

University of Business and Technology in Kosovo UBT Knowledge Center

UBT International Conference

2018 UBT International Conference

Oct 27th, 9:00 AM - 10:30 AM

Combining Cryptographic Primitives According to Security Metrics and Vulnerabilities in Real Systems

Blerina Çeliku

Fan S. Noli University, blerinaceliku81@gmail.com


Rafail Prodani

Fan S. Noli University, rprodani@yahoo.com

Emis Simo

Fan S. Noli University, emis.simo@live.com

Follow this and additional works at: <https://knowledgecenter.ubt-uni.net/conference>

 Part of the [Computer Sciences Commons](#), and the [Digital Communications and Networking Commons](#)

Recommended Citation

Çeliku, Blerina; Prodani, Rafail; and Simo, Emis, "Combining Cryptographic Primitives According to Security Metrics and Vulnerabilities in Real Systems" (2018). *UBT International Conference*. 86.

<https://knowledgecenter.ubt-uni.net/conference/2018/all-events/86>

This Event is brought to you for free and open access by the Publication and Journals at UBT Knowledge Center. It has been accepted for inclusion in UBT International Conference by an authorized administrator of UBT Knowledge Center. For more information, please contact knowledge.center@ubt-uni.net.

Combining Cryptographic Primitives According to Security Metrics and Vulnerabilities in Real Systems

Blerina Çeliku¹, Rafail Prodani², Emis Simo³

¹Department of Mathematics, Informatics and Physics, “Fan S. Noli University” blerinaceliku81@gmail.com

²Department of Mathematics, Informatics and Physics, “Fan S. Noli University” rprodani@yahoo.com

³Department of Mathematics, Informatics and Physics, “Fan S. Noli University” emis.simo@live.com

Abstract. There are so many applications and data that flow during our daily activities, either personal or institutional ones. Also the companies and business do transactions or their real operations through the Web and other Internet facilities. Security breaches are costing individuals or companies millions so information security has to be a major priority. There are several forms of security technology available, but encryption is one that everyday computer users should know about. Encryption and the performance of cryptographic algorithms are variable according to implemented platforms, software and hardware components or application scenarios. According to specific security metrics and requirements we have to use algorithms even in a combined manner that should be more efficient and best suited. Multi-level encryption and hybrid encryption should be a fine solution to protect our data. In this paper we specify the security musts according to Alfa software using cryptographic algorithms.

Keywords: Information Security, Encryption, Security breach, Alpha software.

1 Introduction

Information security is a common subject to many companies, institutions and nowadays more and more to individuals. Cryptography plays a crucial role for protecting data and minimizing security problems that we face every day. Cryptographic techniques are of two types-Symmetric and Asymmetric. Symmetric, if both the sender and the receiver of information are using the same private key. Cryptographic technique is asymmetric, if sender and receiver are using different keys, typically a public for encryption and a private for decryption. The security level of the encryption algorithm should depend on the size of the key space, secrecy of the key, length of the key, initialization vector and how they all work together.

According to [1] there are three main criteria should be considered at the same level of importance to evaluate new cryptosystems: how much it eases implementation, level of security, and efficiency. Cryptography provides a number of security goals to ensure the privacy of data, non - alteration of data and so on. Following are the various goals of cryptography [2]: Confidentiality, Authentication, Integrity, Non Repudiation and Access Control. The encryption techniques are changing rapidly as computer technology has such a tremendous increase. Security products are being developed to address the security needs of an information intensive society and according to [3] there are many security objectives related to information security that can be achieved using different cryptographic algorithms and schemes. Symmetric or asymmetric they vary according to their characteristics and general performance and security measured experimentally or given from cryptographic literature dealing with vulnerabilities or several cryptographic attacks.

Encryption can apply a level of security virtually impossible when it is implemented correctly. In practice, most failures in cryptographic systems derive not from weaknesses in the algorithms used but rather from the exploitation of subtle flaws in the way the algorithms are implemented or through the exploitation of interactions between algorithm implementations and the environments in which they operate.

2 Cryptographic Algorithms and Security Assumptions

Technology and the tremendous changes about computers have a high impact in the way a cryptographic algorithm is implemented and in its performance. There are many advantages and disadvantages of both categories of encryption; however encryption according to NIST is one of the core security technologies that build guidelines around. The standard implementation process of a cryptographic standard revolves around nine key values, which include *usability, integrity, transparency, and global acceptability*. There are new forms of encryption nowadays such as hybrid encryption where we use the best features of a specific algorithm such as scalability, speed of encryption or decryption, throughput, power etc. Another form of encryption has to do with format preserving which makes long strings of numbers indecipherable in both binary and decimal formats; and instead of fitting the data to the environment, the new format adapts the data into the environment. Heartland Payments Systems switched to format-preserving encryption after a 2009 hack, which saw more than 130 million credit and debit card numbers compromised [4].

According to their structure and use scenario cryptographic algorithms behave differently and have specific performance metrics on various applications. Asymmetric encryption is the natural tool to use when we want to allow for confidential transmissions between any two users among a big population. Symmetric algorithms are very fast in nature and they run faster than asymmetric key algorithms such as RSA etc. and the memory requirement of symmetric algorithms is lesser than asymmetric encryption algorithms. However, combining cryptographic primitives has been resulted to be the most efficient scheme to protect sensitive data.

Security assumptions about using cryptographic algorithms and protocols deal with developing robust security definitions that reflect reality as accurately as possible.

We need to rely on different cryptographic assumptions such as the assumption that factoring a number n into its two prime factors p and q is difficult.

There are several constraints according to specific algorithms that direct their use on specific application scenarios. RSA has some operational constraints. Due to these constraints, we do not usually encrypt data *directly* with RSA; instead, we select a small sequence of random bytes, which we call *session key*. We encrypt the session key with RSA; and then we use the session key with a *symmetric* encryption algorithm to process the whole message. This is called *hybrid encryption*. Some recent variants of RSA (with the 'OAEP padding' from PKCS#1 v2.0) internally use hash functions. Hash functions are good "randomizers" and this makes them appropriate for building more elaborate cryptographic algorithms with good security features [5].

2.1 Hardware versus Software Implementing of Algorithms

However, for all practical applications, performance and the cost of implementation are also important concerns. A data encryption algorithm would not be of much use if it is secure enough but slow in performance because it is a common practice to embed encryption algorithms in other applications such as e-commerce, banking, and online transaction processing applications. A software implementation of a cryptography scheme provides the benefits of flexibility, speed of implementation, and lower cost over time. Competitive hardware encryption cannot be updated without replacing the microcontroller, which is costly and complicated [6]. Speed of encryption, throughput, cost of implementation, energy consumption etc. are the main concerns on evaluating efficient and proper encryption scheme.

There is a trade-off between efficiency and protection cost of a software based encryption versus a hardware based encryption. Software- based encryption solutions use a distributed key storage mechanism: keys are stored on the application and database servers on which the data to be encrypted resides.

For large organizations as key management complexity increases the risk of not-backing up a key or losing a key increases exponentially; furthermore this approach poses security vulnerabilities because of incorrect configuration of application and database servers. There are several security platforms that manage key operations that use robust cryptographic algorithms such as RSA, AES, 3-DES that can be used by different application and database servers. Furthermore we can add an extra security level with hardware security compliant module. Software-based encryption solutions generally provide one implementation option: deploying encryption at the database layer although sometimes it can change according to infrastructure requirements or security objectives. All cryptographic operations are performed on the application or the database server's CPU and this doesn't effect on scalability. Generally SW based solutions require a smaller initial investment than HW based solutions but this doesn't mean that the overall cost is not changing especially in complex enterprise environments. If we have a smaller enterprise the requirements, security objectives and main data security threats are easier to specify.

3 Business Data, SQL Vulnerabilities and System Attacks

Nowadays our life and business operations are evolving tightly with different web applications and in a continuous manner we are using several of them to fulfill our daily needs such as shopping, banking, ticket booking etc.

The WWW has evolved from a system that delivers static pages to a platform that supports distributed applications, referred to as web applications, and has become one amongst the foremost rife technologies for data and service delivery. Multiple services are available via single click through various web applications; there is no need to stand in long queues at the banks or market to buy for the modern trends. As web applications are increasingly used to deliver essential services, they become a valuable target for security attacks. Many web applications interact with back-end database systems, which may store sensitive or confidential information such as related to finance, health etc. [7].

One of the most serious attacks on web applications is known as SQLIA (SQL injection attack) (OWASP Top 10 2013). SQLIA is considered a severe of attack affecting confidentiality, integrity and availability of information. The best way to find out if an application is vulnerable to injection is to verify that all use of interpreters clearly separates untrusted data from the command or query. For SQL calls, this means using bind variables in all prepared statements and stored procedures, and avoiding dynamic queries. SQL injections are attacks by which an attacker changes the structure of the original SQL query by injecting SQL code in the input fields of the web form in order to obtain unauthorized access to the database.

SQL injection allows attacker to create, read, update, modify or delete data stored within the back-end database; and the malicious user injects SQL commands into SQL statements through the input of online webpage. This kind of attack against Card Systems on June 2005 put out of business these credit card payment processing companies and a great number of unencrypted credit cards were stolen or exposed. The countermeasures techniques to detect and prevent these attacks are static, dynamic and hybrid. There are so many different attacks against data that are either in storage or in transmission therefore effective security mechanisms seem to be very important. Man in the Middle attack is another kind of network attacks that is very common nowadays on communications between two parties, often client/server situations. In this attack, a third party pretends to be the server that a client is trying to connect to, and when the client connects, sends its request to the actual server it wants to connect to. It takes the response the actual server sent back to it and sends it back to the client. In this paper we describe the vulnerabilities that exist in our system with Alpha software and SQL Server and the main attacks (MITM) that may occur in this scenario.

4 Crypto in Practice and Hybrid Encryption

4.1 Alpha Software Analysis

There are a large number of electronic transactions, including e-commerce; e-banking, e-voting, e-learning and e-health among others can be conducted online at any time and from anywhere.

All these applications are exposed to hacking attempts and security-related problems. We are simulating an MITM attack against a system that uses “Kontabiliteti Alfa” software and there are several SQL vulnerabilities that are present. We have to prevent the disclosure of our sensitive transactional data using cryptographic algorithms in a dual operation and the database attacks can be prevented applying encryption techniques in a hybrid integrated approach.

There are many tools that help us to do the prevention and encryption of data communication link or data encryption in database and generally protocols include some form of endpoint authentication specifically to prevent attacks. We have a LAN with computers that operate with the software Alpha. After we have worked with this program we noticed that there are some vulnerabilities when execute several queries. The program and the SQL server operate together and the data is in open during linking and in storage. That is why we have to protect these data. The SQL statements can be executed with encrypted connection string but we try to do that without encryption to understand the vulnerabilities. From the client computer we execute a .NET script for executing some queries such as: (*"SELECT pike_shitje, ref, nr_dok, date_in, date_out, kursi, klienti_kodi, klienti_emertimi, klienti_detyrimi, menyre_pag, zbritje, monedha_baze, shuma, totali FROM a_reg_shitje WHERE ref_id>0 and ref_active=true", con*)). To demonstrate these vulnerabilities we simulate the MITM attack and after that we proceed with use of algorithms.

4.2 Simulation of MITM Attack and Use of Encryption

The most widely used forms of MITM attacks include ARP cache poisoning, DNS spoofing, HTTP session hijacking, passing the hash, and more; and from the real practice most of the victim machines are Windows-based hosts. One of the oldest forms of modern MITM attack, ARP cache poisoning allows an attacker on the same subnet as its victims to eavesdrop on all network traffic between the victims and it takes advantage of the insecure nature of the ARP protocol.

This is a real threat on modern networks and furthermore very difficult to detect and defend against [8]. There are three laptops in our experiment that perform MITM attack and some tools such as Ettercap, which has both Windows and Linux versions and a great deal of functionality in many types of MITM attacks. Intrusion Detection Systems could be a fine way to pick up most forms of the ARP cache poisoning and DNS spoofing. We have a client computer, a server and the attacker and they have their IP values: Server IP: 192.168.1.2, Client IP: 192.168.1.4 and Attacker IP: 192.168.1.3. After the basic configurations we execute the *sqlinject.sh* (-o [original SQL query] -i [new SQL query] -s [MSSQL Server IP] -c [SQL Client IP]) script that creates an Ettercap filter that will identify a SQL string and replace it with a new one. The script compiled the filter and run Ettercap with the filter loaded. Ettercap performs an ARP spoofing attack against the specified IP addresses automatically as it is shown in Figure 1. After MITM attack executing we have to use an encrypted connection and for that reason we have used the public key cryptography such as RSA.

```

Applications ▾ Places ▾ Terminal ▾ Sun Mar 26, 22:56:13
root@hostname: ~
File Edit View Search Terminal Help
root@hostname:~# sh SQLInject.sh -o "SELECT kod_klient,kod_produkt,ref_id,ref_kod,monedha,totali,shuma,nr_dok,nr_prot FROM a_shitje_fat_ref1 WHERE kod_klient=1" -i "CREATE LOGIN hacker WITH PASSWORD='password01'" -c 192.168.1.4 -s 192.168.1.2
Opening ettercap..
Ettercap filter on
Sniffing and filtering td.query
Query found
!SELECT kod_klient,kod_produkt,ref_id,ref_kod,monedha,totali,shuma,nr_dok,nr_prot FROM a_shitje_fat_ref1 WHERE kod_klient=1
Replacing and injecting new query
CREATE LOGIN hacker WITH PASSWORD="password01"
Converting original query to hex...
Converting new query to hex...
Writing ettercap filter now...

Completed Successfully!
root@hostname:~# █

```

Fig.1 Ettercap Sniffing and filtering td.query

For database encryption we have used a master key, a certificate and a symmetric key algorithm such as 3-DES or AES. RSA has some disadvantages when we have large blocks of data to transmit therefore RSA is best suited for use in conjunction with a secret-key cipher such as AES or others [9]. This is a hybrid cryptosystem that can be used on different application scenarios.

There are so many companies in our country that operate on their business using Kontabiliteti Alfa software. If the data of clients gets compromised then all the business integrity and reputation gets a big negative impact.

There are so many cloud services that have become common during daily business operations. Also there are many flaws, vulnerabilities and attack holes present as we already treat in this paper. That is why we study in specific MITM attacks as the most common ones and also give the right protection issues. We have to be careful especially to secure our internal machines.

If we have to apply a kind of encryption we already know how specific cryptographic algorithms behave in certain cases and according to crypto literature we can choose the most efficient one. In the future, we plan to propose an efficient encryption scheme when we use a cloud service.

References

1. M. El-Wahed, S. Mesbah, and A. Shoukry, "Efficiency and Security of Some Image Encryption Algorithms", Proceedings of the World Congress on Engineering 2008 Vol. I WCE 2008, July 2 - 4, 2008, London, U.K.
2. E. Surya, C.Diviya "A Survey on Symmetric Key Encryption Algorithms", International Journal of Computer Science & Communication Networks, Vol 2(4), ISSN: 2249-5789.
3. A. J. Menezes, Paul C. van Oorschot, Scott A. Vanstone "Handbook of Applied Cryptography".
4. New-nist-guide-provides-new-standard-for-protection-pii; <https://www.fedscoop.com/>.
5. Five common encryption algorithms and the unbreakable of the future, www.storagecraft.com.
6. Pratap et al., International Journal of Advanced Research in Computer Science and Software Engineering 2(9), Sep - 2012, pp. 196-201 © 2012, IJARCSSE All Rights Reserved Page | 201.
7. Mira K. Sadar, Pritish A. Tijare, Swapnil N. Sawalkar, "Securing Web Application against SQL Injection Attack, a Review", International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169 Vol: 2 Issue: 3.
8. Understanding Man-in-the-Middle-Attacks-ARP Cache Poisoning, <http://techgenix.com>.
9. Brian Reindel, "A Hybrid Cryptosystem Using Java, AES (secret-key) and RSA (public-key) Encryption", Sept. 21, 2015.