# Password Reduction Method and Secure Communication between Devices

Onur Çakırgöz, Süleyman Sevinç
Computer Engineering Department, Dokuz Eylul University,
İzmir, Turkey
{onurcakirgoz, suleyman.sevinc}@cs.deu.edu.tr

***Abstract*—Establishing and maintaining a secure and secret communication between devices is one of the fundamental problems. Unsecure communication can contain numerous vulnerabilities and these vulnerabilities are open doors which enables malicious people realizing their goals. Moreover, in some cases, these vulnerabilities can lead to fatal consequences. The initial step required to ensure secure communication, which has importance at this level, is authentication. The most commonly used method for authentication is the username and password pair. In an environment with many devices, there are two alternatives for any device to verify its identity to other devices. In the first alternative, each device has only one unique password and uses this password to authenticate itself to all other devices. It is obvious that the first alternative is not safe. In the second alternative, each device has as many passwords as the number of other devices and uses the corresponding password to authenticate itself to the device it wants to communicate with. At this point, managing a certain number of passwords is a separate problem. In this study, starting from the question "Can we reduce many different passwords to a single password with a mathematical function?", we have developed a password reduction method which is based on the Chinese remainder theorem. Chinese remainder theorem is about finding a solution to the system of simultaneous congruencies and expresses that this system of simultaneous congruencies has a unique solution, and this unique solution is the unique password that the device should know, namely the reduced password, according to our method. The prime numbers in the modulo processes are distributed to the other devices. On the other hand, the remainders of the modulo processes, namely many passwords initially generated, are discarded. In addition, within the scope of this study, a secure authentication and communication protocol has been developed by using encryption, various mathematical and logical operations and one-way hash function to perform the communication in a confidential manner.***

***Keywords*—*password reduction; identification; authentication; chinese remainder theorem; communication***

## I. Introduction

Use of passwords is an ancient method of proving identification and still the most widely used authentication method for devices and electronic services [1]. Password authentication is economical and relatively easy to use, which are probably the main driving factors behind its popularity and extensive usage. However, as the number of electronic devices is on the increase so is the number of passwords individual devices must manage [2].

Increase in the number of mobile devices that require a device to maintain more than one password has consequences:

- Each mobile device must select many passwords, at least one for each device.
- Each mobile device must safely store many passwords.

Selection of many passwords is actually an easy task for mobile devices and passwords must be randomly chosen for obvious security reasons. Because, non-random and weak passwords are known to be prone to dictionary attacks [1]-[5]. Passwords selected by a device for various other devices must be sufficiently far from each other. In the worst-case, when a device accesses n devices by a single password, a malicious device can impersonate it and have illegal access to these n devices.

From an information theoretical point of view, user (person) authentication and device authentication are analogous. For user authentication, various measures have been developed by service providers and by researchers to make password-based authentication securer. Enforcing stronger passwords [1], [2], [10], one time password protocol [6], [21], enforcing renewal of passwords periodically, authentication protocols which aim to prevent certain attacks [3]-[6], remote authentication protocols using smart cards [12]-[16], account locking based on number of unsuccessful trials are a few of those. Because of the analogy between user authentication and device authentication, some of these techniques can be adapted to device authentication with minor alterations.

Other approaches for management of password-based authentication exist. Kerberos [7]-[9] is a well-known password-based authentication protocol which provides secure access to network resources where service providers trust a centralized authority which is charged for the management of user authentication. SAML [11] is an approach which allows trusted parties to refer their authenticated users to other services. These latter approaches require a prior agreement between service providers to enable their users for single point authentication. These methods aren’t applicable to mobile devices, especially for simple and low-capacity ones.

Password reduction method presented in this article reduces many passwords created for different mobile devices to a single password through a mathematical procedure. Individual passwords when needed can be obtained through an inverse conversion procedure. Thus, without any loss of security, this approach has the potential to increase the usability of password-based authentication.

Although password reduction method provides managing many passwords through a unique password, it is not enough for a secure authentication process. To be able to perform authentication securely, we need a protocol which should be compatible with password reduction method. Our secure authentication protocol is developed upon password reduction method (modulo operation) and naturally it is completely compatible with password reduction method. Accordingly, it enables a device using a unique password for different mobile devices securely. Additionally, it ensures identity verification for both sides. By this protocol, a device can authenticate itself to different devices with a unique password securely.

## II. Mathematical Basis

A password is a string of symbols drawn from an alphabet. These symbols can be letters, numbers or punctuation marks. Let k be the size of the alphabet. Then we treat each password as an integer base-k. Conversion of a password string to a base-10 number can be done using standard base conversion techniques. In our experiments, for base-10 conversions, we use Horner's method, described originally to compute the value of a polynomial at a given point, details of which can be found in standard textbooks of mathematics. Reverse conversions from base-10 are done using the widely used method of continuous division.

Clearly the strength (unpredictability) of a password depends on the length and the choice of symbols that make of it. However, treating each possible password as an integer, allows us to bring in the techniques of number theory. From an information theoretical point of view, strictly speaking, it would be just the same whether we represent a password as a string of symbols or an equivalent base-10 number.

## III. Formulating the Problem

The problem is to reduce n different passwords to a single password. Since each password, in our approach, is treated as a number we can re-define the problem as identifying a function:

$$f: Z^n \to Z \tag{1}$$

For example, function f can be defined as a simple arithmetic multiplication of n passwords. However, in general, it would be very inefficient to extract individual passwords because factoring an integer is not an easy task. So, function f, defined as in (2), must be invertible at a constant cost.

$$f^{-1}: Z \to Z^n \tag{2}$$

Then, the relationship between the unique password and individual passwords can be defined such that,

$$f(z_1, z_2, \cdots, z_n) = Y \tag{3}$$

$$f^{-1}(Y, x_i) = z_i \tag{4}$$

Here Y represents the single password and $x_i$ represents additional knowledge needed for inversions. Ideally, no additional knowledge would be required.

## IV. Password Reduction Method

There may be a number of classes of f functions satisfying the requirements above. One such class involves computing polynomials. For example, n passwords may constitute the coefficients of a given polynomial. Therefore, the value of this polynomial at a given point would constitute X, the single password. However, this choice of function f would not have the desired properties.

The password reduction method that will be used in our study is based on an ancient theorem which is frequently used in number theory. This theorem is known as Chinese remainder theorem and was originated by a Chinese mathematician Sun Tzu. Chinese remainder theorem has found place widely in the literature [17]-[19]. In 2007, S. Iftene presented a multi-authority e-voting scheme based on CRT [20]. In 2010, J. C. Patra et al. proposed a novel Chinese remainder theorem (CRT)-based technique for digital watermarking [22]. In 2011, S. K. Kim et al. proposed new modular exponentiation and CRT recombination algorithms secure against all known power and fault attacks [23]. In 2014, C. C. Chang et al. proposed a novel multi-image threshold sharing scheme based on Chinese remainder theorem and Lagrange interpolation [24]. Also in 2014, K. Kaya, and A. A. Selçuk proposed a new threshold scheme for the Digital Signature Standard (DSS) using Asmuth–Bloom secret sharing based on the Chinese Remainder Theorem (CRT) [25].

Chinese Remainder Theorem is about finding a solution to the system of simultaneous congruencies. Suppose that *X*, *a* and *p* are positive integers. Then, (5) defines a congruence.

$$X \equiv a (mod\ p) \tag{5}$$

A system of simultaneous congruencies is defined in (6). Here $p_1$, $p_2$, ..., $p_n$ should be pairwise coprimes. Then, this system of simultaneous congruencies has a unique solution X (mod r).

$$X \equiv a_i\ (mod\ p_i)\ (i = 1,2,\cdots,n) \tag{6}$$

Given,

$$r = \prod_{i=1}^{n} p_i \tag{7}$$

Let,

$$M_i = \prod_{j=1, j\neq i}^{n} p_j\ \ (i = 1,2,\cdots,n) \tag{8}$$

Then, *X* is computed as in (9):

$$X = (\sum_{i=1}^{n} a_i M_i (M_i^{-1} mod\ p_i))\ (mod\ r) \tag{9}$$

Extracting individual passwords from X is straightforward.

Procedure outlined above is called the reverse direction computation of X. Forward direction is also possible. The device chooses, for X, a big integer, then generates n random prime numbers large enough and lastly it computes $a_i$ 's as above from X.

Since $p_i$'s are required for extraction of individual passwords from the single password X, we need to store them somewhere. We think that they can be handed over along with $a_i$'s to each mobile device, respectively. At time of authentication, devices may be queried for these random primes.

Any device with the knowledge of a single pair ($a_i$ , $p_i$) cannot guess any of the other $a_j$ where $i \neq j$. In addition, since individual passwords, i.e. $a_i$'s, are computed using random primes, they are not likely to be prone to dictionary attacks. Moreover, these passwords are very likely to be distributed over the password space randomly.

TABLE I. THE SYMBOLS USED IN SECURE SCHEME

| Symbol | Explanation |
|---|---|
| D | Device |
| $D_i$ | I'th device |
| ID | Device-id |
| h( ) | Secure One-way hash function |
| $\oplus$ | Exclusive-OR operation |
| X | Unique password |
| x | Left-half of the unique password |
| $c_j$,$c_{j+1}$ | Challenges (Randomly generated integers) |
| $a_i$ | I'th individual password for $D_i$ |
| $p_i$ | I'th individual prime number for $D_i$ |
| r | The multiplication of individual primes |
| $PU_d$ | Public key randomly generated by D |
| $PR_d$ | Private key randomly generated by D |
| SK | Symmetric key |
| D(PU, M ) | Decrypt M using PU |
| E(PU, M ) | Encrypt M using PU |
| % | Mod operator |
| \|\| | Concatenation |
| $I_f$ | Information field at the beginning of the message |

## V. AUTHENTICATION PROTOCOL

Before devising the authentication protocol, we should consider password reduction method in the perspective of a malicious person. Unfortunately, password reduction method depicted above is vulnerable to some attacks. These attacks are:

- A malicious device $D_i$ may send other prime numbers to the device and may store the received $a_i$'s. Then, it may try to compute X by using the ($a_i$ , $p_i$) pairs.
- If a malicious device sends a $p_i$ which is greater than X, it can obtain X easily.

The second attack can be eliminated by checking whether the prime number is greater than X. Moreover, if the device stores r, both attacks aforementioned can be eliminated.

The main reason causing the above vulnerabilities is the storage of $a_i$ and $p_i$ in an open format. Besides, a device listening to this transmission can capture the messages and obtain authentication information readily. To eliminate these deficiencies, $a_i$ and $p_i$ should be concealed in some way. At this point, some cryptographic and mathematical means such as one-way hash function, xor operation, concatenation, encryption and challenge-response can be used to conceal.

Taking the vulnerabilities of the password reduction method into consideration, we have developed a secure and efficient authentication protocol. The secure authentication protocol with any device consists of two phases. These phases are registration phase and login phase. The symbols used in secure scheme are defined in Table 1.

### A. *Registration Phase*

We assume that the following transaction takes place over a secure channel.

Device D generates a $c_j$ value and calculates following:

1. $A_{i,j} = h(a_i \oplus (x \oplus c_j))$,
2. $AV_{i,j} = h(A_{i,j})$,
3. $P_{i,j} = E(x, (p_i \oplus h(x \oplus c_j)))$,

For xor operations above, we assume that the operands have equal bit-length.

Here, $A_{i,j}$, $AV_{i,j}$ and $P_{i,j}$ are called pseudo-password for jth session, pseudo-password verification information for jth session and pseudo-prime number for jth session, respectively.

$AV_{i,j}$, $P_{i,j}$ and $c_j$ are sent to the device $D_i$ over a secure channel for registration and are placed in a local database by the device $D_i$. Note that $A_{i,j}$ isn't sent. Later, the device $D_i$ will use this information to authenticate the device D.

### B. *Login Phase*

When device D wants to login to device $D_i$, the following sequence of events take place:

1. Generate public-private key pair($PU_d$ , $PR_d$) randomly.
2. Compute $MES_1 = PU_d \parallel ID$. Here, the length of $PU_d$ is fixed. Hence, $PU_d$ and ID can be easily distinguished and obtained.
3. Send $MES_1$ to device $D_i$.

After device $D_i$ receives the message $MES_1$, it performs the following operations:

1. Separate $MES_1$, obtain $PU_d$ and ID.
2. Get pseudo-prime number $P_{i,j}$ and challenge $c_j$ which

are related to ID from local database.

3. Generate symmetric key SK, randomly.
4. Compute $MES_2 = E(PU_d, (c_j \parallel SK \parallel P_{i,j}))$. Here, the lengths of $c_j$ and SK are fixed.
5. Send $MES_2$ to device D.

When device D receives the message $MES_2$ from device $D_i$, it performs the following steps:

1. Decrypt $MES_2$. $D(PR_d, MES_2) = (c_j \parallel SK \parallel P_{i,j})$.
2. Separate decrypted $MES_2$, obtain $c_j$, SK and $P_{i,j}$.
3. Decrypt $P_{i,j}$. $D(x, P_{i,j}) = (p_i \oplus h(x \oplus c_j))$. Since device D knows x and $c_j$, it can compute $p_i$ as $(p_i \oplus h(x \oplus c_j) \oplus h(x \oplus c_j))$.
4. Perform the operation (r % $p_i$). Here, remember that r is the multiplication of individual primes and is stored by the Device D. If the result of the operation isn't zero, stop the session. If $p_i$ is valid, authenticate device $D_i$, perform (X % $p_i$) and obtain $a_i$.
5. Compute the pseudo-password for the current session. $(A_{i,j} = h(a_i \oplus (x \oplus c_j)))$
6. Generate new challenge $c_{j+1}$ which will be used for the next session, randomly.
7. Compute the pseudo-password verification information to be used for the next session. $AV_{i,j+1} = h(h(a_i \oplus (x \oplus c_{j+1})))$
8. Compute the pseudo-prime number to be used for the next session. $P_{i,j+1} = E(x, (p_i \oplus h(x \oplus c_{j+1})))$
9. Compute $MES_3 = E(SK, (I_f \parallel A_{i,j} \parallel c_{j+1} \parallel AV_{i,j+1} \parallel P_{i,j+1}))$. Here, the information field ($I_f$) stores the lengths of each information in the message and its length is also fixed.
10. Send $MES_3$ to device $D_i$.

After device $D_i$ receives the message $MES_3$ from device D, it performs the following steps:

1. Decrypt $MES_3$. $D(SK, MES_3) = (I_f \parallel A_{i,j} \parallel c_{j+1} \parallel AV_{i,j+1} \parallel P_{i,j+1})$.
2. Separate decrypted $MES_3$ by using $I_f$, obtain $A_{i,j}$, $c_{j+1}$, $AV_{i,j+1}$ and $P_{i,j+1}$.
3. Compute $AV_{i,j}$ with received $A_{i,j}$. $AV_{i,j} = h(A_{i,j})$.
4. Check the computed value with the stored pseudo-password verification information. If they are equal, authenticate device D. From now on, device D and device $D_i$ can communicate each-other in an encrypted manner by using SK. Otherwise reject the authentication request.
5. If device D is authenticated, replace $c_j$ with $c_{j+1}$, $AV_{i,j}$ with $AV_{i,j+1}$ and $P_{i,j}$ with $P_{i,j+1}$, immediately.

The main factors making our authentication protocol secure can be expressed as the randomly generated keys, randomly generated challenge, pseudo-password and pseudo-prime number. Since the pseudo-password and the pseudo-prime number are created by using randomly generated challenge, they are specific for the session, namely in each session, a different pseudo-password and a different pseudo-prime number are utilized. Naturally, the submitted and received messages differ in each session. Although many means depicted above to strengthen our protocol, we should consider and evaluate it from the perspective of an adversary. First of all, any device $D_i$ holding the authentication information can't gather any information about the values of $a_i$ and $p_i$ from $A_{i,j}$, $AV_{i,j}$ and $P_{i,j}$. Because $A_{i,j}$ and $P_{i,j}$ are in scrambled form and they were deliberatively devised in order to hide $a_i$ and $p_i$, respectively. The device $D_i$ knows $c_j$, which is used in both $A_{i,j}$ and $P_{i,j}$, and to make them specific for the session; but this information isn't sufficient to get $a_i$ and $p_i$. On the other hand, brute-force approaches might take months even with very strong computers. Also, a malicious device $D_i$ can't use any authentication information, either stored in its local database or obtained from the sessions, to impersonate a device, because authentication information is specific for both devices and sessions. By the specificity of authentication information for the session, another possible threat is eliminated as well. Namely, a malicious device listening to the environment can capture the messages, but it can't impersonate device D by using the captured messages.

## VI. CONCLUSION

Password Reduction method is designed to make password-based authentication more practical and easier to manage for electrical mobility devices which have to define and maintain many passwords for many devices. The approach uses well-known theorem in basic number theory; Chinese Remainder Theorem. Both forward and reverse direction computations are possible. Reverse direction computation enables constructing the single password X out of many passwords that exist. Forward direction computation allows picking X and computes from it one password for each mobile device.

Password Reduction technique does not introduce new risks for authentication. Since devices can generate X, in forward direction, individual passwords created from X are likely to be random enough and would not be prone to dictionary attacks.

As a second contribution to the literature, we have developed a secure and efficient authentication protocol for electrical mobility devices, which is resistant to attacks. According to our authentication protocol, a mobile device can communicate securely with another mobile device over an insecure band. This is achieved with a symmetric key, which is generated randomly during the authentication process. After a successful login phase, this symmetric key is utilized to transmit and receive messages securely until the end of the connection. The other advantages of our protocol are; a device never reveals its unique password(X), individual passwords($a_i$) and prime numbers($p_i$). So, in short, we have achieved managing many passwords via a unique password. Also we have achieved authentication with multiple devices via same password in a secure manner.

## REFERENCES

[1] C. Herley, and P. Van Oorschot, "A Research Agenda Acknowledging the Persistence of Passwords," *IEEE Security and Privacy Magazine*, vol. 10, no. 1, pp. 28-36, 2012.

[2] S. Furnell, “Getting past passwords,” *Computer Fraud & Security*, vol. 2013, no. 4, pp. 8-13, Apr. 2013.

[3] L. Gong, M. Lomas, R. Needham, and J. Saltzer, “Protecting Poorly Chosen Secrets from Guessing Attacks,” *IEEE Journal on Seclected Areas in Communications*, vol. 11, no. 5, pp. 648-656, 1993.

[4] S. Bellovin, and M. Merritt, “Augmented Encrypted Key Exchange: A Password-Based Protocol Secure Against Dictionary Attacks and Password File Compromise,” *Proceedings of the 1st ACM conference on Computer and communications security*, pp. 244-250, 1993.

[5] S. Bellovin, and M. Merritt, “Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks,” *In: Proc. IEEE Symposium on Research in Security and Privacy*, pp.72-84, 1992.

[6] T. Tsuji, and A. Shimizu, “One-Time Password Authentication Protocol against Theft Attacks,” *IEICE TRANSACTIONS on Communications*, vol.E87-B, no.3, pp. 523-529, 2004.

[7] J. G. Steiner. B. C. Neuman. and I. Schiller, “Kerberos: An authentication service for open network systems, *Proc. Winter 1988 Usenix Conference*, pp. 191-201. Feb. 1988.

[8] S. P. Miller, B. C. Neuman, J. I. Schiller, and J. H. Saltzer, *Section E.2.1: Kerberos Authentication and Authorization System*, M.I.T. Project Athena, Cambridge, Massachusetts (December 21, 1987).

[9] J. T. Kohl, B. C. Neuman, and T Y. T'so. The evolution of the Kerberos authentication system, Distributed Open Systems, IEEE Computer Society Press, pp. 78-94, 1994.

[10] M. Bellare and P. Rogaway. Entity Authentication and Key Distribution. CRYPTO '93, LNCS 773, pages 232-249. Springer-Verlag, Berlin, 1994.

[11] Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0 available at http://docs.oasis-open.org/security/saml/v2.0/ 15 March 2005

[12] H. Y. Chien, J. K. Jan, and Y. H. Tseng, “An efficient and practical solution to remote authentication: smart card,” *Computers and Security*, vol. 21, no. 4, pp. 372–375, 2002.

[13] W. S. Juang, “Efficient password authenticated key agreement using smart cards,” *Computers and Security*, vol. 23, pp. 167-73, 2004.

[14] M. Kumar, “A new secure remote user authentication scheme with smart cards,” *International Journal of Network Security*, vol. 11, no. 2, pp. 88-93, 2010.

[15] L. Yang, J. F. Ma, and Q. Jiang, “Mutual authentication scheme with smart cards and password under trusted computing,” *International Journal of Network Security*, vol. 14, no. 3, pp. 156-163, 2012.

[16] R. Ramasamy and A. P. Muniyandi, “An efficient password authentication scheme for smart card,” *International Journal of Network Security*, vol. 14, no. 3, pp. 180-186, 2012.

[17] Koblitz, N. A Course in Number Theory and Cryptography, pp. 21, Springer, 1994.

[18] C. Ding, D. Pei, & A. Salomaa, “Chinese Remainder Theorem: Applications in Computing, Coding, Cryptography,” *World Scientific Publishing*, 1-224, 1996.

[19] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). Introduction to Algorithms, *Second Edition. MIT Press and McGraw-Hill, Section 31.5: The Chinese remainder theorem*, pp. 873–876.

[20] S. Iftene, “General Secret Sharing Based on the Chinese Remainder Theorem with Applications in E-Voting,” *Electronic Notes in Theoretical Computer Science (ENTCS)*, vol. 186, pp. 67–84, 2007.

[21] L. Gong, J. Pan, B. Liu, and S. Zhao, “A novel one-time password mutual authentication scheme on sharing renewed finite random sub-passwords,” *Journal of Computer and System Sciences*, vol. 79, no. 1, pp. 122-130, Feb. 2013.

[22] J. C. Patra, A. Karthik, and C. Bornand, “A novel CRT-based watermarking technique for authentication of multimedia contents,” *Digital Signal Processing*, vol. 20, pp. 442-453, 2010.

[23] S. K. Kim, T. H. Kim, D. G. Han, and S. Hong, “An efficient CRT-RSA algorithm secure against power and fault attacks,” *The Journal of Systems and Software*, vol. 84, no. 10, pp. 1660-1669, 2011.

[24] C. C. Chang, N. T. Huynh, and H. D. Le, “Lossless and unlimited multi-image sharing based on Chinese remainder theorem and Lagrange interpolation,” *Signal Processing*, vol. 99, pp. 159-170, 2014.

[25] K. Kaya, and A. A. Selçuk, “Sharing DSS by the Chinese Remainder Theorem,” *Journal of Computational and Applied Mathematics*, vol. 259, pp. 495-502, 2014.