

Opdracht 4 - Security

Verslag

Analyse 8 - Advanced Databases (2014-2015)

Namen:	HoYe Lam, Rinesh Ramadhin
Studentnummer:	0876814, 0882447
Klas:	INF2D
Vak:	INFANL01-8
Opdracht:	Security
Datum:	2 – 06 - 2015

Inhoud

Database structuur	2
Database roles, views & users.....	3
Database vullen	4
SQL injections	5
Database secured	9

Database structuur

```
-- Database: postgres
-- DROP DATABASE postgres;

DROP SCHEMA public CASCADE;
CREATE SCHEMA public;

CREATE DATABASE postgres
  WITH OWNER = postgres
       ENCODING = 'UTF8'
       TABLESPACE = pg_default
       LC_COLLATE = 'English_United States.1252'
       LC_CTYPE = 'English_United States.1252'
       CONNECTION LIMIT = -1;

COMMENT ON DATABASE postgres
  IS 'default administrative connection database';

CREATE TABLE Student(
  studentnummer          VARCHAR(7)   NOT NULL PRIMARY KEY CHECK (length(studentnummer) = 7)
, wachtwoord             VARCHAR(45)  NOT NULL
, naam                   VARCHAR(45)  NOT NULL
, klas                   VARCHAR(45)
, ingeschreven           BOOLEAN      NOT NULL
);
```

Database roles, views & users

```
DROP OWNED BY student_role;
REVOKE ALL ON student FROM student_role;
DROP USER student_user;
DROP ROLE student_role;

CREATE ROLE student_role;
GRANT CONNECT ON DATABASE postgres TO student_role;
GRANT USAGE ON SCHEMA public TO student_role;
GRANT SELECT ON student TO student_role;
GRANT SELECT ON INF0D_view TO student_role;
GRANT SELECT ON INF1D_view TO student_role;
GRANT SELECT ON INF2D_view TO student_role;
GRANT SELECT ON INF3D_view TO student_role;
GRANT SELECT ON INF4D_view TO student_role;

CREATE USER student_user WITH PASSWORD '1234';
GRANT student_role to student_user;

CREATE OR REPLACE VIEW INF1D_view AS
SELECT studentnummer, naam, klas, ingeschreven FROM student WHERE klas = 'INF1D' AND ingeschreven = 'true';

CREATE OR REPLACE VIEW INF2D_view AS
SELECT studentnummer, naam, klas, ingeschreven FROM student WHERE klas = 'INF1D' AND ingeschreven = 'true';

CREATE OR REPLACE VIEW INF3D_view AS
SELECT studentnummer, naam, klas, ingeschreven FROM student WHERE klas = 'INF1D' AND ingeschreven = 'true';

CREATE OR REPLACE VIEW INF4D_view AS
SELECT studentnummer, naam, klas, ingeschreven FROM student WHERE klas = 'INF1D' AND ingeschreven = 'true';

CREATE OR REPLACE VIEW INF0D_view AS
SELECT studentnummer, naam, klas, ingeschreven FROM student WHERE klas = 'INF1D' AND ingeschreven = 'true';
```

Database vullen

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;
import java.util.Random;

public class DataInvoegen {
    //Functie Variables
    public static Random rand = new Random();

    public static void main(String[] args) {
        // Maak en start thread 1
        new Thread(new Runnable() {
            @Override
            public void run() {
                Connection connection = Applicatie.connect();

                // Begin 600 iteraties van student en klas toevoegen
                for (int i = 0; i < 600; i++) {

                    // random string
                    char[] chars = "abcdefghijklmnopqrstuvwxyz".toCharArray();
                    StringBuilder sb = new StringBuilder();
                    Random random = new Random();
                    for (int nu = 0; nu < 6; nu++) {
                        char c = chars[random.nextInt(chars.length)];
                        sb.append(c);
                    }

                    // transactie
                    try {
                        Statement st = (Statement) connection.createStatement();

                        int studentnummer = 1876814 + i;
                        String klas = "INF" + rand.nextInt(5) + "D";
                        String naam = sb.toString();

                        int inge = rand.nextInt(2)+1;
                        if (inge == 1){
                            st.executeUpdate("INSERT INTO student " + "VALUES ('"
                                + studentnummer + "','" + "1234'" + " " + naam + "','"
                                + klas + "','true');"");
                        }
                        else{
                            st.executeUpdate("INSERT INTO student " + "VALUES ('"
                                + studentnummer + "','" + "1234'" + " " + naam + "','"
                                + klas + "','false');"");
                        }
                    } catch (SQLException e1) {
                        // TODO Auto-generated catch block
                        e1.printStackTrace();
                    }
                }

                try {
                    System.out.println("Done!");
                    connection.close();
                } catch (SQLException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
            }
        }, "Thread 1").start();
    }
}
```

Voor toelichting zie verslag opdracht 3.

SQL injections

```
import java.io.Console;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Random;
import java.util.Scanner;

public class Applicatie {
    // Connectie variables
    public static String url = "jdbc:postgresql://127.0.0.1:5432/postgres";
    public static String username = "student_user";
    public static String password = "1234";

    // SQL Variables
    public static Statement st1;
    public static String studentnummer;
    public static String wachtwoord;
    public static String klas;
    public static String ingeschreven;
    public static String naam;
    public static String opties;
    public static String opties_gebruiker_ingelogd;
    public static String wachtwoord_oud;
    public static String wachtwoord_nieuw;

    // Applicatie variables
    public static Scanner inputReader = new Scanner(System.in);
    public static Console console = System.console();

    public static void main(String[] args) {
        // Connect met Database
        try {
            connect();
        } catch (Exception e) {
            System.out.println("Database Offline");
        }

        // kies een optie
        opties();
    }

    // Opties voor wat de gebruiker wilt doen
    public static void opties() {
        System.out
            .println("Type 'login' voor inloggen, 'klas' voor klas informatie of 'stop' om de applicatie te stoppen");

        opties = inputReader.nextLine();

        if (opties.equals("login")) {
            login();
        }
        if (opties.equals("klas")) {
            dataKlas();
        }
        if (opties.equals("stop")) {
            System.exit(0);
        } else {
            System.out.println("Verkeerde input, probeer het nog is");
            opties();
        }
    }

    //
    public static void opties_gebruiker_ingelogd() {
        System.out
            .println("Type 'gegevens' voor gegevens, 'wachtwoord' om wachtwoord te wijzigen of 'home' om terug naar het begin te gaan");

        opties_gebruiker_ingelogd = inputReader.nextLine();

        if (opties_gebruiker_ingelogd.equals("gegevens")) {
            getGegevens_gebruiker();
        }
        if (opties_gebruiker_ingelogd.equals("wachtwoord")) {
            wachtwoord_wijzigen();
        }
        if (opties_gebruiker_ingelogd.equals("home")) {

```

```

        opties();
    } else {
        System.out.println("Verkeerde input, probeer het nog is");
        opties();
    }
}

// wijzig wachtwoord dmV insert
public static void wachtwoord_wijzigen() {
    System.out.println("Voer uw huidige wachtwoord in : ");
    wachtwoord_oud = inputReader.nextLine();
    try {
        if (wachtwoord_oud.equals(wachtwoord)) {
            System.out.println("Voer uw nieuwe wachtwoord in : ");
            wachtwoord_nieuw = inputReader.nextLine();
            String wachtwoord_wijzigen = ("UPDATE student SET wachtwoord = '"
                + wachtwoord_nieuw
                + "' WHERE studentnummer = '"
                + studentnummer + "';");
            System.out.println(wachtwoord_wijzigen);
            st1.executeUpdate(wachtwoord_wijzigen);
            System.out.println("Succes! Uw wachtwoord is gewijzigd! ");
            opties_gebruiker_ingelogd();
        }
        else{
            System.out.println("Onjuist wachtwoord probeer het opnieuw : ");
            wachtwoord_wijzigen();
        }
    } catch (Exception e) {
        System.out
            .println("Er was iets fouts gegaan probeer het opnieuw : ");
        opties_gebruiker_ingelogd();
    }
}

// haal gegevens van de gebruiker op
public static void getGegevens_gebruiker() {
    String test = ("SELECT * FROM student WHERE studentnummer = '"
        + studentnummer + "' AND wachtwoord = '" + wachtwoord + "';");
    ResultSet z;
    System.out.println(test);

    try {
        z = st1.executeQuery(test);

        if (z.next()) {
            studentnummer = z.getString("studentnummer");
            naam = z.getString("naam");
            klas = z.getString("klas");
            ingeschreven = z.getString("ingeschreven");
        }

        System.out.println("Student : " + studentnummer);
        System.out.println("Naam : " + naam);
        System.out.println("In Klas : " + klas);

        if (ingeschreven.equals("f")) {
            System.out.println("is nog niet ingeschreven!");
        } else {
            System.out.println("is ingeschreven!");
            System.out.println("");
        }
        opties_gebruiker_ingelogd();
    } catch (Exception e) {
        System.out.println("Er was iets fouts gegaan probeer het nog is");
        opties();
    }
}

// Login
public static void login() {
    Connection connection = connect();

    System.out.println("Voer uw studentnummer in: ");
    studentnummer = inputReader.nextLine();

    System.out.println("Voer uw wachtwoord in: ");
    wachtwoord = inputReader.nextLine();

    try {
        st1 = (Statement) connection.createStatement();
        ResultSet z;
        String login_query = ("SELECT * FROM student WHERE studentnummer = '"
            + studentnummer + "' AND wachtwoord = '" + wachtwoord + "';");
    }
}

```

```

        System.out.println(login_query);
        z = st1.executeQuery(login_query);

        if (z.next()) {
            studentnummer = z.getString("studentnummer");
            naam = z.getString("naam");
            klas = z.getString("klas");
            ingeschreven = z.getString("ingeschreven");
        }
        if (naam == null) {
            System.out
                .println("Geen gebruiker gevonden, probeer het nog is");
            login();
        } else {
            opties_gebruiker_ingelogd();
        }
        // terug naar opties
        opties();
    } catch (Exception e) {
        System.out.println(e);
        System.out.println("Er was iets fout gegaan, probeer het nog is");
        opties();
    }
}

// Haal data binnen een klas
public static void dataKlas() {
    Connection connection = connect();

    studentnummer = null;
    naam = null;
    klas = null;
    ingeschreven = null;

    System.out.println("Voer klas in: ");
    klas = inputReader.nextLine();

    try {

        st1 = (Statement) connection.createStatement();
        ResultSet z;

        String getKlas = ("SELECT * FROM student WHERE klas = '" + klas
            + "' AND ingeschreven = 'true'");
        System.out.println(getKlas);
        z = st1.executeQuery(getKlas);

        if (z.next()) {
            studentnummer = z.getString("studentnummer");
            naam = z.getString("naam");
            klas = z.getString("klas");
            ingeschreven = z.getString("ingeschreven");
        }

        if (naam != null) {
            while (z.next()) {
                studentnummer = z.getString("studentnummer");
                naam = z.getString("naam");
                klas = z.getString("klas");
                ingeschreven = z.getString("ingeschreven");

                System.out.println("Student : " + studentnummer);
                System.out.println("Naam : " + naam);
                System.out.println("In Klas : " + klas);
                if (ingeschreven.equals("f")) {
                    System.out.println("is nog niet ingeschreven!");
                    System.out.println("");
                } else {
                    System.out.println("is ingeschreven!");
                    System.out.println("");
                }
            }
        } else {
            System.out.println("Geen klas gevonden, probeer het nog is");
        }
        opties();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

// Connect functie
public static Connection connect() {

```



```

// maak verbinding met postgres

// maak verbinding met de driver
try {
    Class.forName("org.postgresql.Driver");
} catch (ClassNotFoundException e) {
    throw new RuntimeException(
        "Cannot find the driver in the classpath!", e);
}

// maak verbinding met de database
Connection connection = null;
try {
    System.out.println("Connecting database...");
    connection = DriverManager.getConnection(url, username, password);
    System.out.println("Database connected!");
} catch (Exception e) {
}
return connection;
}
}

```

Database secured

```
import java.io.Console;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Scanner;

public class Applicatie_secured {
    // Connectie variables
    public static String url = "jdbc:postgresql://127.0.0.1:5432/postgres";
    public static String username = "postgres";
    public static String password = "hoye";
    public static Connection connection = connect();

    // SQL Variables
    public static Statement st1;
    public static String studentnummer;
    public static String wachtwoord;
    public static String klas;
    public static String ingeschreven;
    public static String naam;
    public static String opties;
    public static String opties_gebruiker_ingelogd;
    public static String wachtwoord_oud;
    public static String wachtwoord_nieuw;

    // Applicatie variables
    public static Scanner inputReader = new Scanner(System.in);
    public static Console console = System.console();

    public static void main(String[] args) {
        // Connect met Database
        try {
            connect();
        } catch (Exception e) {
            System.out.println("Database Offline");
        }

        // kies een optie
        opties();
    }

    // Opties voor wat de gebruiker wilt doen
    public static void opties() {
        System.out
            .println("Type 'login' voor inloggen, 'klas' voor klas informatie of 'stop' om de applicatie te stoppen");

        opties = inputReader.nextLine();

        if (opties.equals("login")) {
            login();
        }
        if (opties.equals("klas")) {
            dataKlas();
        }
        if (opties.equals("stop")) {
            System.exit(0);
        } else {
            System.out.println("Verkeerde input, probeer het nog is");
            opties();
        }
    }

    public static void opties_gebruiker_ingelogd() {
        System.out
            .println("Type 'gegevens' voor gegevens, 'wachtwoord' om wachtwoord te wijzigen of 'home' om terug naar het begin te gaan");

        opties_gebruiker_ingelogd = inputReader.nextLine();

        if (opties_gebruiker_ingelogd.equals("gegevens")) {
            getGegevens_gebruiker();
        }
        if (opties_gebruiker_ingelogd.equals("wachtwoord")) {
            wachtwoord_wijzigen();
        }
    }
}
```

```

        if (opties_gebruiker_ingelogd.equals("home")) {
            opties();
        } else {
            System.out.println("Verkeerde input, probeer het nog is");
            opties();
        }
    }

    // wijzig wachtwoord dmv update
    public static void wachtwoord_wijzigen() {
        System.out.println("Voer uw huidige wachtwoord in : ");
        wachtwoord_oud = inputReader.nextLine();
        try {
            if (wachtwoord_oud.equals(wachtwoord)) {
                System.out.println("Voer uw nieuwe wachtwoord in : ");
                wachtwoord_nieuw = inputReader.nextLine();
                PreparedStatement ps = connection.prepareStatement("UPDATE student SET wachtwoord = ? WHERE
studentnummer = ?");
                ps.setString(1, wachtwoord_nieuw);
                ps.setString(2, studentnummer);
                ps.executeUpdate();
                System.out.println("Succes! Uw wachtwoord is gewijzigd! ");
                opties_gebruiker_ingelogd();
            }
            else{
                System.out.println("Onjuist wachtwoord probeer het opnieuw : ");
                wachtwoord_wijzigen();
            }
        } catch (Exception e) {
            System.out
                .println("Er was iets fouts gegaan probeer het opnieuw : ");
            opties_gebruiker_ingelogd();
        }
    }

    // haalt gegevens op van de gebruiker
    public static void getGegevens_gebruiker() {
        ResultSet gegevens_gebruiker;

        try {
            PreparedStatement ps = connection.prepareStatement("SELECT * FROM student WHERE studentnummer = ? AND
wachtwoord = ?");
            ps.setString(1, studentnummer);
            ps.setString(2, wachtwoord);
            gegevens_gebruiker = ps.executeQuery();

            if (gegevens_gebruiker.next()) {
                studentnummer = gegevens_gebruiker.getString("studentnummer");
                naam = gegevens_gebruiker.getString("naam");
                klas = gegevens_gebruiker.getString("klas");
                ingeschreven = gegevens_gebruiker.getString("ingeschreven");
            }

            System.out.println("Student : " + studentnummer);
            System.out.println("Naam : " + naam);
            System.out.println("In Klas : " + klas);

            if (ingeschreven.equals("f")) {
                System.out.println("is nog niet ingeschreven!");
            } else {
                System.out.println("is ingeschreven!");
                System.out.println("");
            }
            opties_gebruiker_ingelogd();
        } catch (Exception e) {
            e.printStackTrace();
            System.out.println("Er was iets fouts gegaan probeer het nog is");
            opties();
        }
    }

    // Login
    public static void login() {

        System.out.println("Voer uw studentnummer in: ");
        studentnummer = inputReader.nextLine();

        System.out.println("Voer uw wachtwoord in: ");
        wachtwoord = inputReader.nextLine();

        try {
            PreparedStatement ps = connection.prepareStatement("SELECT * FROM student WHERE studentnummer = ? AND
wachtwoord = ?");
            ps.setString(1, studentnummer);

```

```

        ps.setString(2, wachtwoord);

        ResultSet z;

        z = ps.executeQuery();

        if (z.next()) {
            studentnummer = z.getString("studentnummer");
            naam = z.getString("naam");
            klas = z.getString("klas");
            ingeschreven = z.getString("ingeschreven");
        }
        if (naam == null) {
            System.out
                .println("Geen gebruiker gevonden, probeer het nog is");
            login();
        } else {
            opties_gebruiker_ingelogd();
        }

        // terug naar opties
        opties();

    } catch (Exception e) {
        System.out.println(e);
        System.out.println("Er was iets fout gegaan, probeer het nog is");
        opties();
    }
}

// Haal data binnen een klas
public static void dataKlas() {

    studentnummer = null;
    naam = null;
    klas = null;
    ingeschreven = null;

    System.out.println("Voer klas in: ");
    klas = inputReader.nextLine();

    try {

        PreparedStatement ps = connection.prepareStatement("SELECT * FROM "+klas+"_view WHERE ingeschreven =
'true'");
        ResultSet z;

        //String getKlas = ("SELECT * FROM "+klas+" WHERE klas = '" + klas
        // + "' AND ingeschreven = 'true'");
        //System.out.println(getKlas);
        System.out.println(ps);
        z = ps.executeQuery();

        if (z.next()) {
            studentnummer = z.getString("studentnummer");
            naam = z.getString("naam");
            klas = z.getString("klas");
            ingeschreven = z.getString("ingeschreven");
        }

        if (naam != null) {
            while (z.next()) {
                studentnummer = z.getString("studentnummer");
                naam = z.getString("naam");
                klas = z.getString("klas");
                ingeschreven = z.getString("ingeschreven");

                System.out.println("Student : " + studentnummer);
                System.out.println("Naam : " + naam);
                System.out.println("In Klas : " + klas);
                if (ingeschreven.equals("f")) {
                    System.out.println("is nog niet ingeschreven!");
                    System.out.println("");
                } else {
                    System.out.println("is ingeschreven!");
                    System.out.println("");
                }
            }
        } else {
            System.out.println("Geen klas gevonden, probeer het nog is");
        }
        opties();

    } catch (Exception e) {
        System.out.println("Verkeerde input, probeer het nog is");
    }
}

```

```

        opties();
    }
}

// Connect functie
public static Connection connect() {
    // maak verbinding met postgres

    // maak verbinding met de driver
    try {
        Class.forName("org.postgresql.Driver");
    } catch (ClassNotFoundException e) {
        throw new RuntimeException(
            "Cannot find the driver in the classpath!", e);
    }

    // maak verbinding met de database
    Connection connection = null;
    try {
        System.out.println("Connecting database...");
        connection = DriverManager.getConnection(url, username, password);
        System.out.println("Database connected!");
    } catch (Exception e) {
    }
    return connection;
}
}

```