Your Trusty Python Package:

TTPs of attacks on OSS in Python

Leonid Akinin
Security Architect – EPAM Systems

November 2023

Disclaimer

This material was created for educational purposes and contains examples of source code that can be weaponized for malicious purposes. The author and the company (EPAM Systems and all its affiliates) are not responsible and/or liable for any damage of any kind, to any human or organization, caused by the misuse of these materials. Use only for educational purposes and at own risk.

Contents

01

Why this topic is important?

Machine-Learning Python package compromised in supply chain attack

by Cedric Pernet in Developer on January 4, 2023, 12:00 PM EST

A nightly build version of a machine-learning framework dependency has been compromised. The package ran malicious code on affected systems and stole data from unsuspecting users.

Software supply chain attacks caused PyPI to temporarily suspend new users and projects

Technical | May 24 2023 | 1 min read

In the past several months, the Python Package Index (PyPI), the official third-party repository for Python packages, has faced a surge in malicious users and projects. One of these software supply chain attacks–a malicious package that was uploaded to PyPI–was found by the Apiiro AI risk engine in December 2022.

Six Malicious Python Packages in the PyPI Targeting Windows Users

6,237 people reacted | 16 | 11 min. read

SHARE

By Shaul Ben Hai
July 11, 2023 at 6:00 AM
Category: Cloud
Tags: Cloud Security, Cortex EDR, Cortex XDR, malicious code, open source, Open Source Software, Prisma Cloud, Python, WildFire

ReversingLabs Blog

Threat Research | August 31, 2023

VMConnect supply chain attack continues, evidence points to North Korea

ReversingLabs researchers discovered more packages that are part of the previously identified VMConnect campaign, as well as evidence linking the campaign to North Korea's Lazarus Group.

BLOG AUTHOR
Karlo Zanki, Reverse Engineer at ReversingLabs. READ MORE...

WHY THIS TOPIC IS IMPORTANT?

"...make them believe, that offensive operations, often times, is the surest, if not the only (in some cases) means of defence"

George Washington, 1799

"The only real defence is active defence"

Mao Zedong

"To know your Enemy, you must become your Enemy"

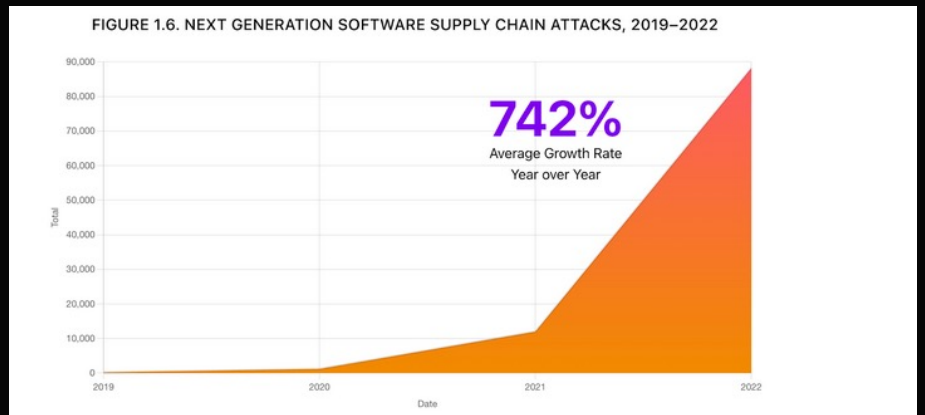
Sun Tzu, "Art of War", 5th century BC

03

History of supply-chain attacks

- First supply chain attacks date back to 2017 with initial campaigns targeting dockerhub, npm and pypi
- The “SolarWind Attack” was first high-profile attack that drew attention of cyber experts and authorities
- Rapid grows since 2020
- Supply chain attacks became one of the favorite vectors for major APTs due to traditional lack of control over development environments
- Attacks range from opportunistic to precisely planned and tailored towards specific organization

Malicious Packages in NPM and PyPI | Source: ReversingLabs



Source: https://www.sonatype.com/state-of-the-software-supply-chain/open-source-supply-demand-security

04

TTPs in supply-chain attacks

TTP

Tactics

"Why?" – the reason an attacker performs the action

- Initial access
- Perimeter bypass
- Data exfiltration
- Ransomware

Techniques

"How?" – how an attacker performs the action

- Uploading malicious packages to repositories
- Typosquatting
- Starjacking
- Injection of malicious code through dev credentials compromise

Procedures

Step-by-step application of techniques

Supply-Chain Compromise

- Project/Repository infiltration
- Dependency Infiltration
- Infiltration of private PyPi repositories and servers
- Distribution through public GitHub projects, FTP servers, etc.
- Typosquatting
- Starjacking

Defense Evasion

Payload obfuscation:

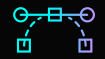
- Encoding
- Encryption
- Bytecode
- Embedding binaries

Traffic obfuscation:

- DNS Exfiltration
- Proxying/Tunnelling

Installation & Delivery

- __init__.py
- setup.py
- dropper



Exfiltration & C2

- Info stealers
- RATs

Supply-chain compromise

Public project/repository infiltration

- Transfer of ownership
- Official channels of contribution

Dependency Infiltration

- Infiltration of a project that provides dependency for the main target

Attacks on private PyPi servers/proxies

- Poorly managed PyPi servers
- Wide-open and over-permissive PyPi repositories
- Vulnerable PyPi servers

Public GitHub repos and FTP servers

- Package distributed as source-code(no egg/wheel)

Typosquatting

- Malicious packages registered using naming patterns similar to legitimate projects

Starjacking

- Utilization of a technical flaw in PyPi ecosystem that allows and attacker to make a references to an arbitrary GitHub source effectively stealing rating stars of that repository

Starjacking demo

Defense Evasion - Obfuscation

Payload obfuscation:

- Encoding – encoding strings to base64, UNICODE, etc.
- Encryption – encrypting payload strings
- Bytecode – embedding bytecode
- Embedding binary executables – embedding executables written in a different language

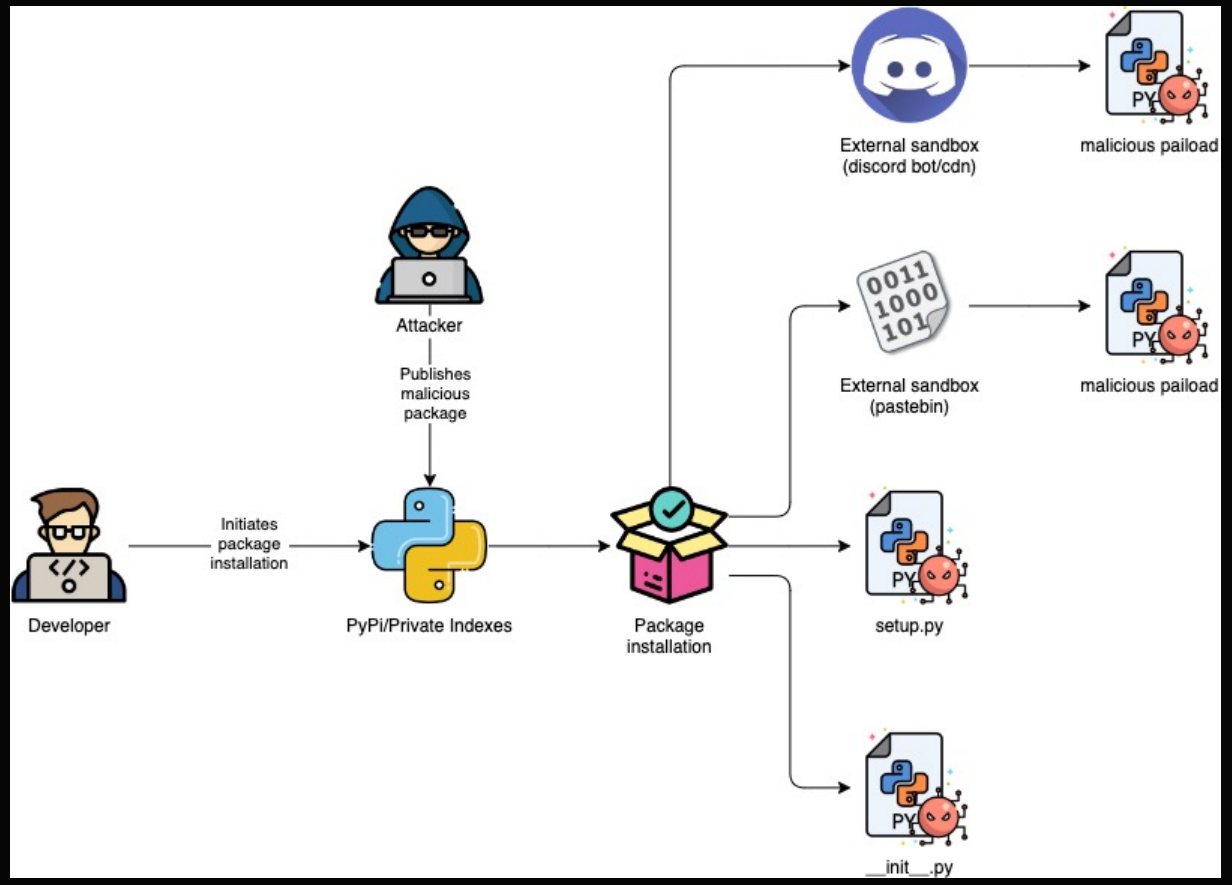
Traffic obfuscation:

- DNS Exfiltration – method of exfiltrating data through DNS tunnelling
- Proxying/Tunnelling – using interim proxies or anonymizers to hide/passthrough traffic

Payload obfuscation demo

Installation & Delivery

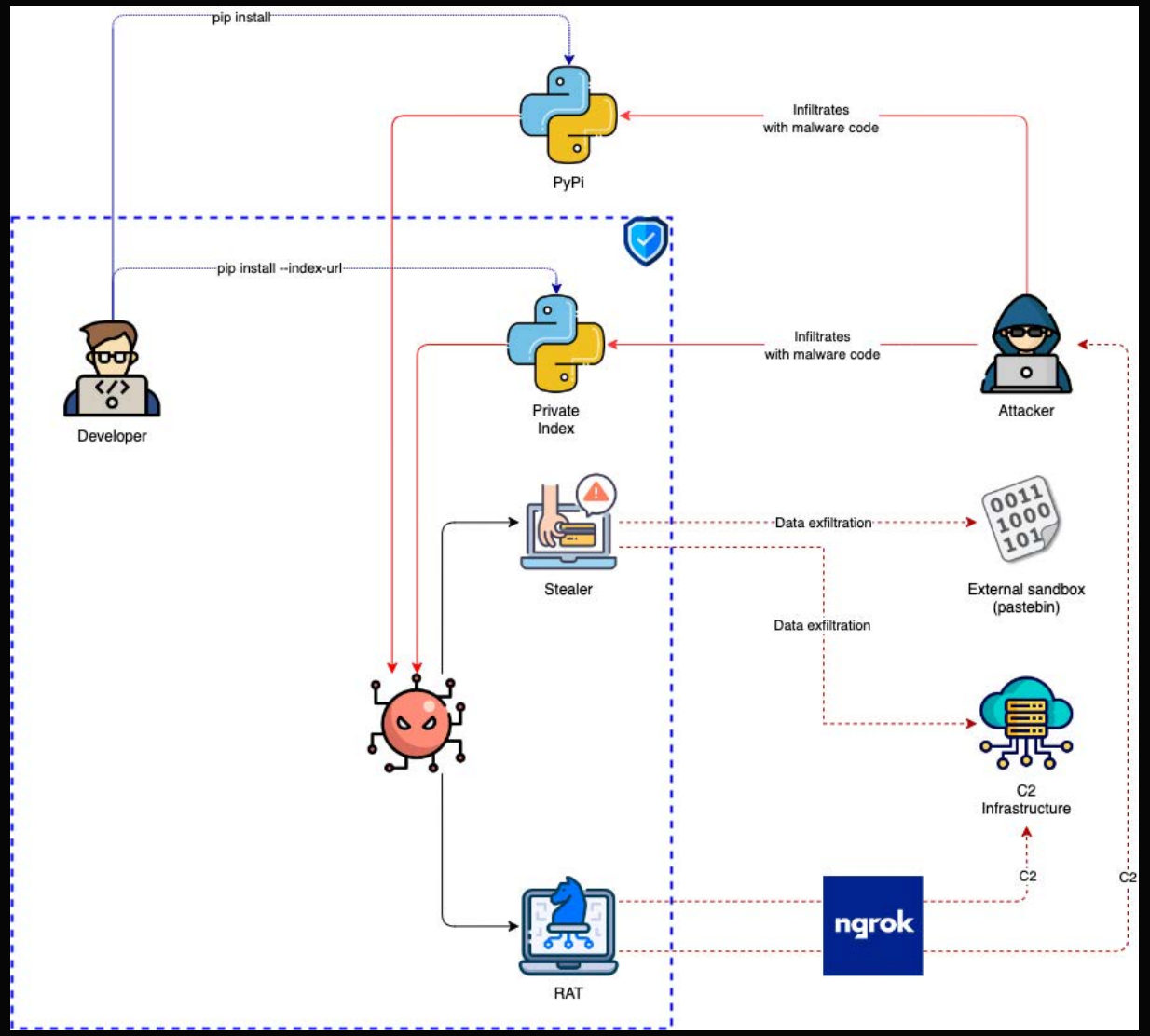
- __init__.py – payload invoked when package is imported
- setup.py – payload invoked during installation
- droppers – payload is delivered from the outside (external sandboxes) when package is imported or installed:
    - Discord
    - Pastebin
    - Telegram bots
    - …

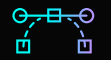


Installation & delivery demo

Exfiltration and C2

- Info stealers
- Remote Access Trojans (RATs)





Exfiltration & C2 demo

05

Defences

Pre-supply protection

- Individual development sandboxes
- Avoiding shared development servers
- Review project details and reputation
- Code review (manual/grep/semgrep)
- Package quarantine
- Avoiding projects that are not published on PyPi
- Fixed versions of dependencies
- Restrict direct downloads of dependencies (use private PyPi servers as trusted proxies)
- SCA on pre-commits
- AV/EDR

Post-supply protection

- Dev sandboxes
- Traffic monitoring
- Principle of least privilege on build agents/nodes
- Semgrep + SCA + SBOM integrated into CI/CD pipelines
- AV/EDR

06

Credits and references

Credits:

- EPAM Systems LTD for supporting my initiatives on security researching and public speaking
- EvilBunnyWrote for all the CTF events and helping with the researching
- www.flaticon.com for icons and graphics used in this presentation

References and additional reading:

1. ActiveState - How to Mitigate the 3 Most Common Python Supply Chain Threats
2. Apiiro - Software supply chain attacks caused PyPI to temporarily suspend new users and projects
3. Exploit-Notes - Python Yaml Privilege Escalation
4. Fortinet - Supply Chain Attacks: Examples and Countermeasures
5. Hinty.io - DNS exfiltration of data: step-by-step simple guide
6. Linux Foundation - The Rising Threat of Software Supply Chain Attacks: Managing Dependencies of Open Source projects
7. Reversinglabs Blog - A (Partial) History of Software Supply Chain Attacks
8. Reversinglabs Blog - VMConnect supply chain attack continues, evidence
9. Sonatype - A History of Software Supply Chain Attacks
10. Semgrep - Code injection prevention for Python
11. TechRepublic - Machine-Learning Python package compromised in supply chain attack
12. Unit 42 - Six Malicious Python Packages in the PyPI Targeting Windows

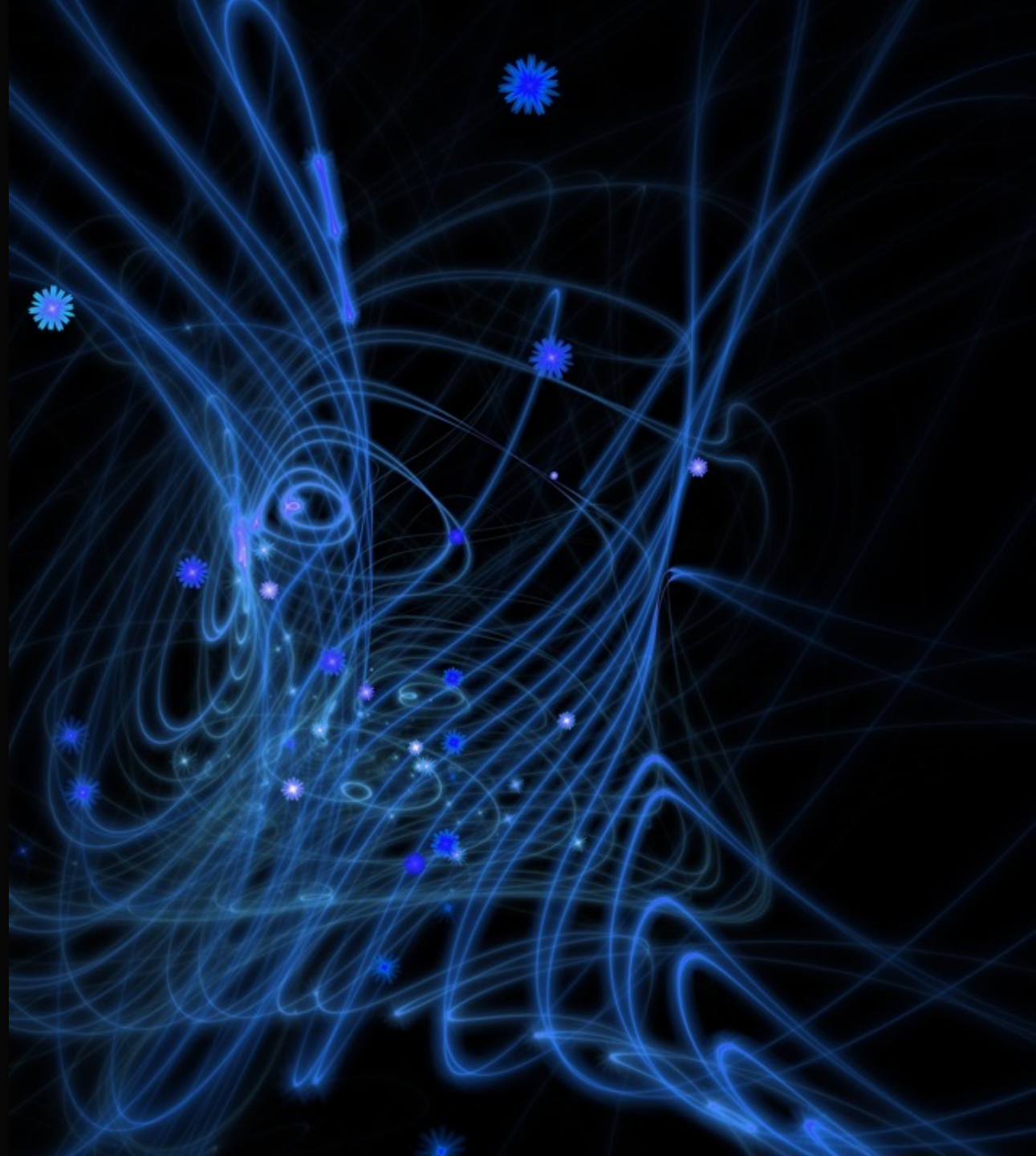
Thank you!

For questions and comments please get in touch

Leonid Akinin

Security Architect

leonid_akinin@epam.com



‹epam›