



链滴

关于一次不明原因的数据丢失，请大佬分析一下

作者: [MatrixCain](#)

原文链接: <https://ld246.com/article/1665755188634>

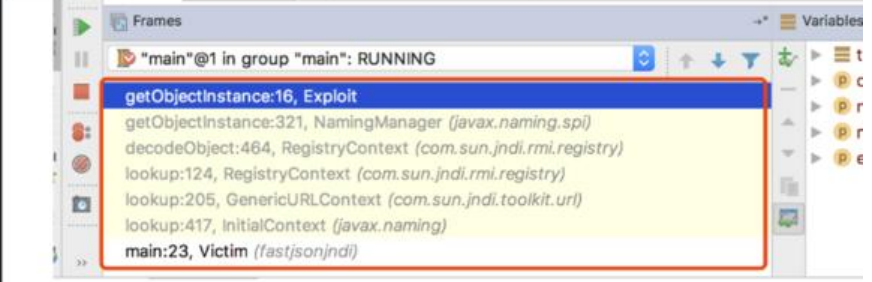
来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

事情的起因是这样的，下午上班和同事聊到java JNDI的几种攻击方法，我想起我在今年7月份的时在思源笔记上写了挺长一篇分析的文章（以下为我恢复后存的图片）

三 大纲	
✕ JNDI的利用	
H2 恶意类编写	
H2 JNDI漏洞利用分析	4
H3 什么是JNDI	
H3 JNDI中涉及的概念	3
H4 Naming是个什么东西	5
H5 javax.naming中的Context接口	
H5 javax.naming中的Name接口	
H5 javax.naming中的binding类	
H5 javax.naming中的reference类	
H5 javax.naming中的InitialContext类	
H4 JNDI范例	
H4 JNDI动态协议转换	
H3 JNDI原理探索	3
H4 在不传入env来指定ContextFactory时，initialContext()是怎么决定用什么处理查询内容的	
H4 决定执行查询的Context后lookup的逻辑又是怎样的	
H4 ☆JNDI注入核心点，java的JNDI是如何处理恶意查询结果的	
H3 JNDI核心原理详细分析 低版本利用与高版本利用分析	3
H4 具体情况举例之 --- 加载远程codebase中的reference	3
H5 适用前提	
H5 举例分析	
H5 java>8u191 修复原理，对使用codebase进行URLClassLoader加载前进行trustURLCodebase判断	
H4 具体利用情况举例之 --- 反序列化恶意返回数据中的serializedData	2
H5 适用前提	
H5 举例分析	
H4 具体利用情况举例之 --- 恶意的Reference Factory工厂类，并利用这个本地的Factory类执行命令	4
H5 前言	
H5 适用前提	
H5 举例分析	
H5 复现踩坑点	
H2 高版本利用	
H2 参考	

我下午发现这篇文章内容相当短只有



调用链:

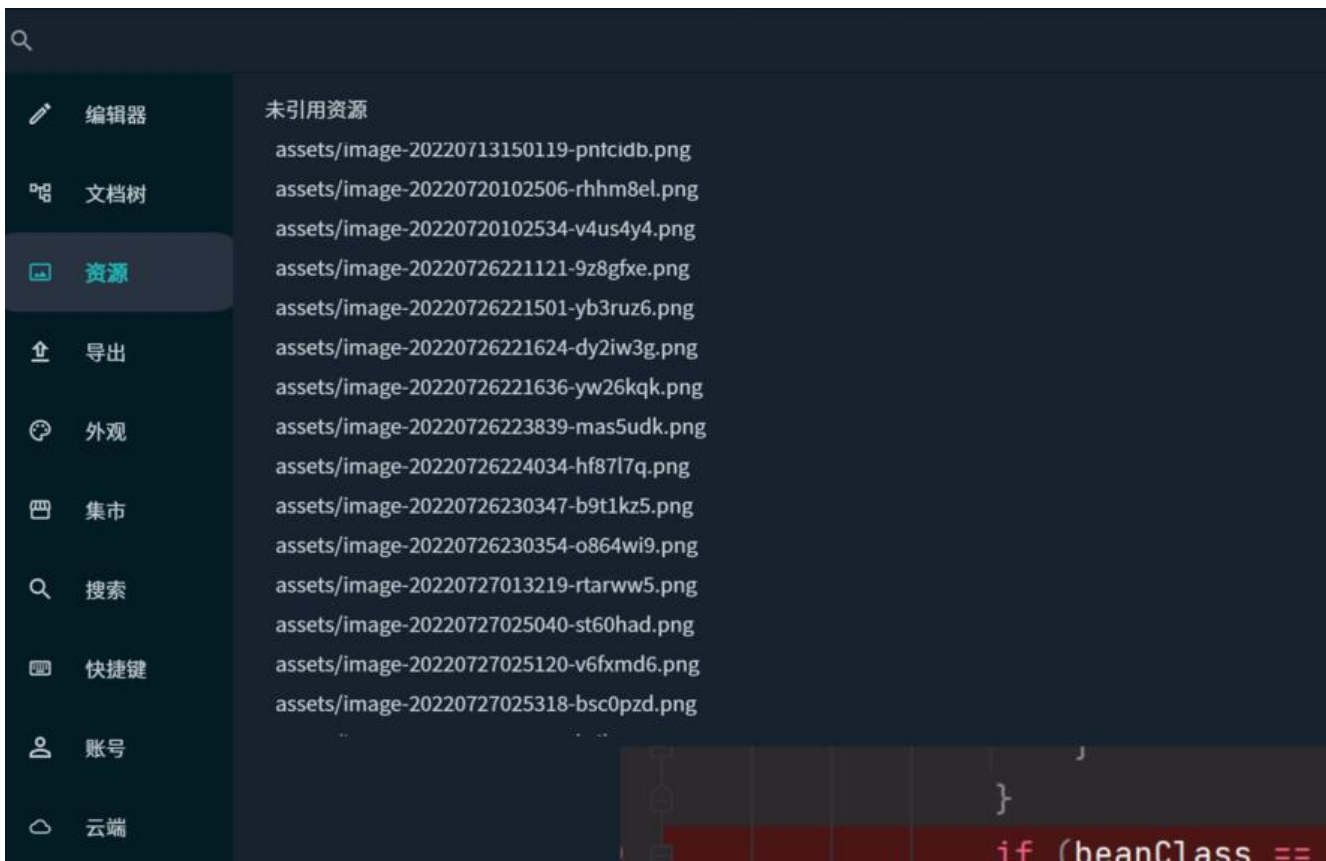
- -> RegistryContext.decodeObject()
- -> NamingManager.getObjectInstance()
- -> factory.getObjectInstance()

Tips: JNDI 查找远程对象时InitialContext.lookup(URL)的参数UF URL。

使用ObjectFactory对拿到的引用对象执行实例化因而RCE

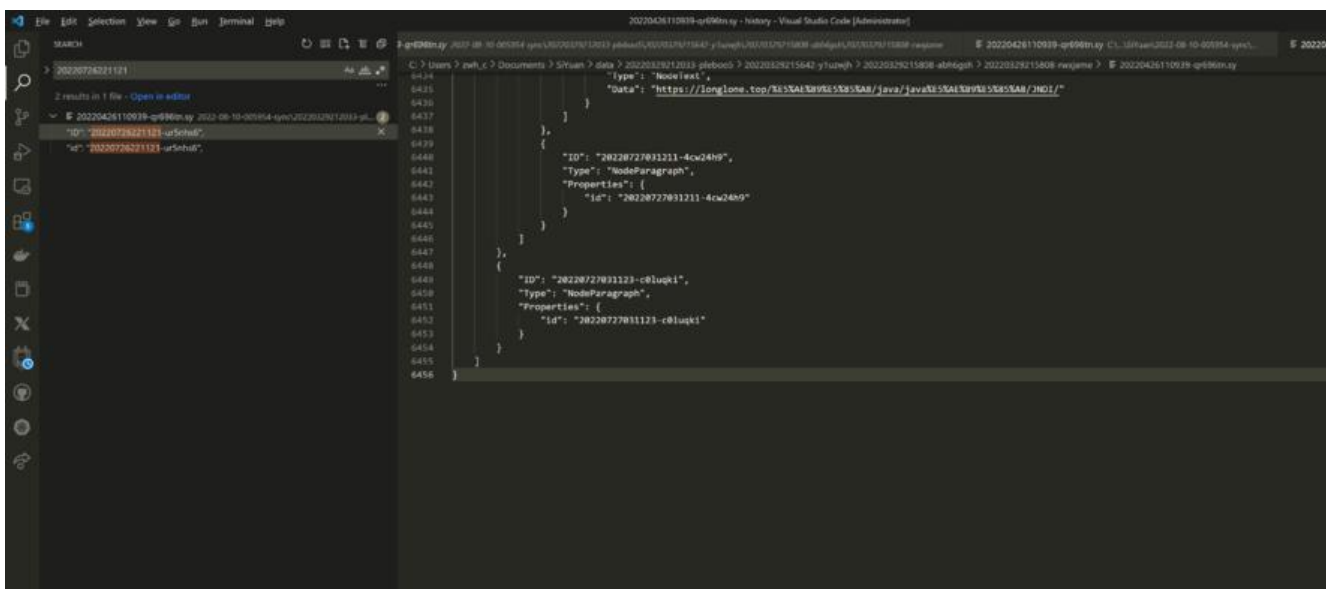
这么多，这篇文章其实我是分了两次写的，这个笔记很早就建立了，当时刚开始接触java安全相关，有深入学习，这是最早写的内容。在暑假也就是7月我用了一个晚上进行内容补充和深入学习。

可是今天下午再打开这篇文章发现内容只有上图这么多，当即我就知道不对劲，晚上回家后在另两台备查看发现都是如此，我甚至一度一味我是不是记错了，直到我打开了未引用资源、



发现了相当多未引用资源，对于我来说未引用资源一般都是截图截的不合适，例如粘贴到了思源笔记现截图大小等不合适，然后就会删掉，因此不太可能在同一天产生特别多未引用资源，。我选择几个开一看，发现果然就是当时截的图。

这时候其实我已经不抱太大希望能找回我的笔记，我在坚果云和另两台设备尝试查找发现备份时间为3天早已超时，笔记最后一次改动是今年7月。但是我的台式机开了365天备份数据，因此我用vscode开整个history文件夹，并利用文件内容搜功能定位到了当时写的完整笔记



可以看到行数相当多

我于是根据文件名和文件夹路径找到对应的data文件中的笔记，将不完整的笔记内容覆盖。重新打开源笔记后，能看到笔记大纲和部分内容，但是点击更下层内容提示找不到id，遂进行重建索引，之后

据就恢复了。

可以说我还是很幸运地，但是现在我就是担心会其他笔记未来也会出现莫名其妙的丢失情况，就像我知识记录下来放在书架上，我不希望等再次查阅时，书架上的书变成了空白的或者残缺的。

我的恢复过程如下，希望帮到可能遇到和我相同情况的用户

发现笔记为老版本的不完整笔记，且我的所有设备的思源笔记上的对应笔记均为老版本 -> 前往未引资源查找定位当时写笔记时引用的图片 -> 使用vscode打开思源笔记的history文件夹 -> 全局搜索图文件名关键字 -> 定位到完整笔记文件和对应路径 -> 将不完整笔记内容替换为完整笔记内容(CTRL+A -> CTRL+C -> CTRL+V) -> 打开思源笔记 -> 重建索引 -> 笔记成功恢复