Requirements Elicitation

Introduction

Requirements elicitation is the practice of researching, gathering and analysing requirements from key stakeholders of a project. This step is crucial to the overall success of the project and must be a thorough process. Our team used two primary methods of requirements elicitation:

1. Background Research
2. Stakeholder Interview

Background Research

To prepare ourselves for the stakeholder interviews, and to add context to our knowledge of the project, our team conducted in-depth background research into the domain of data normalisation. As part of our background research into the context of the project, we used key information initially provided as part of the project summary. The project exists in the sector of cybersecurity monitoring, and requires some prior knowledge of cybersecurity monitoring tools, ETL (Extract, Transform, Load) tools, data streaming tools and data normalisation techniques in industry.

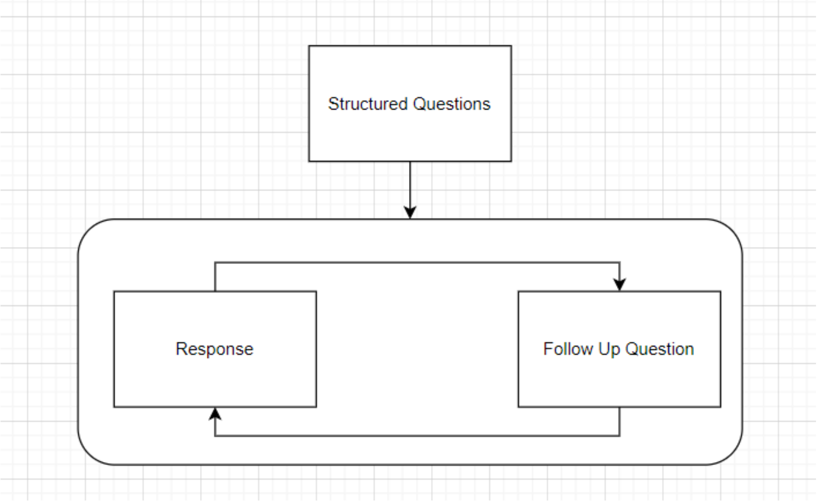
We began this by investigating the two core tools specified in the project summary, Splunk and Elastic. We investigated the purpose and capabilities of these tools, along with the current data normalisation approaches used by users of these tools. We also investigated popular log streaming platforms currently available in the market, like Cribl and Splunk Data Stream processor, with a focus on their capabilities to normalise and stream data to multiple locations. We also read a few publications and articles related to data normalisation to give us context on how these problems are currently being tackled.

Interview Strategy:

For our interview, we wanted to make sure we interviewed stakeholders across the business and technical side of the project, as this would give us both the technology and business context of the proposed project. Our strategy for the interview was simple. We prepared a set of questions to which the entire engineering team contributed. Our structured questions were kept open-ended, to ensure they would stimulate discussion and details from the client. Once consolidated, these were then split into three major areas:

1. Project Overview
2. Understanding System-As-Is
3. Understanding System-To-Be

The concept was to use structured questions to guide the session, and our metric for the success of the interview was to receive detailed answers to the prepared questions. However, we wanted the discussion to be unstructured and free-flowing. To allow for this, the conversation did not follow the strict flow of the prepared questions. The interviewer used provided answers as a launching point for new discussions on relevant topics, and asked follow-ups and clarifications based on the current thread of conversation, even if this meant some questions were answered out of order. As a result, we received in-depth answers and also learnt new information that was not originally considered.



- Open-ended prepared and structured question used to initiate discussion on topic
- Unstructured discussion and spontaneous follow-up questions used to investigate topic in-depth

To ensure we accurately recorded the conversation, we asked the client and were granted permission to record our call over zoom. Furthermore, we had 2 team members acting as minute takers, to ensure no aspect of the conversation was missed. Once consolidated and analysed, a summarised version of the minutes, with action items dependant on the engineering team and the client will be sent to the client for verification.

Interview Roles and Responsibilities:

We appointed a single interviewer to facilitate the meeting to ensure clear lines of communications with the client. The role of the interviewer was to facilitate the discussion, ensure all requisite topics were covered, ask appropriate follow ups and ensure an in depth discussion on topics. Two members were assigned as minute takers to ensure that no information would be lost once they consolidated their information. The rest of the team was assigned the observer role, which was to watch the meeting and supply the interviewer with follow up questions. To this end, we made sure that the interviewer had access to a private communication channel via Slack with the rest of the team. This meant that any spontaneous follow-up questions by team members could also be addressed in the meeting without cluttering the communications.

Team Member	Role
Sayyaf	Interviewer
Sothea-Roth	Observer
Ben	Observer
Tina	Observer
Mustafa	Minute Taker
Michael	Minute Taker

Questions Asked & the Client's Responses

Question	Answer
Initial Questions	
What problem are we looking to address?	• Telstra has a large security centre that keeps Telstra secure and also sells services to customers. • Telstra has log file data coming in, stored for analysis. Data can only be analysed after it has been normalised, currently a manual process, and is a bottle neck. • Every device in the network (laptop, endpoint tools on user laptops, router, switch, server, firewall, software stack, load balancer, AWS, Cloudwatch, database logs, authentication systems, windows, event logs) sends log files into "several very large repositories". Each device has their own format, and needs to be adjusted to be compatible • Over 100,000 network devices sending unstructured and unnormalised data • Current system has many input files, and once processed it goes to 3 primary platforms: Splunk (and Splunk ES), Elastic, and Cloudera
What are the objectives of the system to be built? What do you expect the system to accomplish?	• Document industry practices, and create a tool to automate the process • Leverage pre-existing industry standards, algorithms, and solutions. • The output of our product should be processible by Spunk, Elastic and the cloud application
Understanding the System As Is	
Could you describe the system as is? How is this problem being solved today?	• Starts in data ingestion, where the source data is gathered. Data is then passed into Splunk & Elastic where the data is processed. • Currently, data fields are being labelled manually • Way to figure out how to process is to find the device's specification, which lays out what the device records • In Spunk & Elastic the data is processed, usually via a lookup, and educated guess. • Elastic is used for a product that Telstra sells • Splunk is used internally (at a larger scale)

What are the pain points?	- Data is received in different file formats from different applications (Syslog, Windows Event Logs, Linux Logs) - Sometimes the format of a log file from a device changes, so ideally the system should be able to detect these changes - Data must be normalised & ingested in order to be manipulated. Currently this is a manual effort. - When the log files go to all 3 systems (Splunk, Elastic, Cloudera), each team (3 total) will manually name columns and fields in their own way - Labour intensive and prone to human error - Servers are running a new OS that hasn't been seen before. It has log files that are relevant from a security perspective. User behaviour data is recorded e.g. logged in, logged out, did abc. Data is helpful for security to see if something odd is happening. Ingesting the log file is worthless though because they don't know what a user ID looks like. - Need to extract common fields that have security value given the files are from different vendors with different formats. Need for normalisation.
Is the entire process manual?	- At the moment, the creation of the way of processing a new input file is manual
Who are the users of this platform?	- Primary Users: Data Analysists & Data Scientists, with varying level of expertise, from entry-level (such as graduate) to expert - Secondary Users: People who use the results of the system. Data is fully normalised, ingested, formatted, outputted when it reaches these users. - Also cybersecurity team has 10-20 users who may access final data
Could you show us an example of the system's Input and Output?	- Not at the moment, at a later interview
What level of scale should the end product be able to handle?	- Number of peak users - just a 'handful', as the other analysists use the results, not the system itself - Amount of daily data ingest - about 3-5 TB of data daily - Types of log files - sys log, database log, firewall log, event log
What is the priority of the output services, relative to each other?	- Spunk/Elastic, then Cloudera last (need to contact Patrina from Telstra to find the top priority) - For each output service, there is a team dedicated to it, with their own processes.
Understanding the System To Be	
What is the preferred application type? (Web, Mobile, Desktop, Existing Tech)	- Unsure at the moment, something for us to think what is best
What business environment will this system be used in?	- Analysist/Engineer will parse data, validate recommendations, then pass data into the product, and get the formatted data out
What is the nature of system to-be?	- An automated or semi-automated tool - Tool can be inputted to system and has interactive interface to provide recommended format for data views after normalisation - At this point of time, system to-be is just to create format of data. Preference is not to change ingestion process (Splunk and Elastic have their own). - Maps out source data to a more consumable format like a database with column and rows. Objective is to line up values up into columns. (Currently all done manually) - Detection of unexpected format changes from the source. - Primary concern is to map out format for normalisation, but beneficial to periodically check if anything has changed with source system file formats - Platform/system agnostic
What are the initial thoughts on what may be wanted from SWEN90009?	- A paper prototype - Proposal or recommendation. Different approaches to address problem. - Interface agreement for the three Telstra teams using data. Example: - Step 1 - A script which separates out values. Parse data into consumable format. - Step 2 - Standardise the fields (different vendors call them differently). - This would require manual lookup from documentation, or guessing the meaning of a data item - A review or literature survey of tools that are already solving similar problems - What is the value of these tools: Pros and Cons - Analysis on how effective the tools are. May need to use Telstra's data for this analysis - Inference and ML model, seeing how much mapping we are doing as percentage successfully

Questions to Lead Data Engineer:

Question Answer	
Can the scope be reduced to just IP addresses?	No, with 100,000+ devices, the scope is much wider
What do we need to do, and what is the scope?	Need a tool to assist engineers with data normalisation to speed up the process. An automated system would not be trusted, but a tool that suggests what each field is would be helpful. Potentially also identify existing files changing formats. Suggestion of tools is also good, but there is no fixed solution as the current area is still being researched.
How are Splunk add-ons and Logstash configured for normalisation	They are all config files: • Splunk uses conf files with regular expressions • Elastic uses Grok Can just go into the backend and configure them there.
When investigating the log files, what software do they currently use?	Notepad++, though Splunk has its own interface
Is there documentation on the current normalisation process, or do people just use their own things?	People try to normalise everything, rather than what is needed. There is information online on what sort of files Splunk and Elastic create/use to match fields with types
How does the system handle changes to the log files	Telstra has Elastic ingest pipelines set up on existing Elastic clusters. These are used to continuously normalise incoming data. Splunk is still new, the team doesn't have anything set up