CertesSecret

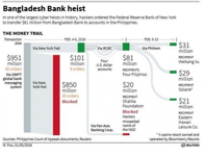
Resected and Secreted

Nitin K N[1] Sharath S Chellappa[2] Chetan Purohit[3]
Madhusudan Thyagarajan[4]

[4]Professor Electronics and Communication Department
PES Institute of Technology

3rd Aug, 2017

Outline



#	Count	Ciphertext	Plaintext
1.	1911938	EQ7fIpT7i/Q=	123456
2.	446162	j9p+HwtWWT86aMjgZFLzYg==	123456789
3.	345834	L8qbAD3jl3jioxG6CatHBw==	password
4.	211659	BB4e6X+b2xLioxG6CatHBw==	adobe123
5.	201580	j9p+HwtWWT/ioxG6CatHBw==	12345678
6.	130832	5djv7ZCI2ws=	qwerty
7.	124253	dQi0asWPYvQ=	1234567
8.	113884	7LqYzKVeq8I=	111111
9.	83411	PMDTbP0LZxu03SwrFUvYGA==	photoshop
10.	82694	e6MPXQ5G6a8=	123123
11.	76910	j9p+HwtWWT8/HeZN+3oiCQ==	1234567890
12.	76186	diQ+ie23vAA=	000000
13.	70791	kCcUSCmonEA=	abc123
14.	61453	ukxzEcXU6Pw=	1234
15.	56744	5wEAInH22i4=	adobe1
16.	54651	WqflwJFYW3+PszVFZo1Ggg==	macromedia
17.	48850	hjAYsdUA4+k=	azerty
18.	47142	rpkvF+oZzQvioxG6CatHBw==	iloveyou
19.	44281	xz6PIeGzr6g=	aaaaaa
20.	43670	Ypsmk6AXQTk=	654321
21.	43497	4V+mGczxDEA=	12345
22.	37407	yp2KLbBiQXs=	666666
23.	[illegible]	[illegible]	sunshine

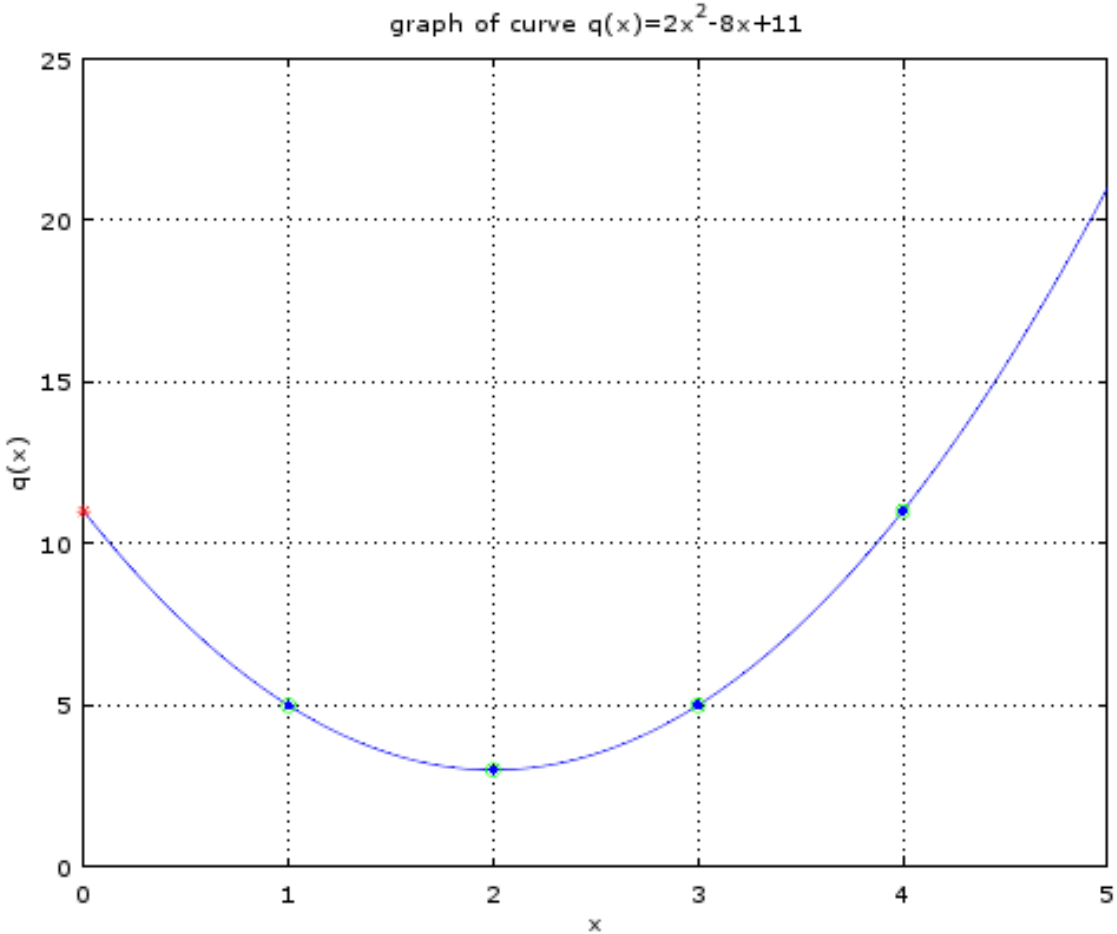
Adobe

SONY SINCE THE NOVEMBER, 2014 HACK

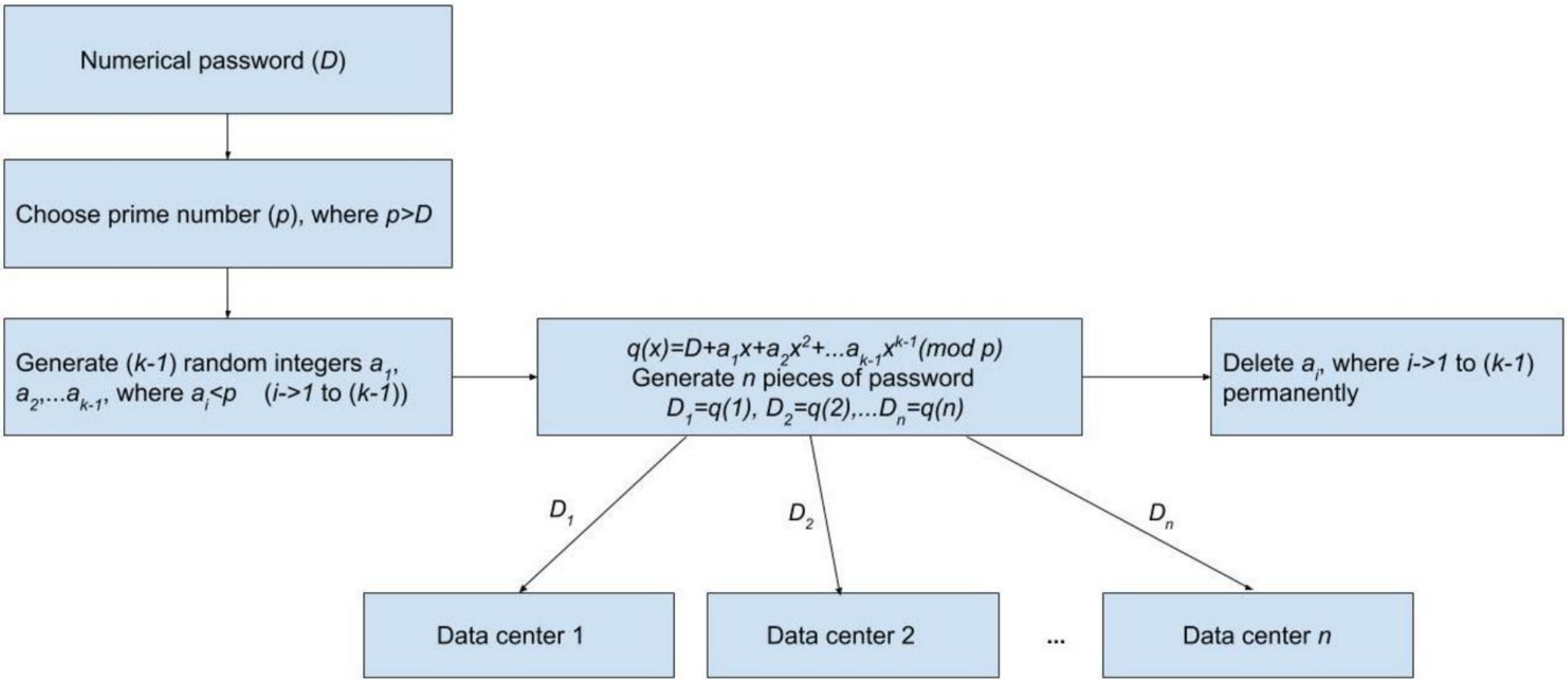
- Could pay as much as $8 million to some 50,000 current and former employees whose privacy was breached.
- Fired Sony Pictures Entertainment head, Amy Pascal.
- Released only 6 movies in the first eight months of 2015; all were box office duds.
- Ends 2015 on high note with the release of "Spectre," "The Night Before," and "Concussion," which was nominated for a Golden Globe.

Splitting of Password

Given k points in the 2-dimensional plane $(x_1, y_1), (x_2, y_2) \ldots (x_k, y_k)$. With distinct x_i's ,there is only one polynomial $q(x)$ of degree $k-1$ such that $q(x_i) = y_i$ for all i.



Splitting of Password



Splitting of Password

Reasons for choice of this algorithm:

- Construction of robust key management that can function when
  - security breaches expose some pieces of password.
  - misfortune destroys some pieces.
- Mathematical approach where the pieces of the password have no correlation with each other discouraging statistical attacks.
- Flexibility in design by choice of k and n.

Architecture

Password Storage

User is asked to enter his Login Id and Name.

App sends the encrypted login id the Bank Server

Enc{Login id +Name+MAC ID}}

OTP + Security Question

Enc{OTP + Security Question}}

Bank decrypts the packet, stores the login and sends a security question along with an OTP back to the user.

OTP + Answer + Password

User enters OTP, answer to the Security Question and his password into the respective field entries in the app.

The App sends the user password, answer, login id and OTP and One-time-key to the Bank Server.

Enc{OTP + Answer + User Password + login id + One time key}}

The Bank server sends the location pass to the app so that it can be hard coded for future use.

Bank pass to be hard coded into the app

The Bank Server checks the answer and now splits the hashed password and stores the different pieces in different servers with the Login id being the query point. The splitting is done based on a map present at Bank Server which maps each login id with a bank pass which gives location of servers.

Enc{Password Piece} + Hashed(login id)

The user gets an acknowledgement that his password has been accepted.

1

Architecture

Password Retrieval

User clicks on the button to request access to his account

App sends the encrypted login id the Bank Server

Enc{Login id}

OTP

OTP

Bank decrypts the login id, stores the login and sends an OTP back to the user.

The Bank server correlates OTP with the login id. This correlation expires in a finite time.

User enters OTP and his password into the respective field entries in the app.

OTP + Password

The App sends the hash(user password + OTP), location pass, login id and OTP to the Bank Server.

Enc{OTP+login ID + Hash(User Password+OTP) + One time key}

The Bank Sever decrypts the packet to get the login id and OTP, gets the location of the servers having pieces from the Bank Pass and sends the login Id to each of the servers after checking if the OTP matches.

Hash(Login id) + One time key to each server

The bank server combines all the pieces together and checks with the password entered by the user. If the password on the server and the password match, the Bank Server sets up a secure connection with the app to allow for the user to access his account

Individual servers return the pieces for the queried login id.

Each piece is sent to Bank Server

The user gets to access his account over a secure connection.

Compromise of Stakeholders

Compromise of Mobile device

The adversary has no idea about the password so the system cannot be broken unless he tries all possible passwords exhaustively.

$$\text{Probability of guessing } p \text{ digit pin } = 10^p \tag{1}$$

This is the theoretical security the System hopes to achieve so compromise of mobile device does not break down the system.

Compromise of Stakeholders

Compromise of Data Center

The adversary has one piece of the password. He needs atleast k pieces to reconstruct the password. The possible way is to guess the remaining $k-1$ piece where each piece is atleast the same length or longer than the actual password

$$\text{Probability of guessing 1 password piece} = 10^p \quad (2)$$

$$\text{Probability of guessing } (k-1) \text{ pieces} = (k-1) * 10^p \quad (3)$$

While the probability of guessing password is 10^p

$$(k-1) * 10^p >> 10^p \quad (4)$$

The adversary has better chances of guessing the password than to guess multiple pieces and then reform the password. Breaking into one data center will not decrease the security of the cryptosystem.

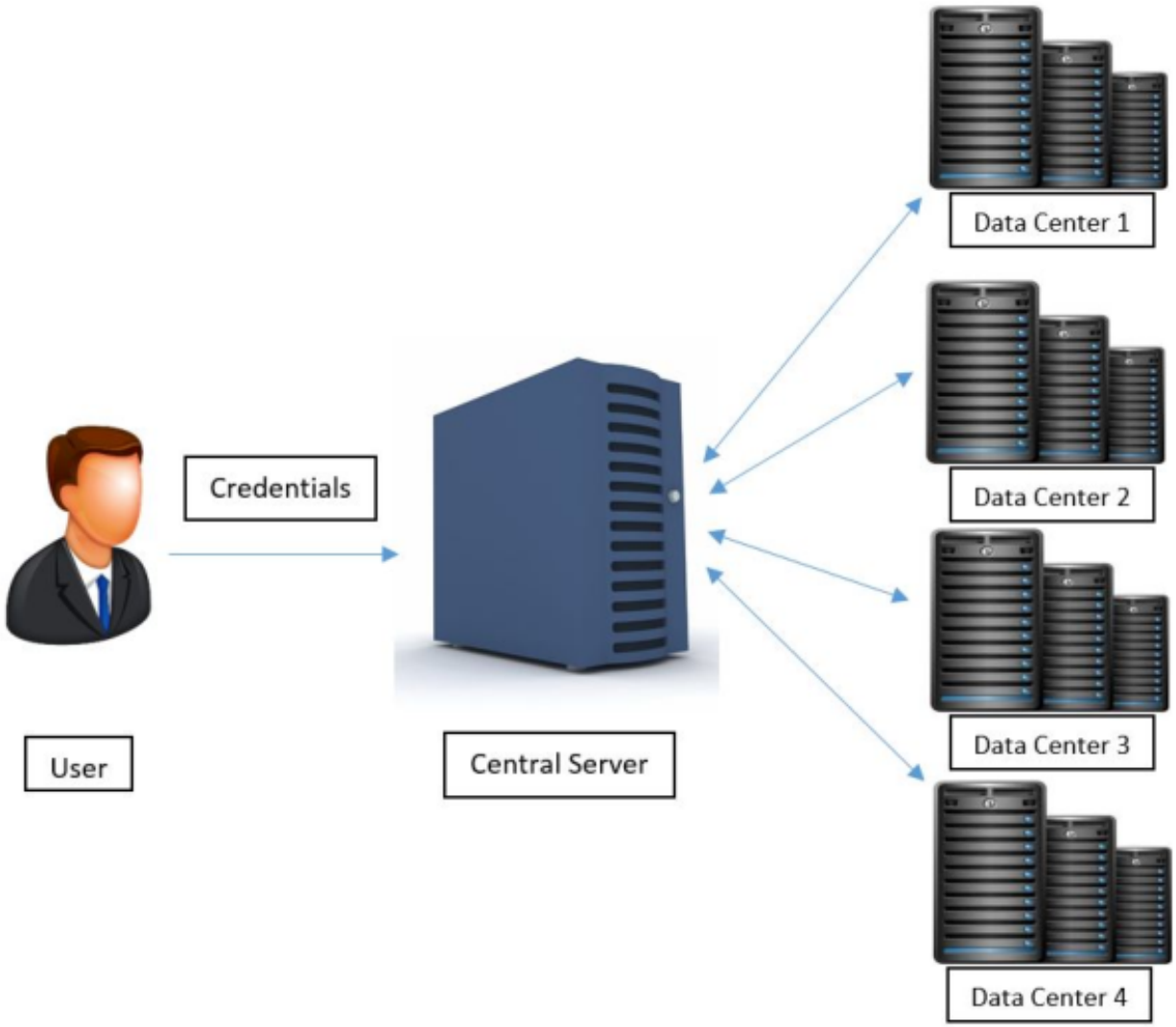
Applications of the Core Solution

Looking at our base algorithm and the current implementation, we have identified a few areas where our solution could help secure the day to day functioning of the Air Force. Some of these applications are as follows:

1. Access Control
2. Multiple Authorization
3. Document Access and Retrieval
4. Multi Password Multi User Single Realtime Access

Applications of the Core Solution

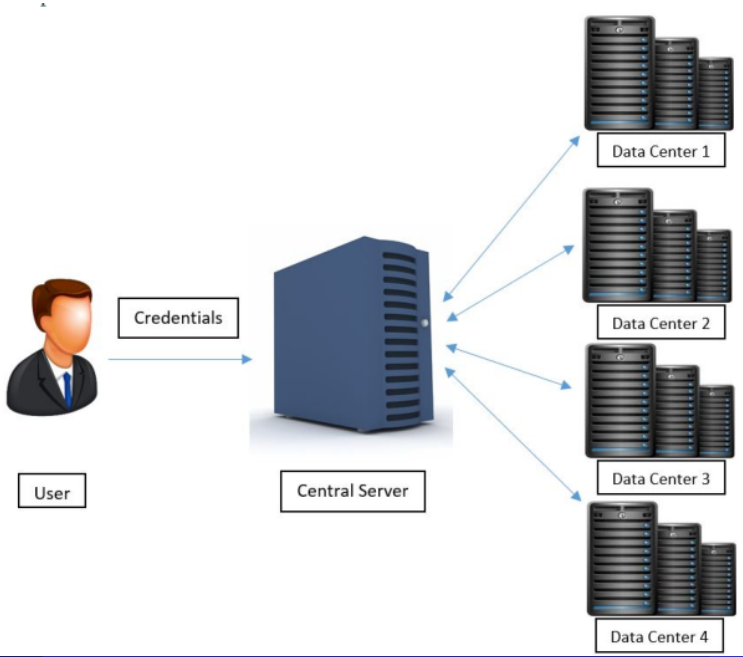
Access Control



Applications of the Core Solution

Access Control

- Bread and Butter Implementation of the core solution
- Access to all systems can be done using a single credential which will be secreted in the core solution.
- This will ensure that all applications can be accessed only by the individual and if required only through his device.
- The system will not have any record of any credentials, thereby making it unbreakable.



Applications of the Core Solution

Document Access and Retrieval

This application of the core solution can be best explained with a Gif.

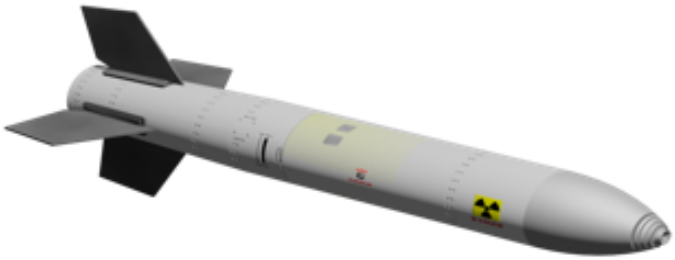
Applications of the Core Solution

Document Access and Retrieval

- Document is secreted by being encrypted with a key which is split and shared with the owner of the document.
- For any additional personnel wishing to access the current document, the base algorithm extrapolates the graph giving rise to more points/pieces which are stored in memory locations.
- Memory locations are shared with the additional personnel thereby granting them access to the documents.
- Lapse of the time allotted causes the memory locations to be cleared thereby revoking access.
- Mapping between the memory locations and the user credential in the database, would be able to show the personnel who has most recently accessed the document.

Applications of the Core Solution

Multiple Authorization



Applications of the Core Solution

Multiple Authorization

- Extremely useful for a case of a resource which would require authentication from multiple parties.
- Assuming a situation where access to the resource is critical with the absence of one of the access granting parties.
- Core Solution generates multiple pieces of the key and distributes the memory locations among multiple parties.
- Core Solution enables access to the critical resource by allowing access when the number of password pieces generated by the algorithm crosses a given threshold set thereby enabling access to the critical resource.
- This is highly useful when the requirement is for collaborative clearance of resources like clearance of Aircraft Air worthiness, flight clearance and launch of nuclear weapons etc.

Multi Password Multi User Single Real Time Access

Applications of the Core Solution

Multi Password Multi User Single Real Time Access

- Extremely useful in the case of Work Scheduling
- The Core Solution could be used to manufacture multiple pieces of the key distributed to multiple parties
- Usage of all the keys by the authorized parties,will allow complete clearance to the document .
- Completion of the task by an authorized party would be the reason for the party to grant clearance by using his or her key.
- Non usage of any one key of an authorized member will affect the complete clearance of the document.

Conclusion

- We have truly attempted to create an virtually unbreakable key management scheme looking to protect the secret holistically
- The proposed system has been implemented as a Proof of Concept for a Bank. Here we have looked to implement a cryptosystem where any one successful attack on the cryptosystem would never compromise the system.
- Eliminated the possibility of internal compromise by now using the Bank/Central Server as a processing unit rather than as a storage unit.
- Would like to implement our Core Solution as a Proof of Concept to understand and explore ways of providing security solutions to the Indian Air Force.

Questions?