

# Java and the Crypto API

Computer Security Lecture 4

Tom Chothia

# Today's Lecture

- Object Oriented programming
- Java
- Java Crypto API

# How do we define the characteristics of an object?

In your program, you define “**Classes**”.

You use “**Classes**” to create “**Objects**”.

You retain references to “**Objects**”

You call “**Methods**” to perform computation.

You use “**Methods**” to manipulate data held in “**Fields**” within an Object.

# See handout

Hello world in Eclipse

Counter

N.B. one class per file.

# Public/Private

The key word `public` makes a field or method accessible outside the class.

`private` stops all direct access.

It's good practice to use `private` for parts of the class you might want to change.

# Constructors

Constructors are used to make a new object.

Declare a constructor by:

```
public <className> (arguments) {  
    ...  
}
```

# Objects and Methods

Always think about running methods on objects.

- e.g. given Number objects “x” and “y”  
`x.add(y)` is more OO than `x = add(x, y)`

Why write methods against objects?

How do you decide which class should contain a method?

# Static Methods.

But not all methods in Java are called on objects,... what's going on here?

Sometimes methods are required that don't run against a specific object.

- Initial program method ("main")
- Factory methods
- Methods that are not object specific

Any methods or fields that are not related to a specific object are declared as "static".

# Static Field

A static field is one that is the same for all objects. E.g. static pi.

A static method is one that is the same for all objects.

Static methods can't refer to none static fields.  
Why?

Static methods don't need to be called on a particular objects. (Classname.method())

# Jar Files

- If you want to bundle classes together, use a Java Archive (JAR) file.
- A JAR file is a zipped directory of java class files.
- To run a jar file type: `java -jar filename`
- To create a jar file use: `jar cf jar-file input-files`
- Or in Eclipse: File>Export>Java>Jar

# APIs

- The best aspect of Java is all its libraries, or “Application Programming Interfaces (APIs)”
- For a list standard packages see:  
<http://java.sun.com/javase/6/docs/api/>
- Third party APIs exist for almost any functions.

# APIs are really important

- APIs are Application Programming Interfaces.
- They are libraries of useful programs that do most of the work for us.
- A lot of programming Java is using the right API.

# Useful APIs for Crypto

`javax.crypto.Cipher:`

- the Cipher object does the encryption.

`java.security.Key`

- a cryptographic key

`java.security.KeyFactory`

- Turn bytes into Key Objects.

Also `RSAPublicKey`, `X509EncodedKeySpec`,...

(remember cmd-shift-O in Eclipse).

# java.security.KeyGenerator

Create the object with:

```
KeyGenerator kg =  
    KeyGenerator.getInstance(<Type>);
```

Give the key length (if needed):

```
kg.initialize(1024);
```

Read out the key:

```
Key key = kg.genKeyPair();
```

# java.security.KeyPairGenerator

Create the object with:

```
KeyPairGenerator kg =  
    KeyPairGenerator.getInstance(<Crypto Type>);
```

Key the key length: `kg.initialize(1024);`

Read out the keys:

```
KeyPair keypair = kg.genKeyPair();  
PrivateKey privateKey = keypair.getPrivate();  
PublicKey publicKey = keypair.getPublic();
```

# Encryption In Java

Steps to encrypt data in Java (see example code):

- Import package
- Create a cipher object
- Initiate the cipher object with the scheme you want in encrypt or decrypt mode.
- Pass the object the data you want to encrypt.
- Read the cipher text out.
- Decrypt in the same way.

# Saving a Key

- We can read and write the bytes of a key to a file.
- This is a bad idea.
- We want to
  - protect read access to private keys,
  - and make sure the public ones are real.

# Java keytool

- Most Java programs use existing keys rather than create keys themselves.
- The keytool command can be used to generate keys outside Java.

# The KeyStore Class

- A KeyStore holds password protected private keys and public keys as certificates.

- Make keystores using the keytool e.g.

```
keytool -genkey -keyalg RSA
```

```
-keypass password -alias mykey
```

```
-storepass storepass
```

```
-keystore myKeyStore
```

# Next Lecture:

## Access Control:

Different models of access control.

Access control in Linux.

The Confused Deputy problem.