FYP Project Proposal Form 2019/2020

Student Name: Povilas Kubilius	**Student Number: C16370803**
Mobile Number: 087 337 8177	**Supervisor:**
Programme Code: DT228	
Project Title: Investigating the use of digital images to generate cryptographic keys	

Summary (approximately 200 words)

The goal of my project is to improve security of cryptographic keys. I plan to accomplish this by using digital images as seeds or salts for cryptographic algorithms.

My program will take a digital image as an input, then by looking at certain color frequencies values, it will hash out a certain unique value which will be used as a seed to be able to generate a unique cryptographic key. This key then can be used with standard encryption algorithm like AES or RSA to encrypt files, internet traffic and secure virtual machines in the cloud.

I want to create a website application where anyone can upload their image and get their key generated, as a proof of concept. The user could also download my program to be used offline to be used securely.

The main advantage of using digital images to generate keys as opposed to letting pseudo-random algorithms generate unique keys is that it will still be impractical to brute force but the key can always be replicated by the original creator because they can use the same image, which only original creators knows which image they used. This way it may be necessary to store the RSA key in some plaintext file where it’s possible for an attack to steal.

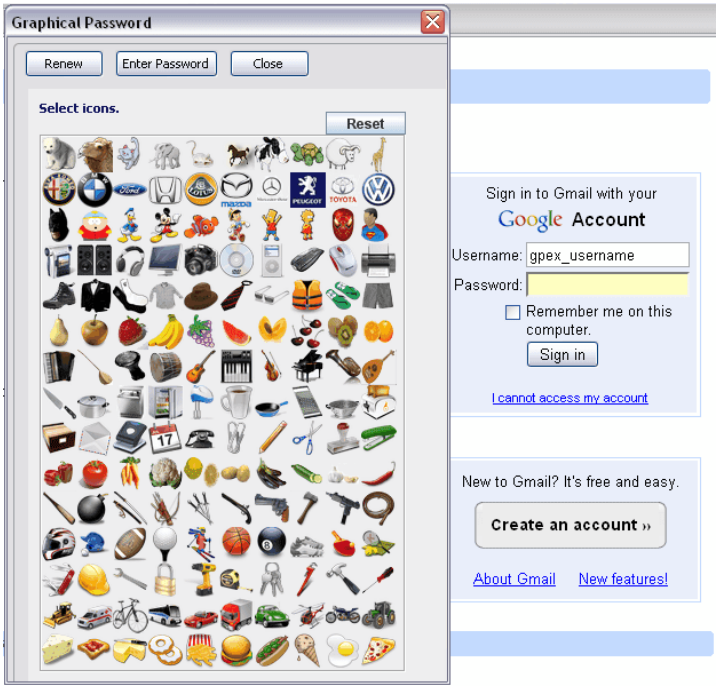
FYP Project Proposal Form 2019/2020

Background (and References)

I always had an interest in cyber security. Cryptography is the core of security. Systems that allow only authorized users access, systems to prevent others seeing the contents of your internet packets and more. The biggest weakness of most computer systems are weak passwords. People's passwords are too easy to brute force. Using advanced types of passwords like RSA keys are a lot more secure because they are generated randomly [1], and each key is practically unique. Inevitably, the more secure a system, the more inconvenient it is to use. When you generate an a cryptographic key like RSA key, it's kept on a file. In cases where the file lost, you get locked out of your protected system with no way to regenerate your RSA key.

In *Secrets and Lies: Digital Security in a Networked World by Bruce Schneier (2000),* the author mentioned there were systems that used biometrics, like your fingerprint, to generate cryptographic keys. The main failure of this system was that secure systems should regularly change their keys. Humans have 10 fingers, so the number of possible keys generated from fingerprints is 10. Another system used human passwords to generate keys. This becomes as secure as a regular password, a hacker can use the same algorithm, use a dictionary attack to brute force the key. Passwords and passphrases can also be hard to remember, if forgotten can be locked out system.

Humans remember pictures better than words [2]. The use of graphical passwords is an alternative system to word passwords. This works by having a user select an image when they are registering. Then when they use the system, instead inputting a password for a PIN, they must select the image they chose at registration out of an array of other images [3]. This works great instead of a pin or possibility to login into a website. It doesn't apply to cases where you use cryptographic keys to encrypt files and messages like public key cryptography or SSH keys.



The approach I want to take is also use images as password, but in the context of cryptographic keys. There is a way proposed of using images to generate cryptographic keys [4]. By looking at the color frequency of red, green and blue in the digital image, you can derive a unique value the image. Then using this value it's possible to hash out an encryption. Then using this key, encrypt files or use to SSH into remote machine. I think this is an interesting approach to cryptography and so far, I have not seen this method be implemented.

References

[1] Public Key Cryptography and the RSA Cryptosystem, Nuh Aydin.
https://digital.kenyon.edu/cgi/viewcontent.cgi?article=1001&context=math_pubs
[2] Neural correlates of the episodic encoding of pictures and words, Cheryl L. Grady, Anthony R. McIntosh, M. Natasha Rajah, and Fergus I. M. Craik,
https://www.pnas.org/content/95/5/2703
[3] Method and system for producing a graphical password, and a terminal device,
https://patents.google.com/patent/US7376899B2/en
[4] A Proposed Method for Generating a Private Key Using Digital Color Image, Wisam Abed Shukur
https://www.ripublication.com/ijaer17/ijaerv12n16_110.pdf

FYP Project Proposal Form 2019/2020

Proposed Approach

At the core level, my program takes in a digital image and returns a cryptographic key. I want to build a whole website to make this program available to everyone and make it easy to use. This will be mainly a client-server architecture.
For the website I will need to use HTML\CSS and JavaScript.
On the server side, I haven't yet fully decided, but I would like to implement this in Python, or Java if need be.
The website should be accessible from PC, laptop and mobile devices, anything with a web browser.
The main use case would be a user going to my web app, uploading an image to my server. This will be handled by the of secure data transfer protocols like SSL/TLS when transmitting data over the internet. The Java servlet handles the uploaded and uses the python framework that was used to implement the image file processing algorithm to get the color frequency values and use that to hash out cryptographic keys like AES key or using RSA to make public/private keys and SSH keys. This then will be passed out of the server and still using SSL/TLS send back the cryptographic keys to the client where it will be displayed for the user. On the web app, there will also be an option to download an executable form of my program.
I want to also have an offline version of the application. The offline version will have a graphical user interface with is easy to use. Take an image stored locally on the computer and then get the color frequency values and use that to hash out cryptographic keys like before, then display the keys for the user to be able to copy paste into a text file or simply use the "Save" to save the key in the right file format, ready to be used.

I want to use Feature Driven Development. Since it's a relatively small projected managed only by myself, I think it is most appropriate as opposed to Agile or Test-Driven Development. Although Feature Driven Development is still very similar to Agile, in fact it takes all coding best praises and put them together into a cohesive whole.

This a fully software project, so there is no need for any additional hardware.

To evaluate the project, I will get people who are tech savvy and other who are not to see how easy and comprehensible is to use the web application and user interface n the offline version.
I will also test with multiple different type of images such black and white, all single solid color, or complex high-resolution color images to evaluate which images generate the most random cryptographic keys. I will also test with different image formats like JPEG and PNG. Evaluate if slightly compressing or changing image format will affect intrinsic color values in the image that will generate a new key instead of regenerating the original key as in the goal of this project.

FYP Project Proposal Form 2019/2020

Deliverables

Interim Report
A project dissertation
Front end: A web app that will be publicly available
Back end: A server that takes image inputs from front end, generates key and send back to front end
Offline version of the program with an easy to use user-interface

Technical Requirements

Laptop
Website hosting site
Uses of programming languages, Python, Java and JavaScript
Frameworks like Java Servlets and Django (python web framework)

FYP Project Proposal Form 2019/2020

Project Reviews – Please include reviews of two of LAST 2 years projects from either DT228, DT282 or DT211C.

Project 1
Title: Education Tool for Web-Based Vulnerabilities

Student: Cormac Kelly

Description (brief):

Interesting project scans your Java files for possible SQ L Injection vulnerabilities. It is designed as an education tool. I like the way it is a web application, making it accessible and easy by the user. It encourages to design code with security in mind and using this tool as quick test for any obvious security flaws pertaining to SQL Injection. I like the idea behind the project, to raise awareness about computer security and encouraging to write secure code.

The project used many technologies and languages. For the code base, Python, Java and JavaScript were used. These are well suited and straightforward languages to use to make a web application and the server back end. These languages also have graphical user interface libraries to make the program easily accessible.

The I share similar intent with this project, raising computer security awareness and making it more accessible to public via web applications. I would like to also use the same technologies and architecture for my project and web application and educate people on computer security.

.

FYP Project Proposal Form 2019/2020

Project 2
Title: Secure Document Sharing

Student: Owen Kane

Description (brief):

This project creates a secure online system to create, edit and share documents over the internet. It uses client-side AES encryption algorithm to encrypt the files before they are sent over the internet. This way the data will never be sent in plain text format for any man-in-the-middle to see the contents of the data in case where they are sniffing and capturing passing packets online.

This is a good approach to file sharing. This increases the privacy and security of data from being access by unauthorized users. The technologies used are also like what I want use, Python, Java and JavaScript in a client-server architecture. Any transition of data between the tiers in the architecture use a secure encrypted transfer protocol, SSL/TLS. SSL is used when data is retrieved from the database to the server, and again when data is sent from server to client and vice versa. This a good approach, with I'll have do the same in my own project.

The project was very well tested. Used multiple types of tests, such as ad-hoc testing, unit testing and integration testing. Testing is vital to any coding project, but more so to project with computer security as possible bugs in the guys can expose vulnerabilities and opportunities for hackers to steal confidential or sensitive data.

Proposal Sign off:

Student Signature:	**Date:**
Lecturer Signature:	**Date:**