podem resultar em estados conflitantes, como dois sentidos em verde simultaneamente, aumentando o risco de acidentes de trânsito. Portanto, serão implementados mecanismos de redundância e estratégias de operação autônoma, permitindo que os semáforos mantenham um funcionamento seguro mesmo em caso de falhas de comunicação ou sincronização. Assim, a segurança e confiabilidade do sistema é garantida.

2.3 - Ataques Cibernéticos a Dispositivos IoT

Dispositivos IoT expostos sem autenticação ou criptografia podem ser comprometidos, permitindo a manipulação de dados ou a interrupção de serviços essenciais, como iluminação pública e controle de tráfego. Assim, os arquivos .txt utilizados para armazenar dados sensíveis nos protótipos serão criptografados, garantindo a confidencialidade e integridade das informações. Também serão adotadas técnicas de autenticação entre dispositivos, assegurando que apenas fontes confiáveis possam enviar ou receber dados. Essas medidas reduzem significativamente a superfície de ataque do sistema e fortalecem a proteção contra invasões ou modificações indevidas.

2.4 - Sobrecarga em Grandes Eventos

Eventos de grande porte, como manifestações ou shows, têm o potencial de gerar picos expressivos e repentinos no fluxo de dados processados, excedendo a capacidade operacional planejada. Tal sobrecarga causa demoras, perdas de dados ou suspensão momentânea de serviços cruciais. Visando prever estas situações, foram efetuados testes de carga com o gerador, o que possibilitará estimar os limites concretos da infraestrutura e alocar os recursos de forma apropriada. Assim, torna-se viável implementar táticas como o balanceamento dinâmico da carga, garantindo a preservação da qualidade do serviço mesmo durante momentos de alta demanda.

3 - Estratégias de Mitigação

Com base na análise das métricas e riscos, foram definidas estratégias para aumentar a segurança, a confiabilidade e a conformidade legal do sistema. Essas estratégias combinam medidas técnicas e operacionais.

3.1 - Criptografia Ponta a Ponta

Todos os dados de sensores e cidadãos e arquivos de texto que contenham dados pessoais, como o Registro_usuarios.txt, devem ser criptografados, e o acesso a essas informações precisa ser restrito e autenticado, a fim de impedir que informações sensíveis sejam acessadas indevidamente.

3.2 - Redundância e Fallback Seguro

Para garantir disponibilidade elevada, componentes críticos devem possuir mecanismos de redundância e capacidade de operação autônoma em caso de falhas de comunicação. O controle de semáforos é um exemplo em que fallback local é imprescindível.

3.3 - Simulações de Carga

A realização periódica de testes de estresse permite identificar gargalos e avaliar a capacidade de resposta do sistema diante de cenários extremos. O uso do load_generator.cpp exemplifica uma abordagem prática de alta carga para esse tipo de teste.

3.4 - Firewalls e Monitoramento de Rede

Para diminuir as áreas vulneráveis e assegurar a identificação imediata de atividades incomuns, é fundamental usar firewalls e mecanismos de identificação e prevenção de invasões. Além disso, a validação recíproca entre aparelhos IoT aumenta a proteção nas trocas de dados entre as partes espalhadas do sistema.
É crucial que os serviços de registro e validação de usuários atuem em locais isolados e seguros, com restrições de acesso severas. A utilização da autenticação de múltiplos fatores (MFA), principalmente para usuários com permissões administrativas, oferece uma proteção extra, atenuando o perigo de acesso indevido a dados e operações delicadas.

4 - Funcionamento geral e segurança do sistema

Os módulos de sensores (qualidade do ar, ruído e consumo energético) geram dados contínuos que podem ser usados para avaliar ingestão, armazenamento e tratamento de grandes volumes de dados.

O módulo de semáforos fornece um caso de uso crítico, onde latência, disponibilidade e mecanismos de fallback seguro são determinantes para manter o fluxo urbano e evitar falhas operacionais.

O portal de reclamações exemplifica a importância de proteção de dados pessoais e autenticação de usuários, além de destacar a necessidade de respostas rápidas e rastreabilidade nas comunicações entre cidadão e gestão pública.

O gerador de carga permite simular cenários extremos e mensurar métricas de qualidade, como tempo de resposta e tolerância a falhas, de forma objetiva e controlada.

O sistema de cadastro de usuários complementa o ecossistema de protótipos ao introduzir a camada de identificação, autenticação e controle de acesso. Ele permite diferenciar perfis de Cidadãos (Tipo C) e Gestores (Tipo G), garantindo permissões distintas conforme o papel de cada um na plataforma.

O sistema valida CPFs previne duplicações e realiza autenticação por senha, armazenando as informações em um arquivo estruturado (Registro_usuarios.txt), que funciona como um banco de dados local simplificado. Além disso, oferece mecanismos de recuperação de senha e checagem de credenciais, possibilitando testar fluxos de login e segurança em integração com outros módulos.