

Irreversible Encryption in for Securing ATM Transactions

18BIT0267 Ritika Sengar
18BIT0272 Priyal Bhardwaj
18BIT0321 Ruchita Nathani
18BIT0473 Tamanna Srivastava

Information Security Analysis & Audit
CSE3501 - G2 Slot
Prof. Thandeewaran R.
J-Component Report





Project Report

Outline

[Abstract](#)

[Problem Statement](#)

[Literature Survey](#)

[Our Proposal](#)

[Architecture](#)

[Working](#)

[Code - home.py](#)

[Code - auth.py](#)

[Implementation](#)

[Conclusion](#)



Abstract

ATMs allow to make deposits and withdraw money and even you can print a statement, view your account balance and even transfer money between your accounts. ATMs, if properly secured, are safe and most convenient way to manage our money.

To protect our money and transaction we need to safeguard them from different type of attacks. Nowadays due to development in technology, new ATM machines are being built up with more and more security. But to destroy this security level, threats are being imposed. Regardless of enhancement in the automation, still ATM are prone to thefts and frauds.



Problem statement

The present ATM model uses a card and a PIN. This is a very simple ATM model and can easily be attacked. The attacks can in form of stolen cards, duplicity of cards or due to statically given PINS.

To improve the security of the ATM model, biometric technique was also introduced in which user has to give biometric first and then after verifying he has to enter the PIN. This improved security but later it failed as there were different errors found in this technique.

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|--|---|-----------------------------------|--|---|--|
| 1. | Enhanced security for ATM machine with OTP and Facial recognition features | https://www.sciencedirect.com/science/article/pii/S1877050915004093 | Procedia Computer Science Vol. 45 | Mohsin Karovaliya, Saifali Karedia, Sharad Oza , D.R. Kalbande Dr. | Features like face recognition and One-Time Password (OTP) are used for the enhancement of security of accounts and privacy of users. Face recognition technology helps the machine to identify each and every user uniquely thus making face as a key. Moreover, the randomly generated OTP frees the user from remembering PINs as it itself acts as a PIN. | Retina recognition not yet implemented due to higher cost. Attackers can intercept the OTP as well so a security measure against that was not addressed. |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|--|---|---|-----------------------------------|---|---|
| 2. | A Review on Secure ATM by Image Processing | http://tmu.ac.in/college-of-computing-sciences-and-it/wp-content/uploads/sites/17/2016/10/ICAC-1604155.pdf | International Conference on Advance Computing | Shivendra Dwivedi, Ranjana Sharma | ATM model can be developed that is more reliable in providing security by using facial recognition software. This type of security model can be used to minimise ATM frauds | The flaws in face recognition technique like the inability to detect face when beard, aging, glasses and caps can be rectified and eliminated or reduced. |

Literature Survey

| S.No. | Paper | Link | Journals | Authors | Abstract | Research Gap |
|-------|--|---|---|---------------------------------|--|--|
| 3. | Combating Automated Teller Machine Frauds through Biometrics | http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.445.2592&rep=rep1&type=pdf | International Journal of Emerging Technology & Advanced Engineering | Adeniyi Akanni, Oludele Awodele | The combined biometric features approach is to serve the purpose both the identification and authentication that card and PIN do. While iris replaces the card, fingers are used to do the authentication. | If the cost of retina or iris recognition reduces, it can be used. |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|---|---|---------------------------|------------------------|--|--|
| 4. | Biometrics and Smart Cards in Identity Management | https://pdfs.semanticscholar.org/3167/b3145a1ad762b8782b34a99a2a88bc84e05e.pdf | Computer Security Journal | Bart Jacobs, Eric Poll | The card can protect the information, it cannot easily be cloned, and even if a card is lost or stolen, the protection it provides remains in place. | <p>When trying to match a stored biometric with one freshly obtained, there is always the chance of false matches and false non-matches.</p> <p>Face Recognition: between 2 and 10 persons out of 100 may not be verified.</p> |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|---|---|--|--|--|---|
| 5. | Facial verification technology for use in ATM transactions. | https://www.semanticscholar.org/paper/Facial-Verification-Technology-for-Use-In-Atm-Eze/22915640d7141a27ea16a45f916dda492ad1b3d3 | American Journal of Engineering Research | Okereke, E. Ihekweaba, G. & Okpara, F.K. | A system which incorporates facial recognition technology into the identity verification process used in ATMs was proposed | 1. The proposed system was not built as an improvement on the existing system. 2. The study relied on open-source facial recognition program and did not discuss the local features that will be analyzed for the facial verification process |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|--|---|---|--|--|---|
| 6. | Improving Security Levels In ATMs Using Multifactor Authentication | https://www.researchgate.net/publication/301536500_Improving_Security_Levels_In_Automatic_Teller_Machines_ATM_Using_Multifactor_Authentication | International Journal of Emerging Technology & Advanced Engineering | Twum, Frimpong & Nti, Isaac kofi & Asante Michael. | Increase the security and make the system more immune to attacks using Multi-factor Authentication | Addition of fingerprint might seem a good idea, but it is an expensive process to implement from scratch and there are multiple methods to forge a fingerprint. |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|---|---|--|------------------------------------|---|--|
| 7. | ATM Security Using Fingerprint Biometric Identifier: An Investigative Study | https://www.researchgate.net/publication/267265597_ATM_Security_Using_Fingerprint_Biometric_Identifier_An_Investigative_Study | International Journal of Advanced Computer Science & Application | Onyesolu, Moses & Ezeani, Ignatius | A fingerprint biometric technique is fused with the ATM for person authentication to ameliorate the security level. | Transmitting the fingerprint value to the authentication unit, the encryption will take place ambiguously as the pixel values corresponding to the fingerprint will not always correspond to the same binary values each and every time. Hence any sort of man in the middle attack will easily get the fingerprint value without even a cryptanalytic attack. |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|--|---|--|---|--|--|
| 8. | Towards understanding ATM security - A field study of real world ATM use | https://www.researchgate.net/publication/221166442_Towards_understanding_ATM_security_-_A_field_study_of_real_world_ATM_use | ACM International Conference Proceedings | De Luca Alexander, Langheinrich , Marc, Hussmann Heinrich | Field observations were performed in six different locations in two central European cities, Munich (Germany) and Delft (the Netherlands). We chose ATMs that were available 24 hours a day, seven days a week, and which were located outside. This allowed for unobtrusively observing actual ATM interactions | The limitation of the survey are that to get the info the users have to be disturbed from general course as it is not collected from a specific focus group and hence dependent upon many human factors. |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|--|---|---|--|--|---|
| 9. | "A New Vision for ATM Security Management": The Security Management Platform | https://www.researchgate.net/publication/312435426_A_New_Vision_for_ATM_Security_Management_The_Security_Management_Platform_per/f201403211395407076.pdf | 11th International Conference on Availability, Reliability and Security (ARES) | Claudio Porretti, Denis Kolev, Raoul Laheije | A new vision for ATM Security Management that is proposed by the GAMMA project, and implemented by its “core” prototype called Security Management Platform. | Decision makers might find it too easy to accept vulnerabilities if mitigating them takes necessary resources from accomplishing a business objective |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|--|---|--|--------------------|---|--|
| 10. | Enhanced ATM Security using Biometrics | Oko, S. and Oruh, J. (2012): Enhanced ATM security system using biometrics. IJCSI International Journal of Computer Science Issues, September 2012. Vol. 9, Issue 5, No 3, pp. 352-357. | IJCSI International Journal of Computer Science Issues | Oko S. and Oruh, J | Developed an ATM based fingerprint verification and simulated it for ATM operations by incorporating the fingerprints of users into the bank"s database | The system developed was inefficient because there was no finger print matching algorithm. |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|---|---|---|-----------------------------|---|---|
| 11. | Enhancing ATM security using fingerprint and GSM technology | https://www.researchgate.net/publication/315483165_Enhancing_ATM_Security_using_Fingerprint_and_GSM_Technology | International Journal of Computer Application | V. Padmapriya , S. Prakasam | A combination of fingerprint biometric token and GSM technology | <p>1. A nominee or third party"s finger print was incorporated in the architecture.</p> <p>2. There is a discord between the main user and the nominee user in the proposed system architecture</p> |

Literature Survey

| S.No. | Paper | Link | Journal | Authors | Abstract | Research Gap |
|-------|---|---|--------------------------------------|-------------|--|---|
| 12. | Card duplication and crime prevention using biometrics. | https://www.researchgate.net/publication/271296818_Credit_Card_Duplication_and_Crime_Prevention_Using_Biometrics | IOSR Journal of Computer Engineering | Prithika M. | Proposed using the Iris Recognition and Palm Vein (IRPV) recognition technology to prevent card duplication and crimes via the ATM | 1) a reliable scanner and registration device are not enough to create a trustworthy system 2)The system is very expensive |

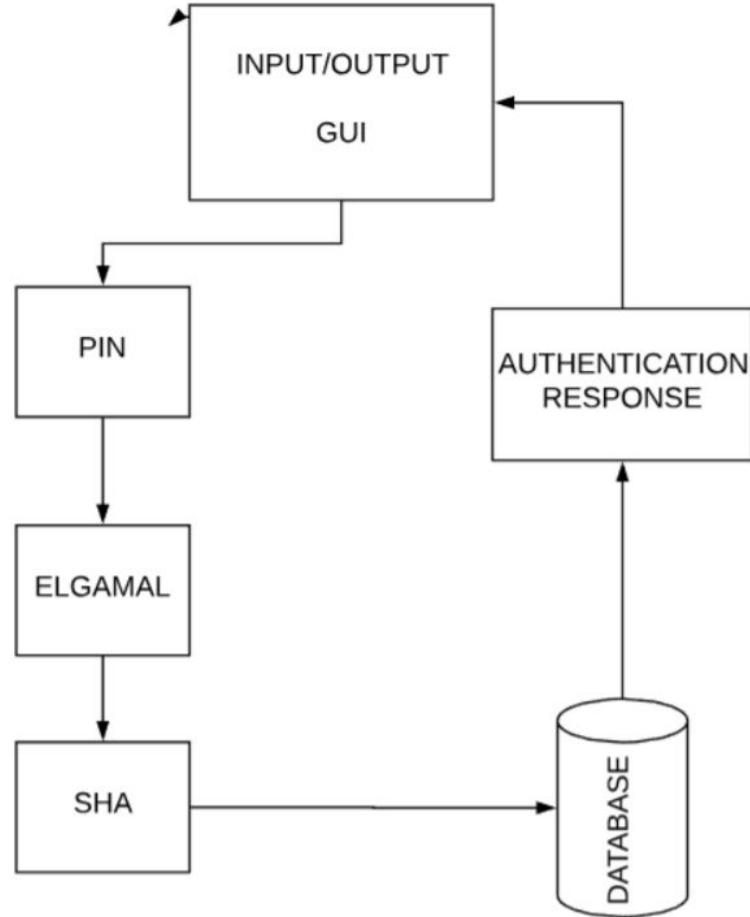
Our Proposal



We propose a different and much more secure way of protecting ATM machine. In our proposed model, first user enters the pin, this pin is encrypted using **elgamal cryptographic algorithm** and then the resulted output is given as input to SHA-512 Hash.

This is a secure way of transmission as it protects from **man in the middle attack** because if the attackers tries to decode, he will get only the hash value and will not be able to get the actual pin.

Architecture



Working



Elgamal encryption, in cryptography, is an asymmetric (public) encryption technique. As the name suggests, it uses asymmetric key encryption wherein there exists two separate keys, a combination of public and private keys each for the both the parties involved in the communication.

This technique is based on the complexity of calculating discrete logarithm in a cyclic group, i.e., it prevents us from computing the value of g^ak , even if know the values of ga and gk .

In a given communication using Elgamal technique, if we assume A to be the sender and B, the receiver, the following steps could be listed out that shall be executed:

1. Generation of public and private key

- a. The user selects a very large number 'q' and a cyclic group F_q .
- b. An element 'g' is selected from the cyclic group F_q and an element a such that $\text{GCD}(a, q) = 1$.
- c. $h = g^a$ is then computed.
- d. The values ' F_q ', 'h', 'q' and 'g' becomes the public key and the value 'a' is retained as the private key.

Working



2. Encryption at the sender site

- a. The sender selects a random integer 'k' from the cyclic group F_q such that $\text{GCD}(k, q) = 1$.
- b. Following this, the values $p = g^k$ and $s = h^k = g^{ak}$ are computed.
- c. The values 's' and 'M' are then multiplied together where 'M' is the message.
- d. Finally, the tuple $(p, M \times s) = (g^k, M \times s)$ is sent as the encrypted message to the receiver.

3. Decryption at the receiver site

- a. The receiver calculates the value $s = p^a = g^{ak}$.
- b. Following this the value $M \times s$ is divided by 's' to obtain 'M'.

SHA (Secure Hash Algorithms) are a family of cryptographic algorithms to ensure the authentication of data. It functions by converting a given message into a hash value or a message digest of a definite size (a fixed size string generated from the message). The algorithm that does the same comprises of functions including bitwise operations, modular addition and compression functions. The algorithms work on the principle of a one-way function, i.e., once the messages are transformed into their respective hash values, they cannot be converted back to the original form.

Working



Encrypting passwords or PINs is one of the common applications of the SHA algorithms. This comes as an immediate effect of the fact that the server side has to actually only keep track of the specific hash values corresponding to the entered passwords/PINs of a particular user instead of the actual password/PIN.

The benefit that it provides is that if the database, by any chance gets hacked, the attacker would only get to get their hands on the hash values and not the actual PINs.

In addition to that, SHAs exhibit the avalanche effect wherein modifying even a single character in the message leads to a drastic alteration in the hash value. This prevents the attacker from even finding out the length of the original message, let alone the message itself.

Integration of Elgamal with SHA



In our project, we have integrated the concept of Elgamal encryption with the SHA-256 authentication to provide additional security to the PIN generated for the ATM users.

The user enters the 4-digit PIN which is then encrypted using the Elgamal encryption technique following which it is converted into its corresponding hash value using the SHA-256 algorithm.

This message digest is then finally sent to the server, where it is then compared with the existing hash values in the directory. If a match is found, the corresponding user details are then fetched and the same is then reflected on the client-side output.

[Github Link to Code](#)

CODE - home.py

```
from tkinter import *
from auth import *
from math import pow
window = Tk()
window.attributes('-fullscreen',True)
w = window.winfo_screenwidth()
h = window.winfo_screenheight()
#window.geometry("%dx%d+100+100" % (w, h))
window.title("Welcome to ISAA Bank")
window.configure(background = "LightSeaGreen")
hashDir = {'d84305873370ac353a4aaa3df835104df5a4dd1fe6f88a88fff50ae08564a85c' :
'Priyal' , #7828
'16a05870017ccb4b0f94bb191e8952eb77365e3f1b265d0f46fdde8f1297d820': 'Ritika', #6774
'5ba3df1057fcddf3c60a1699fb0e736634809c0b6dd41d9c6dfd0d6f24ccd79f': 'Ruchita', #8367
'f4df4c0e33686c9bc43272be64bcb4ad4a6e3e24f4db0ea40ba4aa7b0303b615': 'Tamanna'
#5046
}
```

CODE - home.py

```
hashDirLen = len(hashDir)
hashList = list(hashDir.keys())
#nameList = list(hashDir.values())
def authenticate():
    newWin = Toplevel(window)
    newWin.attributes('-fullscreen',True)
    newWin.title("Authentication")
    newWin.configure(background = "LightSeaGreen")
    Label(
        newWin,
        text = " Welcome to ISAA Bank ",
        bg = "LightSeaGreen",
        fg = "DarkCyan",
        font = "Consolas 72"
    ).place(
        x = 90,
        y = 50 )
```


CODE - home.py

```
P = PINEntry.get()
PINEntry.delete(0,END)
if len(P) == 4 and P.isdigit():
    q = 123456789987654321234567898      #random.randint(pow(10,20),pow(10,50))
    g = 23931164504956447807213117212663825326210289577470
    key = gen_key(q)      #Receiver_Private_key
    h = power(g,key,q)
    en_msg, p, out = encrypt(P,q,h,g)
    dr_msg = decrypt(en_msg,p,key,q)
    dmsg = ''.join(dr_msg)
    print("Decrypted Message: ",dmsg)
    flag = 0
    for i in range(len(hashList)):
        if out == hashList[i]:
            flag = 1
            pos = i
            break
```

CODE - home.py

```
if flag == 1:
    Label(
        newWin,
        text = "PIN Matched. Hello, " + hashDir[hashList[pos]] + "!",
        bg = "LightSeaGreen",
        fg = "Navy",
        font = "Consolas 20"
    ).place(
        x = 430,
        y = 300
    )
    Button(
        newWin,
        text = "Back",
        width = 20,
        font = "Calibri 15",
        bg = "LightSeaGreen",
```

CODE - home.py

```
        fg = "White",
        command = newWin.destroy
    ).place(
        x = 550,
        y = 425
    )
else:
    Label(
        newWin,
        text = "No match found. Please try again.",
        bg = "LightSeaGreen",
        fg = "Maroon",
        font = "Consolas 20"
    ).place(
        x = 430,
        y = 300
    )
```

CODE - home.py

```
Button(  
    newWin,  
    text = "Back",  
    width = 20,  
    font = "Calibri 15",  
    bg = "LightSeaGreen",  
    fg = "White",  
    command = newWin.destroy  
)  
.place(  
    x = 550,  
    y = 425  
)  
else:  
    Label(  
        newWin,  
        text = "Invalid PIN. Please try again.",  
        bg = "LightSeaGreen",  
        fg = "Maroon",
```

CODE - home.py

```
font = "Consolas 20"
).place(
    x = 430,
    y = 300
)
Button(
    newWin,
    text = "Back",
    width = 20,
    font = "Calibri 15",
    bg = "LightSeaGreen",
    fg = "White",
    command = newWin.destroy
).place(
    x = 550,
    y = 425
)
```


CODE - home.py

```
Label(  
    text = " Welcome to ISAA Bank ",  
    bg = "LightSeaGreen",  
    fg = "DarkCyan",  
    font = "Consolas 72"  
).place(  
    x = 90,  
    y = 50  
)  
Label(  
    text = "Enter your 4-digit PIN:",  
    bg = "LightSeaGreen",  
    fg = "White",  
    font = "SegoeUILight 20",  
).place(  
    x = 525,  
    y = 250  
)
```

CODE - home.py


```
PINEntry = Entry(
    window,
    width = 4,
    font = "Consolas 50",
    show = "*"
)
PINEntry.place(
    x = 582,
    y = 300
)
PINEntry.focus_set()
Button(
    window,
    text = "Proceed",
    width = 20,
    font = "Calibri 15",
    bg = "LightSeaGreen",
    fg = "White",
```

CODE - home.py



```
    command = authenticate
).place(
    x = 550,
    y = 425
)
Button(
    window,
    text = "Quit",
    width = 20,
    font = "Calibri 15",
    bg = "LightSeaGreen",
    fg = "White",
    command = window.destroy
).place(
    x = 550,
    y = 475
)
window.mainloop()
```


CODE - auth.py



```
import random
from math import pow
import hashlib
import time
start = time.time()
a = 7
def gcd(a, b):
    if a < b:
        return gcd(b, a)
    elif a % b == 0:
        return b
    else:
        return gcd(b, a % b)
# Generating large random numbers
def gen_key(q):
    key = 12345678998765432123456789    #random.randint(pow(10, 20), q)
```

CODE - auth.py

```
while gcd(q, key) != 1:
    key = random.randint(pow(10, 20), q)
return key
# Modular exponentiation
def power(a, b, c):
    x = 1
    y = a
    while b > 0:
        if b % 2 == 0:
            x = (x * y) % c
        y = (y * y) % c
        b = int(b / 2)
    return x % c
# Asymmetric encryption
def encrypt(msg, q, h, g):
    en_msg = []
    k = 19 #gen_key(q) # Private key for sender
    s = power(h, k, q)
```

CODE - auth.py

```
p = power(g, k, q)
for i in range(0, len(msg)):
    en_msg.append(msg[i])
print("g^k used : ", p)
print("g^ak used : ", s)
for i in range(0, len(en_msg)):
    en_msg[i] = s * ord(en_msg[i])
print('Encrypted pin : ', en_msg)
output = hashlib.sha256(str(en_msg[1]).encode('utf-8')).hexdigest()
print('Hash generated : ',output)
return en_msg, p, output
def decrypt(en_msg, p, key, q):
    dr_msg = []
    h = power(p, key, q)
    for i in range(0, len(en_msg)):
        dr_msg.append(chr(int(int(en_msg[i])/h)))
    return dr_msg
```

CODE - auth.py

```
# Driver code
def main():
    msg = input('Enter 4 digit pin : ')
    end = time.time()
    print("Original Message : ", msg)
    q = 123456789987654321234567898          #random.randint(pow(10, 20), pow(10, 50))
    g = 23931164504956447807213117212663825326210289577470
    key = gen_key(q)          # Private key for receiver
    h = power(g, key, q)
    print("g used : ", g)
    print("g^a used : ", h)
    en_msg, p, out = encrypt(msg, q, h, g)
    dr_msg = decrypt(en_msg, p, key, q)
    dmsg = ''.join(dr_msg)
    print("Decrypted Message : ", dmsg)
    print(end-start)
if __name__ == '__main__':
    main()
```

Implementation - home

Welcome to ISAA Bank

Enter your 4-digit PIN:

Proceed

Quit

Implementation - 3 digit invalid pin

Welcome to ISAA Bank

Enter your 4-digit PIN:

Proceed

Quit

Implementation - 3 digit invalid pin



Welcome to ISAA Bank

Invalid PIN. Please try again.

Back

Implementation - 4 digit valid and correct pin

Welcome to ISAA Bank

Enter your 4-digit PIN:

* * * *

Proceed

Quit

Implementation - User 1



Welcome to ISAA Bank

PIN Matched. Hello, Ruchita!

Back

Implementation - User 2



Welcome to ISAA Bank

PIN Matched. Hello, Priyal!

Back

Implementation- User 3



Welcome to ISAA Bank

PIN Matched. Hello, Ritika!

Back

Implementation - User 4



Welcome to ISAA Bank

PIN Matched. Hello, Tamanna!

Back

Implementation - Command prompt execution



```
g^k used : 34215135440877585739649990
g^ak used : 53029511344900323266701110
Encrypted pin : [2916623123969517779668561050, 2969652635314418102935262160, 2651475567245016163335055500, 2969652635314418102935262160]
Hash generated : d84305873370ac353a4aaa3df835104df5a4dd1fe6f88a88fff50ae08564a85c
Decrypted Message: 7828
g^k used : 34215135440877585739649990
g^ak used : 53029511344900323266701110
Encrypted pin : [2810564101279717133135158830, 2545416544555215516801653280, 2757534589934816809868457720, 2863593612624617456401859940]
Hash generated : f4df4c0e33686c9bc43272be64bcb4ad4a6e3e24f4db0ea40ba4aa7b0303b615
Decrypted Message: 5046
g^k used : 34215135440877585739649990
g^ak used : 53029511344900323266701110
Encrypted pin : [2863593612624617456401859940, 2916623123969517779668561050, 2916623123969517779668561050, 2757534589934816809868457720]
Hash generated : 16a05870017ccb4b0f94bb191e8952eb77365e3f1b265d0f46fdde8f1297d820
Decrypted Message: 6774
g^k used : 34215135440877585739649990
g^ak used : 53029511344900323266701110
Encrypted pin : [2969652635314418102935262160, 2704505078589916486601756610, 2863593612624617456401859940, 2916623123969517779668561050]
Hash generated : 5ba3df1057fcddf3c60a1699fb0e736634809c0b6dd41d9c6dfd0d6f24ccd79f
Decrypted Message: 8367
```

Implementation - Valid but incorrect pin



Welcome to ISAA Bank

No match found. Please try again.

Back

Conclusion



It is hence seen that using elgamal with hashing proves helpful as it reduces man in the middle attacks to a great extent owing to the fact that hash are irreversible. The 4 digit pin is converted to a fixed 256 bit hash value based upon which the pin is verified. Since strong and weak collision resistance follows hence it is highly improbable to find another pin with the same hash value. Even if the attacker gets a hold of the hash value he cannot find the original pin, hence keeping the identity of the user intact.

The only shortcoming with this technique is that it is not permanent and while the system becomes scalable then the chances of collision becomes higher. Elgamal with hash gave the best time values for run time hence is used for the model here. As far as future work is concerned, using biometric identities is becoming the need of the hour but the obvious shortcomings were mentioned in the literature survey. To overcome this a hybrid biometric authentication system could be made for future which cannot be forged and is hundred percent immune to external attacks.

THANK YOU