Q4: You are required to produce an ideal postmortem/root cause analysis report for the problem reported below. Create a document with the relevant structure and content and store your file in a shared location where Crossover team can access and download it for evaluation. And add your sharable link in the answer field of this question.

Problem Statement :

A Jira issue (AESEDI-53447) was logged that the customer data was not sent from AES EDI. 486,000 records were affected. The investigation showed that the file with the data was sent, but it did not get processed due to an issue with the AES CIS service (Jira Issue No: AESCIS-38263) at that point of time. The same issue affected EDI to CIS monitoring service working on the CIS side. So the missed records were not discovered automatically. The file was resent and the issue got resolved. The missing records were discovered at 11:56 AM, processed at 1:00 PM.

Date: 03/08/2020

Author: Naresh

Status : Resolved

Problem statement: The customer data was not sent from AES EDI. The investigation showed that the file with the data was sent, but it did not get processed due to an issue with the AES CIS service.

Analysis: EDI to CIS monitoring service got affected, the issue is with the CIS service. CIS service either Busy not accepting the requests or CIS service is down.

Resolution: once the CIS service re starts and the files resent then the issue got resolved.

Action items:

1) Implement Queuing mechanism at the service side schedule for every one sec.

2) Health Monitoring of CIS service run every one sec

3) Health Monitoring of EDI to CIS data ingestion service run every one sec.

Prepare dashboard of all the above monitoring services and raise alerts if the monitoring services got affected.

4) Implement stress test of CIS service and come up with through put rate of CIS service.
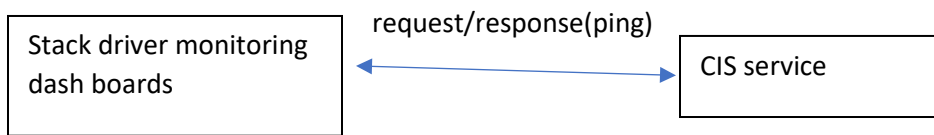
Time lines:

I Implemented stack driver monitoring and Dashboards for Go Micro services.

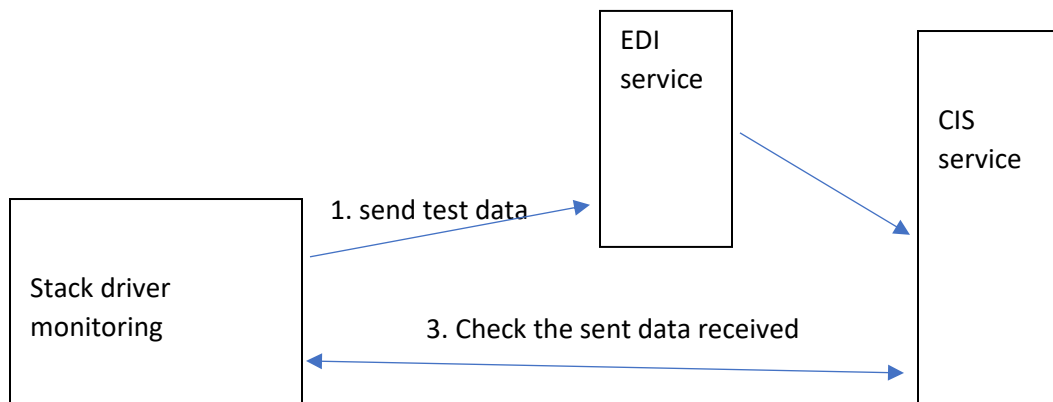1) Health Monitoring of CIS service

        Request and check response

        Stack driver monitoring and maintain SLI (schedule health check every one sec)

        and SLO (98%)

request/response(ping)

Stack driver monitoring dash boards	←→	CIS service

2) Monitoring of Data ingestion from EDI to CIS service

```
                                    EDI
                                    service          CIS
                                                     service

 Stack driver        1. send test data →
 monitoring

                     3. Check the sent data received
                     ←→
```

3) Implement Queuing mechanism in between EDI service and CIS service

```
                        PUB/ SUB
    EDI    →                          →    CIS
                        Publisher/Subscriber      service
```