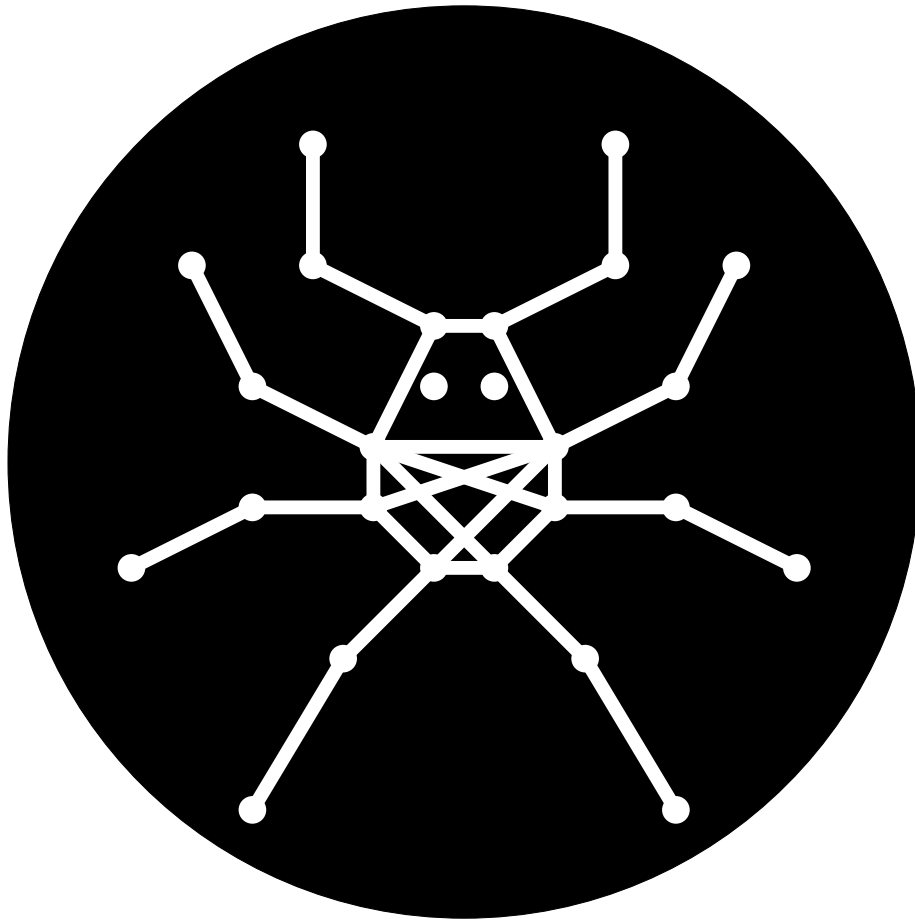GROUP PROJECT

Project Darknet – Group Report

Johann Beleites, Ibtehaj Nadeem, Farah Kiran Patel,
Josh Treon, Augustin Zidek

Group Report

Due to the uncertainty and expected evolution over time of such an experimental project, we decided that any solution we implemented should be fully modular and extensible. This facilitates future development by the client as much as it made adapting our system over time a lot easier. For example, after development had started we were provided with a CSV file containing information about thousands of people who had booked tickets through the Spektrix online booking system. It was possible for us to process the data and store it in our database by building just one extra module, since we had sufficiently encapsulated each component of our system. This project has helped us to realise the importance of carefully designing solutions that are as flexible as possible.

Throughout the timeframe of the project, all team members have been able to communicate with each other freely and without conflict. This allowed us to collaborate effectively to produce a system design that we were all satisfied with. Communication with our client, which was of particular importance for a continually evolving project, was fluid and we were provided with comprehensive information whenever it was needed. It is clear from this experience that the ability to communicate is critical in the realm of software development.

Although our system is very extensible for future development, we could have better prepared certain aspects before committing them to the design. During the development process, we discovered that it would not be possible to extract personal information from Facebook in the way we had assumed would work. This was unfortunate, as Facebook was originally intended to be the primary source from which we collected data. We also found that although the Twitter API was less restrictive with regards to Twitter content, the data we could collect was largely unreliable. We have learned from this incident that when designing a system, any assumptions about external components – in this case the Facebook API and the Twitter API – should be checked so that the design can be modified before starting on its implementation.

One area that proved slightly troublesome was the issue of determining the legality of some of the proposed methods of obtaining data. We had initially contemplated staging a man-in-the-middle attack, whereby the internet traffic of audience members who had connected to a Wi-fi hotspot in the theatre would be monitored. The legality of such an action was unclear, and as a result we had to decide to omit the idea from our project. As well as considering the technical challenges that a software development task brings about, we are now aware that such a task has to be viewed from a wider context, including legal and ethical implications.

We have completed the entire back-end of the system we set out to construct. Although the depth of personal information we had initially anticipated we would be able to gather was not possible to obtain in the time that we had, the project should still be considered an overall success.

Breakdown of Work

We decided against assigning a single project manager, instead choosing to spread the associated responsibilities amongst the team. This turned out to be a very effective approach for a small group such as ours, as technical, creative and administrative work could be distributed more easily to match preferences. Although many tasks were individually assigned, the team collaborated on a very large proportion of the project; design decisions were made together, and many group sessions outside the allocated times were held, including a three hour meeting at Starbucks on a Sunday morning! Below are the additional contributions made by each team member.

Ibtehaj was assigned the role of client liaison. This included sending written deliverables to the client and the group project organizers. He also created UML class diagrams to assist in designing the system. He studied the documentation of the database system used in this project and provided a database server. All the code interacting directly with the database (executing SQL queries, creating tables etc.) were written by him, as well as some utility classes used throughout the system.

Augustin was assigned the role documentation manager since he has got a lot of experience with the LaTeX typesetting system. That means he was responsible for the final layout of the Specification, the reports and also the presentation. He has also designed the project logo. Concerning the coding part of the project, Augustin has created the image storage utility, that downloads and stores uniquely named images on the hard drive, so that they can be efficiently linked with the data in the database. Augustin also designed and coded all GUIs that are used in the project and developed the layer that delegates the user input and actions to the frontt- and back-end layers.

Josh came up with a design for the system that greatly simplified dependencies between components, thus increasing the extensibility and modularity of the project. During meetings, he supplied ideas to the client about ways in which audience members' data could be used during the performance. Josh also wrote the code responsible for managing the data collectors as well as generating the global table of attributes used in the database.

Farah was responsible for writing the project plan and progress report and due to this helped with some of the organization of internally setting deadlines during meetings. She implemented classes to parse data from Spektrix CSV files in the backend. In the frontend, she dealt mostly with generating the two effects. She implemented the report generation for the report effect. She also implemented the picture wall effect that retrieves pictures and forms a collage to display the images collected. She wrote tests for all segments of code that she produced.

Johann's responsibilities during this project mainly lay in the area of data collection. Initially he researched the feasibility – both of technical and legal nature – of a Man-In-The-Middle (MITM) attack on a wireless hotspot during the performance to gain guests' data. This part of

the project was not continued to the implementation phase due to the connected legal issues, however. Further, Johann investigated what and how data could about guests could be obtained from the social networking sites Facebook and Twitter. He implemented a Facebook data collector, which tries to find guests on Facebook and store any obtained data. Also he implemented a Twitter demonstration, which fetches tweets from twitter users. Apart from input during and outside of meetings to the system design Johann also created a small logging framework for the project, intended to help when debugging during development and maintenance of the product. For the code written suitable tests were written and performed, of course.