Process Historian with Cassandra: 24 hour challenge

***Abstract*—Process historian is a time-series database that stores readings from, for example, SCADA devices, IoT sensors, and the like. Such database should be scalable, durable (should have certain resilience to node failures) and support fast write operations. In this short document we describe our 24 hour challange in building such database using Cassandra NoSQL, masterless database. We use Python Flask as a IoT facing web server. The web server implements simple REST API for the intergration with IoT devices.**

I. Introduction

Process Historian is a must in modern IoT applications: such database is used for storing the time-series readings from various sensors. Process Historian should hold the following properties: (i) it should be durable to node failures, (ii) it should be scalable horizontally, (iii) it should guarantee fast writes to the disk, (iv) it should have clean design.

In what follows we present MVP for such database that uses Cassandara NoSQL database and Python Flask user facing web service. The solution is a 24 hours challenge and the source codes can be found in [1].

In Figure 1 we show rather abstract architecture of our deployment. Thus in the setup we had 4 nodes deployed in the DigitalOcean cloud: (i) 3 nodes for Cassandra cluster; (ii) MySQL, Nginx, and single REST API server were deployed on single computing node; (iii) we had multiple data generators running on local machine.

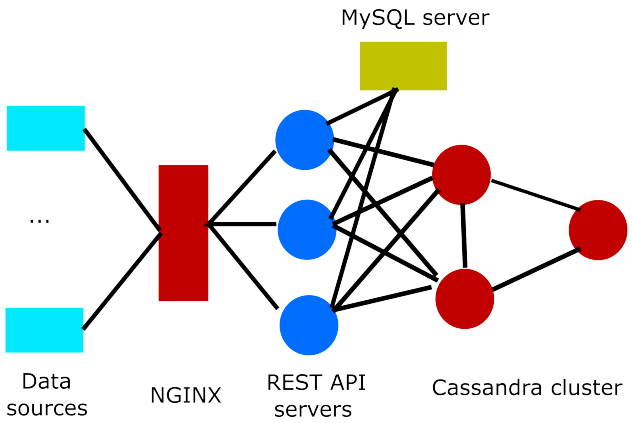

Fig. 1: Process Historian Architecture

II. Data collection methodology

III. Data processing and basic results

IV. Conclusions

In this short paper we have played a bit with the packets which were captured in a small enterprise. Our primary goal was to analyse the interactions of computers and build the

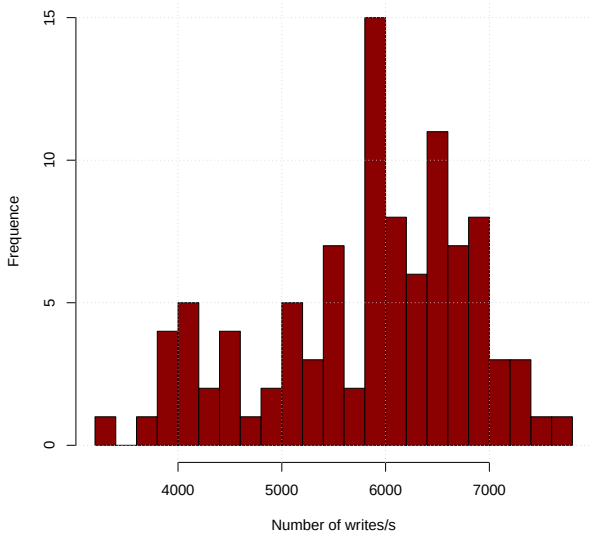

Fig. 2: Number of datapoints written per second

statistics for the traffic which was captured for several hours. Our key findings are the following: (i) major traffic in the network is HTTPS and HTTP, (ii) antivirus solutions consume considerable amount of bandwidth, (iii) computers in the network mainly interact with the default gateway and few servers, such as NFS server and mail server.

References

[1] D. Kuptsov. Simple Process Historian database. https://github.com/dmitriykuptsov/process-historian-cassandra.

TABLE I: Distribution of non IPv4 frames

Protocol	**Number of frames**	**Fraction (%)**
ARP	136857	48.8
Cisco Shared Spanning Tree Protocol	119385	42.5
IPv6	17034	6.1
Spanning Tree Protocol (IEEE 802.1D)	4774	1.7
IPX	1164	0.4
Cisco Loop	957	0.3
Cisco CDP/VTP	510	0.2