

Java

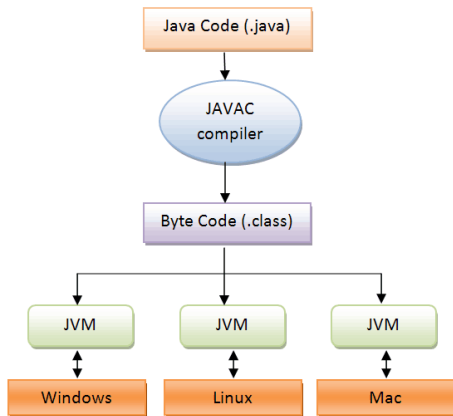
Massimo Dong

November 27, 2017

Outline

- 1 The Java Language
 - Java Virtual Machine(JVM)
- 2 Java Memory Management
 - References & Garbage Collection
 - Array Check
 - *Heartbleed
- 3 Compile & Run Time Check
 - Exceptions
 - Type Safety
 - Modifiers
 - Undefined Behaviors
- 4 Java Rich Internet Applications
 - Web Start Applications
 - Sandbox
 - Security Manager
 - Signing

Java Virtual Machine(JVM)



Outline

- 1 The Java Language
 - Java Virtual Machine(JVM)
- 2 Java Memory Management
 - References & Garbage Collection
 - Array Check
 - *Heartbleed
- 3 Compile & Run Time Check
 - Exceptions
 - Type Safety
 - Modifiers
 - Undefined Behaviors
- 4 Java Rich Internet Applications
 - Web Start Applications
 - Sandbox
 - Security Manager
 - Signing

References vs Pointers

Java

```
ProblemSolving ref = new ProblemSolving();  
...
```

C++

```
ProblemSolving *ptr = new ProblemSolving;  
...  
delete ptr;
```

Java

```
static void foo(ProblemSolving ref){
    ref.UD = 2;
}

public static void main(String[] argv){
    ProblemSolving ref = new ProblemSolving();
    foo(ref);
    System.out.println(ref.UD);
}
```

Java

```
static void foo(ProblemSolving ref){
    ref.UD = 2;
}

public static void main(String[] argv){
    ProblemSolving ref = new ProblemSolving();
    foo(ref);
    System.out.println(ref.UD);
}
```

2

Java Arrays

An object is a class instance or an array.

Java Arrays

An object is a class instance or an array.

Java

```
public class ArrayDemo{
    public static void main(String[] argv){
        int[] A = new int[10];
        System.out.println(A instanceof Object);
    }
}
```

Java Arrays

An object is a class instance or an array.

Java

```
public class ArrayDemo{
    public static void main(String[] argv){
        int[] A = new int[10];
        System.out.println(A instanceof Object);
    }
}
```

true

Array Check

Java

```
public class ArrayDemo{  
    public static void main(String[] argv){  
        int[] A = new int[10];  
        System.out.println(A[233]);  
    }  
}
```

Array Check

Java

```
public class ArrayDemo{  
    public static void main(String[] argv){  
        int[] A = new int[10];  
        System.out.println(A[233]);  
    }  
}
```

Exception in thread "main"

```
java.lang.ArrayIndexOutOfBoundsException: 233  
    at ArrayDemo.main(ArrayDemo.java:4)
```

Heartbleed

Heartbleed



Heartbleed

C/C++

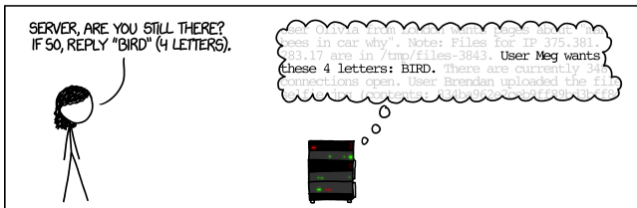
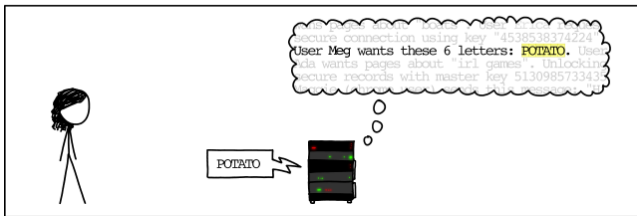
```
memcpy(bp, pl, payload);
```

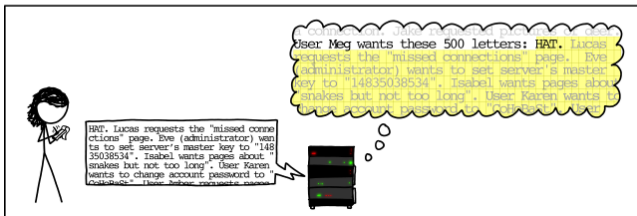
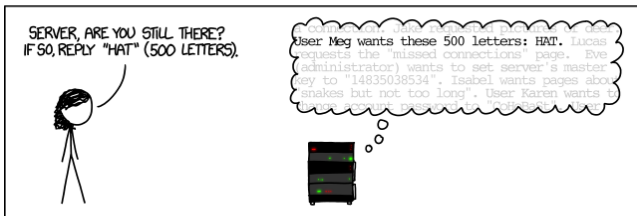
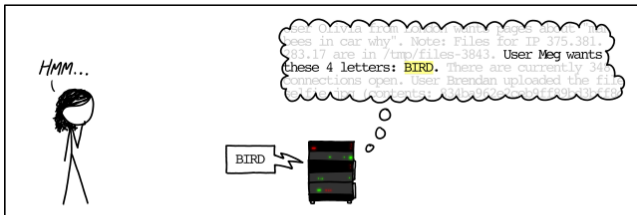

C/C++

```
memcpy(bp, pl, payload);
```

```
pl = "ProblemSolving"  
payload = 64 * 1024;
```

HOW THE HEARTBLEED BUG WORKS:





Outline

- 1 The Java Language
 - Java Virtual Machine(JVM)
- 2 Java Memory Management
 - References & Garbage Collection
 - Array Check
 - *Heartbleed
- 3 **Compile & Run Time Check**
 - **Exceptions**
 - **Type Safety**
 - **Modifiers**
 - **Undefined Behaviors**
- 4 Java Rich Internet Applications
 - Web Start Applications
 - Sandbox
 - Security Manager
 - Signing

Exceptions

Java

```
public static void main(String[] argv){
    FileInputStream in = new FileInputStream("data.in");
}
```

Exceptions

Java

```
public static void main(String[] argv){
    FileInputStream in = new FileInputStream("data.in");
}
```

```
error: unreported exception FileNotFoundException;
      must be caught or declared to be thrown
    FileInputStream in = new FileInputStream("data.in");
                        ^
```

Exceptions

Java

```
try{
    FileInputStream in = new FileInputStream("data.in");
}catch(FileNotFoundException E){
    System.out.println("File Not Found!");
}
```

Exceptions

Java

```
try{
    FileInputStream in = new FileInputStream("data.in");
}catch(FileNotFoundException E){
    System.out.println("File Not Found!");
}
```

Java

```
public static void main(String[] argv)
    throws FileNotFoundException{
    FileInputStream in = new FileInputStream("data.in");
}
```


C

```
int main(){  
    void (*foo)();  
    foo = 233;  
    foo();  
}
```

C

```
int main(){
    void (*foo)();
    foo = 233;
    foo();
}
```

warning:

assignment makes pointer from integer without a cast

```
foo = 233;
```

^

Type Safety

Java

```
public static void main(String[] argv){  
    A a;  
    B b;  
    a = b;  
}
```

C

```
int main(){  
    struct A *a;  
    struct B *b;  
    a = b;  
}
```

Access Modifiers

- private
- public
- protected

Non Access Modifiers

- static
- final
- abstract
- synchronized and volatile

Java

```
public static void main(String[] argv){  
    final int a;  
    a = 39;  
}
```

Java

```
public static void main(String[] argv){  
    final int a;  
    a = 39;  
}
```

```
public static void main(String[] argv){  
    final int a;  
    a = 39;  
    a = 40;  
}
```

Java

```
public static void main(String[] argv){  
    final int a;  
    a = 39;  
}
```

```
public static void main(String[] argv){  
    final int a;  
    a = 39;  
    a = 40;  
}
```

error: variable a might already have been assigned

Undefined Behaviors

Undefined Behaviors

C

```
i = i++ + 1;
```

Undefined Behaviors

C

```
i = i++ + 1;
```

Undefined Behavior

Undefined Behaviors

C

```
i = i++ + 1;
```

Undefined Behavior

Java

```
i = i++ + 1;
```

Undefined Behaviors

C

```
i = i++ + 1;
```

Undefined Behavior

Java

```
i = i++ + 1;
```

1

Outline

- 1 The Java Language
 - Java Virtual Machine(JVM)
- 2 Java Memory Management
 - References & Garbage Collection
 - Array Check
 - *Heartbleed
- 3 Compile & Run Time Check
 - Exceptions
 - Type Safety
 - Modifiers
 - Undefined Behaviors
- 4 Java Rich Internet Applications
 - Web Start Applications
 - Sandbox
 - Security Manager
 - Signing

Demo

Java

```
try{
    FileInputStream in = new FileInputStream("input.txt");
} catch (Exception E){
    System.out.println(E);
}
```


Java

```
try{
    FileInputStream in = new FileInputStream("input.txt");
}catch(Exception E){
    System.out.println(E);
}
```

```
java.security.AccessControlException:
access denied ("java.io.FilePermission" "input.txt" "read")
```

Java

```
SecurityManager security = System.getSecurityManager();
if(security != null){
    try{
        security.checkRead("input.txt");
    }catch(Exception E){
        System.out.println(E);
        return;
    }
}
```

Java

```
SecurityManager security = System.getSecurityManager();
if(security != null){
    try{
        security.checkRead("input.txt");
    }catch(Exception E){
        System.out.println(E);
        return;
    }
}
```

```
$javaws -nosecurity demo_no_codebase.jnlp
```


Demo

- <https://docs.oracle.com>
- <https://blogs.oracle.com>