[SYSS-2019-039] Smartbear ReadyAPI/SoapUI Pro/jProductivity Licensing Unsafe Deserialization

From: Moritz Bechler <moritz.bechler () syss de>
Date: Tue, 19 May 2020 10:04:52 +0200

```
Advisory ID: SYSS-2019-039
Product: Protection Licensing Toolkit, SoapUI/LoadUI/ServiceV Pro
Manufacturer: jProductivity LLC, SmartBear Software
Affected Version(s): - ReadyAPI 3.2.5
Tested Version(s): ReadyAPI 3.2.5
Vulnerability Type: Unsafe deserialization/remote code execution (CWE-502)
Risk Level: High
Solution Status: Open
Manufacturer Notification: 2019-09-02
Public Disclosure: 2020-05-18
CVE Reference: CVE-2020-12835
Author of Advisory: Moritz Bechler, SySS GmbH

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Overview:

jProductivity Protection! is a solution for software vendors to
implement licensing checks and management in their products.

The manufacturer describes the product as follows (see [1]):

"Protection! - is a powerful multi-platform Licensing Toolkit and License
Manager that provides the ability to add licensing into custom applications
or components only allowing the permitted use according to the supplied
license."


ReadyAPI is a suite of web service testing tools. It is using
the jProductivity Protection licensing solution.

The manufacturer describes the product as follows (see [2]):

"The ReadyAPI platform accelerates functional, security, and load testing
of RESTful, SOAP, GraphQL and other web services right inside your CI/CD
pipeline."

The jProductivity Protection Licensing Toolkit is using RMI-based
network protocols to communicate with its network license server.
These protocols are susceptible to deserialization attacks, which
in the case of ReadyAPI can be exploited to gain remote code execution
on the client side.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Vulnerability Details:

When trying to check out a remote floating license, the client
softare, ReadyAPI, contacts the Licensing Server using the
Java RMI protocol on port 1099. As there is no transport security,
this service can be impersonated by an attacker in a suitable
position on the network.

Java RMI, and the underlying JRMP protocol, heavily relies on
Java serialization to transport method arguments, return values
and exception data.
Java serialization has been shown ([5]) to in many cases
allow the execution of arbitrary code when certain specially
crafted object graphs are reconstructed during deserialization.

ReadyAPI contains multiple libraries with published gadgets
that can be exploited in this way.

While the license server suffers from the same vulnerability,
no gadgets were identified that lead to direct code execution.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Proof of Concept (PoC):

Setup a JRMP/RMI service that returns a malicious serialized object
graph. In this case, a gadget from the commons-beanutils library is
used to get command execution. Other options exist on the ReadyAPI
classpath.

========================================================================
$ java -DproperXalan=true \
 -cp commons-beanutils-1.9.3.jar:target/ysoserial-0.0.6-SNAPSHOT-all.jar
  ysoserial.exploit.JRMPListener 1099 CommonsBeanutils1 gnome-calculator
* Opening JRMP listener on 1099
Have connection from /192.168.56.102:34834
Reading message...
Sending return with payload for obj [0:0:0, 0]
Closing connection
========================================================================

When trying to check out a floating license from the rogue server,
RMI calls are made which results in the deserialization of the
attacker-provided serialized data. Here, this causes the gnome-calculator
program to be run.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Solution:

Avoid using Java serialization-based network prococols like RMI and
deserializing untrusted data in general.
If they cannot be avoided, strict whitelist-based filtering allowing only
the neccessary object types should be performed.

Other users of the jProductivity Protection Licensing Server are likely
affected as well.

There is no vendor patch available as of now.

Mitigation in ReadyAPI may be possible adding the following serialization
filter to bin/ready-api.sh (however, this may break other features):

JAVA_OPTS="$JAVA_OPTS -Djdk.serialFilter=java.util.*;java.security.*;
java.lang.*;sun.security.**;com.jp.protection.pub.**;dev.util.collections.*;
com.jp.protection.pub.pro.lserver.rmi.**;java.rmi.**;sun.rmi.**;!*"
```

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Disclosure Timeline:

2019-08-08: Vulnerability discovered
2019-09-02: Vulnerability reported to manufacturer
2019-10-10: On inquiry, "early 2020" is mentioned as the fix timeline
2020-01-30: Requested an update, no reply
2020-03-20: Another inquiry, no clear timeline provided
2020-04-15: Final 4 week deadline set, mitigation suggested
2020-05-18: Public disclosure of vulnerability

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

References:

[1] Product website for jProductivity Protection!
    http://www.jproductivity.com/products/protection/
[2] Product website for ReadyAPI
    https://smartbear.com/product/ready-api/
[3] SySS Security Advisory SYSS-2019-039

https://www.syss.de/fileadmin/dokumente/Publikationen/Advisories/SYSS-2019-039.txt
[4] SySS Responsible Disclosure Policy
    https://www.syss.de/en/news/responsible-disclosure-policy/
[5] ysoserial, "Marshalling Pickles: how deserializing objects will ruin
your day"
    https://github.com/frohoff/ysoserial/

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Credits:

This security vulnerability was found by Moritz Bechler of SySS GmbH.

E-Mail: moritz.bechler () syss de
Public Key:
https://www.syss.de/fileadmin/dokumente/PGPKeys/Moritz_Bechler.asc
Key ID: 0x768EFE2BB3E53DDA
Key Fingerprint: 2C8F F101 9D77 BDE6 465E  CCC2 768E FE2B B3E5 3DDA

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Disclaimer:

The information provided in this security advisory is provided "as is"
and without warranty of any kind. Details of this security advisory may
be updated in order to provide as accurate information as possible. The
latest version of this security advisory is available on the SySS website.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Copyright:

Creative Commons - Attribution (by) - Version 3.0
URL: http://creativecommons.org/licenses/by/3.0/deed.en
```

Attachment: **smime.p7s**
Description: S/MIME Cryptographic Signature

By Date By Thread

Current thread:

[SYSS-2019-039] Smartbear ReadyAPI/SoapUI Pro/jProductivity Licensing Unsafe Deserialization *Moritz Bechler (May 19)*

Site Search

Nmap Security Scanner
Ref Guide
Install Guide
Docs
Download
Nmap OEM

Npcap packet capture
User's Guide
API docs
Download
Npcap OEM

Security Lists
Nmap Announce
Nmap Dev
Full Disclosure
Open Source Security
BreachExchange

Security Tools
Vuln scanners
Password audit
Web scanners
Wireless
Exploitation

About
About/Contact
Privacy
Advertising
Nmap Public Source License