NTNU
Institutt for datateknikk
og informasjonsvitenskap

Exam

TDT4237 – Software Security (Programvaresikkerhet)

Monday December 10th 2012, 15:00 - 19:00

Oppgaven er utarbeidet av faglærer Lillian Røstad og kvalitetssikrer er Torbjørn Skramstad.
Kontaktperson under eksamen er Lillian Røstad (mobil 994 00 628)

Språkform: Engelsk/Bokmål/Nynorsk

Tillatte hjelpemidler: D

Ingen trykte eller håndskrevne hjelpemidler tillatt. Bestemt, enkel kalkulator tillatt.
No printed or hand-written materials allowed. Approved calculator allowed.

Results available (sensurfrist): Thursday January 10th 2013

Read each task carefully. Identify what the task asks for.
If you find that information is missing in a task you are free to make the assumptions you consider necessary, but remember to explain and clearly mark your assumptions.

Les oppgaveteksten nøye. Finn ut hva det spørres etter i hver oppgave.
Dersom du mener at opplysninger mangler i en oppgaveformulering gjør kort rede for de antagelser og forutsetninger som du finner det nødvendig å gjøre.

ENGLISH

Task 1 (30%)

a) Compare XSS and CSRF. For both of these attacks, explain why they are possible (vulnerability) and how they are performed - using an example.

b) Explain the Needham-Schroeder protocol.

c) Compare DAC (Discretionary Access Control) and MAC (Mandatory Access Control) – what are the main differences?

d) What is the simple security property?

e) Compare the Biba and Bell-LaPadula policy models – what are the main differences?

f) List the seven touchpoints of software security, in order of effectiveness.

Task 2 (30%) – Code Quiz

On the next page is a set of Java-based code excerpts. For each case your task is to identify security vulnerabilities in the code.

For every vulnerability you identify, you shall:

1. Explain how it could be exploited, and
2. Explain how you would fix the code. This part of your answer should include the corrected code.

Case 1 :

```
import java.security.MessageDigest;

 public byte[] getHash(String password) throws NoSuchAlgorithmException {
       MessageDigest digest = MessageDigest.getInstance("SHA-1");
       digest.reset();
       byte[] input = digest.digest(password.getBytes("UTF-8"));
       return input;
 }
```

Case 2:

```
public final Connection getConnection() throws SQLException {
  return DriverManager.getConnection(
      "jdbc:mysql://localhost/dbName",
      "username", "password");
}
```

Case 3:

```
import java.util.Random;

Random number = new Random(123L);
//...
for (int i = 0; i < 20; i++) {
  // Generate another random integer in the range [0, 20]
  int n = number.nextInt(21);
  System.out.println(n);
}
```

Case 4:

```
class SecurityIOException extends IOException {/* ... */};

try {
  FileInputStream fis =
      new FileInputStream(System.getenv("APPDATA") + args[0]);
} catch (FileNotFoundException e) {
  // Log the exception
  throw new SecurityIOException();
}
```

Case 5:

```
private void createXMLStream(BufferedOutputStream outStream,
                             String quantity) throws IOException {
  String xmlString;
  xmlString = "<item>\n<description>Widget</description>\n" +
              "<price>500.0</price>\n" +
              "<quantity>" + quantity + "</quantity></item>";
  outStream.write(xmlString.getBytes());
  outStream.flush();
}
```

Task 3 (20%) – E-voting

Several countries around the world have in recent years explored and experimented with e-voting. The basic idea behind e-voting is to allow votes to be cast online in elections, and the motivation is of course that making voting easier and more accessible will help increase participation in elections.

Security concerns is the number one reason why e-voting is still not widespread. Your company still believes e-voting is the future, and is eager to develop a product for online voting. Your task is to figure out how this can be done, and what security requirements such a product needs to fulfill

Use the Risk Management Framework (RMF) and perform one iteration for your e-voting system. The result from this initial iteration of the RMF should be a set of security requirements.

Task 4 (20%) – Futuristic: computer science exams on computers

Security concerns is also the number one reason why exams like the one you are working on right now is still performed using pen and paper. Suppose that NTNU have selected three potential systems to be used to provide exams such as this one on computers. As part of the evaluation process for the candidate systems, NTNU has created a hacking contest. Students are invited to create teams and they will have access to all three systems for one month. The systems are deployed online, and are available for testing to anyone with a valid NTNU-account. A key feature of all systems is that they should be able to automatically detect plagiarism.

The grand prize is tickets and travel expenses for the entire team to attend DEF CON in Las Vegas, one of the world's largest hacking conventions.

Your team really wants to win. You realize that you need a plan, because as the quote goes - *failing to plan equals planning to fail!*

Create a test-plan that will help your team win this contest. Explain the techniques you use.

BOKMÅL

Oppgave 1 (30%)

a) Sammenlign XSS og CSRF. Forklar for begge angrepene hvorfor de er mulige (sårbarhet) og hvordan et angrep utføres. Bruk eksempler.

b) Forklar Needham-Schroeder protokollen.

c) Sammenlign DAC (Discretionary Access Control) og MAC (Mandatory Access Control) – hva er hovedforskjellene mellom disse to?

d) Hva er "the simple security property"?

e) Sammenlign Biba og Bell-LaPadula policy modellene – hva er hovedforskjellene?

f) List de syv "touchpoints of software security", ordnet etter effekt.

Oppgave 2 (30%) – CodeQuiz

På den neste siden i oppgaven finner du et sett med Java-baserte utdrag fra kildekode. For hvert av disse er oppgaven din å identifisere sårbarheter i koden.

For hver sårbarhet du identifiserer, skal du:

1. Forklare hvordan sårbarheten kan utnyttes, og
2. Forklare hvordan du ville rettet koden. Denne delen av svaret ditt skal inkludere den rettede koden.

Case 1 :

```
import java.security.MessageDigest;

 public byte[] getHash(String password) throws NoSuchAlgorithmException {
       MessageDigest digest = MessageDigest.getInstance("SHA-1");
       digest.reset();
       byte[] input = digest.digest(password.getBytes("UTF-8"));
       return input;
 }
```

Case 2:

```
public final Connection getConnection() throws SQLException {
  return DriverManager.getConnection(
      "jdbc:mysql://localhost/dbName",
      "username", "password");
}
```

Case 3:

```
import java.util.Random;

Random number = new Random(123L);
//...
for (int i = 0; i < 20; i++) {
  // Generate another random integer in the range [0, 20]
  int n = number.nextInt(21);
  System.out.println(n);
}
```

Case 4:

```
class SecurityIOException extends IOException {/* ... */};

try {
  FileInputStream fis =
      new FileInputStream(System.getenv("APPDATA") + args[0]);
} catch (FileNotFoundException e) {
  // Log the exception
  throw new SecurityIOException();
}
```

Case 5:

```
private void createXMLStream(BufferedOutputStream outStream,
                             String quantity) throws IOException {
  String xmlString;
  xmlString = "<item>\n<description>Widget</description>\n" +
              "<price>500.0</price>\n" +
              "<quantity>" + quantity + "</quantity></item>";
  outStream.write(xmlString.getBytes());
  outStream.flush();
}
```

Oppgave 3 (20%) – E-valg

Flere land har de senere årene eksperimentert med online stemmegiving - e-valg. Hovedtanken bak e-valg er at man skal kunne stemme på nett. Motivasjonen kommer selvfølgelig fra at man ønsker å øke deltagerraten i valg ved å gjøre det enklere for folk å stemme.

Sikkerhetsbekymringer er den viktigste årsaken til at e-valg fortsatt ikke er utbredt. Men din bedrift har troen på at dette er fremtiden, og ønsker å utvikle et produkt for e-valg. Din oppgave er å finne ut hvordan dette kan gjøres, og hvilke sikkerhetskrav et slikt produkt må oppfylle.

Bruk Risk Management Framework (RMF) og gjennomfør én iterasjon for e-valgsystemet deres. Resultat fra denne initielle gjennomføringen av RMF skal være et sett med sikkerhetskrav.

Oppgave 4 (20%) – Futuristic: eksamen i datateknikk på datamaskin

Sikkerhetsbekymringer er også den viktigste årsaken til at eksamener som den du avlegger akkurat nå, fortsatt gjennomføres med penn og papir. Anta at NTNU har valgt ut tre kandidatsystemer som skal kunne brukes til å tilby eksamener, slike som denne, på data. Som en del av evalueringen av kandidatsystemene, har NTNU laget en hacking-konkurranse. Studenter inviteres til å delta ved å opprette team, og teamene vil ha tilgang til alle tre systemene i én måned. Systemene settes opp online, og vil være tilgjengelig for alle med en gyldig NTNU-konto. En viktig egenskap ved alle systemene er at de skal være i stand til å automatisk avsløre juks (plagiat).

Hovedpremien er reise og opphold til DEF CON i Las Vegas for hele teamet. DEF CON er en av verdens største konferanser for hackere.

Laget ditt vil virkelig vinne. Og du er godt kjent med begrepet - *failing to plan equals planning to fail!*

Lag en testplan som vil hjelpe laget ditt å vinne. Forklar teknikkene du bruker.

NYNORSK

Oppgåve 1 (30%)

a) Samanlikn XSS og CSRF. Forklår for begge angrepa kvifor dei er mogelege (sårbarheit) og korleis eit angrep vert utført. Bruk eksempel.
b) Forklår Needham-Schroeder protokollen.
c) Samanlikn DAC (Discretionary Access Control) og MAC (Mandatory Access Control) – kva er hovudforskjellane mellom desse to?
d) Kva er "the simple security property"?
e) Samanlikn Biba og Bell-LaPadula policy modellane – kva er hovudforskjellane?
f) List dei sju "touchpoints of software security", ordna etter effekt.

Oppgåve 2 (30%) – CodeQuiz

På den neste sida i oppgåva finn du eit sett med Java-baserte utdrag frå kjeldekode. For kvart av desse er oppgåva di å finne sårbarheter i koden.

For kvar sårbarhet du finn, skal du:

1. Forklåre korleis sårbarheten kan utnyttast, og
2. Forklåre korleis du ville retta koden. Denne delen av svaret ditt skal inkludere den retta koden.

Case 1 :

```
import java.security.MessageDigest;

 public byte[] getHash(String password) throws NoSuchAlgorithmException {
       MessageDigest digest = MessageDigest.getInstance("SHA-1");
       digest.reset();
       byte[] input = digest.digest(password.getBytes("UTF-8"));
       return input;
 }
```

Case 2:

```
public final Connection getConnection() throws SQLException {
  return DriverManager.getConnection(
      "jdbc:mysql://localhost/dbName",
      "username", "password");
}
```

Case 3:

```
import java.util.Random;

Random number = new Random(123L);
//...
for (int i = 0; i < 20; i++) {
  // Generate another random integer in the range [0, 20]
  int n = number.nextInt(21);
  System.out.println(n);
}
```

Case 4:

```
class SecurityIOException extends IOException {/* ... */};

try {
  FileInputStream fis =
      new FileInputStream(System.getenv("APPDATA") + args[0]);
} catch (FileNotFoundException e) {
  // Log the exception
  throw new SecurityIOException();
}
```

Case 5:

```
private void createXMLStream(BufferedOutputStream outStream,
                             String quantity) throws IOException {
  String xmlString;
  xmlString = "<item>\n<description>Widget</description>\n" +
              "<price>500.0</price>\n" +
              "<quantity>" + quantity + "</quantity></item>";
  outStream.write(xmlString.getBytes());
  outStream.flush();
}
```

Oppgåve 3 (20%) – E-val

Fleire land har de seinare åra eksperimentert med online stemmegjeving - e-val. Hovudtanken bak e-val er at ein skal kunne stemme på nett. Motivasjonen kjem sjølvsagt frå at ein ynskjer å auke deltakarprosenten i val ved å gjere det enklare for folk å stemme.

Sikkerheitsgrunnar er den viktigaste årsaken til at e-val fortsatt ikkje er utbredt. Men din bedrift har trua på at dette er framtida, og ynskjer å utvikle eit produkt for e-val. Di oppgåve er å finna ut korleis dette kan gjeras, og kva for sikkerheitskrav eit slikt produkt må fylla.

Bruk Risk Management Framework (RMF) og gjennomfør éin iterasjon for e-valsystemet. Resultat frå denne initielle gjennomføringa av RMF skal være eit sett med sikkerheitskrav.

Oppgåve 4 (20%) – Futuristic: eksamen i datateknikk på datamaskin

Sikkerhetsbekymringar er og den viktigaste årsaken til at eksamenar som den du svarar på nå, fortsatt gjennomføras med penn og papir. Anta at NTNU har valt tre kandidatsystem som skal kunne brukast til å tilby eksamenar, slike som denne, på datamaskin. Som ein del av evalueringa av kandidatsystema, har NTNU laga ein hacking-konkurranse. Studentar vert inviterte til å delta ved å opprette team, og teama vil ha tilgang til alle tre systema i éin månad. Systema vert sett opp online, og vil være tilgjengelige for alle med ein gyldig NTNU-konto. Ein viktig eigenskap ved alle systema er at dei skal væra i stand til å automatisk avsløre juks (plagiering).

Hovudpremien er reise og opphald til DEF CON i Las Vegas for heile teamet. DEF CON er ein av verdas største konferansar for hackere.

Laget ditt vil verkelig vinne. Og du er godt kjent med omgrepet - *failing to plan equals planning to fail!*

Lag ein testplan som vil hjelpe laget ditt å vinne. Forklår teknikkane du bruker.