Project plan for a degree project

PA2512: RESEARCH METHODOLOGIES IN SOFTWARE ENGINEERING

Version NUMBER - February 21, 2014

Thesis	Tentative title	ON THE APPLICABILITY OF AN AKS-CLASS ALGORITHM IN RSA KEY GENERATION
	Classification	SECURITY AND PRIVACY CRYPTOGRAPHY PUBLIC KEY (ASYMMETRIC) TECHNIQUES*
Student 1	Name	SAI PRASHANTH JOSYULA
	e-Mail	sajo14@student.bth.se
	Social security nr	920728P834
Student 2	Name	JAYA KRISHNA RAAVI
	e-Mail	jara14@student.bth.se
	Social security nr	930705P558
Student 3	Name	RAM KUMAR CHILLA
	e-Mail	rach14@student.bth.se
	Social security nr	930726P652
Supervisor	Name and title	
	e-Mail	
	Department	
External	Name and title	
	e-Mail	
	Company/HEI	

**2012 ACM Computing Classification System: www.acm.org/about/class/2012*

1 Introduction

Public key cryptographic algorithms are based on difficult mathematical problems that have inefficient solutions. RSA algorithm (Rivest-Shamir-Adleman algorithm) is the most widely used public key cryptographic algorithm. The security of the data encrypted with the RSA algorithm rests in part on the difficulty of factoring large numbers [1]. The RSA algorithm consists of three main algorithms: key generation, encryption and decryption. The initial step of key pair generation phase is to generate large random prime numbers.

Primality testing algorithms are used to determine if a specified number is a prime number. These algorithms can either be probabilistic or deterministic. Miller-Rabin algorithm is the mainstream primality testing algorithm that is based on probability theory [2]. This algorithm is widely used in the key generation phase of RSA. It is also the recommended algorithm in Digital Signature Standard (DSS) by The National

Institute of Standards and Technology (NIST) [2]. In spite of being a fast primality test, the result produced by the Miller-Rabin algorithm is only probabilistic. A deterministic primality testing algorithm is required if the result of the primality test is to be deterministic. The choice of the primality testing algorithm is crucial as it determines the run time of the RSA key generation phase.

M. Agrawal et al, present a primality testing algorithm that runs in polynomial time [3]. The algorithm is known as AKS algorithm (Agrawal-Kayal-Saxena algorithm). AKS algorithm is the first deterministic, rigorous, polynomial-time primality proving algorithm. Several works are currently being done to improve the AKS algorithm [4][5]. But few mathematicians and computer scientists consider this algorithm to be mainly of theoretical interest [6]. They are of the opinion that it is unfeasible to apply this algorithm on numbers that are of a size of interest for practical purposes [6]. However none of the papers that state such opinions actually invalidate the applicability of AKS algorithm for practical purposes. Hence this is a research gap that remains unfilled. Due to this research gap there may be possible practical applications of AKS algorithm that are yet to be discovered. By analyzing the applicability of this algorithm in RSA key generation, we try to fill this research gap. By filling this research gap, new areas of investigation may also open up.

R. Crandall et al, refer to the "AKS class" of algorithms which are improved versions of the original AKS algorithm [4]. Carl et al, present an updated version of their paper that explains an AKS variant algorithm [5]. Among the AKS class of algorithms, we will identify the algorithm that has greatest efficiency for large input sizes. Our work analyzes the applicability of this algorithm for primality testing in RSA key generation phase by analyzing the results of its comparison with the Miller-Rabin algorithm. By the end of this research, we endeavor to fill the identified research gap.

2 Aim and objectives

The main aim of the research is to analyze the applicability of an efficient AKS class algorithm in RSA key generation phase, by comparing its performance with the Miller Rabin algorithm.

The major objectives of the research are:

a) To compare the performance of the AKS class algorithm with the Miller-Rabin algorithm.

b) To discuss the applicability of the AKS class algorithm in RSA key generation phase, based on the results of the comparison.

3 Research questions

1) Why is the Miller-Rabin algorithm widely used in RSA key generation in spite of the availability of a deterministic polynomial time algorithm as an alternative?

2) How does the use of AKS algorithm for primality testing in RSA key generation phase affect the phase?

4 Method

This research will be based on quantitative methodology to achieve our major research objectives and research aim. The quantitative research methodology is used for collecting quantitative data from controlled experiments and is well suitable for comparison and statistical analysis [7]. As we propose to analyze the results of comparison between the two chosen algorithms, performing an experiment is suitable for our research project. Hence we will use experiment as quantitative methodology to analyze the applicability of the AKS class algorithm in RSA key generation, by comparing its performance with the Miller-Rabin algorithm.

The input size and the actual input will be taken as the independent variables and the running time will be taken as the dependent variable. The independent variables are carefully manipulated under known, tightly defined and controlled conditions [8]. The software implementation and the hardware platform will be kept unchanged. As our input sizes, we will choose the most practical and common key sizes in RSA key generation phase. We will choose a random sample of inputs for each input size. We will have a collection of samples, each one pertaining to a specific input size. We carry out the experiment with these samples and record the running times of both the algorithms.

We perceive threats to external validity, statistical conclusion validity, construct validity and context validity. The risks and mitigation methods are described in detail in the following section that deals with risks.

We expect to analyze the results of our research by carrying out a statistical analysis using SPSS (Statistical Package for the Social Sciences). We will plot two types of graphs. One of the types of graphs will have input size as the independent variable and running time as the dependent variable. Another type of graph will have actual input as the independent variable (for a fixed input size) and running time as the dependent variable. We will plot such graphs for both the algorithms and compare them. This analysis method allows us to easily compare performance of both the algorithms by examining the slope of the graphs. Hence this analysis method is well suitable for our particular research project and research methodology.

5 Expected outcomes

The expected outcomes of the research are:

a) A comparison of performance of the AKS class algorithm and the Miller-Rabin algorithm.

b) An analysis of the applicability of the AKS class algorithm in RSA key generation phase, which is derived from the results of the comparison.

6 Time and activity plan

The time plan for our activities throughout the course period is as follows:

20140121: Group meeting to discuss about possible research topics

20140125: Select keywords and search strings on the selected research topic
20140127: Search studies based on selected keywords and search strings
20140129: Study the selected studies
20140131: Start writing the first draft for the systematic literature review
20140202: Group meeting to discuss about the systematic literature review
20140204: Group meeting to finalize the systematic literature review
20140207: Submission of systematic literature review
20140210: Group meeting to discuss about the research proposal
20140212: Start writing the first draft for the research proposal
20140213: Group meeting to discuss about the research proposal draft
20140214: Start writing the final research proposal
20140217: Group meeting to finalize the research proposal
20140221: Submission of research proposal
20140226: Group meeting to discuss about the research paper
20140301: Start writing the first draft copy of the research paper
20140302: Group meeting to discuss about the research paper draft
20140304: Correct the research paper draft
20140306: Write the final research paper
20140311: Revise and check the research paper
20140314: Submit the research paper

7 Risk management

TABLE I
IDENTIFICATION OF RISKS, DESCRIPTION AND MITIGATION STRATEGIES

Risks	**Description**	**Risk Mitigations**
Software implementation bias (External Validity Threat)	The software implementation of one algorithm may be better than that of the other algorithm.	We will try to reduce the bias that occurs due to difference in software implementations of both the algorithms.
Hardware platform bias (External Validity Threat)	The hardware platform may be biased towards a particular algorithm.	We will try to reduce the advantages that an algorithm may have over another algorithm due to the hardware platform on which they are implemented.
Sample selection bias (External Validity Threat)	The chosen samples may be biased or non-random.	We will try to randomize the study samples so that they represent the general population.
Unreliable measurements (Statistical Conclusion Validity Threat)	While measuring the variables, large amounts of measurement errors can lead to incorrect conclusions.	We will try to measure the variables using reliable methods and avoid measurement errors.

Convergent validity threat to applicability of the algorithm (Construct Validity Threat)	The measures of running time may not be related to applicability of the algorithm to the expected degree.	To some extent the practical applicability of the algorithm also depends upon the space complexity of the algorithm. We will try to consider this while making conclusions.
Context Validity Threat	The results of the experiment may not be applied in real world scenario to the expected degree.	We will try to set up the experiment in such a way that the results are applicable when RSA key generation phase is considered too.

References

[1] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[2] Li Dongjiang, Wang Yandan, and Chen Hong, "The research on key generation in RSA public-key cryptosystem," in *2012 Fourth International Conference on Computational and Information Sciences (ICCIS), 17-19 Aug. 2012*, 2012, pp. 578–80.

[3] M. Agrawal, N. Kayal, and N. Saxena, "PRIMES is in P," *Ann. Math.*, pp. 781–793, 2004.

[4] R. E. Crandall and J. S. Papadopoulos, "On the implementation of AKS-class primality tests," *Adv. Comput. Group Apple Comput. E Univ. Md. Coll. Park*, 2003.

[5] H. W. Lenstra Jr and C. Pomerance, *Primality testing with Gaussian periods (2011)*. .

[6] P. Berrizbeitia and A. Olivieri, "A Generalization of Miller's Primality Theorem," *Proc. Am. Math. Soc.*, vol. 136, no. 9, pp. 3095–3104, Sep. 2008.

[7] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering*. Springer, 2012.

[8] L. Blaxter, C. Hughes, and M. Tight, *How to research*. McGraw-Hill International, 2010.