Weekly Planning Diary

**The Challenge**

“Create **a raspberry pi, NPCap and Wireshark** based **packet sniffing network monitor** that can be attached to **wired and wireless networks** and **passively monitor network communications**. Time permitting explore how anomalous transmissions can be detected using machine learning based modelling of normal network behaviour. This project also requires an analysis of the data protection and privacy implications of the technology.”

**Week 1**

By the end of this project I would like to be able to develop a device that can be connected into any network and provide network diagnostic information. I want the product to be easy to use and its information very readable to other people. I would like it to alarm whenever it detects a breach in the network, as well as provide ways of optimising the network. Ideally the device , as simple as connecting the device to a computer in order to begin. This **should benefit a lot of network administrators**, as it will provide a **quick way of analysing the network**, without putting in too much effort.

**Week 2**

I’ve only taken the first steps into the project however I still see the plan envisaged in Week 1. I do wish to add that the device created should **passively monitor the network**, so there is no need for human input (minus to start/stop the device and initial setup).My priority is to enable NPCap to run on my laptop aswell as running it on the raspberry pi. This will enable the collection of packets from the network and thus the ability to **passively monitor network communications.**

**Week 3**

In order to better plan, I have adopted a mentality of working from the product back to where I am at currently with the project.

When I imagine my device in operation I two distinct options. Either I could create a device which uses machine learning to detect anomalies in the network or I could create a device which would use multiple raspberry pis in order to fully analyse what is happening in the network. In either case I will have to find a way of wirelessly transferring the packet files from the raspberry pi(s) to my laptop. Once I solve that issue, I will be able to collect and analyse the packet information and feed that into an algorithm. Both options will be valuable to network administrators. However, the machine learning approach will provide a more proactive aid in cybersecurity, being able to detect if something is wrong. The other approach will provide a more detailed analysis of the network via the use of multiple raspberry pis, it will be able to analyse specific routes in the network and how optimised they are.

So, my priority must be to figure out how **to send the packet files to my laptop**. Also, this week I need to read around the two different options. It would be of benefit to **know what existing products** of this nature there are and their strengths & weaknesses.

**Week 4**

I have found a way to transmit files from the raspberry pi to my laptop (Flask) and after researching into existing products it seems that the **most valuable** course of action is to go with the **machine learning option**. This is because there are multiple products on the market which generically analyse the network, however there are none that explicitly state that they use machine learning. Further the machine learning option has only a handful of academic papers on it, so if I made a tutorial on the subject, it would be **extremely valuable** as it **would be a world first.**

Additional Tutorial (Packet Occurrence Barchart)

Another thing I could create to add value would be a video/web tutorial on how to create a Packet Occurrence Barchart in python. This would be valuable as a quick google search yields no results for something like it and I had to circumvent a python library(pyshark) in order to obtain the correct data type for creation of the barchart.

**Week 5**

I plan to run the machine learning algorithm in this week. At the time of writing I am going through the tedious process of cleaning the data.

Once this is done, I will take a step back and analyse my current situation. I will find good academic papers on the subject, to see if they can recommend a better machine learning algorithm.

On Wednesday the 19th of February, I gave birth to my first model. From this, it is now time to take a step back and analyse where I’m at in terms of the project and how far I’ve come

It is now six weeks since the beginning of the semester and 5 weeks since beginning the project. So far, I’ve used/done many different technologies/feats which I have never done before, these include (but not limited to and in no order).

* Dual Booting a Laptop.
* Learning Linux Commands.
* Emulation.
* NPCap
* Setup of a raspberry pi
* C Code
* Webpage Creation
* Pandas Data frames
* Flask Web server
* cURL
* Creating Packets
* Creating a Machine Learning Model

I am amazed at how much progress I have made in 5 weeks aswell as how much I have learnt. There is a danger now of losing site of what is important, that being the creation of something valuable. I have made a model, however since there is no validation data set, I cannot tell if I have made a good model. I believe that obtaining a suitable data to run the algorithm on is of upmost importance now, as with this I can prove that this model is valuable.

There are a few options that I can pursue in order to obtain good datasets (in the form of pcap files). I can

1. Search online for datasets provided by institutions.
2. Ask academics at Queens for datasets
3. Obtain datasets from real attacks on honeypot computers.

The plan now is to obtain these datasets, and conduct the experiment, proving that the model works.

In an optimal setting, I would like to use the system in a real-world company with the help of penetration testers to see if the model worked or not. Perhaps even a real attack upon a system would be useful aswell. I say this because right now the program has only worked with static files, so it would be nice to see it working in real time. Alas, we do not live in search a perfect world.

**Week 6**

Datasets obtain the first task is to shift through them and decide which one would be best to use in order to test my model. Then following this, writing code that will test my model.

In terms of the future, I would like the graphical representation of my model to work correctly. This would be very useful as it would show what decisions the model is making “under the hood”.

Over the last week I collated a few dataset resources which may be of value to anyone attempting this project on the future. I shall make either a website or a repo for these datasets so others can have easy access to them.

My end game plan is to be able to sell a box(s) to a company which would handle all their internal security issues. The device itself will have the ability to cut network access to affected computers and essentially quarantine them. The machine learning aspect of the device will decide which computer is affected and by what attack it is being affected by.  
  
The device itself will be a “plug and play” type of device where a user requires no technical knowledge to set it up. Ideally, for every error a step by step guide should be created so that customers themselves will be able to fix their internal networking issues. Failing that a customer could call a network of engineers who could help resolve the issue.

**Week 7**

At the end of the module I would like to achieve a dynamic network security device which uses machine learning to detect threats. I believe I can hit this goal, if I remove the final bug from my current system. I’m unsure that I will be able to create the physical device itself, nor allow it to break a network connection. I will have to test my model, so I will have to come up with a suitable testing strategy. I have already obtained enough datasets for the project. The required tests to be done on the model can be done within the time frame.

Who would benefit from my project?

As a network manager I want reassurance that my network is secure. Both internally and externally. Right now, there are plenty of cyber security products that deal with external threats, yet very little/no security products for internal threats. I would like a physical device for peace of mind. This device would detect attacked computers on the network and remove them from the network, so that the computes themselves are quarantined from the rest of the network. This device would then provide me with a detailed report of what had occurred, along with possible steps to fix the issue. It would also be kept up to date with all the latest threats and would not conflict with the external network protection software. I would also like the device to alert me whenever the device disconnects a computer from the network, either by a siren or ,even better, ring or send a text to my phone. This way I can more easily uphold my responsibility of protection the network.

As a high-level member of a company, I would like to keep my company’s secrets, secret. I am already protected from external attacks, yet in order to safeguard against an untrustworthy employee, I would like this device protecting my most valuable digital assets. It is much easier to launch an attack upon these assets if the attacker already has access to the internal network. This device should alert me to an attempted attack and will be able to provide me with the address of where the attack came from. With this I can investigate the device.

I have created a function in python which parses packet data and returns useful information back to the user. This is my magnum opus of the coding for this challenge so far. This function has many uses outside of machine learning, such as plotting graphs of network data, visualising networks and mis configuration detection. I also have made another function which calculates packets per second for a given transmission.

In terms of prioritising what is of most benefit to people, I believe that packaging my current code into a library for people to use is of upmost importance. This means that even if I fail at implementing the model, I will still be able let others use my code in order to achieve that. Updating my GitHub will also be a priority. This ensures that if I do not finish my project, those who come after me will have a head start. Essentially, leaving my work in a state that can benefit someone is of the greatest priority right now.

As my main audience is students interested in my project, the best community to find these people and share my work would be, in my opinion, on a subreddit, perhaps r/Python or r/MachineLearning.

**Week 8**

The plan has not changed from Week 7. What has changed from Week 7 is that I have finished practical work, and so will be enacting on the promises of cleaning up the code and creating a python library detailed in Week 7.

**Week 9**

I have encountered a minor setback with the Week 7 plan. In turns out that creating a python module is much harder than I first thought. I will continue to try and move past this. I am also focusing on the final hand in, having given thought to the Testing strategy and Market research.

**Week 10**

Market Research and Testing is the main priority right now. I will produce a series of drafts which will hopefully produce an excellent Market Research and Testing section(s). I also want to test my system, so I plan to install Kali Linux on my laptop. After that I intend to use Kali Linux’s inbuilt attacks to test my system.

**Week 11**

With Market Research and Testing being in the final stages now, I now turn my attention to ensuring the project is in a state that someone could feasibly continue with it. This means uploading my code to GitHub, along with an accompanying how-to guide.

**Summary of Lasting Value:**

Below I have listed all things I have created which are of lasting value aswell as the links to them.

**Project Related**

* NPCap How-to guide (link to hosted NPCap how-to guide). Despite not being used in the final version of the project, the library itself was poorly documented, especially for beginners. This How-to guide should help those getting set up with NPCap. It is intended for beginners, and as a result of this the guide goes over how-to setup a Visual Studio Project.
* How to use PyShark video guide:(<https://www.youtube.com/watch?v=dL3X4IAZVlA>) PyShark is the library which I have used in order to read the packet files. It is essentially a wrapper for tshark. In the video I go over how to read packet files and how to interpret that data into a bar chart graph. This guide is intended for beginners and as a result, the “only summaries” property is set to true. This means that the packet data read in is only the bare minimum (no byte data displayed etc), thus avoiding beginners feeling overwhelmed. Hosted on YouTube as it is the most popular video sharing platform, thus easiest for beginners to see/view as these people should be used to the layout of the website.
* How to use tShark video guide (<https://www.youtube.com/watch?v=NvKiyHCRXrU&t=338s>). tShark is the library I use on the raspberry pi in order to sniff the packets. In the video I go over how most of the keywords and operation modes work in the tshark library. This should help reduce the confusion caused by tshark’s multiple different parameters and run modes. This guide was intended for those looking at what the tshark library is capable of. Therefore, I go over each run mode. Hosted on YouTube as it is the most popular video sharing platform, thus easiest for beginners to see/view as these people should be used to the layout of the website.
* Machine Learning Packet Analysing Python Code (<https://github.com/bcheevers/RaspPiPacketMachineLearning/blob/master/DecisionTreeCode/DecisionTree.py>)

This is the Machine Learning aspect of the system and is probably the most valuable thing produced. This code opens the door on using machine learning with packet data, especially for newcomers. Hosted on GitHub for ease of download and for the possibility of forking into other projects.

* Guide for the Machine learning Packet Analysing Python Code (Word Doc)

(<https://github.com/bcheevers/RaspPiPacketMachineLearning/blob/master/DecisionTreeCode/Guide%20to%20Decision%20Tree%20Packet%20Algorithm.docx>). Created for beginners, this guide goes through the code function by function. The guide features images of the code to ensure the reader is not lost.

* Guide for the Machine learning Packet Analysing Python Code (Video) (<https://www.youtube.com/watch?v=6VqTpga0_j4>)

The video is intended for beginners in this field. It is NOT intended for those who are learning the python language. As a result of this, little time is spent on the fundamentals (for loops, variables , importing etc).

* Flask Server for Raspberry Pi code (aswell as command to start the script) .(<https://github.com/bcheevers/RaspPiPacketMachineLearning/tree/master/Flask>) .This script is primarily for people who just want a simple flask server to run on their raspberry pi so that they can do simply file sending. This is helpful for those who cannot ssh into their pi, perhaps because they are not on the same network as it. More of a helpful utility. It is independent, which means it can be put into any project. Hosted on GitHub for the potential forking.
* Blog Post (Link to blog post). The Blog post provides my story throughout this module, filled with insights and challenges. It also links to my garnered packet file data sets. The Blog Post also includes a Testing and Analysis system, which details potential issues with the system and how to fix them. This section also includes ideas on how to develop a testing system which will test the extent to which the solution fixes the issue.Aswell as this there is a User Analysis section which details what how data about how the user is using the system can be obtained. This document will be vitally useful for those wishing to carry the project forward.
* Social Media Post (Link to Social Media post) This post will increase the visibility of the project. I will use r/Python as the platform for this post as similar projects have been well received: <https://www.reddit.com/r/Python/comments/dfdtbs/automated_brightness_on_a_laptop_using_machine/>,<https://www.reddit.com/r/Python/comments/fudlk3/i_made_my_first_machine_learning_project_in/>. Further justification for this can be seen in the Daily Diary under the Market Analysis (March 28th) section.

The structure of these posts seems quite simple. They consist of an image or video of the program, along with a catchy headline. I will adopt this form, as this has been the most well received.

**Module Related**

* How-to make a How-to Webpage Guide(<https://github.com/bcheevers/CompSciModule/tree/master/How-to-Guide>This document was created to aid others in the module. Specifically, it was to aid those who had trouble with assessment 1.
* Diary Templates.( <https://github.com/bcheevers/CompSciModule/tree/master/DiaryTemplates>) Though I did not specifically create these documents, the structure of these documents seemed from my approach to the diary aspect of the module.