SAMSUNG SW ACADEMY FOR YOUTH

對外秘

프로젝트 명세서

MySQL 내장함수를 이용한 Data 암호화

목 차

1. 과제 개요

기업에서 일반적으로 데이터를 저장할 때에는 Database 를 이용합니다. DB 에는 외부 공개 가능 정보, 회사 기밀정보, 이용자 개인정보 등 다양한 종류의 데이터가 저장이 되는데, 외부에 공개할 수 없는 데이터가 DB 에 저장이 되면, 기업에서는 기본적으로 누구나 쉽게 접근하지 못하도록 계정과 권한을 관리하고, 접근 이력을 검토하여 이상 징후를 파악하는 등 중요 정보의 안전한 처리 환경을 만들기 위해 다양한 보호조치를 적용합니다.

DB 에서 개인정보를 처리하는 경우에는 개인정보보호법 제 29 조(안전조치의무) 및 시행령, 법률보충적 행정규칙(개인정보보호위원회 고시 제 2021-2, 3 호)에서 규정하는 보호조치 사항을 모두 적용해야 하며, 필요에 따라 안전성을 높이기 위해 추가적인 보호조치를 적용해야 합니다.

근거	준수 항목	상세 내용
안전성확보조치 제 7 조 제 2 항 기술적관리적보호조치 제 6 조 제 1 항	비밀번호 일방향 암호화	이용자, 개인정보취급자 등의 비밀번호가 노출 또는 위변조되지 않도록 일방향 암호화해서 저장하고, 난수 추가(Salting) 등의 조치를 취한다.
안전성확보조치 제 7 조 제 5 항 기술적관리적보호조치 제 6 조 제 2 항	개인정보 양방향 암호화	주민등록번호, 여권번호, 운전면허번호, 외국인등록번호, 신용카드번호, 계좌번호, 바이오정보는 안전한 알고리즘으로 양방향 암호화하여 저장한다.

이번 과제에서는 법령상 준수해야 하는 보호조치 중 데이터 암호화와 관련된 부분을 MYSQL 에서 제공하는 내장함수를 이용하여 회원관리 기능을 구현해 봅니다.

2. 기본 과제

본 명세서의 기본과제는

1. MYSQL 내장함수
2. 일방향 암호화
3. 양방향 암호화
4. 과제 안내

로 구성되어 있습니다.

1. MYSQL 내장함수

- MYSQL 서버는 다양한 내장함수를 포함하고 있습니다. 그 중 자주 쓰이는 함수를 종류별로 몇가지를 알아보겠습니다.

1) 집계 함수

COUNT(필드명) : NULL 값이 아닌 레코드의 수

SUM(필드명) : 필드명의 합계

AVG(필드명) : 필드명의 평균값

MAX(필드명) : 최대값

MIN(필드명) : 최소값

2) Math 함수

ABS(숫자) : 절대값

MOD(분자, 분모) : 나머지

CEILING : 소수점 올림

FLOOR : 소수점 내림

ROUND(숫자, 자리수) : 숫자를 소수점 이하 자리에서 반올림

3) String 함수

LENGTH(문자열) : 할당된 byte 반환

CONCAT(문자열, 문자열..) : 문자열 합치기

LEFT(문자열, 숫자) : 문자열 좌측부터 숫자 만큼 자르기

RIGHT(문자열, 숫자) : 문자열 우측부터 숫자 만큼 자르기

4) Date 함수

NOW(), SYSDATE(). CURRENT_TIMESTAMP() : 현재 시각 출력

DATE_ADD(날짜, INTERVAL 기준값) : 날짜에서 기준값 만큼 덧셈

ex : SELECT DATE_ADD(NOW() , INTERVAL 1 SECOND);

- 그 외에도 다양한 함수들을 제공하고 있습니다.(공식 Documentation 참고)

https://dev.mysql.com/doc/refman/8.0/en/built-in-function-reference.html

2. 일방향 암호화

- 일방향 암호화란 해쉬 함수를 이용하여 Hash 를 만드는 알고리즘입니다. 해쉬 함수를 이용하면 암호화는 가능하지만 복호화는 불가능합니다. 그렇기 때문에 암호화된 내용을 서로 비교해서 같은지 아닌지로 올바른 데이터인지 확인을 합니다. 대부분의 서비스에서 비밀번호 찾기를 할 경우 과거 비밀번호를 알려주지 않고, 비밀번호 변경만 제공하는 이유가 여기에 있습니다.

- MYSQL 에서는 3 가지 일방향 암호화를 제공하고 있습니다.

  1) MD5 : 128bit hash 로 변환하는 일방향 암호화 알고리즘. 보안에 취약해 개인정보 암호화에 사용하는 것은 권장하지 않음.

  2) SHA-1 : 입력받은 문자열을 160bit 의 Digest 로 변환하는 해쉬 알고리즘. 해독방법이 제시되어 최근에는 사용하지 않음

  3) SHA-2 : SHA-1 해독방법이 제시되어 새롭게 공표된 암호화 해시 함수

- 암호화 함수 : SHA2(문자열, 해쉬값 크기)

  ■ 문자열을 해쉬값 크기(bit)로 된 Hash 로 변환.

  ■ 해쉬값 크기는 224, 256, 384, 512 가 들어감 일반적으로 256, 또는 512 를 사용합니다.

  ■ SHA-2 256bit(SHA-256) : 16 진수 숫자로 이루어진 길이 64 의 문자열 반환 (varchar(64))

  ■ SHA-2 512bit(SHA-512) : 16 진수 숫자로 이루어진 길이 128 의 문자열 반환 (varchar(128))

3. 양방향 암호화

- 양방향 암호화란 데이터에 대한 복호화가 가능한 암호화 방식입니다. 암호화 하여 저장하지만, 필요에 따라 복호화가 가능해야 하는 정보를 저장하는 경우(주민등록번호, 계좌번호 등) 사용합니다.

- 대표적으로 대칭키(DES, 3DES, AES, SEED, ARIA 등), 공개키(RSA) 암호화 방식이 있습니다.

- MYSQL 에서는 대칭키 암호화 알고리즘으로 AES 와 DES 를 제공합니다. 사용방법은 동일하나 8.0.3 버전부터 DES 알고리즘은 제거되어 AES 알고리즘을 사용합니다.

- AES 알고리즘은 블록 알고리즘의 일종으로 특정 문장을 암호화 할 때 고정된 블록 단위로 암호화 합니다. 특정 문장에서 여러 블록이 있을 때, 블록 내의 문장이 동일할 경우 같은 암호화 결과가 나와 원문을 추론할 수 있다는 보안적 이슈에 의해 iv(초기화 벡터, initialization vector)를 사용하여 동일 문장 추론을 방지할 수 있습니다. 이번 과제에서는 iv 를 이용하지 않는 ECB 운용방식으로 진행하겠습니다.

- MYSQL 8.x 에서는 기본적으로 128 비트 인코딩입니다.

- DBMS 기본 모드 확인

  SELECT @@session.block_encryption_mode;

- DBMS 기본 모드 변경(256 비트로 변경 가능)

  SET @@session.block_encryption_mode = 'aes-256-ecb';

- 암호화 함수 : AES_ENCRYPT(문자열, 암호화키)

■ 열을 암호화하여 바이너리 데이터로 반환(ex. 쳑 P,j 쫹믹 s^/뢟)

SELECT AES_ENCRYPT('01012345678', 'enckey');

■ 일반적으로 HEX() 등을 통해 16 진수로 변환하여 저장(바이너리 데이터를 문자열로 사용하는 경우 프로그램상 오류가 발생하는 경우가 발생할 수 있음)

SELECT HEX(AES_ENCRYPT('01012345678', 'enckey'));

■ 암호화키 자체도 단방향 암호화로 암호화시키기도 함

SELECT HEX(AES_ENCRYPT('01012345678', SHA2('enckey', 512)));

- 복호화 함수 : AES_DYCRYPT(암호화된 문자열, 암호화키)

■ 원래 문자열로 복호화

SELECT
AES_DECRYPT(unhex('AB8D1B502C6AA690B9CD735E2F8F5A20'),
'enckey');

SELECT
AES_DECRYPT(unhex('3D0E414D9B8ADA7E252235C89167BB8B'),
sha2('enckey', 512));

4. 과제 안내

- 대부분의 서비스는 회원가입 및 LOGIN 을 구현하게 됩니다. 암호화 함수들을 이용하여 회원정보 관련 부분의 Query 를 작성해 봅니다.

1) 회원 관리 테이블 생성

   회원아이디(평문), 회원비밀번호(단방향 암호화),
   회원계좌번호(양방향암호화) 반드시 포함)

2) 회원가입 : 해당되는 암호화 방식을 사용하여 회원데이터 등록

3) 회원정보조회 : 복호화된 정보를 포함하여 회원정보 조회

4) 회원정보수정 : 해당되는 암호화 방식을 사용하여 회원데이터 수정

5) 로그인 : 올바른 유저인지 조회

6) 비밀번호 변경 : 비밀번호 변경

3. 심화 학습

1. 과제를 위해 암호화에 대해서 간단하게 설명하였으나, 실제로는 훨씬 복잡한 내용들이 있습니다. 이번 기회에 다양한 암호화 알고리즘에 대해 깊이 있게 공부해 봅니다.
2. 다른 DBMS(Oracle, PostgreSQL 등)에서는 어떤 방식으로 암호화를 제공하는지 공부해 봅니다.

4. 산출물 제출

1. https://lab.ssafy.com/s12-study/seasonal_fesw 의 "산출물 제출 가이드" 참조

2. 제출할 내역
   - 작성한 Table CREATE 문
   - 작성한 회원관리 Query 문