

Xinference PyPI Supply Chain Poisoning Warning

nsfocusglobal.com/xinference-pypi-supply-chain-poisoning-warning

NSFOCUS

April 24, 2026



Overview

Recently, NSFOCUS CERT detected that Xinference had suffered supply chain poisoning in its PyPI warehouse. The attacker stole the PyPI release permission credentials of Xinference maintainers and released three consecutive malicious versions implanted with Trojans on April 22, GMT+8. When triggered by the user, it will collect cloud credentials, SSH keys, API tokens, Sensitive data such as database passwords, cryptocurrency wallets and environmental variable configurations are packaged and sent to the attacker's C2 server; The impact is wide, and relevant users are requested to take measures for investigation and protection as soon as possible.

Xinference (Xorbits Inference) is an open source distributed AI model inference framework designed specifically for the deployment and management of language, speech recognition, and multimodal models. It is widely used in AI development, research, and privatized large-scale model deployment.

Reference link: <https://research.jfrog.com/post/xinference-compromise>

Scope of Impact

Affected versions

- Xinference = 2.6.0
- Xinference = 2.6.1
- Xinference = 2.6.2

Note: The total download volume of Xinference in the PyPI repository has exceeded 680,000 times, and users who have installed and used the above 3 malicious versions are affected.

Unaffected version

Xinference <= 2.5.0

Note: Please pay attention to the official version release dynamics, download link:

<https://github.com/xorbitsai/inference/releases>

Event Analysis

Event timeline

- April 13, 2026: Xinference officially released version 2.5.0
- April 22, 2026: Attackers release 3 malicious versions of PyPI, and users report suspicious behavior
- April 22, 2026: JFrog releases analysis report, TeamPCP statement that third parties are using its name to commit crimes

The attacker's supply chain poisoning against Xinference should be achieved by obtaining the PyPI permission credentials of the Xinference maintenance personnel, who implanted malicious code encoded in multiple layers of Base64 in the project's `__init__.py` module file. When the user installs the affected xinference library or executes `import xinference`, The Python interpreter loads `__init__.py`, where the malicious payload is automatically decoded and executed in memory; The decoded payload annotation begins with the `#` hacked by teampcp logo, which may be related to the participant mark that appeared in the recent TeamPCP intrusion incident. However, TeamPCP denied carrying out this attack through its social account and called it a deliberate imitation by a third party. After a user discovered abnormal behavior and asked about it on the project's GitHub issue, Xinference maintainers confirmed and urgently removed the relevant malicious version.

Malicious version behavior

Xinference 3 malicious versions of `__init__.py` execute code when imported into Python, creating a variable named "test" and a payload containing base64 encoding, decoding it and passing it to `subprocess.Popen` for execution:

Stage1

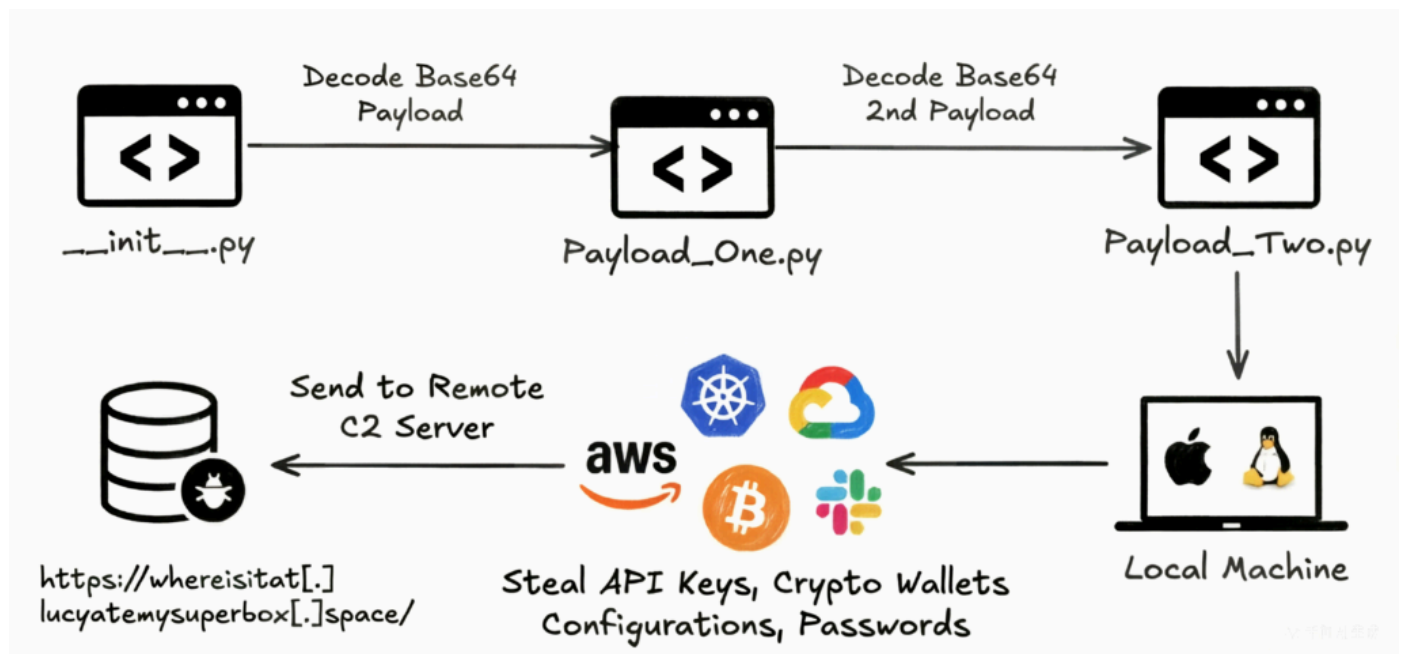
1. Create a temporary directory;
2. Decode the second-stage collector and derive a separate child Python interpreter process
3. Write the standard output of the child process to a temporary file and compress it into `love.tar.gz`
4. Use the curl command with a custom HTTP header `X-QT-SR: 14` to upload the collected data to the attacker's server `https://whereisitat.lucyatemysuperbox.space`
5. Perform exception handling, suppress stdout/stderr and clean up temporary files to achieve concealment

Stage2

1. Obtain the current user, device, IP address, network interface, environment variables and SSH key information of the target host
2. Extract configuration files: .env, .npmrc/.pypirc, AWS, Kubernetes, Google Cloud, Docker, database configuration
3. Collect key credentials: SSH key, SSL certificate, Git credential, AWS/Azure/GCP cloud credential, Slack key, Discord key, Slack key, Discord key, database password
4. Get infrastructure: Terraform, Helm, WireGuard configuration
5. Other information: shell history, cryptocurrency wallet, local account data

AWS-specific exploitation behavior

Actively detect cloud metadata services 169.254.169.254/169.254.170.2, retrieve IMDSv2 tokens and IAM role credentials, capture temporary cloud credentials and try to call Secrets Manager and SSM to obtain sensitive information in the cloud environment.



Trojan execution process (Source: ox.security)

Risk Investigation

Relevant users can conduct troubleshooting according to the following steps:

1. Check the current X inference version:

pip show xinference | grep Version

2. If it is a malicious version, immediately downgrade to the safe version:

pip install xinference==2.5.0 -y

If it is confirmed that there are malicious packages in the environment, the infected assets should be taken offline immediately and the hosts should be physically or logically isolated.

#Clean pip cache:

pip cache purge

Check if there is any residue in the xinference directory under

site-packages: python -c "import site; Print(site.getsitepackages())"

find <the above site-packages path> -path "*xinference*" 2>/dev/null If residual files are found, the entire xinference directory and associated egg-info need to be recursively deleted; at the same time, check whether the ~/.xinference/ configuration directory has been tampered with, clear its contents and rebuild permissions. Finally, verify that there are no xinference entries in the pip list output. Running python -c "import xinference" should report ImportError to ensure complete removal.

Check whether there are abnormal external connection behaviors or C2 communication traces, and whether requests to 169.254.169.254/169.254.170.2 appear in the log; Users can add the configuration: 127.0.0.1 whereisitat.lucyatemysuperbox.space in the host file to block malicious domain names from attackers.

Review server login logs and sensitive files to check for abnormal logins and backdoor legacy:

Check whether the SSH key has been accessed or modified abnormally:

Ls -la ~/.ssh/id_rsa ~/.ssh/authorized_keys 2>/dev/null#Check whether there are abnormal

curl/wget/base64 decoding commands in the shell history:

Grep -E "curl|wget|base64" ~/.bash_history ~/.zsh_history 2>/dev/null#Check whether there is sensitive information in the environment variable:

env | grep -E "SECRET|TOKEN|PASSWORD|KEY"

Summary and Recommendations

This incident has once again sounded the alarm for security in the AI ecosystem and open source community. Supply chain security is no longer just an option for traditional development, but has gradually become the core infrastructure of the AI era. All units must establish a normalized open source component security audit mechanism and build a complete supply chain security governance system to prevent such threats.

It is recommended that affected users take the following measures to deal with it:

1. Revoke and rotate all SSH keys, cloud service certificates, API Keys, certificate keys, k8s tokens, database passwords, cryptocurrency wallets and other credentials;
2. Block and investigate malicious IoCs;
3. Enable two-factor authentication for accounts on platforms such as PyPI and GitHub and rotate secrets regularly;
4. Build an internal private PyPI image, scan it securely to verify the signature before synchronizing;
5. Establish a normalized open source component security audit mechanism and increase multiple approval processes before release;
6. Subscribe to PyPI security alerts, GitHub Security Advisories, etc. for supply chain monitoring.

IOCs

Malicious files

File name: xinference-2.6.0.tar.gz

MD5:971670c10eff28339a085ca50a600e35

File name: xinference-2.6.0-py3-none-any.whl

MD5:3ee893ae46530b92e0d26435fb979d82

File name: xinference-2.6.1.tar.gz

MD5:9b3257e45b27a6bbe4e240e41a3a306f

File name: xinference-2.6.1-py3-none-any.whl

MD5:e291734d46c313a23d676681499f8846

File name: xinference-2.6.2.tar.gz

MD5:484067fd6232f7cdd7b664b33857fc2c

File name: xinference-2.6.2-py3-none-any.whl

MD5:c6ce4e25f7fe3e3bb1eea2e9052483bf

File name: xinference/__init__.py

SHA256: e1e007ce4eab7774785617179d1c01a9381ae83abfd431aae8dba6f82d3ac127

After one decoding SHA256:

077d49fa708f498969d7cdffe701eb64675baaa4968ded9bd97a4936dd56c21c

SHA256 after secondary decoding:

fe17e2ea4012d07d90ecb7793c1b0593a6138d25a393192263e751660ec3cd0

Temporary archive file: love.tar.gz

Text tag string: #hacked by teampcp

Malicious domain name

whereisitat.lucyatemysuperbox.space

Malicious URL

hxxps://whereisitat.lucyatemysuperbox.space

Others

HTTP header: X-QT-SR: 14

Statement

This advisory is only used to describe a potential risk. NSFOCUS does not provide any commitment or promise on this advisory. NSFOCUS and the author will not bear any liability for any direct and/or indirect consequences and losses caused by transmitting and/or using this advisory. NSFOCUS reserves all the rights to modify and interpret this advisory. Please include this statement paragraph when reproducing or transferring this advisory. Do not modify this advisory, add/delete any information to/from it, or use this advisory for commercial purposes without permission from NSFOCUS.

NSFOCUS, a pioneering leader in cybersecurity, is dedicated to safeguarding telecommunications, Internet service providers, hosting providers, and enterprises from sophisticated cyberattacks.

Founded in 2000, NSFOCUS operates globally with over 3000 employees at two headquarters in Beijing, China, and Santa Clara, CA, USA, and over 50 offices worldwide. It has a proven track record of protecting over 25% of the Fortune Global 500 companies, including four of the five largest banks and six of the world's top ten telecommunications companies.

Leveraging technical prowess and innovation, NSFOCUS delivers a comprehensive suite of security solutions, including the Intelligent Security Operations Platform (ISOP) for modern SOC, DDoS Protection, Continuous Threat Exposure Management (CTEM) Service and Web

Application and API Protection (WAAP). All the solutions and services are augmented by the Security Large Language Model (SecLLM), ML, patented algorithms and other cutting-edge research achievements developed by NSFOCUS.