### **Vulnerability Name:**

Unauthorized Arbitrary Operations in Xinference Web Service

### **Vendor:**

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

### **Introduction to Xinference**

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

GitHub: [https://github.com/xorbitsai/inference](https://github.com/xorbitsai/inference" \t "https://www.doubao.com/chat/_blank)  
Official Documentation: [https://inference.readthedocs.io/zh](https://inference.readthedocs.io/zh" \t "https://www.doubao.com/chat/_blank) - cn/latest/index.html

### **Version:**

All current versions

### **Risk and Hazard:**

When Xinference is deployed in the way of "xinference - local --host 0.0.0.0 --port 9997" (this port can be customized) with the default configuration, attackers can access the Web GUI interface without authorization and perform arbitrary operations such as viewing, downloading, deploying, running, and deleting models through interface operations.

### **Vulnerability Reproduction**

1. Deploy Xinference

bash

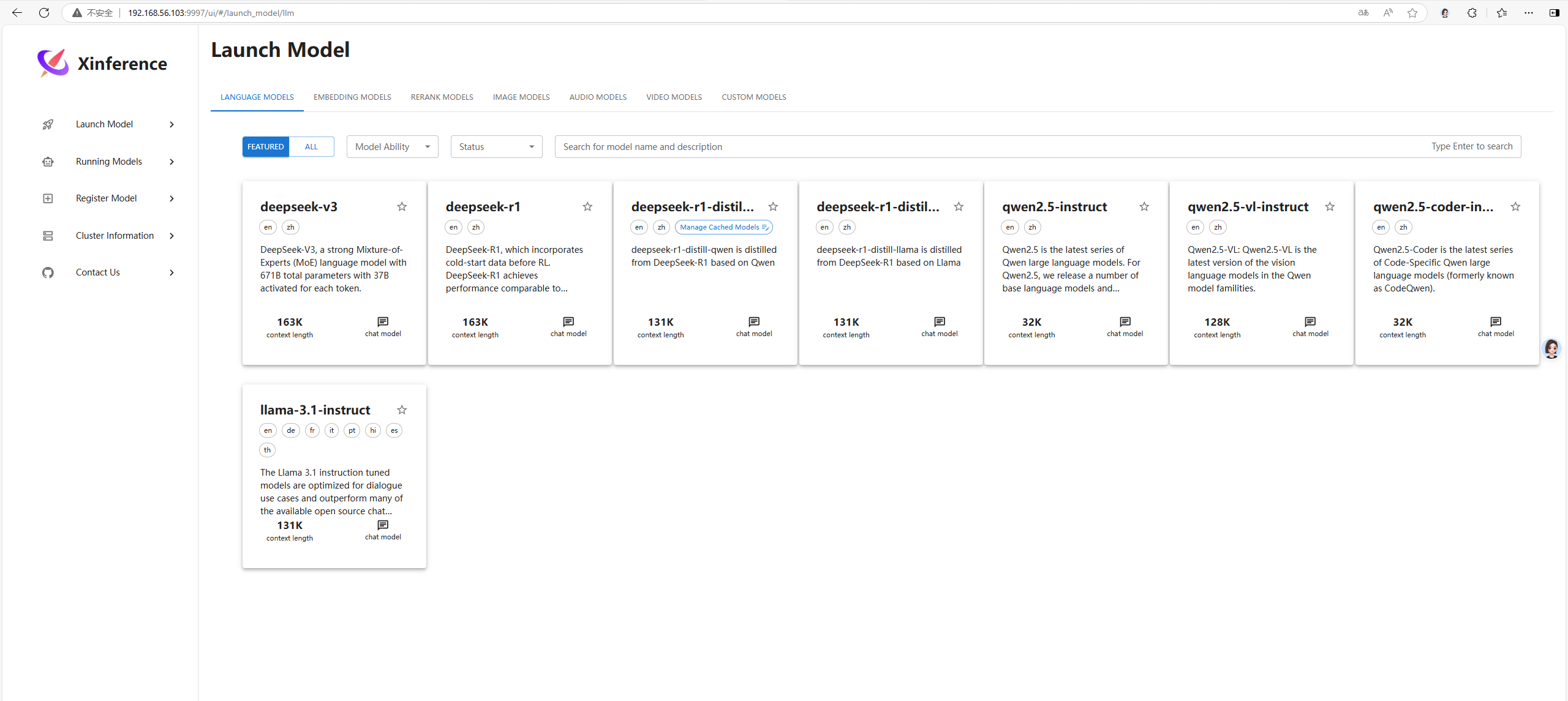
pip install "xinference[all]"

1. Start Xinference

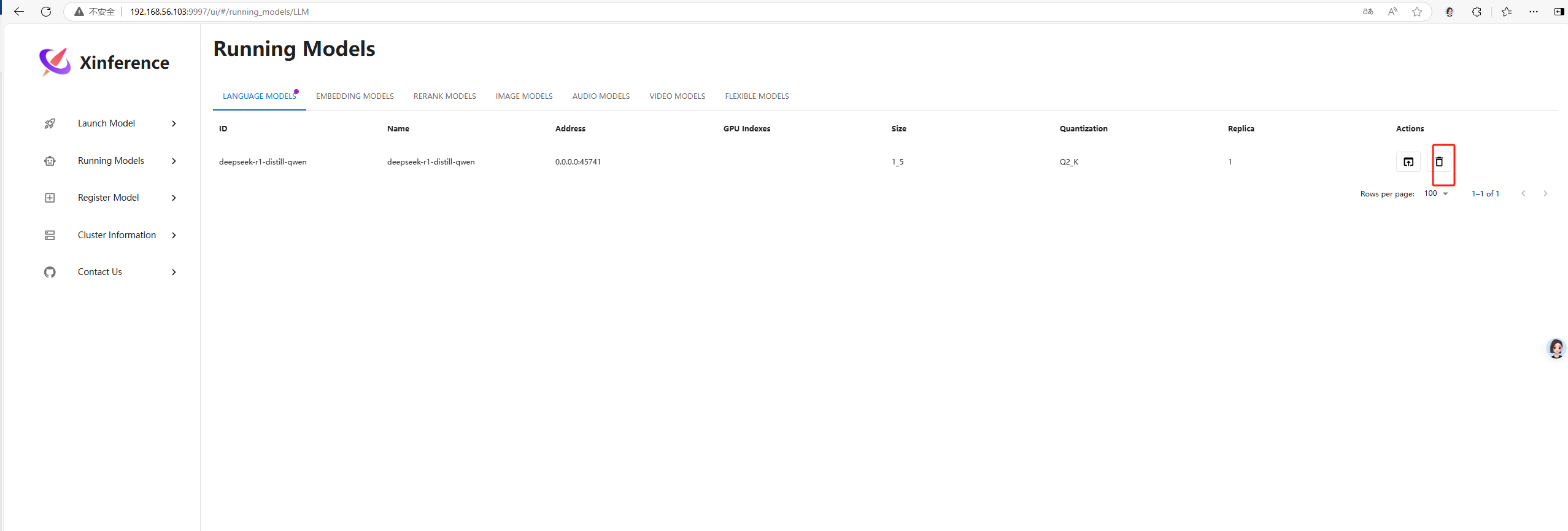
bash

xinference - local --host 0.0.0.0 --port 9997

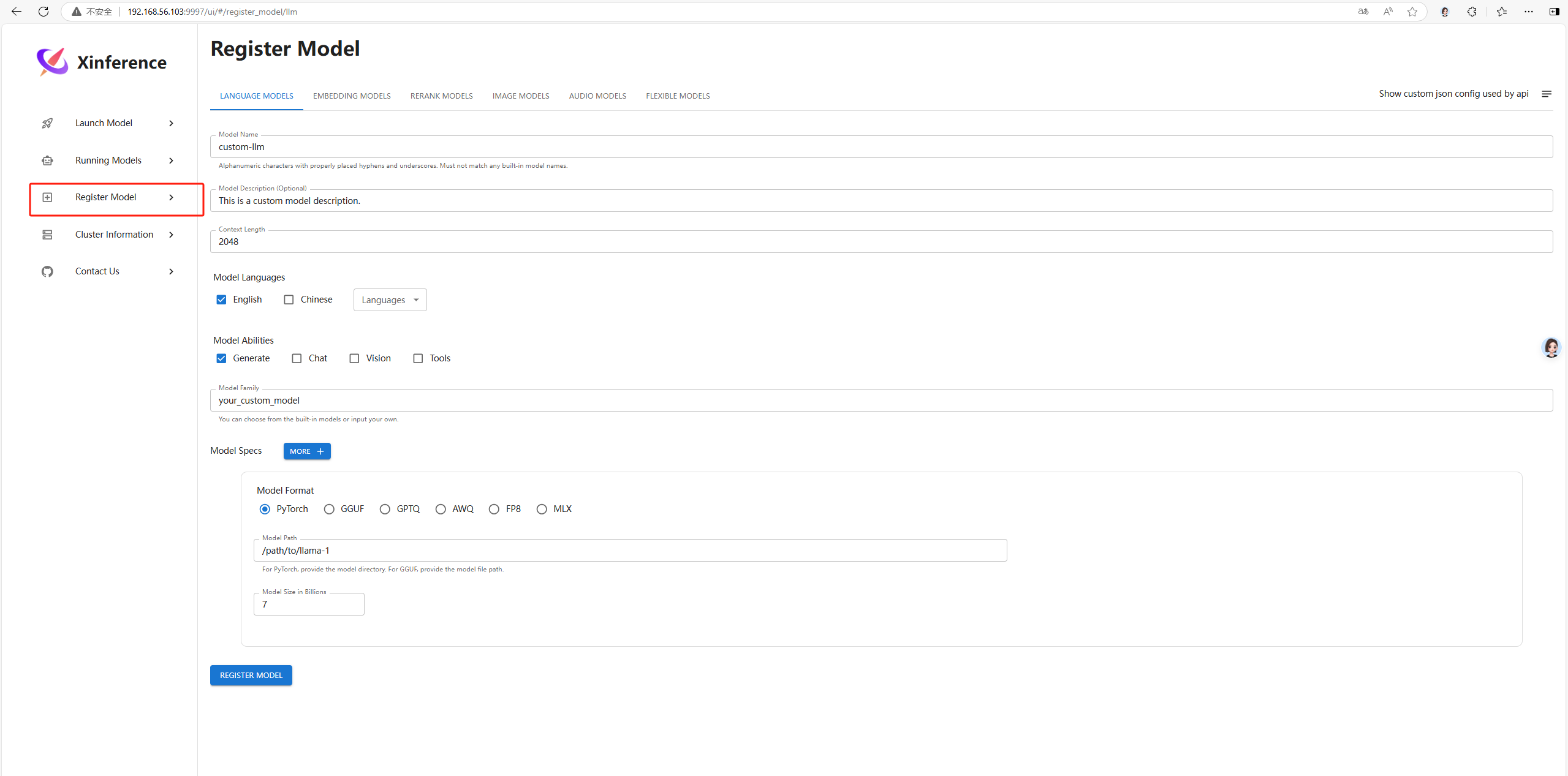
1. Access the Xinference interface
2. Pull and deploy models. On the Launch Model interface, the attacker can click a model and complete a simple configuration to pull any model from the Magic Tower community and complete the installation.



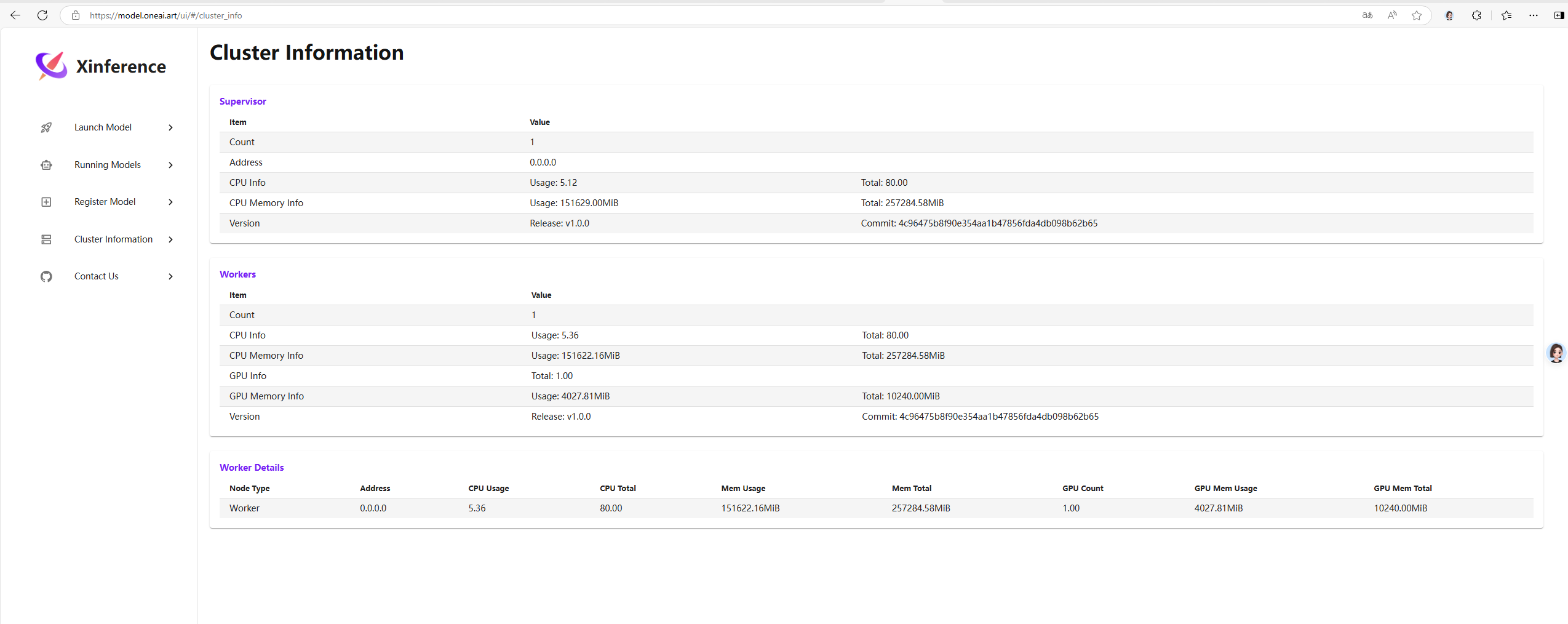
1. Delete the model. On the "Running models" interface, attackers can choose to delete the running models.



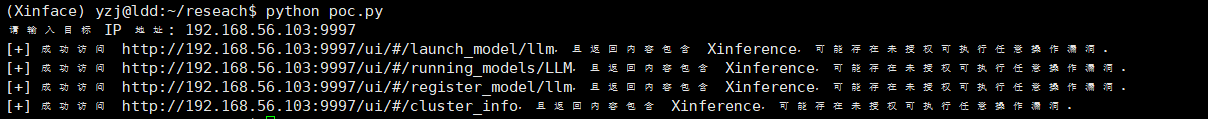
1. Modify the model configuration. On the "Register Model" interface, attackers can modify the relevant configurations or create new ones.



1. View sensitive cluster information. Attackers can view cluster resource information in "Cluster Information".



1. Run the poc.py script of Python and then enter the relevant IP address and port.



### **POC:**

import requests

def check\_url(ip, path):

url = f"http://{ip}{path}"

try:

response = requests.get(url)

if response.status\_code == 200:

if response.text.strip() and "Xinference" in response.text:

print(f"[+] 成功访问 {url}，且返回内容包含 Xinference，可能存在未授权可执行任意操作漏洞。")

return True

else:

print(f"[-] 成功访问 {url}，但返回内容不符合预期。")

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\_\_":

ip = input("请输入目标 IP 地址及端口: ")

paths = [

"/ui/#/launch\_model/llm",

]

for path in paths:

check\_url(ip, path)

### **Asset mapping:**

Search for "title=Xinference".

Approximately 1,300 relevant pieces of information can be obtained.

For example:

http://59.80.34.141:10000/ui/#/launch\_model/llm

https://model.oneai.art/ui/#/cluster\_info

etc.