Chad Ballay

CYBR350-342N

Week 9

08/30/2020

WebGoat – Journal pt 5 (A8-A9)

Please see previous doc for full notes.

<https://github.com/Chad-Ballay/Cybersecurity-work-class/blob/master/classes/CYBR350/week6/journal_notes.docx>

<https://github.com/Chad-Ballay/Cybersecurity-work-class/blob/master/classes/CYBR350/week7/journal_week7.docx>

<https://github.com/Chad-Ballay/Cybersecurity-work-class/blob/master/classes/CYBR350/week8/journal_week8.docx>

# Insecure Deserialization

## Insecure Deserialization

So I was unable to get this to work. The concept of getting data to be converted into a language friendly format that would then be actionable. I barely understand the concept and even after reading other articles I was lost on how to defeat this challenge.

<https://thehackerish.com/tag/insecure-deserialization-webgoat/>

# Vulnerable Components

## Vulnerable Components

The beginning of this one was a straight forward problem description with nothing concrete towards permanent resolution. Modern software requires using thirdparty components. You can onboard them only so quickly within your CICD pipeline for scanning. The iteration of versions also means that you must test continuously. Bugs get introduced, discovered, resolved at different cadences from your own scans.



This very issue is what has kept several proof of concepts from being done in my workplace. All software is broken or at least has to start off being treated as untrustworthy. The route to being trusted is a greater than zero cost for the review and validation. Sometimes the approval took longer than the time allocated for the idea. For this specific CVE my first stop was the NVD writeup itself.

<https://nvd.nist.gov/vuln/detail/CVE-2013-7285>

Still while trying to read several of the writeups I was unable to understand completely what is being done. It looks like the XML snippet we create will be consumed by the XStream() method. When that happens the xml has the ability to cause arbitrary commands to be executed. Even when recreating the existing writeups, I continued to receive a Deserializing Null Object error.



But the underlying issue is at least understandable. Core parts of the technology stack is dependent on third parties that may or may not care about the security of your process. Without vigilance this vector will be exploited and used to gain access to the data.