Javaセキュアコーディングセミナー東京

第2回
数値データの取扱いと入力値の検証
演習

2012年10月14日(日)
JPCERTコーディネーションセンター脆弱性解析チーム
戸田 洋三

Hands-on Exercises

- サンプルコード Unzip を修正しよう
- サンプルコード AltConst を修正しよう

Hands-on Exercise(1)

サンプルコード Unzip を
修正しよう

ZipBombの影響を受けるコード例

```
class Unzip {
  static final int BUFFER = 512;

  public static void main(String[] args) throws FileNotFoundException,IOException {
    BufferedOutputStream dest = null;
    ZipInputStream zis =
        new ZipInputStream(new BufferedInputStream(new FileInputStream(args[0])));
    ZipEntry entry;
    while ((entry = zis.getNextEntry()) != null){
       System.out.println("Extracting: " + entry);
       int count;
       byte data[] = new byte[BUFFER];
       FileOutputStream fos = new FileOutputStream(entry.getName());
       dest = new BufferedOutputStream(fos, BUFFER);
       while ((count=zis.read(data,0,BUFFER)) != -1){
          dest.write(data, 0, count);
       }
       dest.flush();
       dest.close();
    }
    zis.close();
  }
}
```

コード例 Unzip の問題点

(A) 解凍後のサイズをチェックしていない

(B) 例外 ArrayIndexOutOfBoundsException が発生する可能性がある

(C) 既存ファイルを上書きする可能性がある

(D) その他?

これらの問題点を解決せよ!!

Hands-on Exercise(2)

サンプルコード AltConst を
修正しよう

Q. 以下のコードの問題点は何か?

```
class AltConst {
  int i;
  AltConst(){ this(10); }
  AltConst(int i0){
    if (checkarg(i0)) {
      this.i = i0;
    }
  }
  boolean checkarg(int i) throws IllegalArgumentException {
    if (i<0 || 100<i) {
      throw new IllegalArgumentException("arg should be positive < 100.");
    }
    return true;
  }
}
```

(A) コンパイルエラーになる
(B) **checkarg()** による引数のチェックは役に立たない
(C) コンストラクタが二つ定義されている
(D) フィールド **i** に初期化子がない

A. 以下のコードの問題点は何か?

```
class AltConst {
  int i;
  AltConst(){ this(10); }
  AltConst(int i0){
    if (checkarg(i0)) {
      this.i = i0;
    }
  }
  boolean checkarg(int i) throws IllegalArgumentException {
    if (i<0 || 100<i) {
      throw new IllegalArgumentException("arg should be positive < 100.");
    }
    return true;
  }
}
```

(A) コンパイルエラーになる

(B) **checkarg()** による引数のチェックは役に立たない

(C) コンストラクタが二つ定義されている

(D) フィールド **i** に初期化子がない

A. 以下のコードの問題点は何か?

AltConst のサブクラスをつくることにより, checkarg() によるチェックの後でフィールド i の値を変更できる.

```
class attack extends AltConst {
  attack(int arg) {
    super();
    this.i = arg;
  }
  public static void main(String[] args) {
    AltConst a = new attack(101);
    System.out.println("attack.i: " + a.i);
  }
}
```

A. 以下のコードの問題点は何か?

AltConst のサブクラスをつくることにより, checkarg() によるチェックの後でフィールド i の値を変更できる.

```
class attack extends AltConst {
  attack(int arg) {
    super();
    this.i = arg;
  }
  public static void main(String[] args) {
    AltConst a = new attack(101);
    System.out.println("attack.i: " + a.i);
  }
}
```

サブクラス化による攻撃への対策を行え!!