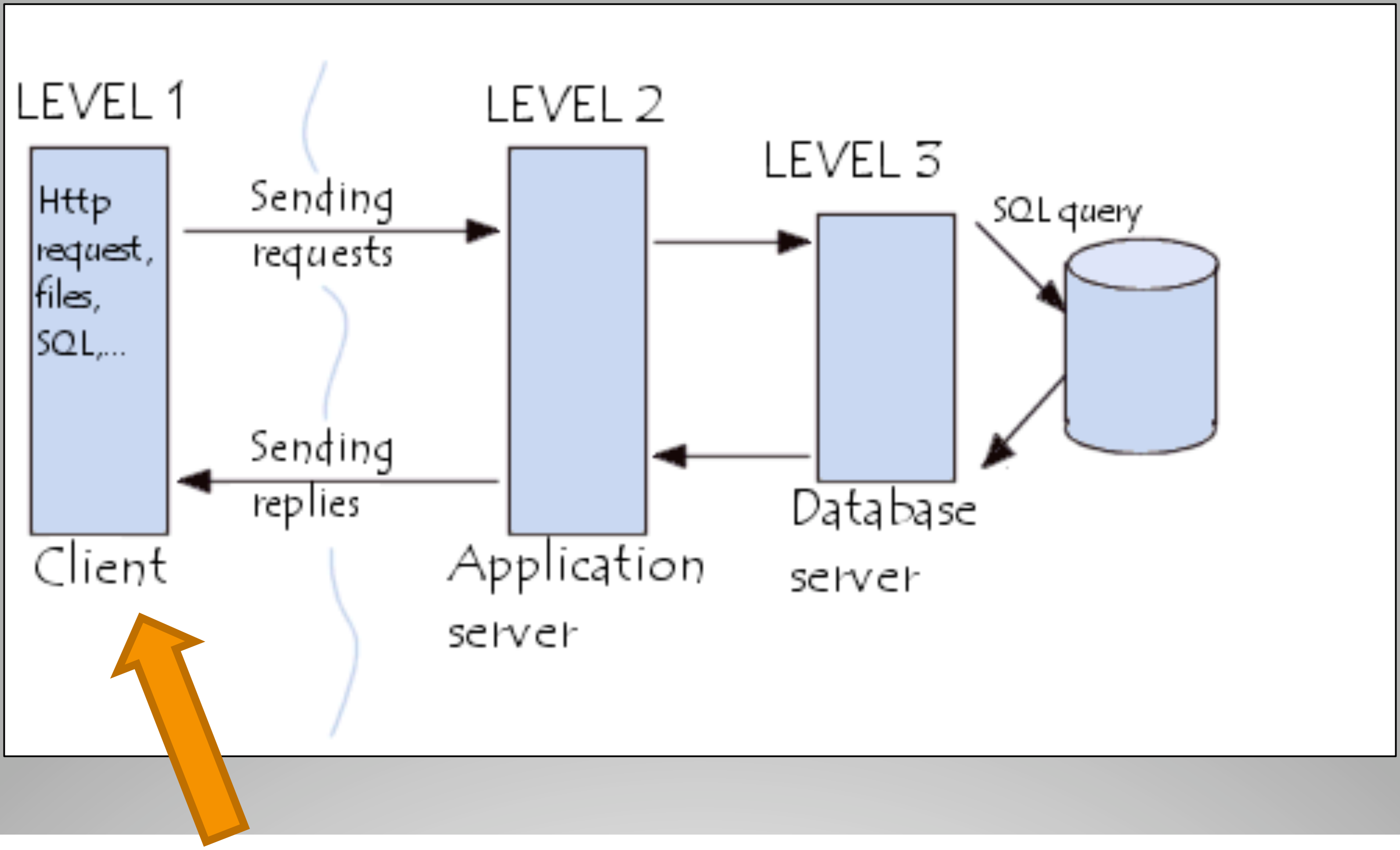
JavaSnoop

how to hack anything in java

arshan dabirsiaghi
director of research
aspect security
http://www.aspectsecurity.com/
http://i8jesus.com/
@nahsra



Any more detail is theoretically irrelevant. A client is a client.

Agenda

- ➡ Why hacking Java apps is practically difficult
- ● Showing how JavaSnoop solves the problem
- ● Demos, videos, details

The following talk occurs between Monday and Friday of any given week. Like, it could be next week or something.

Hey, Security Company X. I want you to test the security of this important applet. Can you do it in 40 hours?

- No problem we do it all the time!! What's an applet again?
- Absolutely. I can scan that with WebInspect, right?

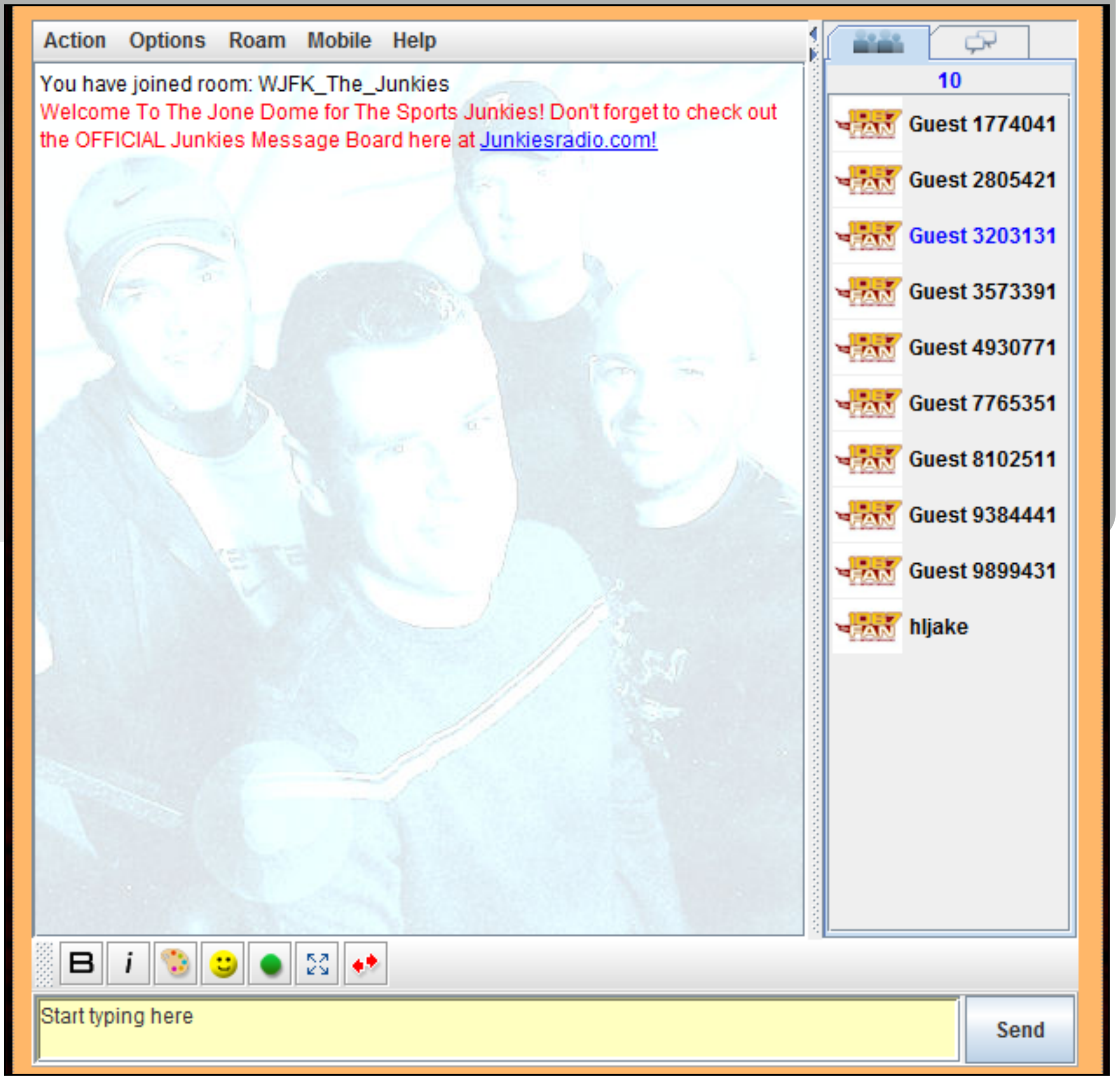
Zero intel on applet.

Looks to be some kind of chat thing.

Not sure about protocols,
exit points, data types.

After eating Panda Express and bitching about lack of useful docs, time left:

38 hours.



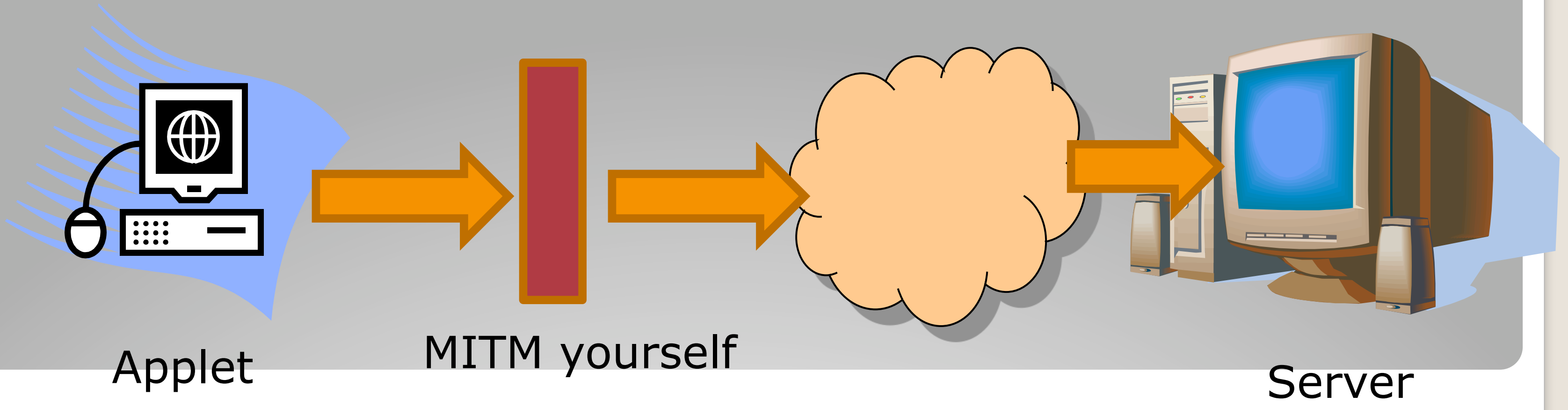
You have joined room: WJFK_The_Junkies
Welcome To The Jone Dome for The Sports Junkies! Don't forget to check out the OFFICIAL Junkies Message Board here at Junkiesradio.com!

Guest 1774041
Guest 2805421
Guest 3203131
Guest 3573391
Guest 4930771
Guest 7765351
Guest 8102511
Guest 9384441
Guest 9899431
hljake

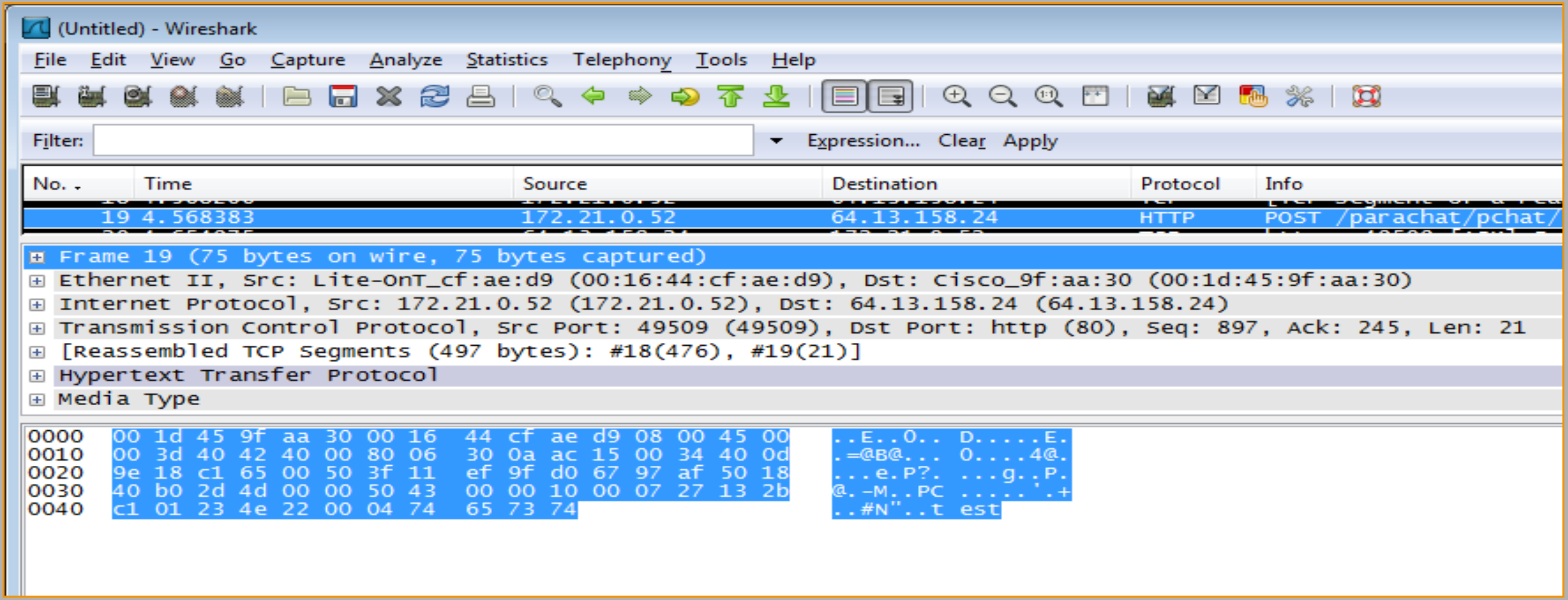
Start typing here

Option #1 (hack the traffic)

1. Pray it uses HTTP
2. Pray it has configurable proxy settings
3. Pray it doesn’t use serialized objects/ layer 7.5 encryption/custom protocols



- I setup Wireshark to look at the data.


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply

```
No. .  Time          Source           Destination       Protocol  Info
   19 4.568383       172.21.0.52      64.13.158.24      HTTP      POST /parachat/pchat/
```

```
⊞ Frame 19 (75 bytes on wire, 75 bytes captured)
⊞ Ethernet II, Src: Lite-OnT_cf:ae:d9 (00:16:44:cf:ae:d9), Dst: Cisco_9f:aa:30 (00:1d:45:9f:aa:30)
⊞ Internet Protocol, Src: 172.21.0.52 (172.21.0.52), Dst: 64.13.158.24 (64.13.158.24)
⊞ Transmission Control Protocol, Src Port: 49509 (49509), Dst Port: http (80), Seq: 897, Ack: 245, Len: 21
⊞ [Reassembled TCP Segments (497 bytes): #18(476), #19(21)]
⊞ Hypertext Transfer Protocol
⊞ Media Type
```

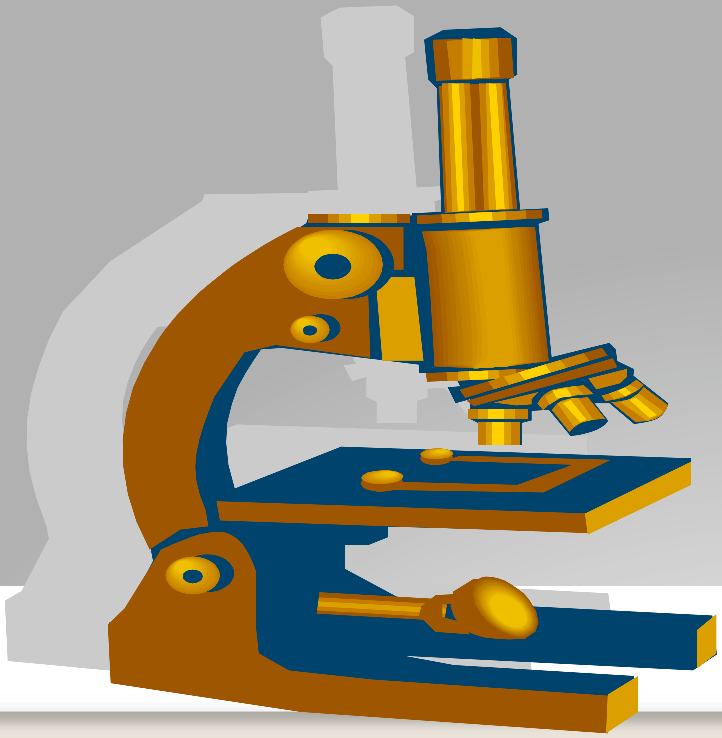
```
0000   00 1d 45 9f aa 30 00 16  44 cf ae d9 08 00 45 00    ..E..0..  D.....E.
0010   00 3d 40 42 40 00 80 06  30 0a ac 15 00 34 40 0d    .=@B@...  0....4@.
0020   9e 18 c1 65 00 50 3f 11  ef 9f d0 67 97 af 50 18    ...e.P?.  ...g..P.
0030   40 b0 2d 4d 00 00 50 43  00 00 10 00 07 27 13 2b    @.-M..PC  .....'.+
0040   c1 01 23 4e 22 00 04 74  65 73 74                   ..#N"..t  est
```

- Crap, it's not HTTP. It's some kind of bizarro protocol. That rules out Ethereal/Middler too.

What am I even looking at?
Never mind, this clearly didn't work.
Time left: 35 hours.

Option #2 (hack the client)

1. Grab classes/jars
2. Decompile them
3. Perform source code review



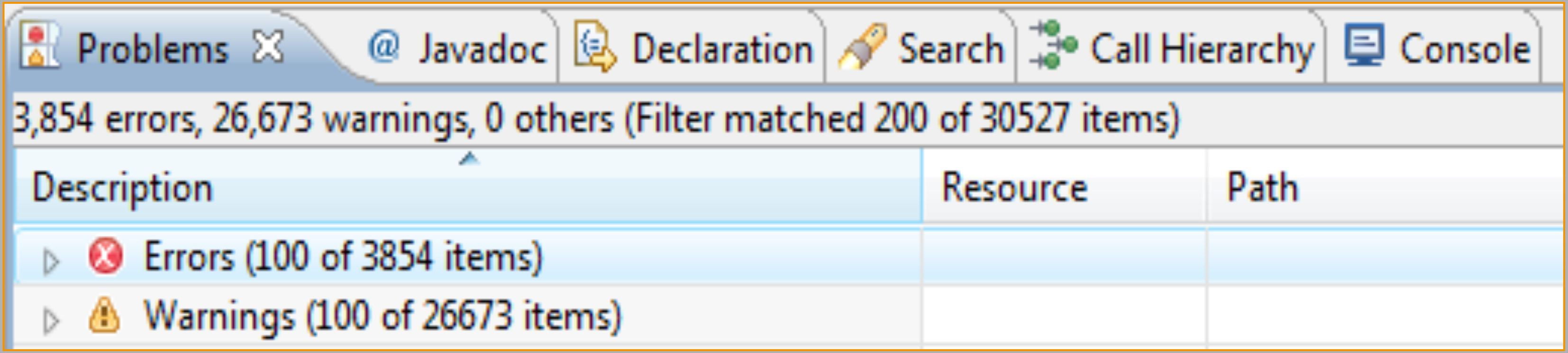
Theoretical next steps:

4. Alter code
5. Recompile evil client
6. Send custom attacks

Real next steps:

4. Alter code
5. Nothing compiles/works
6. Tests never happen or are invalid

1. I download the applet codebase.
2. I decompile the codebase.
3. I load the decompiled code into Eclipse.



3,854 errors, 26,673 warnings, 0 others (Filter matched 200 of 30527 items)

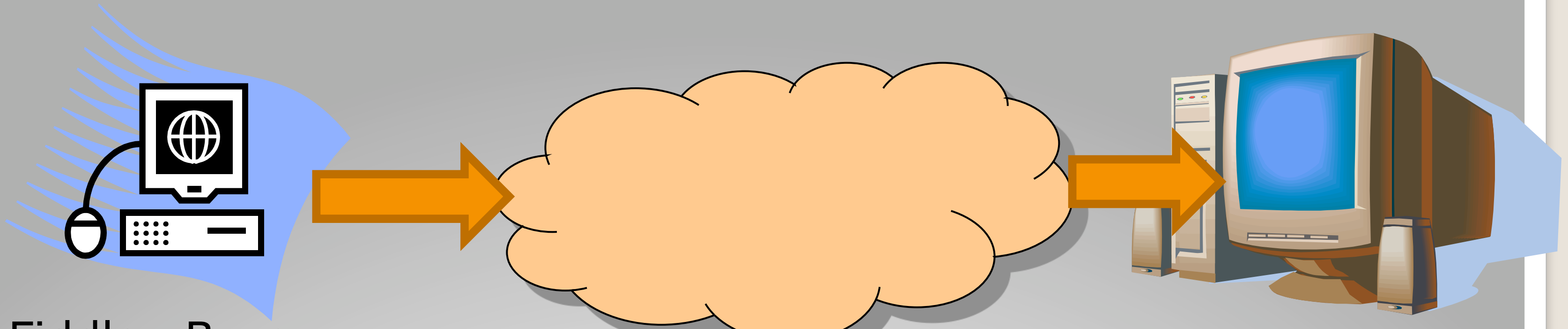
Description	Resource	Path
Errors (100 of 3854 items)		
Warnings (100 of 26673 items)		

- Are you serious? 3800+ errors? Is every single line of code broken?

I don't have that kind of time.
Time left: 31 hours.

Option #3 (hack the server)

1. Pray the endpoints are HTTP
2. Pray it doesn't require client certificates
3. Pray it doesn't use serialized objects/ layer 7.5 encryption/custom protocols



- Tried to talk to the server.
- Not sure about this traffic - some new raw-byte protocol?
- F*#&ing stupid Java s*%#, mother*@#& bananas.
- Entering Mel Gibson rage.

I need some "me" time.
Time left: 27 hours.

We need some inspiration. Anna?

If only there was a “WebScarab” or “Burp”, but for the Java Virtual Machine.

If there was, I could tamper with method parameters like HTTP traffic. That certainly would have made Scary Movie 3 easier to make.

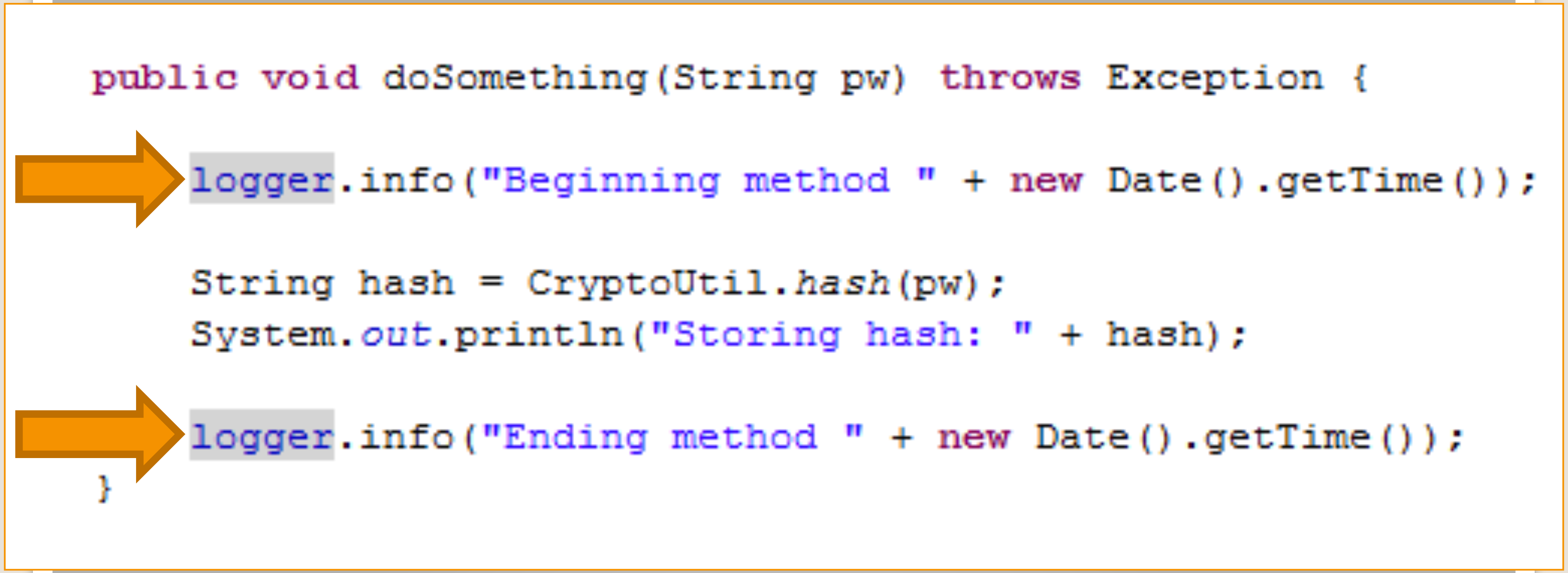
Also, I love you Arshan.

-- Anna Faris

That sounds like something we could do with instrumentation.

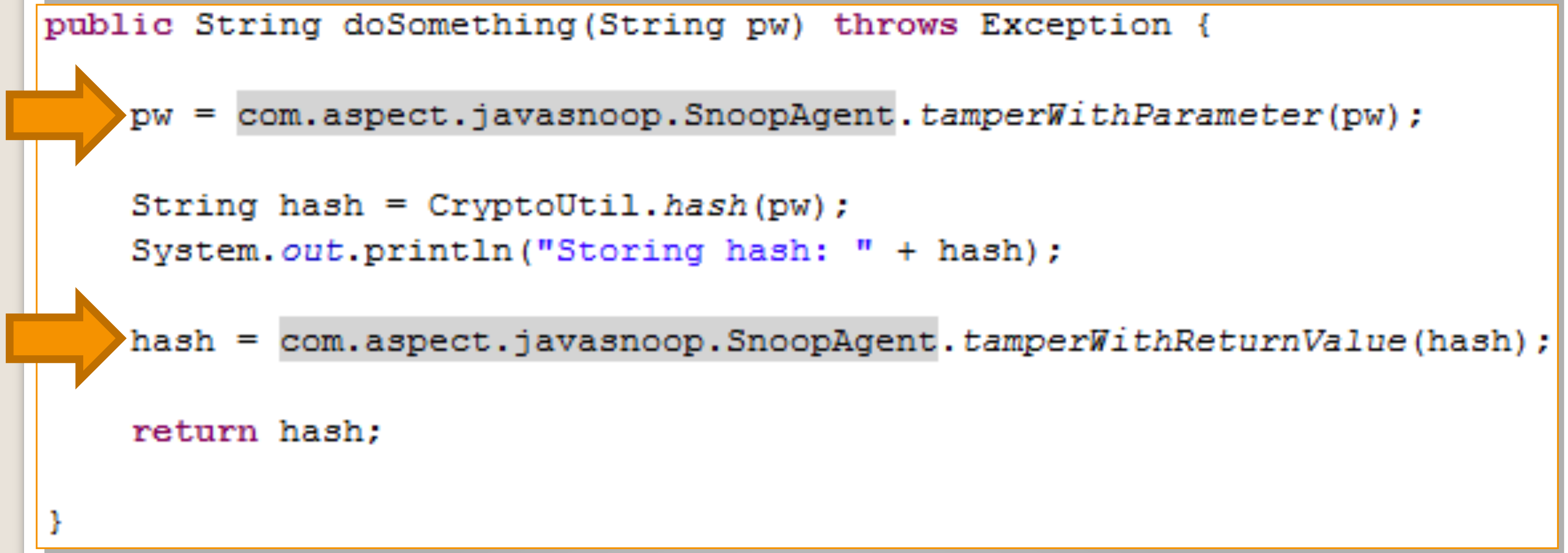
we miss you pdp, come back

What is instrumentation?



```
public void doSomething(String pw) throws Exception {

    logger.info("Beginning method " + new Date().getTime());

    String hash = CryptoUtil.hash(pw);
    System.out.println("Storing hash: " + hash);

    logger.info("Ending method " + new Date().getTime());
}
```

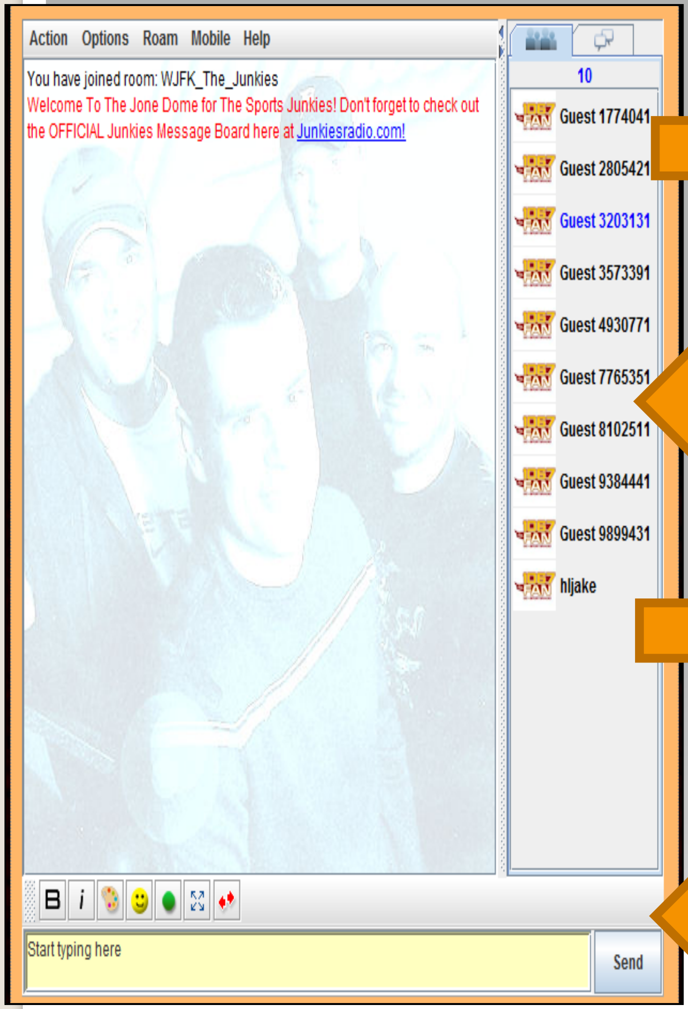
How would instrumentation help?



```
public String doSomething(String pw) throws Exception {

    pw = com.aspect.javasnoop.SnoopAgent.tamperWithParameter(pw);

    String hash = CryptoUtil.hash(pw);
    System.out.println("Storing hash: " + hash);

    hash = com.aspect.javasnoop.SnoopAgent.tamperWithReturnValue(hash);

    return hash;

}
```

Target application

Our evil hacking program (JavaSnoop)



You have joined room: WJFK_The_Junkies
Welcome To The Jone Dome for The Sports Junkies! Don't forget to check out the OFFICIAL Junkies Message Board here at Junkiesradio.com!

Guest 1774041
Guest 2805421
Guest 3203131
Guest 3573391
Guest 4930771
Guest 7765351
Guest 8102511
Guest 9384441
Guest 9899431
hljake

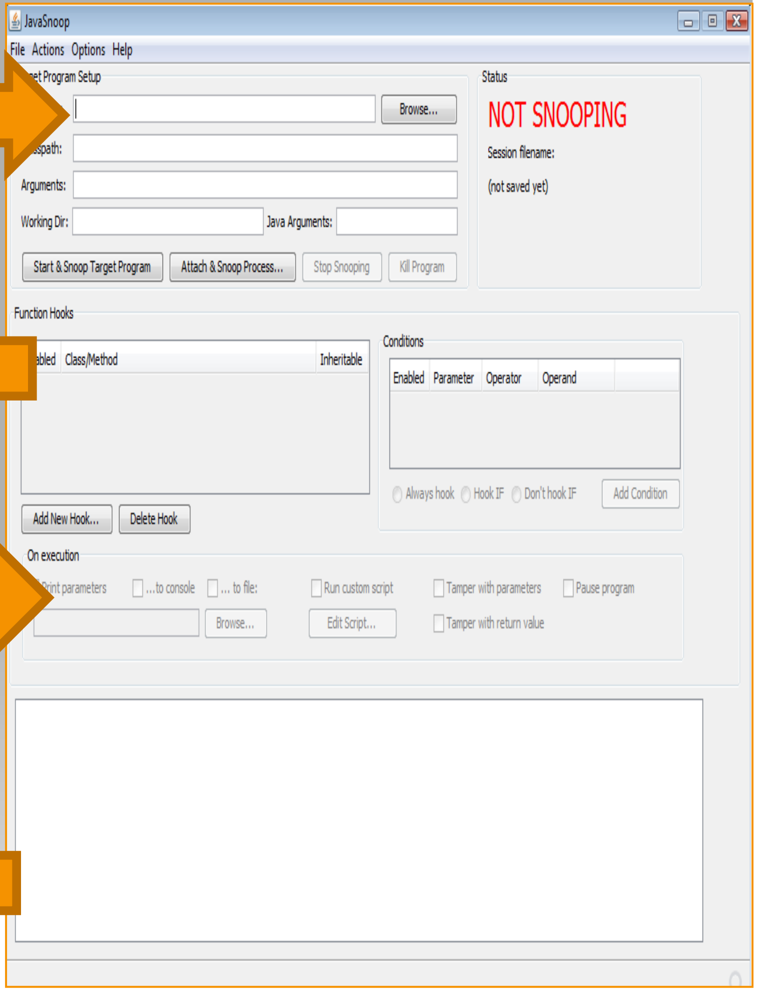
Start typing here

Method parameters

Tampered method parameters

Return value

Tampered return value



Target Program Setup
Classpath:
Arguments:
Working Dir: Java Arguments:

Status
NOT SNOOPING
Session filename:
(not saved yet)

Function Hooks

Enabled	Class/Method	Inheritable

Conditions

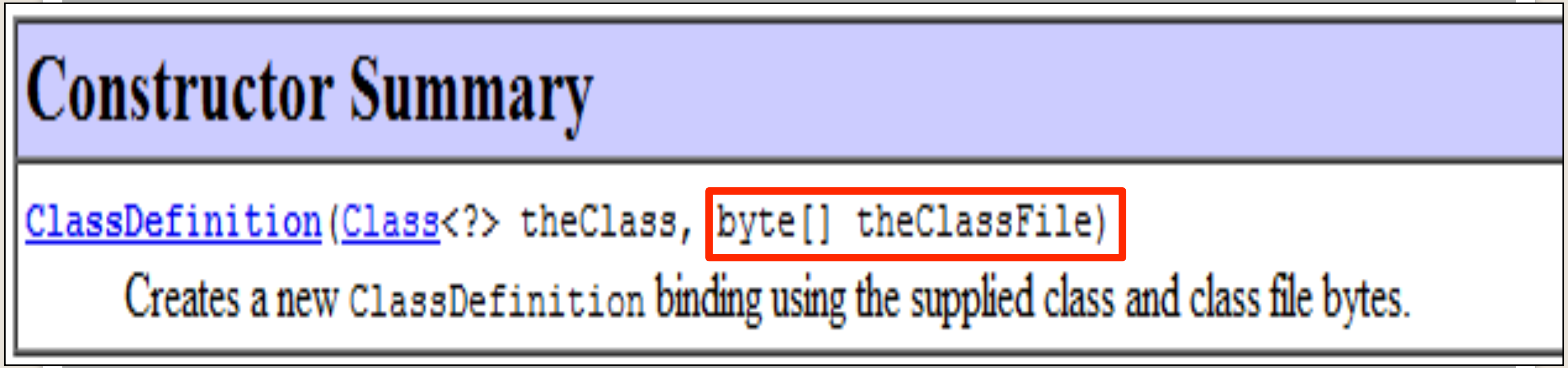
Enabled	Parameter	Operator	Operand

Always hook Hook IF Don't hook IF

On execution
Print parameters ...to console ... to file:
Run custom script
Tamper with parameters
Pause program
Tamper with return value

- Have to read up on instrumentation.
- Time left: 20 hours.
- Am I really good at my job? Maybe I should have stayed in development/ snarky Slashdot commenting.

Number of flaws found: zero.



Constructor Summary

ClassDefinition(Class<?> theClass, byte[] theClassFile)

Creates a new ClassDefinition binding using the supplied class and class file bytes.

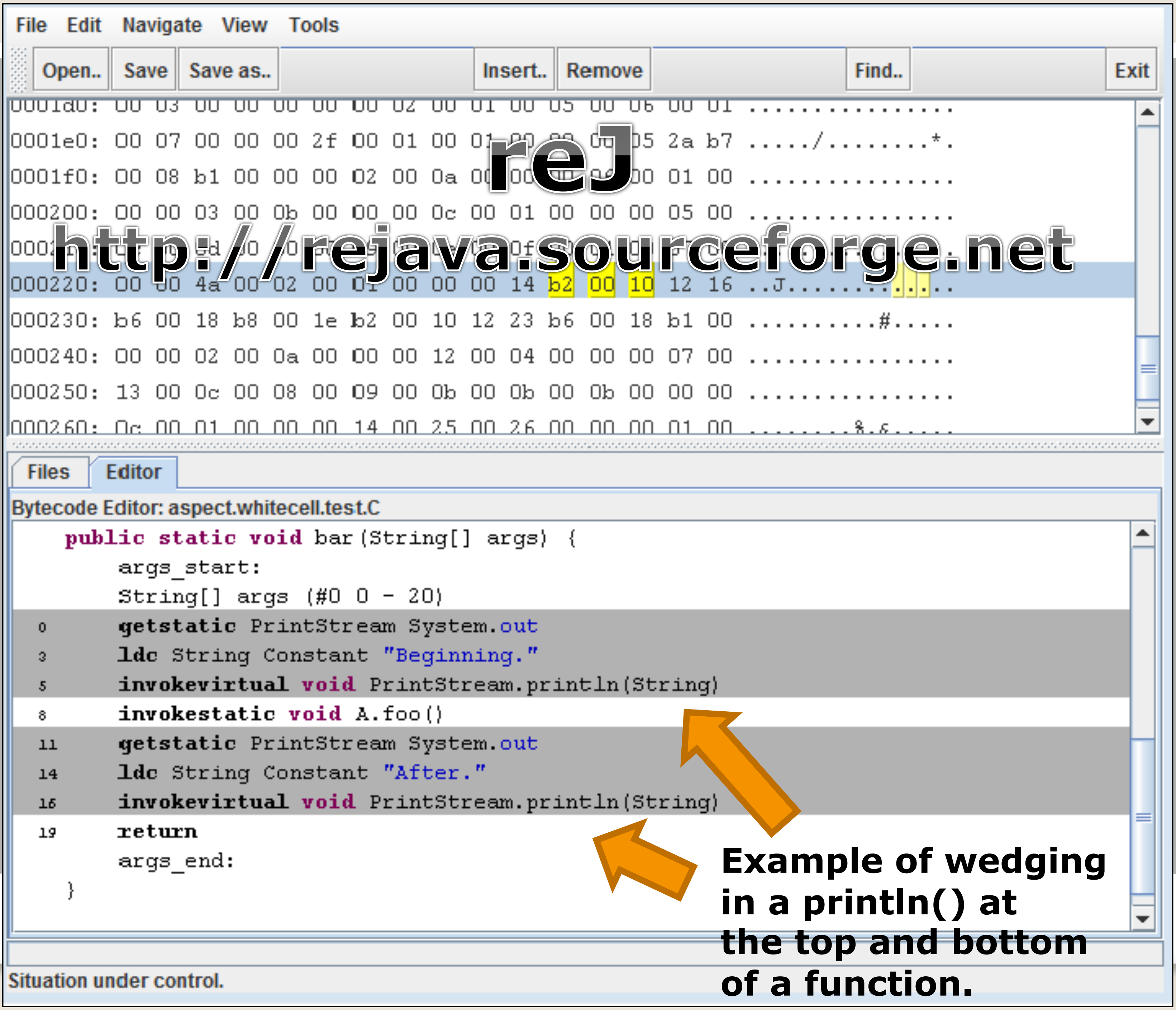
To redefine a class we need the actual raw bytecode. I tried putting in:

`alert(document.cookie)`

…but it didn't work.

reJ

http://rejava.sourceforge.net


File Edit Navigate View Tools
Open.. Save Save as.. Insert.. Remove Find.. Exit
Files Editor

```
0001d0: 00 03 00 00 00 00 00 02 00 01 00 05 00 06 00 01 ................
0001e0: 00 07 00 00 00 2f 00 01 00 01 [illegible] 05 2a b7 ...../........*.
0001f0: 00 08 b1 00 00 00 02 00 0a 00 [illegible] 00 01 00 ................
000200: 00 00 03 00 0b 00 00 00 0c 00 01 00 00 00 05 00 ................
000210: [illegible]
000220: 00 00 4a 00 02 00 01 00 00 00 14 b2 00 10 12 16 ..J.............
000230: b6 00 18 b8 00 1e b2 00 10 12 23 b6 00 18 b1 00 ..........#.....
000240: 00 00 02 00 0a 00 00 00 12 00 04 00 00 00 07 00 ................
000250: 13 00 0c 00 08 00 09 00 0b 00 0b 00 0b 00 00 00 ................
000260: 0c 00 01 00 00 00 14 00 25 00 26 00 00 00 01 00 ........%.&.....
```

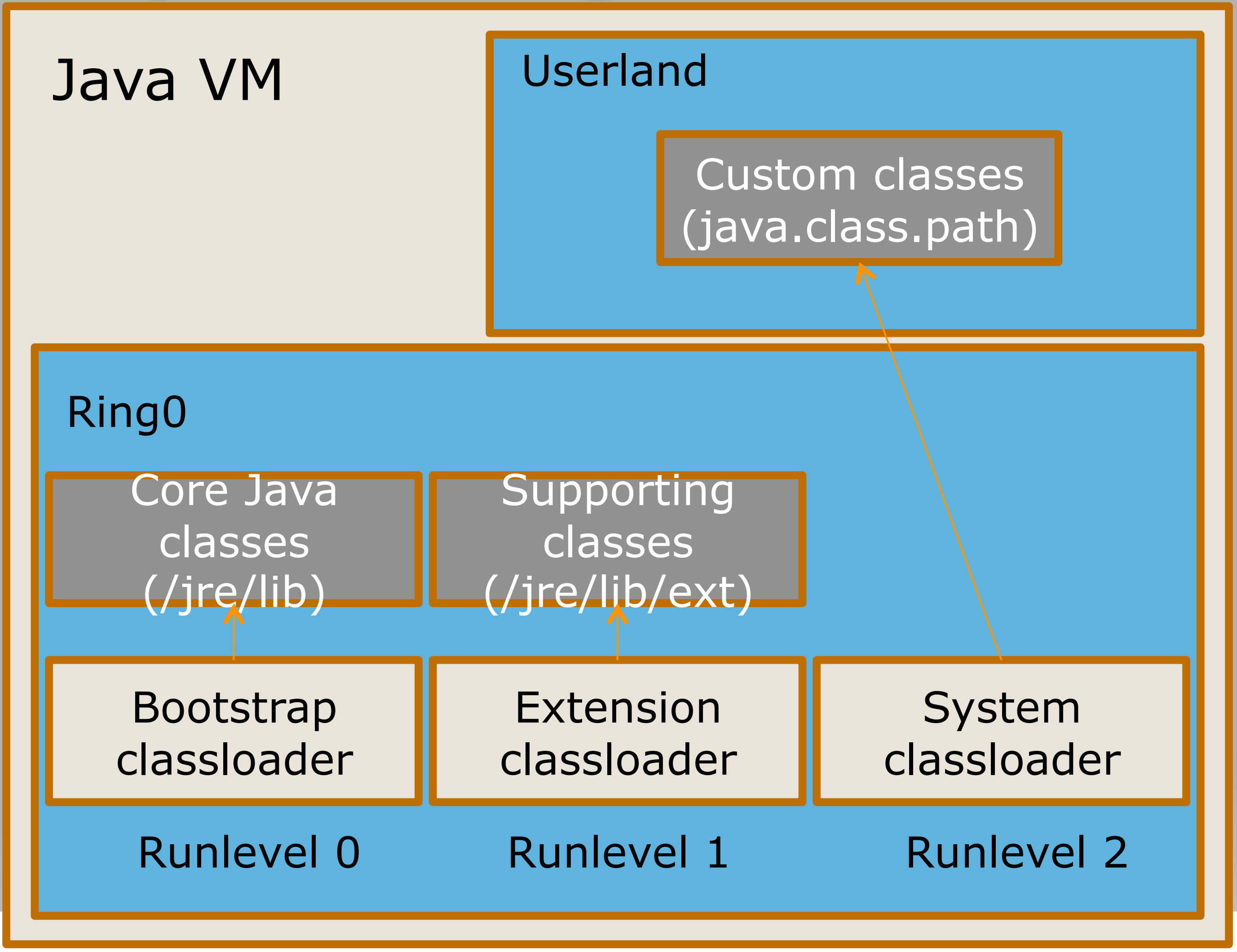
Bytecode Editor: aspect.whitecell.test.C

```
    public static void bar(String[] args) {
        args_start:
        String[] args (#0 0 - 20)
0       getstatic PrintStream System.out
3       ldc String Constant "Beginning."
5       invokevirtual void PrintStream.println(String)
8       invokestatic void A.foo()
11      getstatic PrintStream System.out
14      ldc String Constant "After."
16      invokevirtual void PrintStream.println(String)
19      return
        args_end:
    }
```

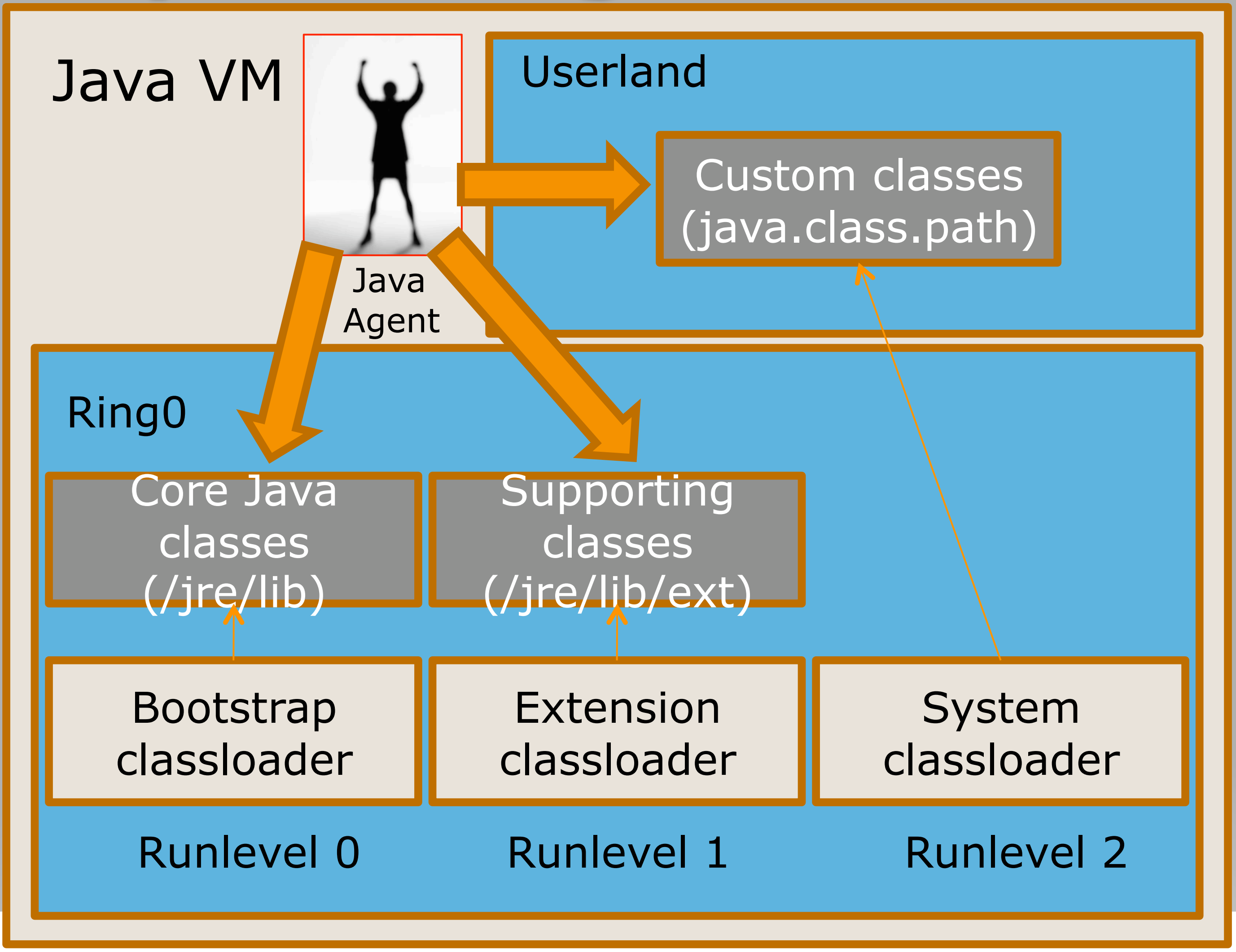
Situation under control.

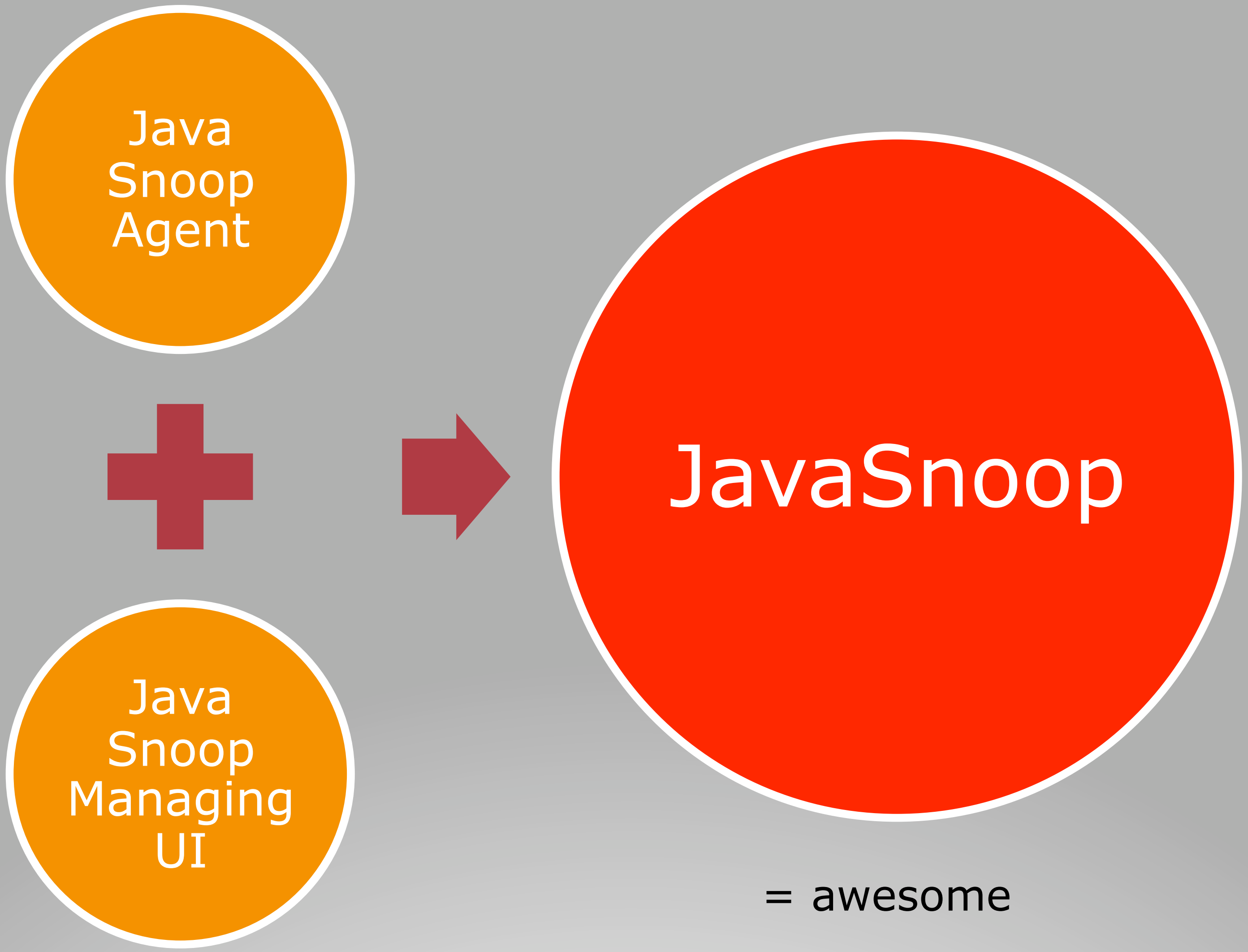
Example of wedging in a println() at the top and bottom of a function.

Dailydaver's guide to Java VM



Dailydaver's guide to Java VM





= awesome

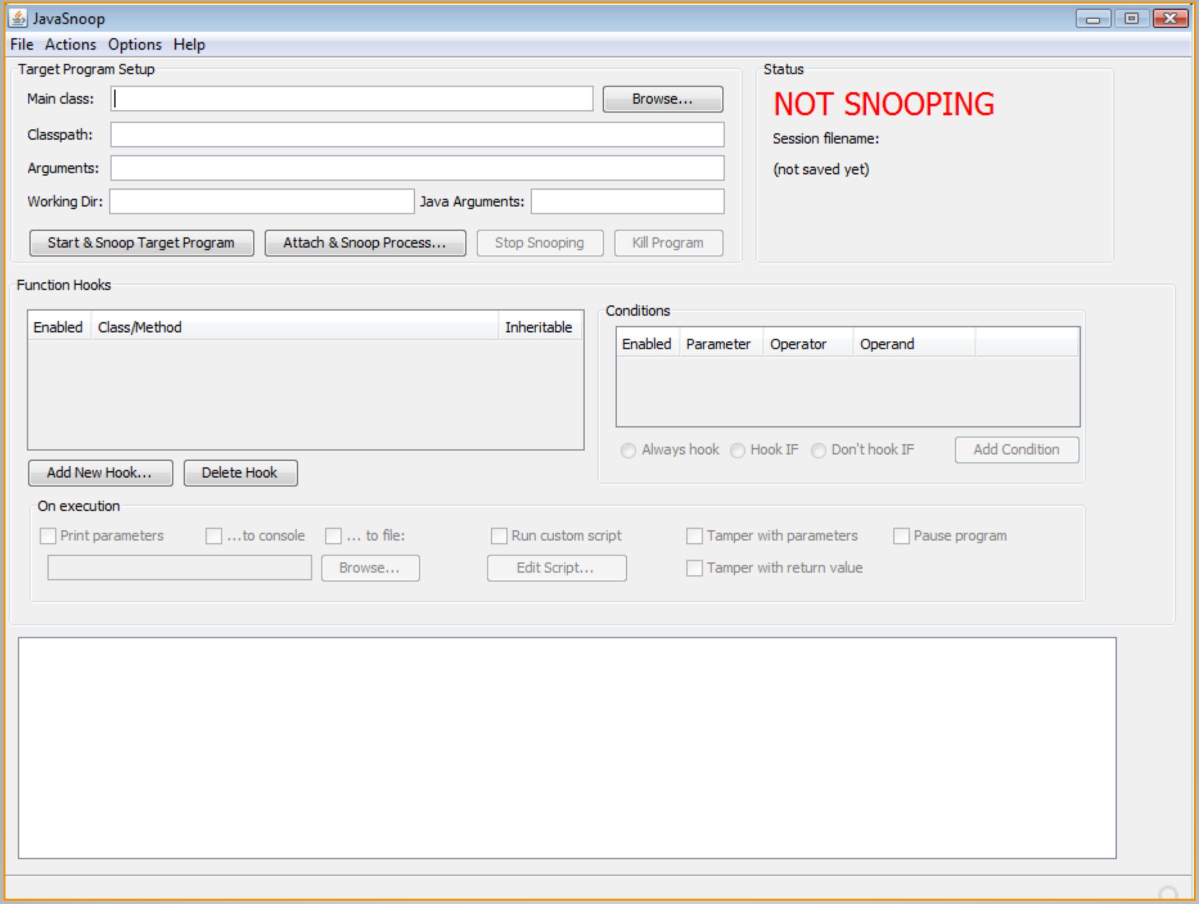
Time left: 12 hours. It's Thursday.

THERE'S NOT ENOUGH TIME

Agenda

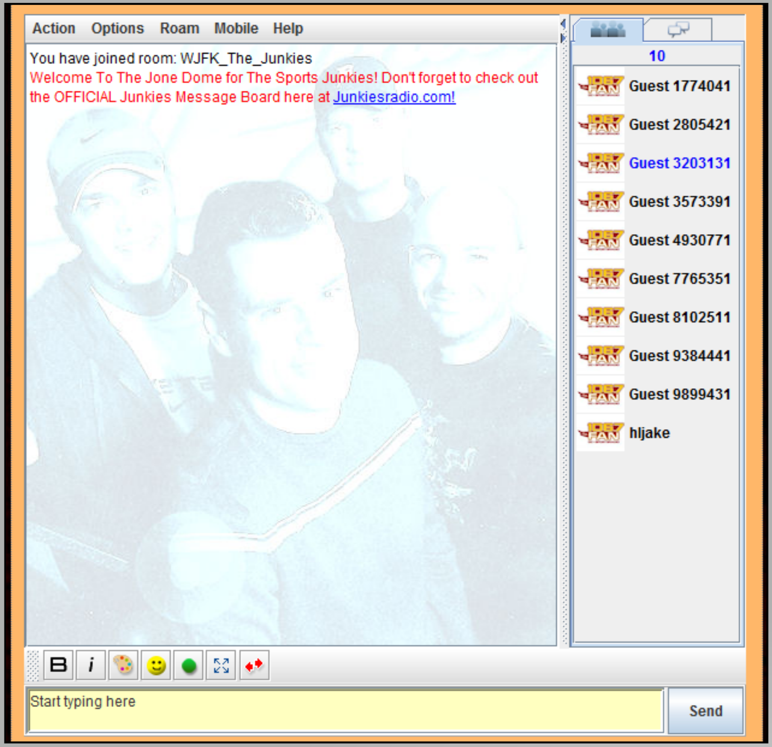
- Why hacking Java apps is practically difficult
- Showing how JavaSnoop solves the problem
- Demos, videos, details

Step #1: Startup JavaSnoop



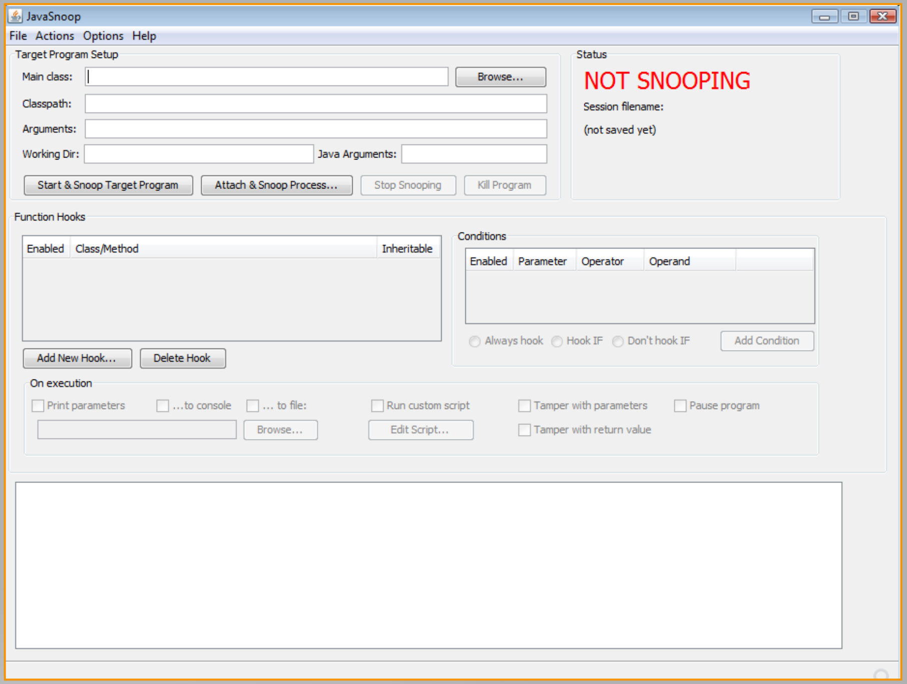
Okay, I can do that.

Step #2: Startup target



You have joined room: WJFK_The_Junkies
Welcome To The Jone Dome for The Sports Junkies! Don't forget to check out the OFFICIAL Junkies Message Board here at Junkiesradio.com!

Guest 1774041
Guest 2805421
Guest 3203131
Guest 3573391
Guest 4930771
Guest 7765351
Guest 8102511
Guest 9384441
Guest 9899431
hljake

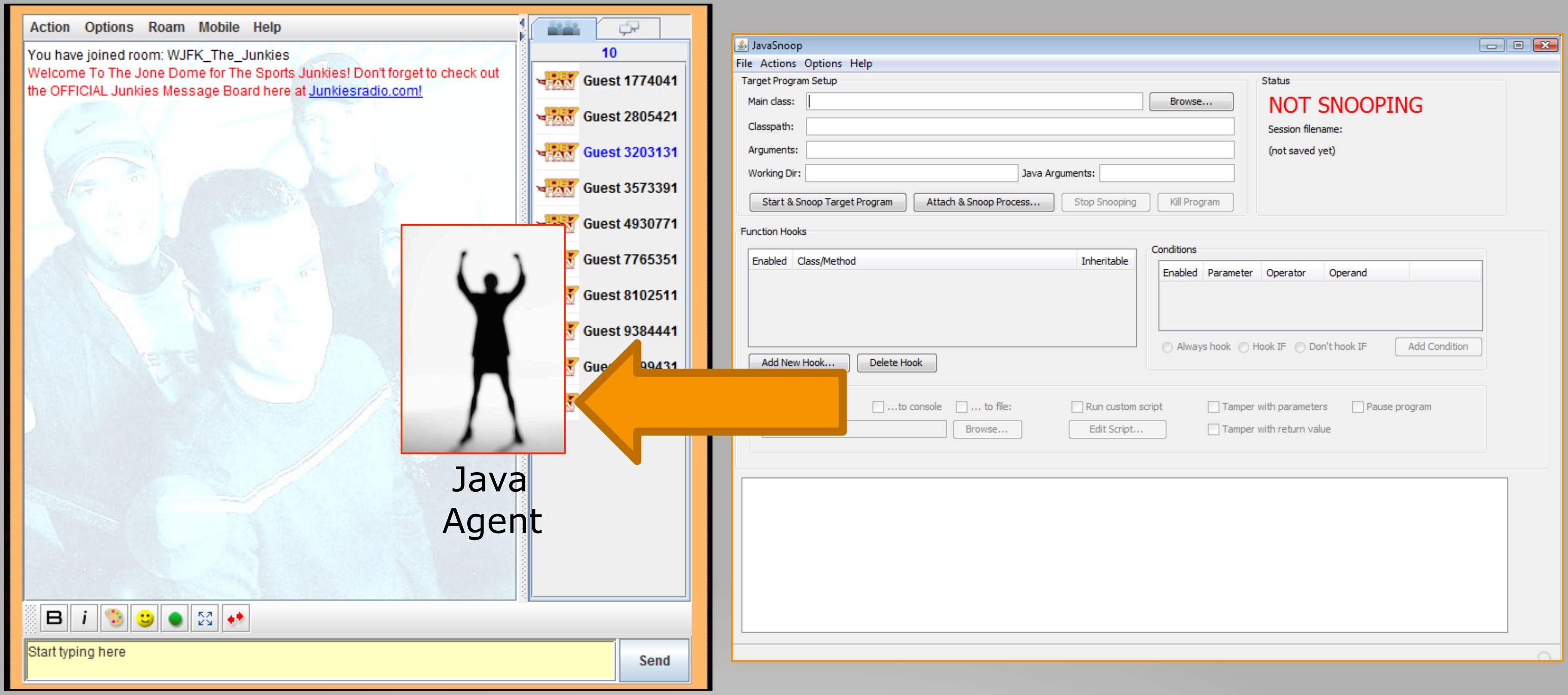


NOT SNOOPING

(not saved yet)

Okay, that's easy too. Can I call myself a hacker now?

Step #3: Attach evil agent to target VM



You have joined room: WJFK_The_Junkies
Welcome To The Jone Dome for The Sports Junkies! Don't forget to check out the OFFICIAL Junkies Message Board here at Junkiesradio.com!

Guest 1774041
Guest 2805421
Guest 3203131
Guest 3573391
Guest 4930771
Guest 7765351
Guest 8102511
Guest 9384441

Start typing here

Target Program Setup
Main class:
Classpath:
Arguments:
Working Dir:
Java Arguments:

Status
NOT SNOOPING
Session filename:
(not saved yet)

Function Hooks

Enabled	Class/Method	Inheritable

Conditions

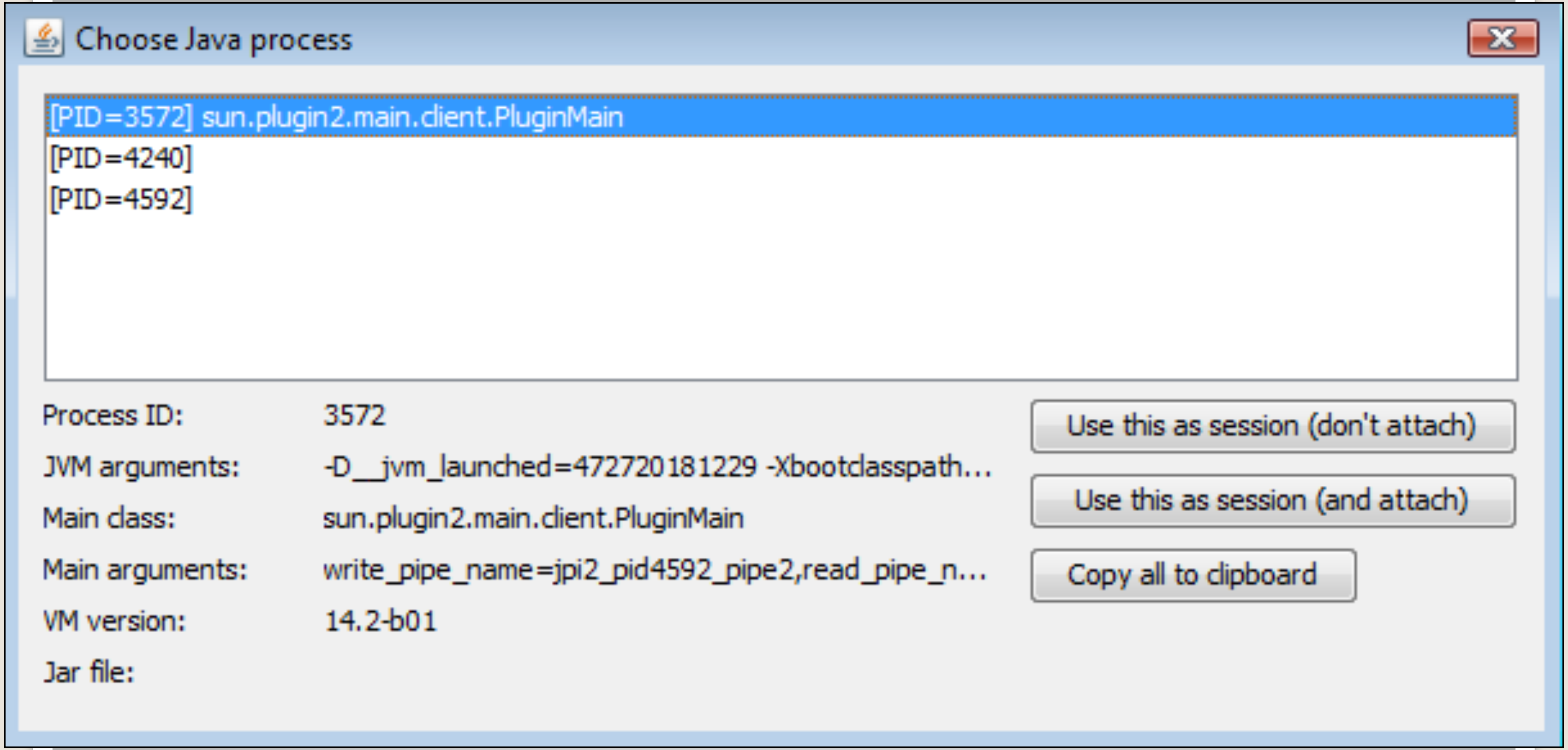
Enabled	Parameter	Operator	Operand

○ Always hook ○ Hook IF ○ Don't hook IF

☐ ...to console ☐ ... to file: ☐ Run custom script ☐ Tamper with parameters ☐ Pause program
☐ Tamper with return value

Hurry up, only 8 hours left.

Aside: how do I know which Java process to target?

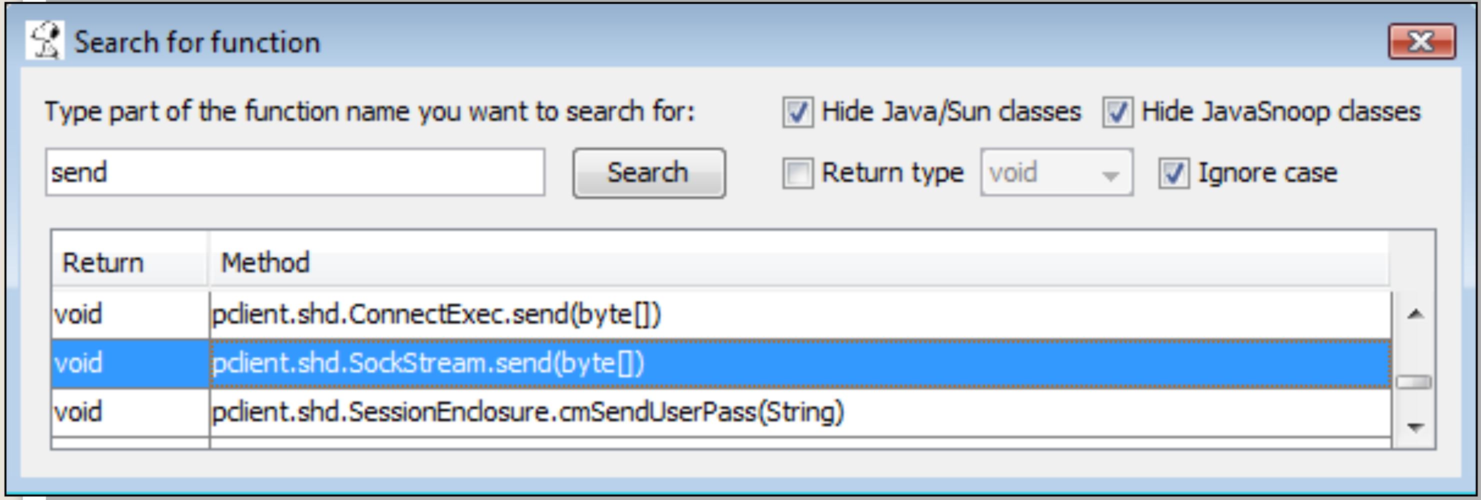


```
[PID=3572] sun.plugin2.main.client.PluginMain
[PID=4240]
[PID=4592]
```

Process ID:	3572
JVM arguments:	-D__jvm_launched=472720181229 -Xbootclasspath...
Main class:	sun.plugin2.main.client.PluginMain
Main arguments:	write_pipe_name=jpi2_pid4592_pipe2,read_pipe_n...
VM version:	14.2-b01
Jar file:	

Use this as session (don't attach)
Use this as session (and attach)
Copy all to clipboard

Step #4: pick a method to hack and how



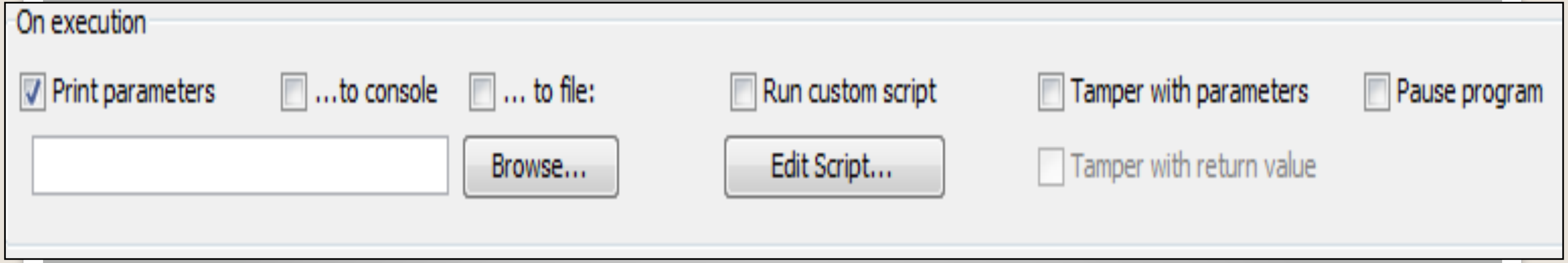
Search for function

Type part of the function name you want to search for: send [Search]

[✓] Hide Java/Sun classes [✓] Hide JavaSnoop classes

[ ] Return type void [✓] Ignore case

Return	Method
void	pclient.shd.ConnectExec.send(byte[])
void	pclient.shd.SockStream.send(byte[])
void	pclient.shd.SessionEnclosure.cmSendUserPass(String)



On execution

[✓] Print parameters [ ] ...to console [ ] ... to file: [Browse...]

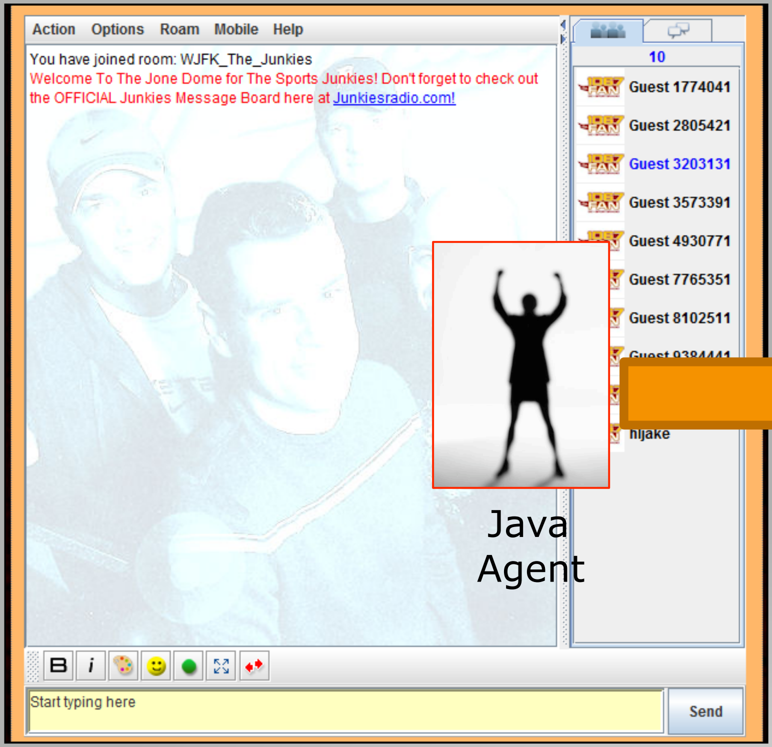
[ ] Run custom script [Edit Script...]

[ ] Tamper with parameters [ ] Pause program

[ ] Tamper with return value

Let's check "Tamper with parameters". Clock is ticking.

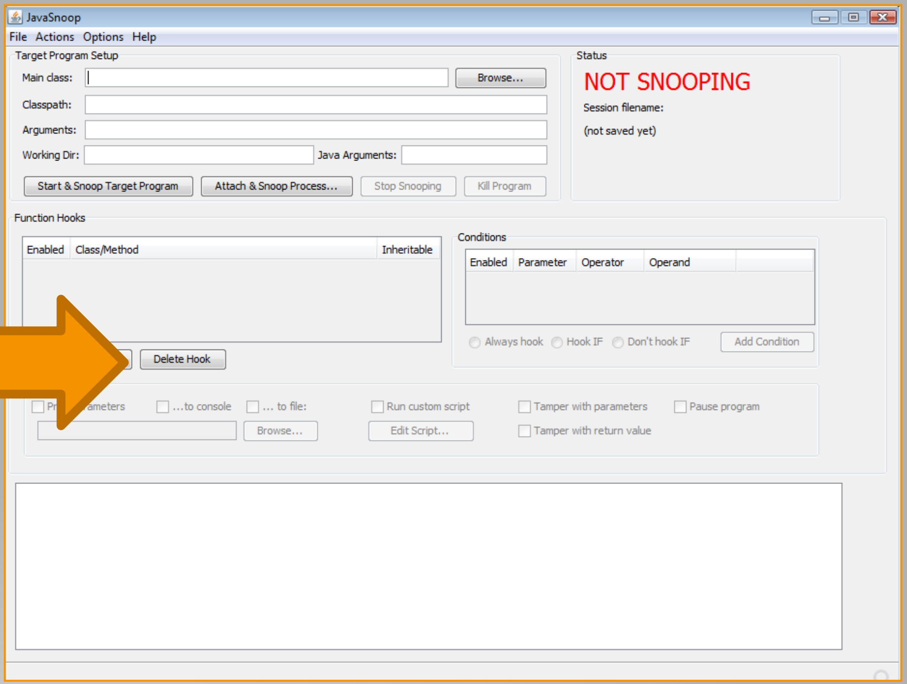
Step #5: JavaSnoop inserts a callback into method, which soon gets called



You have joined room: WJFK_The_Junkies
Welcome To The Jone Dome for The Sports Junkies! Don't forget to check out the OFFICIAL Junkies Message Board here at Junkiesradio.com!

Guest 1774041
Guest 2805421
Guest 3203131
Guest 3573391
Guest 4930771
Guest 7765351
Guest 8102511
Guest 9384441
hljake

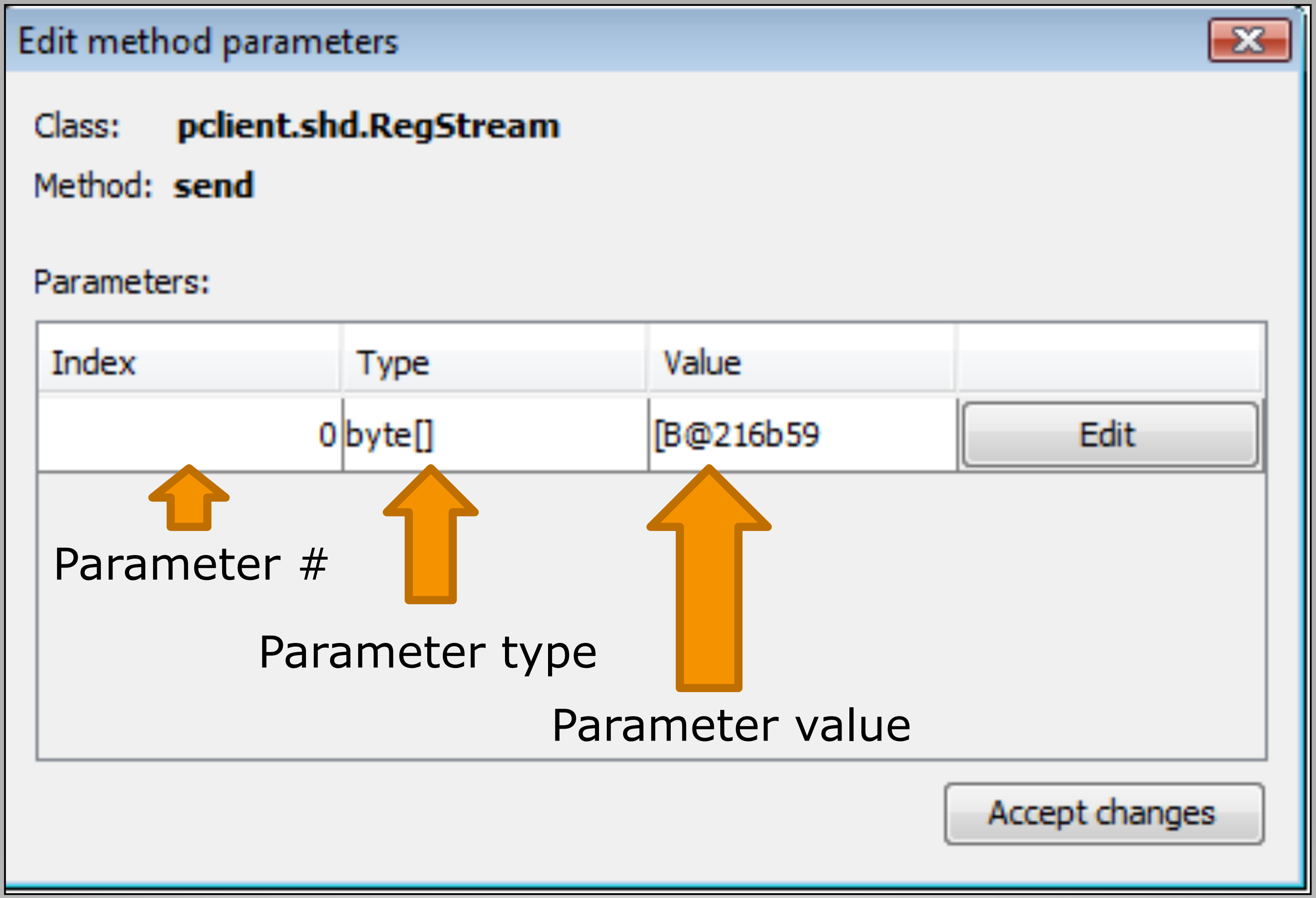
Start typing here



NOT SNOOPING
(not saved yet)

Can I start name dropping yet?
Better yet, will you name drop *me*?

Step #6: Tamper with the data



Edit method parameters

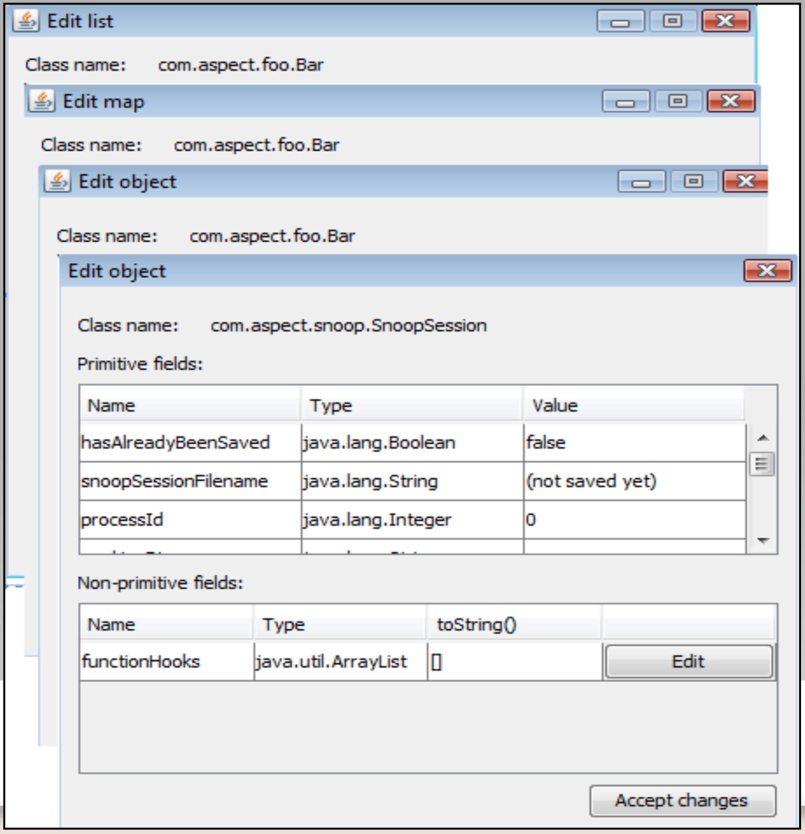
Class: **pclient.shd.RegStream**

Method: **send**

Parameters:

Index	Type	Value	
0	byte[]	[B@216b59	Edit

Aside: JavaSnoop has custom views for editing Lists, Maps, Java primitives, arrays, byte arrays, and even custom objects


Edit list
Class name: com.aspect.foo.Bar
Edit map
Class name: com.aspect.foo.Bar
Edit object
Class name: com.aspect.foo.Bar
Edit object
Edit
Accept changes

Class name: com.aspect.snoop.SnoopSession

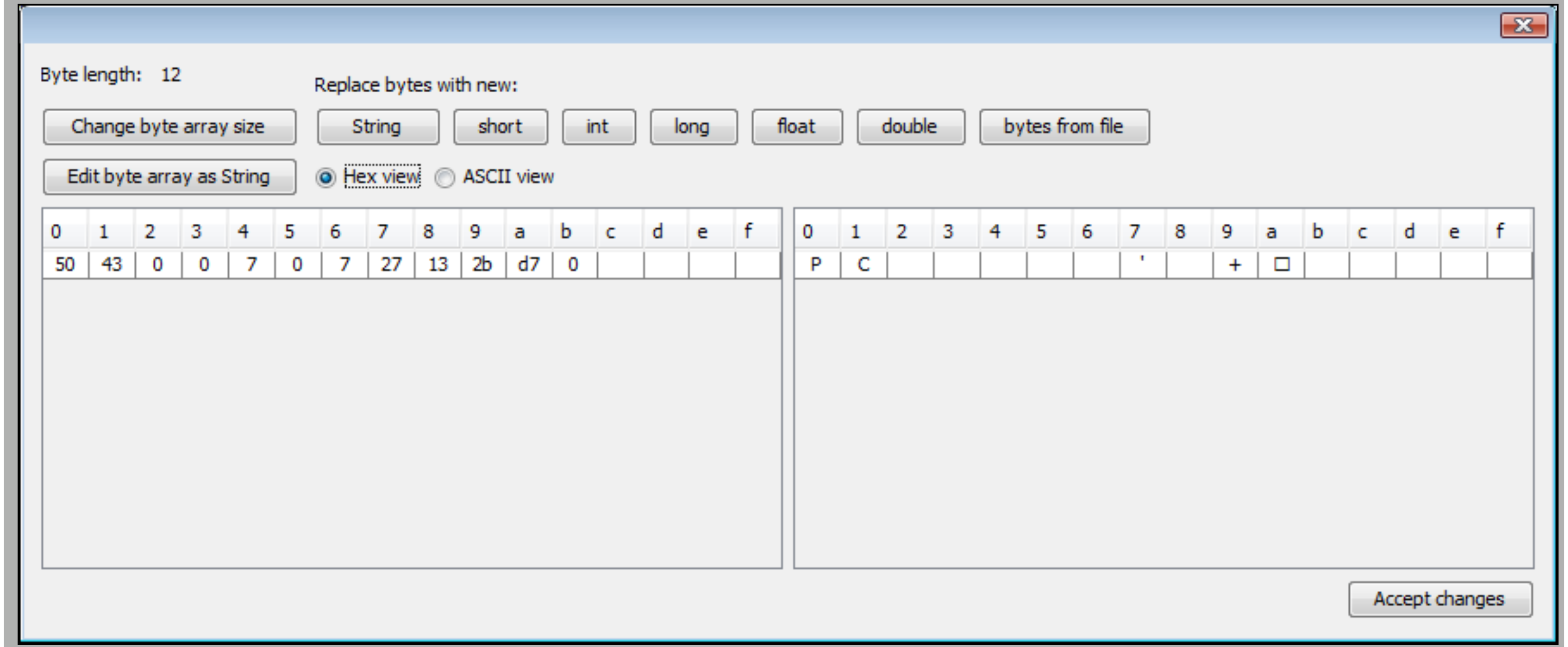
Primitive fields:

Name	Type	Value
hasAlreadyBeenSaved	java.lang.Boolean	false
snoopSessionFilename	java.lang.String	(not saved yet)
processId	java.lang.Integer	0

Non-primitive fields:

Name	Type	toString()	
functionHooks	java.util.ArrayList	[]	Edit

Step #7: Edit that carp.



0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
50	43	0	0	7	0	7	27	13	2b	d7	0				

0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
P	C						'		+	□					

I'll change that byte that contains my user ID, and hopefully the chat message will look like it came from Alice!

Step #8: Profit.

You spoofed the message. A serious flaw.

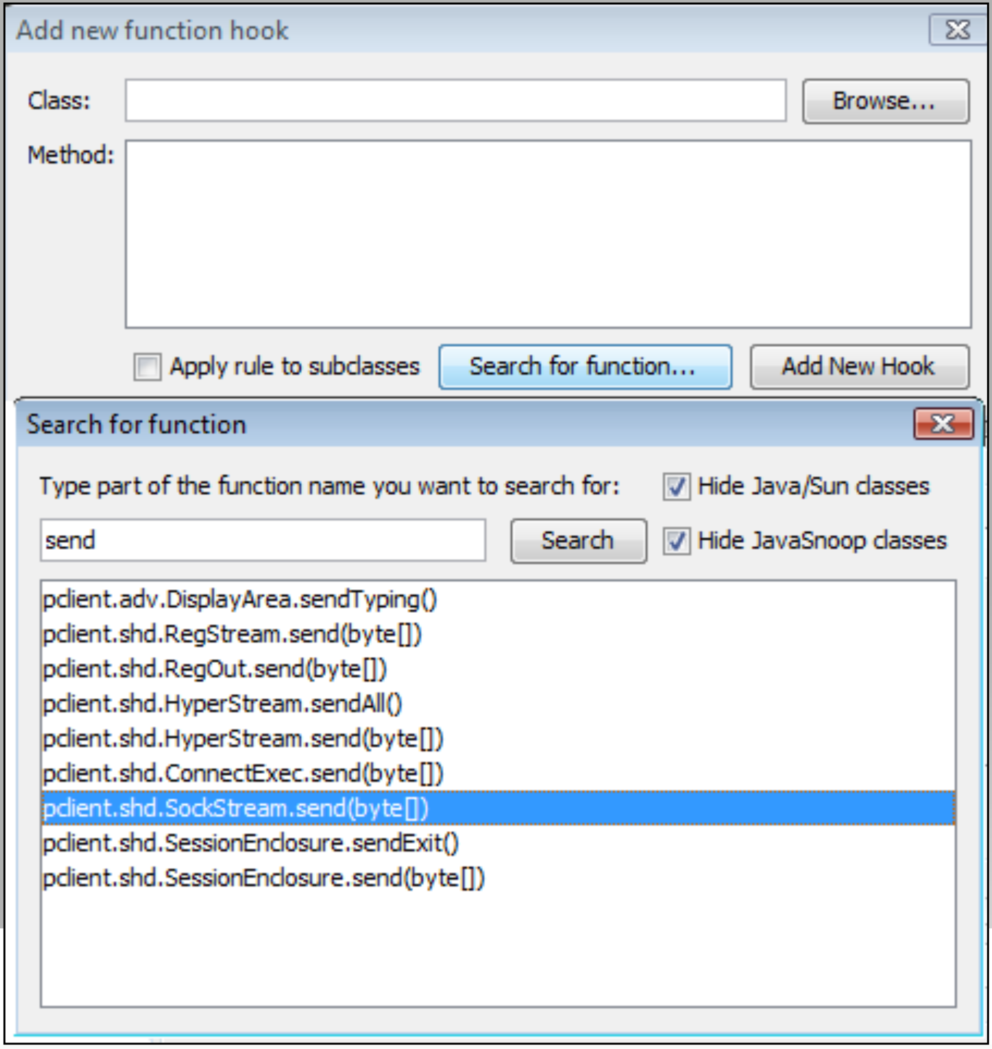
Time left: *2 hours*. That was close.

Agenda

- Why hacking Java apps is practically difficult
- Showing how JavaSnoop solves the problem
- ➡ Demos, videos, details

demo

Aside: How do I know which method to hook? Answer #1



[ ] Apply rule to subclasses

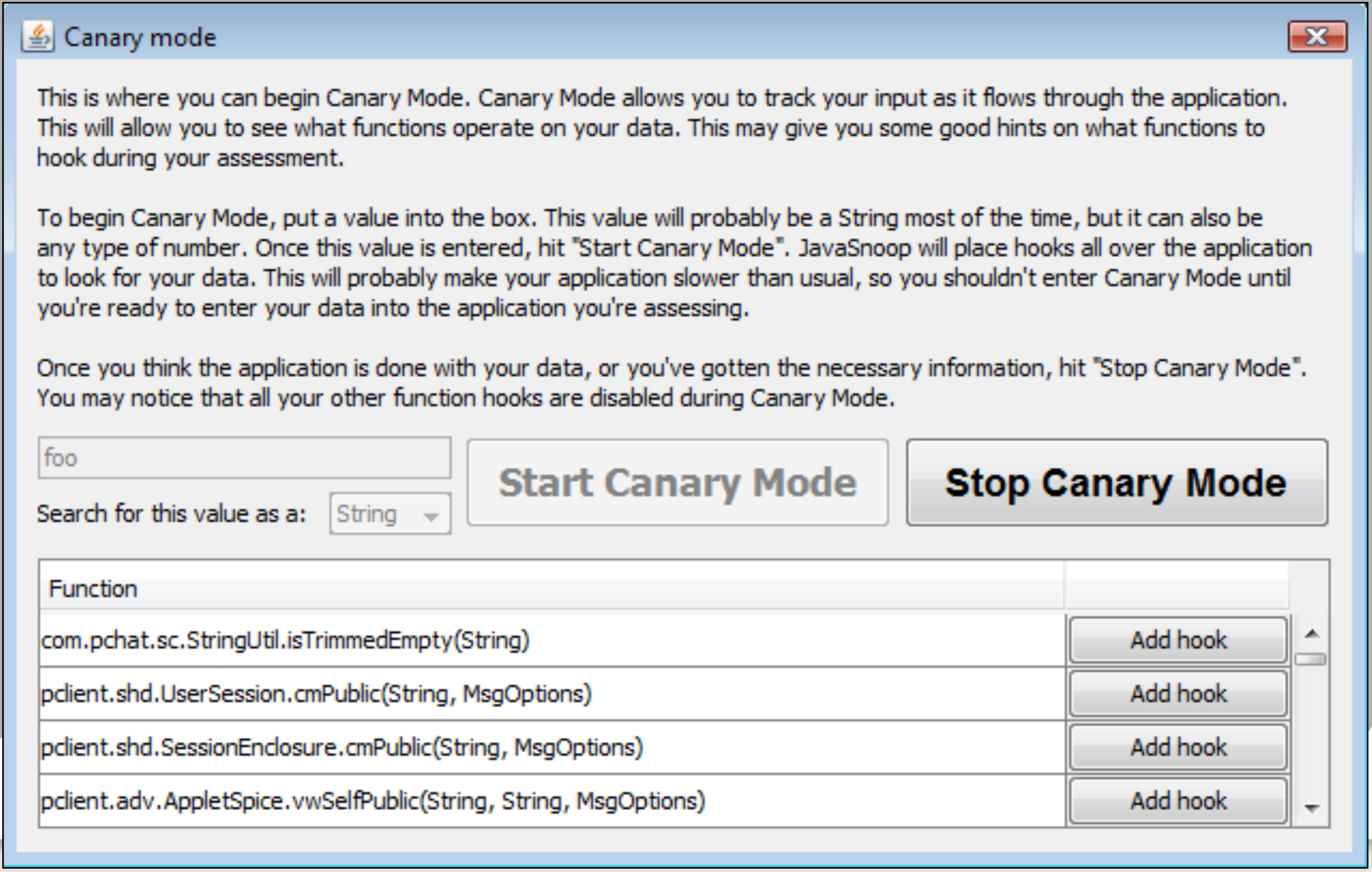
Type part of the function name you want to search for: send

[✓] Hide Java/Sun classes
[✓] Hide JavaSnoop classes

```
pclient.adv.DisplayArea.sendTyping()
pclient.shd.RegStream.send(byte[])
pclient.shd.RegOut.send(byte[])
pclient.shd.HyperStream.sendAll()
pclient.shd.HyperStream.send(byte[])
pclient.shd.ConnectExec.send(byte[])
pclient.shd.SockStream.send(byte[])
pclient.shd.SessionEnclosure.sendExit()
pclient.shd.SessionEnclosure.send(byte[])
```

- Browse classes and their methods
- Search by method name
- Search by return type

Aside: How do I know which method to hook? Answer #2



Canary mode

This is where you can begin Canary Mode. Canary Mode allows you to track your input as it flows through the application. This will allow you to see what functions operate on your data. This may give you some good hints on what functions to hook during your assessment.

To begin Canary Mode, put a value into the box. This value will probably be a String most of the time, but it can also be any type of number. Once this value is entered, hit "Start Canary Mode". JavaSnoop will place hooks all over the application to look for your data. This will probably make your application slower than usual, so you shouldn't enter Canary Mode until you're ready to enter your data into the application you're assessing.

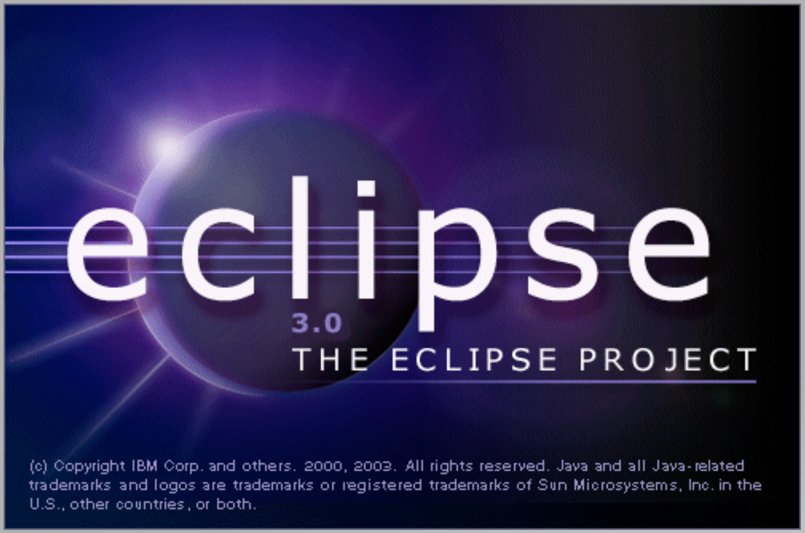
Once you think the application is done with your data, or you've gotten the necessary information, hit "Stop Canary Mode". You may notice that all your other function hooks are disabled during Canary Mode.

foo

Search for this value as a: String

Function	
com.pchat.sc.StringUtil.isTrimmedEmpty(String)	Add hook
pclient.shd.UserSession.cmPublic(String, MsgOptions)	Add hook
pclient.shd.SessionEnclosure.cmPublic(String, MsgOptions)	Add hook
pclient.adv.AppletSpice.vwSelfPublic(String, String, MsgOptions)	Add hook

Aside: How do I know which method to hook? Answer #3

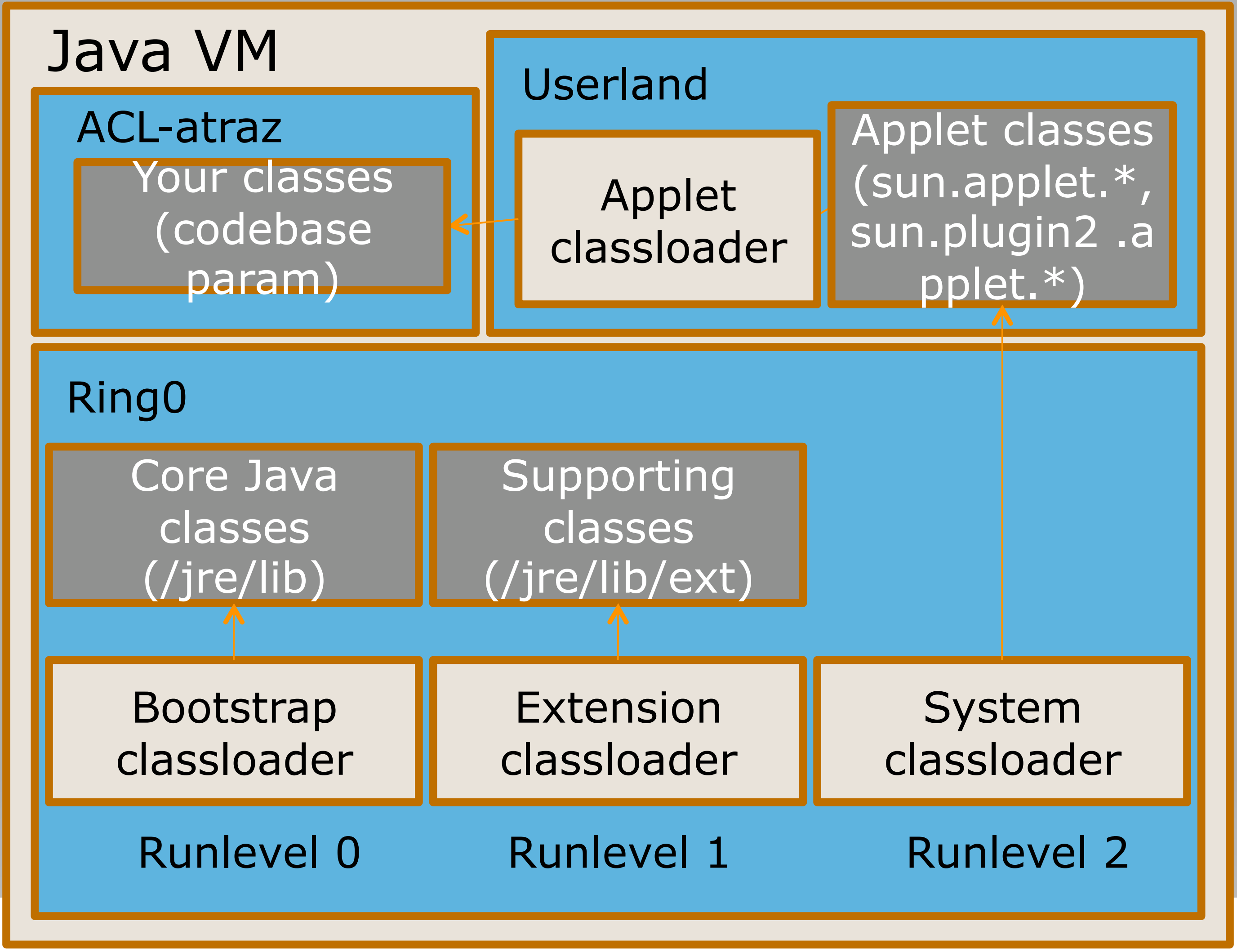


(c) Copyright IBM Corp. and others. 2000, 2003. All rights reserved. Java and all Java-related trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S., other countries, or both.

Rational AppScan

FORTIFY®

Dailydaver's guide to applets


Java VM
ACL-atraz
Your classes (codebase param)
Userland
Applet classloader
Applet classes (sun.applet.*, sun.plugin2 .a pplet.*)
Ring0
Core Java classes (/jre/lib)
Supporting classes (/jre/lib/ext)
Bootstrap classloader
Extension classloader
System classloader
Runlevel 0
Runlevel 1
Runlevel 2

How come JavaSnoop turns off Java security when it runs?

- Remember that evil Java agent we install in our target program?
- That little guy requires a lot of privileges to do the things he does
- Those privileges aren't usually granted to untrusted applets (which is smart)

JavaSnoop doesn’t create new vulnerabilities.

It just makes finding and exploiting flaws in Java apps possible.

And practical.

Supported Operating Systems

- ✓Windows XP/Vista/7
- ✓Mac OSX
- ✓Linux

That’s all.

- Thanks to Dave (Wichers|Anderson|Lindner), Jeff Williams, Nick Sanidas, Mike Fauzy, Jon Passki, Jason Li, Eric Sheridan, basically all the engineers at Aspect Security and Marcin Weilsdfisdfsdklfsdf of GDS for help/feedback/code
- RIP #madcircle #dword
- Check it out for yourself:

http://www.aspectsecurity.com/tools/javasnoop/

Arshan Dabirsiaghi

http://i8jesus.com/

http://twitter.com/nahsra