Ali Amr Osman
Ahmed Emad
Gabry

Secure WhatsApp

In this project we sought out to reinvent the idea of of chatting security. We took a different approach to the common encryption used in WhatsApp and Triple DES Encryption. My SQL connector with java was used in order to store user passwords and tokens. Encryption library was used to encrypt messages and print them on the console. Data encryption is a requirement in the age of cyber criminals and advanced hacking techniques. But the roots of encryption are actually thousands of years old, and encryption in computers even has a pretty long history. The Data Encryption Standard (DES) was developed in the late 1970s and saw widespread use for many years. It wasn't a perfect method then, but still it was used quite heavily. Then shortly after it was developed, an improved version called Triple DES (3DES) was created.

What is 3DES?

In spite of the fact that it's authoritatively known as the Triple Data Encryption Algorithm (3DEA), it is most normally alluded to as 3DES. This is on the grounds that the 3DES calculation utilizes the Data Encryption Standard (DES) figure multiple times to scramble its information.

DES is a symmetric-key calculation dependent on a Feistel arrange. As a symmetric key figure, it utilizes a similar key for both the encryption and decoding forms. The Feistel arrange makes both of these procedures the very same, which results in a calculation that is progressively productive to execute.

DES has both a 64-bit square and key size, however practically speaking, the key just gives 56-bits of security. 3DES was created as a progressively secure option in light of DES's little key length. In 3DES, the DES calculation is gone through multiple times with three keys, anyway it is possibly viewed as secure if three separate keys are utilized.

The employments of 3DES

When the shortcomings of typical DES turned out to be progressively obvious, 3DES was received in a wide scope of uses. It was one of the more ordinarily utilized encryption plots before the ascent of AES.

A few instances of its usage included Microsoft Office, Firefox and EMV installment frameworks. A considerable lot of these stages never again utilize 3DES in light of the fact that there are better choices.

The National Institute of Standards and Technology (NIST) has discharged a draft proposition saying that all types of 3DES will be expostulated through 2023 and prohibited from 2024 ahead. Despite the fact that it's only a draft, the proposition connotes the finish of a time, and it is well past an opportunity to move onto other, progressively secure calculations.

The historical backdrop of 3DES encryption

Since 3DES is gotten from DES, it's ideal to present the prior standard first. In the seventies, the National Bureau of Standards (NBS – it has since been renamed NIST) was looking for a calculation that it could use as a standard for scrambling delicate yet unclassified government data.

The NBS acknowledged recommendations for a standard that would accommodate its prerequisites, however none of the hopefuls from the first round were suitable. It welcomed more entries, and this time IBM sent through a calculation that its group created. The accommodation was gotten from the Lucifer figure that Horst Feistel planned.

In 1975, the IBM calculation was distributed by the NBS as the proposed Data Encryption Standard. The open was welcome to remark on the plan, which pulled in some analysis.

Unmistakable cryptographers, for example, Whitfield Diffie and Martin Hellman, fashioners of the Diffie-Hellman key trade, guaranteed that the key length was excessively short and that the S-boxes had been transformed from their underlying structure.

At the time, numerous in the cryptographic network felt that the NSA had undermined the venture and debilitated the calculation, so it would be the main office that could break DES.

At the point when this was researched by the United States Senate Select Committee on Intelligence, it was discovered that the "NSA persuaded IBM that a decreased key size was adequate; by implication aided the advancement of the S-box structures; and guaranteed that the last DES calculation was, as far as they could possibly know, free from any measurable or scientific shortcoming."

A similar report proceeded to state that the "NSA did not mess with the plan at all." This has been sponsored up by some previous IBM staff who asserted that the DES calculation was structured completely by the IBM group.

The NSA's very own declassified documentation guarantees that the office "worked intimately with IBM to fortify the calculation against all aside from beast power assaults and to reinforce substitution tables… "

Doubts of NSA altering were facilitated in the nineties once differential cryptanalysis was freely found. At the point when the much-censured S-boxes were tried with the new method, they were observed to be more impervious to assault than if they had been picked haphazardly.

This demonstrates the IBM group had definitely thought about the differential cryptanalysis in the seventies, with Steven Levy asserting that the NSA requested that they keep the method mystery so as to ensure national security.

Acclaimed cryptographer Bruce Schneier once joked, "It took the scholastic network two decades to make sense of that the NSA 'changes' really improved the security of DES."

In spite of the underlying inquiries concerning the calculation's security and the NSA's association, the IBM calculation proceeded to be endorsed as the Data Encryption Standard in 1976. It was distributed in 1977 and reaffirmed as the standard in 1983, 1988 and 1993.

At the point when straight cryptanalysis was first distributed in 1994, it began to bring up issues about the security of the calculation. In 1997, NIST declared that is was searching for a calculation to supplant DES. The requirement for another calculation was escalated as innovation grew further and potential assaults became more grounded.

Different splitting endeavors demonstrated that it was less hard to break the calculation than recently suspected. In 1998, distributed.net had the capacity to split DES is 39 days.

By the beginning of 1999, the Electronic Frontier Foundation's Deep Crack had gotten the time down to barely 22 hours. This flagged the finish of DES, since an assault of this nature was currently inside the compass of a well-resourced foe.

The fundamental issue was the little key space, and another calculation was distressfully required. This was an issue, since it would take a few additional years for NIST to settle on the calculation which turned into the substitution standard, the Advanced Encryption Standard (AES).

While the figure for AES was being settled on, 3DES was proposed on as a stopgap measure. It includes running the DES calculation multiple times, with three separate keys. In

1999, DES was reaffirmed, however with 3DES as the perfect calculation. Typical DES was just allowed in heritage applications.

3DES proceeded to turn into a far reaching encryption calculation, despite the fact that its substantial utilization of assets and security impediments have driven it to be supplanted by AES in most of utilization cases.

Understanding the DES calculation

Before we can discuss the subtleties of 3DES, it's critical to comprehend the DES calculation that it's gotten from. So we should begin directly toward the start.

We use encryption to transform our plaintext information into ciphertext, which is data that can't be gotten to by aggressors (as long as we are utilizing proper calculations).

Encryption calculations are basically mind boggling numerical recipes. With regards to encoding something like "How about we go to the shoreline", numerous individuals get befuddled. All things considered, how might you apply math to things like letters and characters?

Encoding the content

Actually PCs don't bargain in letters and characters. Rather, they deal with an arrangement of 0s known as paired. Every 1 or 0 is known as a bit, and a gathering of eight of them are known as a byte.

You can either find it physically, or utilize an online converter to see that in twofold, "We should go to the shoreline" moves toward becoming:

01001100 01100101 01110100 00100111 01110011 00100000 01100111 01101111 00100000 01110100 01101111 00100000 01110100 01101000 01100101 00100000 01100010 01100101 01100001 01100011 01101000

Squares

At the point when information is scrambled, it's partitioned into discrete squares for handling. DES has a 64-bit square size, which basically implies that each square fits a blend of 64 zeros. Our first square (the initial 64 digits of the double appeared) would be:

01001100 01100101 01110100 00100111 01110011 00100000 01100111 01101111

Our second would be:

00100000 01110100 01101111 00100000 01110100 01101000 01100101 00100000

What's more, our last square would be:

01100010 01100101 01100001 01100011 01101000

Tokens

Tokenization, when connected to information security, is the way toward substituting a delicate information component with a non-touchy proportional, alluded to as a token, that has no extraneous or exploitable importance or esteem. The token is a reference (for example identifier) that maps back to the touchy information through a tokenization framework. The mapping from unique information to a token uses techniques which render tokens infeasible to switch without the tokenization framework, for instance utilizing tokens made from irregular numbers.The tokenization framework must be verified and approved utilizing security best practices pertinent to delicate information insurance, secure capacity, review, confirmation and approval. The tokenization framework gives information preparing applications the specialist and interfaces to demand tokens, or detokenize back to delicate information.

The security and hazard decrease advantages of tokenization necessitate that the tokenization framework is coherently disconnected and fragmented from information preparing frameworks and applications that recently handled or put away touchy information supplanted by tokens. Just the tokenization framework can tokenize information to make tokens, or detokenize back to reclaim delicate information under exacting security controls. The token age strategy must be demonstrated to have the property that there is no possible methods through direct assault, cryptanalysis, side channel examination, token mapping table introduction or beast power systems to switch tokens back to live information.

At the point when tokens supplant live information in frameworks, the outcome is limited presentation of delicate information to those applications, stores, individuals and procedures, decreasing danger of trade off or coincidental introduction and unapproved access to touchy information. Applications can work utilizing tokens rather than live information, except for few believed applications expressly allowed to detokenize when carefully important for an endorsed business reason. Tokenization frameworks might be worked in-house inside a protected secluded portion of the server farm, or as an administration from a safe specialist organization.

Tokenization might be utilized to shield delicate information including, for instance, ledgers, budget summaries, therapeutic records, criminal records, driver's licenses, advance applications, stock exchanges, voter enrollments, and different sorts of by and by recognizable data (PII). Tokenization is frequently utilized in Visa handling. The PCI Council characterizes tokenization as "a procedure by which the essential record number (PAN) is supplanted with a surrogate esteem called a token. De-tokenization is the invert procedure of reclaiming a token for its related PAN esteem. The security of an individual token depends transcendently on the infeasibility of deciding the first PAN knowing just the surrogate value". The decision of tokenization as an option in contrast to different methods, for example, encryption will rely upon fluctuating administrative necessities, translation, and acknowledgment by particular reviewing or evaluation substances. This is notwithstanding any specialized, building or operational requirement that tokenization forces in commonsense use.

Authentication is important because it enables organizations to keep their networks secure by permitting only authenticated users (or processes) to access its protected resources, which may include computer systems, networks, databases, websites and other network-based applications or services. As used in the project, a user must insert a correct username and password in order to gain access. Tokens were also used as a one time use, a user can access his account only once before he is locked out. The goal of access control is to minimize the risk of unauthorized access to physical and logical systems. Access control is a fundamental component of security compliance programs that ensures security technology and access control policies are in place to protect confidential information, such as customer data. Most organizations have infrastructure and procedures that limit access to networks, computer systems, applications, files and sensitive data, such as personally identifiable information and intellectual property. Which was used to send private messages within a group to one specific user only. Encryption is widely used on the internet to protect user information being sent between a browser and a server, including passwords, payment information and other personal information that should be considered private. Organizations and individuals also commonly use encryption to protect sensitive data stored on computers, servers and mobile devices like phones or tablets. Therefore encrypted messages were printed on the console to avoid unauthorized access.