Nathanael Asaam

**DESIGN AND DEVELOPMNENT OF AN AUDIT FRAMEWORK AND AN ELECTRONIC SYSTEM FOR DETECTING AND MONITORING THE USAGE OF A COMPUTER NETWORK AND DEVELOPMENT OF A MEDICAL SYSTM USING EVOLUTIONARY COMPUTING**

**INTRODUCTION**

This paper is a research proposal which will be doing investigation into building an anomaly threat detection system using behaviour models and a risk analysis system. Such behaviour models are the heart of machine learning, and evolutionary computing. Some other methods of building such anomaly threat detection systems include the use of statistical models such as time series models, univariate models and mean and standard deviation models. In this paper we describe how to build a usage profile of a system. The aim of building these usage profiles is to be able to detect unusual behavior on the system. As a result, the usage profile will be used as the basic building block for the development of the anomaly threat detection system.

This paper uses regression to determine the usage profiles of a system by studying the relationship between relevant system variables that will be used to formulate the usage profile. The dependent and independent variables for the usage profile can be determined from an audit trail. Additionally, the paper applies hidden markov models to study the various states a computer system can fall into and the various stage transition in order to be able to predict unusual behaviour in the system. Unusual behaviour in this case may be a particular state or a transition from one state to another or the manner in which a particular state transition occurred. With this usage profile which is composed of the usage profile equation and a mean and standard deviation model that capture average usage and its standard deviation and the markov chain model that captures the various states of the system and the various state transition it becomes possible to detect anomaly on the system.

Using linear and nonlinear programming, the usage profile equation can be maximized to determine states of the system and points at which the system is optimal. This can help improve the system’s usage. Also using differential coefficient of the usage profile equation and other statistical models such as the mean and standard deviation model, a threat profile of the system can be developed. When the threat profile equation is minimized using linear and nonlinear programming, it will help prevent threats on the system. The benefit of this research is its application to the development of anomaly threat detection systems and risk analysis systems that can be used for performing computer security risk assessments and analysis. The research proposal will also explore how to build an electronic system for detecting and monitoring the usage of a computer network.

Evolutionary computing uses recombination and mutation to seed new candidates. The process can be iterated until a candidate with sufficient solution is found or a previously set computational limit is reached. Recombination is an operation that is performed on two or more chromosomes. Mutation is an operation that is performed on one chromosome. The field uses concepts that make it possible to solve some complex problems. It derives its origin from biological evolution.

This document seeks to measure medical care information like weight, height, heartbeat, and rate of blood flow using evolutionary computing and genetic programming. Android phones come with a robust operating system that is built on a Linux kernel. It has a programming interface that makes it possible to write java like code to manage the operating system and the device. There are numerous apps and systems developed using the android programming language. The language is composed of java for programming application or system logic and xml for specifying design features such as interface layout, interface design, system resources, and application or system architecture.

**Usage Profile**

If a usage profile of a system can be built, it will become possible to detect unusual behaviour on the system. The method for building such usage profiles involved determining factors of the system that are critical to the system. These factors can be seen as critical system variables that affect the system’s usage. The other thing to consider is determining the way in which you can obtain an abstract representation of the usage profile. The abstract representation of the usage profile can be achieved by the application of behaviour models such as statistical models, machine learning models and cognitive based models.

The first goal of this research paper is to investigate techniques for building a usage profile of a computer system. The aim of building the usage profile is to be able to have a working model that describes the systems behaviour. The second goal of the research is to be able to detect unusual behaviour on the system. Unusual behaviour will be detected as deviation from the usage profile built. The last goal is to be able to build an anomaly threat detection system and a risk analysis system that can be used for detecting threats and performing risk analysis on a system.

**Research Model**

The research model is given as Y=f(Xi) where Y is the usage of a system, and Xi are the various independent variables that can be related to a dependent variable Y to form the equation Y=f(Xi).

**Research Methodology**

The research methodology for developing the threat detection system is given below.

* Study machine learning techniques and algorithms, linear and nonlinear regression and intrusion detection system to know the extent to which the research model can be used to represent the usage of a system.
* Experiment the usage profiles that can be built by using programming as a tool.
* Apply rates of change, linear and nonlinear programming to detect threats that may occur in the system.
* Develop an anomaly threat detection system and a risk analysis system using the result of the experiment.

### BUIDING THE USAGE PROFILE

To build a usage profile, we use a mathematical model that captures the behaviour of the system and a markov chain model that captures various states and transitions in the system. The mathematical model is made up of a usage equation composed of a dependent and independent variables and a statistical model that captures average usage and its standard deviation. The usage equation of the system can be summarized as Y=f (Xi, Ci), where Y is our systems’ usage and Xi are the various independent variables of our system that constitutes the normal usage or behavior of the system.

The markov chain model can be built using various states or points of occurrences in a system and their associated probabilities. The mathematical model can be developed using regression. This paper will describe how to build a usage profile using an authentication model.

### BUILDING THE AUTHENTICATION USAGE MODEL

The critical system variables for an authentication usage model of a system on a network may include the download speed on the network, the upload speed on the network, the size of data sent to the server during authentication, the size of data sent to the client during authentication and the time it takes for a successful authentication. The size of data sent and received from the server are request data and response data respectively. To build the usage profile for the authentication data, we will capture data for all the critical variables at equal time intervals say every 10 minutes while the authentication system is being used. After having a sample of sample size of about 10 we will try to determine the relationship between the dependent variable and the independent variable. As already stated the relationship can be determined using simple or multiple linear regression. In addition to the relationship, we will also determine other statistics that describe the behavior of the authentication system. These statistics are the averages for the various critical variable captured.

### BUILDING THE MARKOV CHAIN MODEL FOR THE AUTHENTICATION SYSTEM

Hidden markov models are machine learning models that are used to model states in a system, the sequence in which they occur and the associated probabilities for each state transition. When a system has a set of states in which it usually falls and it can be predicted or established that each new state is dependent on the previous states, then hidden markov models can be used to learn the state transitions that usually happens in the system. To build the markov chain model we will determine states on the authentication usage model and their associated probabilities. Some of these states include the average usage of the authentication system. This may be abstracted as the average time it takes for a successful authentication. Other states include the minimum and maximum recorded time for a successful authentication and the average time it takes for a failed authentication or the maximum and minimum recorded time for failed authentications. With this information and their associated probabilities of occurrence during a normal day we have more information about the behaviour of the authentication system.

**BUILDING THE ANOMALY THREAT DETECTION SYSTEM**

To build the anomaly threat detection system we will be finding occurrences or states in the system that deviate from the behaviour models we have built which are the regression based model and the markov hidden model. Any programmed system that infers from the two behaviour models built and alerts or blocks occurrences that are not in line with the behaviour seen in the two behaviour models is an anomaly threat detection system that prevents threats on the authentication system. As such, if for some reason the data from the independent variables don’t match up with their dependent variable when substituted into the regression equation obtained then an alert can be signal to show that there is a threat in the authentication system. Also if for some reason the probabilities for a particular state say the average time it takes for a failed authentication or the average time it takes for a successful authentication is increasing then we can signal an alert that there is a threat in the authentication system. Normally, after a model is built, and agreed upon we expect the behaviour of the system in question to conform to the built model when being used. This becomes the basis for signaling an alert.

### USAGE MODELL LIST

### FALSE ALARM DETECTOR

The false alarm detector is a utility that detects normal system usage that otherwise may be deemed threats. Occurrences that meet the criteria for false alarms are normal usage that seems to put the entire usage of the system into a false state of vibration or anarchy. Such usage occurrences are as such prioritized as normal optimal usage. The remedy for the vibrations such usage occurrences cause is delay in other normal usage occurrences in the system.

The state and magnitude of other system occurrences plus the state and magnitude of the normal optimal usage determine the impact of the perceived anarchy. To increase convenience with which the system for which this utility is developed, the average delay time and its standard deviation must be detected. This utility is part of the normal usage. The utility is modelled just like the aggressive usage detector.

### AGGRESSIVE USAGE DETECTOR

This model is a utility that detects aggressive behavior on a system. It is modelled just like the various micro usage models. Various factors that determine aggressive behavior during system usage are used to determine the mathematical representation of this utility. Aggressive behavior includes aggressive use of major system resources, and aggressive use of system components with limited resources.

The average aggressive behavior and its standard deviation are determined. Any system occurrence that indicates the average aggressive behavior, or the average aggressive behavior plus its standard deviation or the average aggressive behavior minus its standard deviation is considered a threat and must be halted, alerted or stored for audit purposes.

### AUTHENTICATION USAGE MODEL

The authentication usage model represents the usage of an authentication system. The independent variables that must be sampled to determine the usage of an authentication system are the average data transmitted during an authentication (x1) and the average network speed for a single authentication (x2). The average data transmitted is the average of request and response data for a single authentication and the average network speed is the average upload and download speed for a single authentication. The dependent variable that must be sampled is the time taken for an authentication (y).

The goal of modelling the dependent and independent variables is to arrive at a mathematical relationship between y and the two independent variables x1 and x2. It is expected that the relationship will be Y=c1(x2/x1), where c1 is a system constant. In addition to that, some system constants that will aid threat analysis must be determined. These are the total number of valid authentications, the expected authentications within a time frame, the minimum authentications within a time frame and the maximum authentications within a time frame.

### SESSION USAGE MODEL

A session usage model represents a single user’s behavior before his session expires. To determine the mathematical model for a user’s session, two main independent variables must be sampled. These are size of session data accumulated (x1), and number of user actions (x2). The dependent variable that must be sampled is time spent before session expires (y). The session usage model is expected to be made up of two micro usage models. The mathematical representation of the micro usage models is an object containing Y1=c1x1 where c1 is a system constants and Y2=c2x2 where c2 is a system constant.

In addition to the two mathematical functions, some system constants that will aid threat analysis must be determined. These include average user actions, average size of data accumulated, average time spent. These constants can be determined from the data set used to determine the usage model.

### MEMORY USAGE MODEL

The memory usage model represents the usage of memory space in a system. The independent variables that must be sampled are number of application programs running (x1), and the number of system processes running (x2). The dependent variable that must be sample is amount of memory space being used(Y). The mathematical relationship between x1, x2, and Y is an object made up of Y1=c1x1 where c1 is the average memory space for programs, and Y2=c1x2 where c2 is the average memory space for processes.

In addition to these, some system constants that aid threat analysis must be determined. These include the minimum and maximum memory space for programs and the minimum and maximum memory space for processes. The mathematical relationship between x1, x2, and y is the memory usage model. When determined, the memory usage model can be used to analyze changes in the memory usage that indicate threats in the system.

### CPU USAGE MODEL

The CPU usage model represents CPU usage in a system. The independent variables that must be sampled are the number of application programs running (x1), and number of system processes running (x2). The dependent variable that must be sampled is amount of CPU power being used (Y). The mathematical relationship between x1, x2, and Y is an object made up of e Y1=c1x1, where c1 is the average CPU power being used for programs, and Y2=c2x2 where c2 is the average CPU power being used for processes. In addition to these, some system constants that aid threat analysis must be determined. These include the minimum and maximum CPU power for programs and the minimum and maximum CPU power for processes. The mathematical relationship between x1, x2 and y is the CPU usage model. When determined, the CPU usage model can be used to analyze changes in the CPU usage that indicate threats in the system.

### HOST USAGE MODEL

The host usage model is composed of three independent variables. Memory usage (x1), session usage (x2) and CPU usage (x3), derived from their respective usage models. The dependent variable that must be sampled in the time host spent on host (Y). Any relationship determined between the dependent Y and the independent variables is the host usage model. The resulting host usage model is an object containing Y1=c1x1, Y2=c2x2 and Y3=c3x3.

### SERVER USAGE MODEL

The server usage model is made up of the CPU time being used, the memory space being used and the number of processes running. These variables are used to form two different micro usage models. As such, there are two dependent variables, CPU time (x1) and memory space (x2). The independent variable for both micro usage models is the number of processes running (Y) . The usage model for the server is made up of Y1=c1x1, where x1 is the CPU time and Y2=c2x2, where x2 is the memory space being used.

### BATTERY USAGE MODEL

The battery usage model is made up of the average usage of CPU, average memory usage and the average usage of how a session behaves in the system. These are the independent variables. The dependent variable is the battery lifespan. The independent variables are derived from their respective micro usage models.

### DEVICE USAGE MODEL

The device usage model is made up of a battery usage model, a host usage model, and the time spent on the device. The usage models that make up the device usage model compute the average micro usage and try to relate that with the time spent on the device. The time spent on the device is the dependent variable.

### PORT USAGE MODEL

The port usage model is made up of the time elapsed during communication, number of processes that use the port and the number of paired ports. The number of paired ports is the dependent variable and the remaining variables are the independent variables.

### NETWORK USAGE MODEL

The network usage model is made up of average port usage (x1), average server usage(x2) , average host usage (x3), and time spent on the network(Y). The first three variables are the independent variables. The last one is the dependent variables. The network usage model is an object made up of Y1=c1x1, Y2=c2x2, Y3=c3x3.

### PROGRAM USAGE MODEL

To determine the program usage model, the dependent and independent variables that must be sampled are time spent using program (Y), and number of process running is the program (x1). In addition to that, the following constants must also be determined. The relationship between Y and x1 is Y1=c1x1.

### RISK ANALYSIS IN A SYSTEM

To do risk analysis in a system, the frequency at which threats in the system occur and the impact they have on the system must be known. When a frequency table is constructed for all threats and their associated impacts stored, it becomes easy to analyze risks associated with a system. When a threat is predicted, the likelihood of the threat occurring in the system can be computed using the threat frequencies.

The impacts various threats have can also be determined based on the types of threats and other parameters such as the number of such threats, the speed at which they occurred and the resources they affected or damaged. Risk in a system is computed as the product of the likelihood of threat occurrence and the impact that threat occurrence has on the system. These concepts are the basics for developing a risk analysis system using the techniques we have discussed so far.

**DESIGN AND DEVELOPMENT OF A SECURITY AUDIT FRAMEWORK**

This section of the document describes how to build a security audit framework. An audit framework can be developed by developing two main components a threat or intrusion detection system and a risk analysis system. The diagram below describes the components of the security audit framework.

Security Audit Framework

Threat Detection System

Risk Analysis System

Threat Profile

Usage Profile

The next section of this document describes how to develop an electronic system for monitoring the usage of a computer network. The next one describes how to develop an expert system for monitoring the usage of web, desktop and network apps.

**DEVELOPMENT OF AN ELECTRONIC SYSTEM FOR DETECTING AND MONITORING THE USAGE OF A COMPUTER NETWORK**

The electronic system that will be developed will be developed using concepts from Computer Organization and Architecture and Embedded Systems. I will also use Boolean algebra as the basics for developing an ALU (Arithmetic and Logic Unit) that forms part of the electronic system. Also an Operating system will be installed on the electronic device to manage it and other apps that will be deployed on it to aid detection and monitoring of the usage of the network. The operating system which this paper proposes is an android system. The diagram below describes the components of the electronic system.

Expert System

Microprocessor

Android OS

**Very Large Scale Integration Systems**

We will develop the electronic system using concepts from Very Large Scale Integrated Systems VLSI. Very Large Scale Integration is the process of creating an integrated circuit by combining hundreds of transistors into a single chip. An electronic system might consist of a CPU, ROM, RAM and Glue Logic.

**Components of a CPU**

The two main components of a CPU are Arithmetic and Logic Unit (ALU) and Control Unit (CU). The ALU performs arithmetic and logical operations and the Control Unit extracts instructions from memory decodes it and executes it.

**Development of Android DA**

A paper entitled development of android DA describes the development of an android DA that includes three categories of functions. The functions are conventional functions of vehicle – mounted devices, functions available by being connected to smartphones and functions specific to Android. “*Being equipped with the Android OS, the Android DA is configured with Tegra 3 microprocessor and the SH microprocessor, including real-time OS that mainly control vehicle signals and System power supply*. “

**Expert System that Detects and Monitors the Usage of the Network**

An expert system that will be used for detecting and monitoring the usage of the computer network will be developed and installed on the electronic system. The expert system can be developed using any programming language such as Java, C++, Python and Android. Since the operating system that will be installed on the electronic system is android we will use the android programming language to develop this expert system.

**DEVELOPMENT OF AN EXPERT SYSTEM FOR MONITORING THE USAGE OF WEB, DESKTOP AND NETWORK APPS**

This section of the research proposal describes how to build an expert system for monitoring web and desktop and network apps. The expert system will be developed on an android device.

**Building the Web Usage System**

To develop the web usage system, we will first define a set of classes that will be used by web apps. This classes will be defined for links, buttons and input html elements. The Web Usage System will be composed of a registration system for registering web apps that will be monitored. The data that will be managed by the registration system include the app name and the url where the app is hosted. We will also use a web framework that collects data from the web app and send it to the expert system for detecting the usage profile of the web app.

**Building the Desktop/Network Usage System**

To develop the desktop/network usage system, we will define a set of abstractions that can be used to map buttons, and inputs in a desktop/network apps. The Desktop/Network System will also be composed of a registration system for registering the desktop/network apps that will be monitored. The data that will be managed by the registration system include app name, and a link to where the app is hosted. We will also use a desktop or network system that collects data from the desktop or network app and send it to the expert system for the detection of the usage profile of the network or desktop app.

**Building the expert system**

The registration for all types of apps including web, desktop and network apps can be combined so that we can capture the app type, the name of the app and where the app is hosted. Also how to monitor the app can be done by using android classes that make it possible to read the html elements, and their contents and be able to know if in the case of buttons and links that an action such as click action has been performed and the page that action leads to. An XML parser can be used if classes for reading html elements are not found. The expert system will also compose of the usage system of the web apps and the usage system of the desktop/network apps. We will be using android Services for monitoring the apps and android AyncTask for monitoring background processes in the expert system. The expert system will also be made up of the web usage system and the desktop/network usage system. The function of the expert system is to learn the usage of the web, desktop or network apps and build a usage profile. It will also monitor the web, desktop and network apps to determine anomalous activities in the web, desktop and network apps.

**Development of Usage System**

The usage system will be developed for an application, for hosts and for the network on which host communicate with other host and on which the application software is being used. As such three usage systems will be developed. These are application usage, host usage and network usage. The usage system would be the expert system that will be used to determine and monitor the network usage.

***Application Usage***

We will build usage profile for an application using its authentication and the behavior of the user before his session expires. As such we will model the authentication and the user session of an app. In this project the medical care system developed using evolutionary computing will be used to demonstrate how a usage profile will be built for an application. The application usage system will be developed in PHP.

***Host Usage***

We will also build a usage profile for a host such as a computer or mobile phone using the usage of memory and the usage of CPU. As such the memory usage and the CPU usage model will be used to determine the host usage. The host usage system will be developed in C++.

***Network Usage***

We will also build a network usage profile using the average behavior of various hosts on the network, the average behavior of servers on the network and the average behavior of ports being used for communication on the various servers and various hosts on the network. The Network usage system will be developed in C++.

**Development of a Medical System using Evolutionary Computing.**

**Evolutionary Computing**

It must be stated that application of statistics, genetic programming and concepts of evolutionary computing can make it easy to developed android systems for medical care. These systems may be made up of hardware and software that can be developed using the concepts that will be studied during this research. Research into developing medical equipments that run on android and have sensors embedded in them may be very beneficial to this study. It will make it possible to measure medical information such as weight, height, temperature and blood pressure of a patient using those equipments. This data together with the database of diseases and their symptoms will help build expert systems for medical consultation using application of statistics, linear and non linear regression, linear and non linear programming, genetic programming and concepts of evolutionary computing.

**Research Model**

The research model is for modelling diseases that a patient has and other diseases he can get. The model is composed of two dependent variables and an independent variable. These variables are basically, the chance that a patient has an illness. The chance is computed on an hundred percent scale based on symptoms and reactions patient is having and other medical parameters such as weight, height, blood pressure, and glucose level of patient. The other factor that affects the research model is the number of years a patient has been having symptoms that point to a particular disease. The research model is given as. Y=aX1+ bX2. Note that a and b are constants.

**Research Questions**

The main questions to be investigated are listed below.

* What are the best and most efficient ways of modelling diseases for development of a medical care system for administering medical care?
* How can a mobile system for administering medical care be developed?

**Objectives of Paper**

The goal of this paper is to develop a mobile system for administering medical care.

The list below summarizes the objectives.

* Developing a Mobile System for Administering Medical Care
* Mapping a set of symptoms to a disease.
* Determining diseases that a patient is likely to acquire based on medical information

**Research Assumptions and Hypotheses**

The assumptions below form the basis of the first part of this research. Based on these assumptions, several hypotheses are formulated as part of the basis for this research.

* Sensors on an android phone can be used to measure temperature, height, weight and blood pressure.
* Weight of a patient can be computed using an accelerometer,
* Blood pressure of a patient can be computed using a senor that measures patient heart beat and blood flow.
* Temperature of patient can be predicted based on the above or knowledge of patient being feverish. It can also be measured using a sensor that measures temperature.
* Height can also be measured based on motion data measured by accelerometer and weight of patient computed using accelerometer.

Based on the above, several hypotheses as listed below can be formulated for conducting this research.

* The first group of hypotheses is based on the extent to which an android system such as a phone or medical equipment that runs on android can be used to provide medical care.
* The second group of hypotheses is based on what diseases can be diagnosed using an android system or an android medical equipment?
* The third group of hypotheses is based on the symptoms or indications that can be detected, verified or predicted for medical care using sensors on an android system?

**Evolutionary Computing**

Evolutionary computing uses recombination and mutation to seed new candidates. The process can be iterated until a candidate with sufficient solution is found or a previously set computational limit is reached. Recombination is an operation that is performed on two or more chromosomes. Mutation is an operation that is performed on one chromosome. The field uses concepts that make it possible to solve some complex problems. It derives its origin from biological evolution. This document seeks to measure medical care information like weight, height, heartbeat, and rate of blood flow using evolutionary computing and genetic programming. Evolutionary computing has several components. These components include evolutionary algorithms, genetic algorithms, evolutionary programming, memetic algorithms and genetic programming.

**Components of Evolutionary Algorithms**

The components of an evolutionary algorithm are representation, Evaluation or fitness function, population, parent selection mechanism, variation operators which are recombination and mutation, survivor selection mechanism (replacement).

**Representation**

Representation includes changing the real world into evolutionary computing world. The possible solution set which is the set of phenotypes is encoded into objects in the evolutionary computing world called genotypes. Many synonyms are used to describe elements of the two space. The genotypes are called chromosomes. Genes are placeholders and alleles describe objects in the place.

**Evaluation or Fitness Function**

The evaluation function forms the basis for selection. It is the requirement to adapt to. It defines what improvements means. From the problem solving perspective, it represents the task to solve in the evolutionary computing context. Technically, it represents a procedure that assigns a quality measure to the genotypes. Typically, the procedure is composed from a quality measure in the genotype space and the reverse representation. Often the problem to solve in an evolutionary algorithm is an optimization problem. In such cases the name objective function is used in the problem context and the fitness function is identical to or a simple transformation of the objective function.

**Population**

The population is a multiset of genotypes. The role of the population is to hold possible solutions. The population is the unit of the evolution. Genotypes are static individual objects, not changing or adapting, it is the population that does.

**CONCLUSION AND DISCUSSION**

To end this discussion, it is worth mentioning that the normal usage models and threat models described in this paper represents a computer system and it associated threats. These threats can be analyzed periodically and audited as part of a computer security audit. This research will fuel development of a threat detection system and a risk analysis system. With an electronic system which is a network gadget that has an expert system developed and installed on it, it will become possible to detect and monitor the usage of a computer network.

**REFERENCES**

Accelerometer <http://en.wikipedia.org/wiki/Accelerometer>[10th February 2015]

Evolutionary Algorithms <http://www.cs.vu.nl/~gusz/ecbook/ecbook-course.html>[20th Febuary 2016]

Introduction to Evolutionary Computing <http://www.cs.vu.nl/~gusz/ecbook/ecbook-course.html>[5th January 2016]

<http://www.dimensionengineering.com/info/accelerometers>

A, A study of information security awareness and practices in Saudi Arabia [26-28 June 2012] http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6285845&contentType=Confere nce+Publications&queryText%3DCyber+Security+Papers+.PLS.+mobile+phones [15th November, 2012]

Cashion J, Protocol for mitigating the risk of hijacking social networking sites [15-18 Oct. 2011] http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6144818&contentType=Confere nce+Publications&queryText%3DCyber+Security+Papers+.PLS.+mobile+phones [15th November, 2012]

Chidambaram Mahadevan (CISA, FCA), Intrusion, Attack, Penetration – Some Issues

Cybersecurity http://whatis.techtarget.com/definition/cybersecurity [14th November 2012]

EthemAlpadin, Introduction to Machine Learning 2nd edition 2010

Jian g Chunfeng, Research and application of behavior encryption [27-31 May 2012] http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6320096&contentType=Confere nce+Publications&queryText%3DCyber+Security+Papers+.PLS.+mobile+phones [15th November, 2012]

Larson, Hostetler, Edwards, Multivariable Calculus 8th edition 2006

Matthew E. Whiteman, Herbert J. Mattoro, Principles of Information Security 2nd edition 2005

Nong Ye, A Markov Chain model of Temporal Behaviour for Anomaly Detection [6-7 June 2017]

Security hole allows anyone to hijack your Skype account using only your email address http://thenextweb.com/microsoft/2012/11/14/security-hole-allows-anyone-to-hijack-yourskypeaccount-using-only-your-email-address/ [12th November, 2012]

Very Large Scale Integration (VLSI) <http://whatis.techtarget.com/definition/Very-Large-Scale-Integration-VLSI> [20th November 2017]

Week 2 Tutorial: Building an ALU <http://minnie.tuhs.org/CompArch/Tutes/week02.html> [20th November 2017]

Glue Logic <http://whatis.techtarget.com/definition/glue-logic> [2oth November 2017]

Central Processing Unit <https://www.webopedia.com/TERM/C/CPU.html> [21st November 2017]

Development of Android DA <https://www.denso-ten.com/business/technicaljournal/pdf/41-7.pdf> [21st November 2017]

Services <https://developer.android.com/guide/components/services.html> [22nd November 2017]

Creating a Background Service <https://developer.android.com/training/run-background-service/create-service.html> [22nd November 2017]