CVE-2025-1550

최준원

Contents

01 개요

02 환경구성

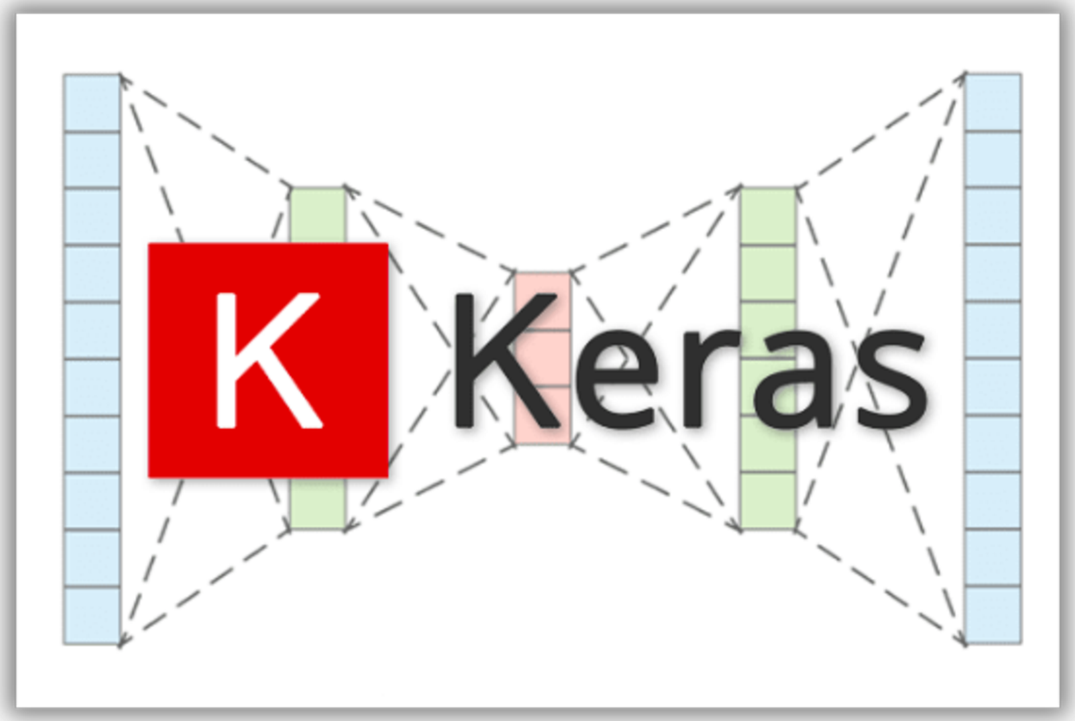
03 시나리오

04 결과

05 대응방안

06 개선사항

01 개요



CVE-2025-1550

신뢰되지 않은 .keras 모델 파일을 로드할 때, 내부 설정을 검증하지 않고 역직렬화하여 RCE가 가능한 취약점

01 개요

발생원인

Keras.models.load_model() 함수가 외부 입력을 신뢰하고 모델 구조를 역직렬화하는 과정에서 모듈 및 클래스 이름을 검증 없이 동적 로딩하면서 해당 취약점 발생

공격자가 조작된 .keras 파일 내부에 악성 모듈을 삽입할 수 있음

01 개요

취약한 버전 정보

Keras

- 초기 권고 : Keras v3계열(3.0.0 이상, 3.9.0미만)이 취약하다고 보고됨
- 추가 권고 : 이후 보안 공지에서 우회 사례가 보고되어 영향범위가 확장(3.11.0미만)된 사례가 보고됨

Tensorflow

- tf.keras를 사용하는 애플리케이션도 내부에 Keras 역직렬화 로직을 포함할 수 있으므로, TensorFlow 배포판에 포함된 Keras 코드/버전을 확인해야함

NAME	VER
Keras	3.9.0 > 3.0.0
tensorflow	2.19.0 >= 2.10.0

02 환경구성

VICTIM

NAME	VER
Ubuntu	22.04(WSL2)
Keras	3.8.0
tensorflow	2.19.0
Flask	3.1.0

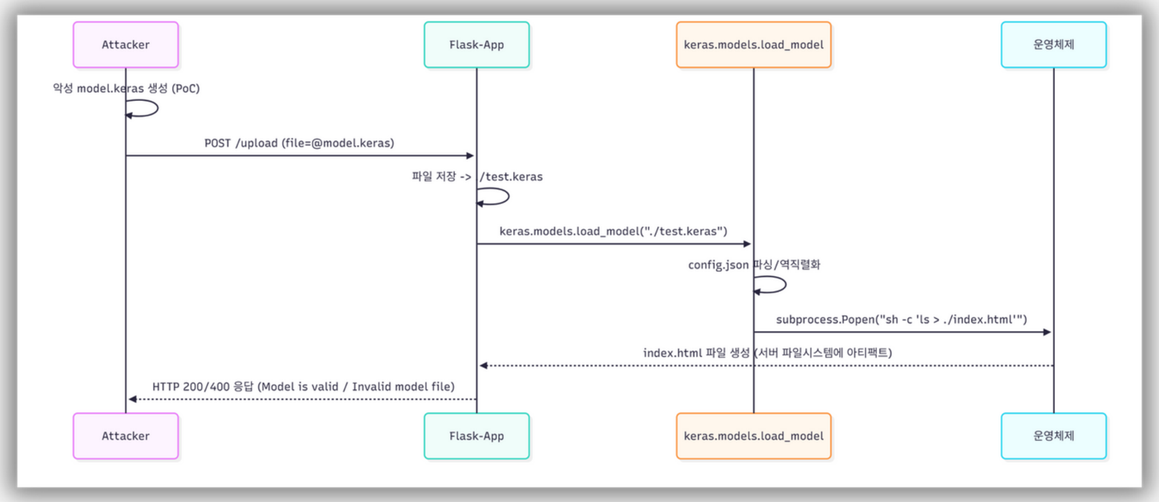
02 환경구성

ATTACKER

NAME	VER
Ubuntu	20.04(WSL2)
Keras	3.8.0
tensorflow	2.19.0

03 시나리오

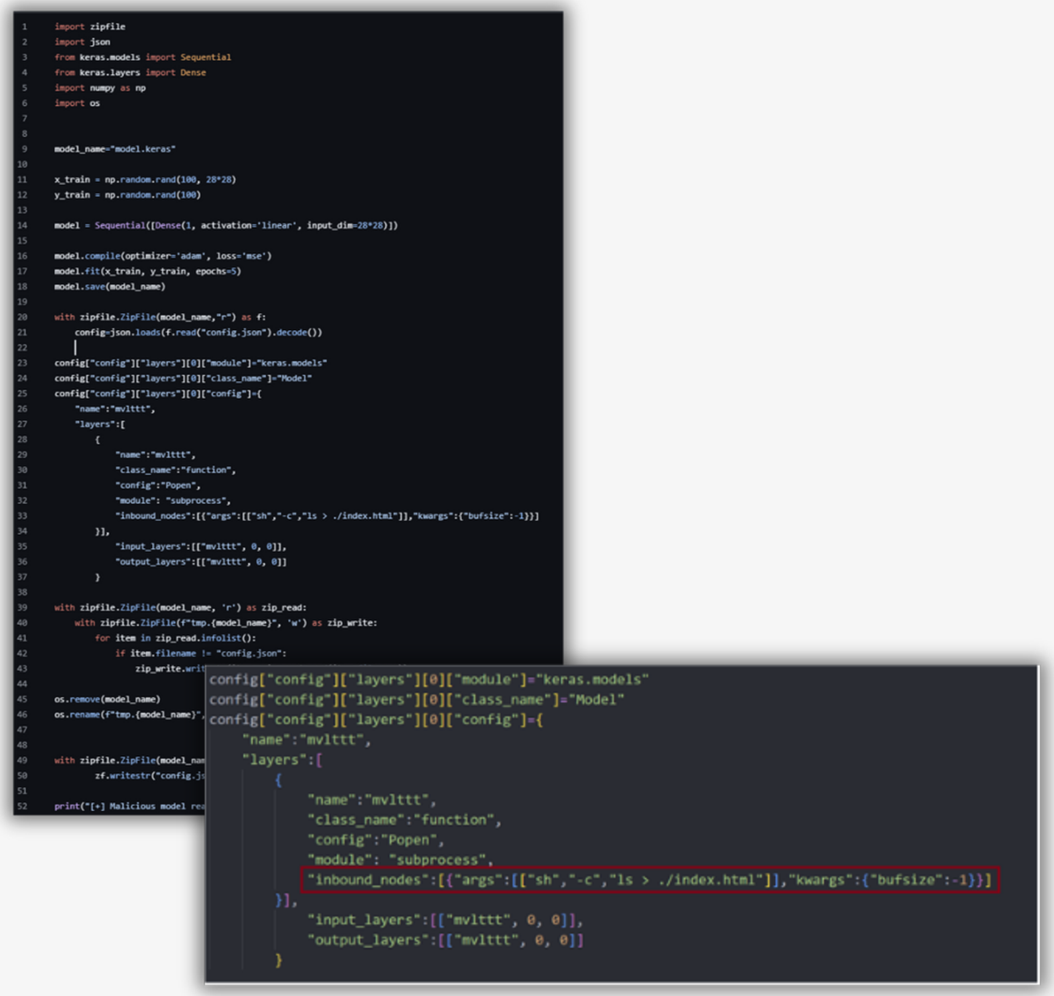
1. 공격자PC에서 PoC를 실행하여 악성 model.keras 아카이브 모델 생성
2. VICTIM 서버 준비
3. 공격자는 피해자 서버에 악성 Keras 모델 업로드
4. 서버는 업로드 파일을 ./test.keras로 저장하고 is_valid_model()함수에서 load_model()을 호출하여 모델 검증 시도
5. load_model()이 config.json을 역직렬화/파싱하는 중 악성 모듈에 의해 악성 스크립트 호출



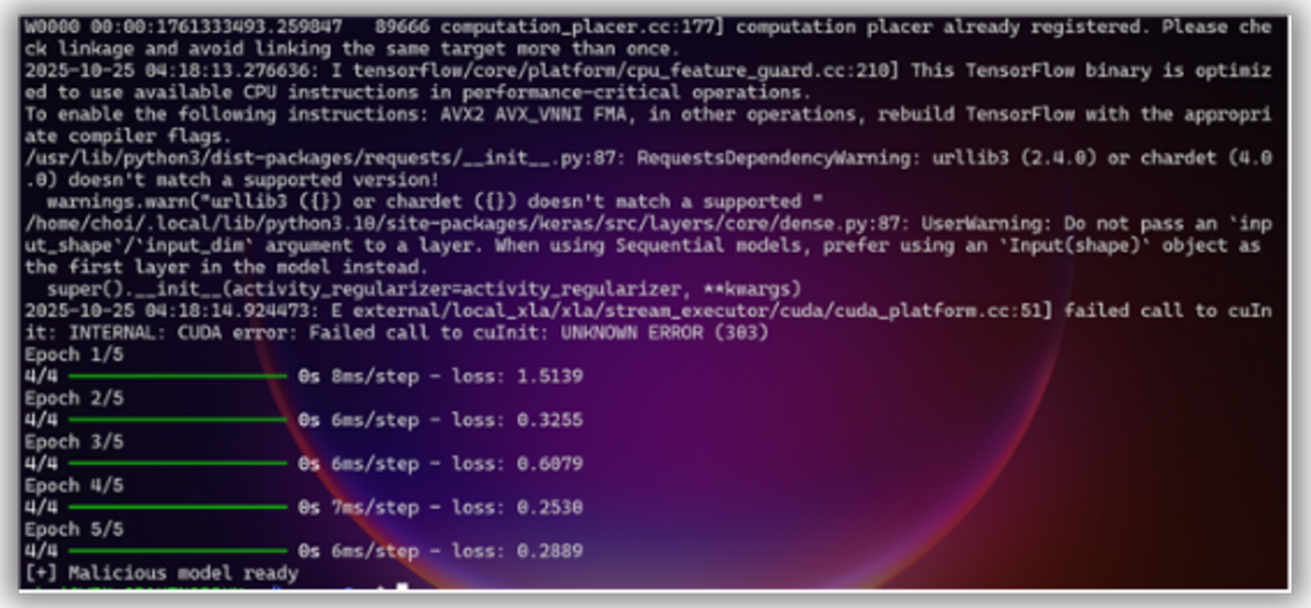
04 결과

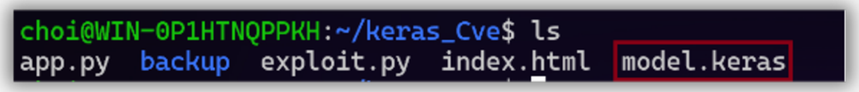
악성 모델 생성

ls 결과를 index.html에 덮어씌우는 악성 스크립트를 삽입한 악성 keras 아카이브 모델을 공격자PC에서 생성



04 결과


W0000 00:00:1761333493.259047 89666 computation_placer.cc:177] computation placer already registered. Please check linkage and avoid linking the same target more than once.
2025-10-25 04:18:13.276636: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 AVX_VNNI FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
/usr/lib/python3/dist-packages/requests/__init__.py:87: RequestsDependencyWarning: urllib3 (2.4.0) or chardet (4.0.0) doesn't match a supported version!
warnings.warn("urllib3 ({}) or chardet ({}) doesn't match a supported "
/home/choi/.local/lib/python3.10/site-packages/keras/src/layers/core/dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
2025-10-25 04:18:14.924473: E external/local_xla/xla/stream_executor/cuda/cuda_platform.cc:51] failed call to cuInit: INTERNAL: CUDA error: Failed call to cuInit: UNKNOWN ERROR (303)
Epoch 1/5
4/4 0s 8ms/step - loss: 1.5139
Epoch 2/5
4/4 0s 6ms/step - loss: 0.3255
Epoch 3/5
4/4 0s 6ms/step - loss: 0.6079
Epoch 4/5
4/4 0s 7ms/step - loss: 0.2530
Epoch 5/5
4/4 0s 6ms/step - loss: 0.2889
[+] Malicious model ready

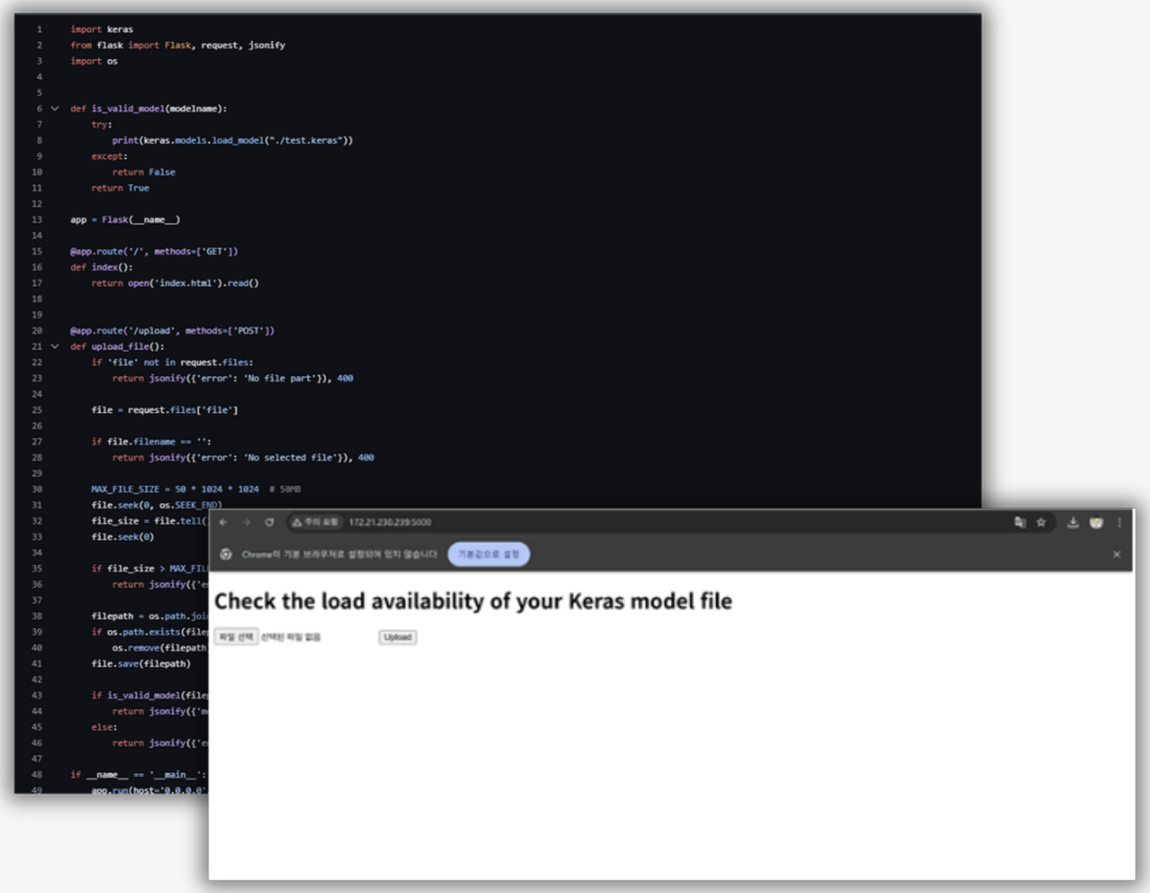

choi@WIN-0P1HTNQPPKH:~/keras_Cve$ ls
app.py backup exploit.py index.html model.keras

04 결과

VICTIM 서버 준비

파일을 업로드하고 .Keras모델을 Safe_mode=True 상태에서 검증하는 피해자측 서버 준비

피해자 서버는 .keras 파일 업로드시 test.keras 라는 파일로 해당 파일을 저장하고 저장직후 test.keras 파일을 로드 한다.

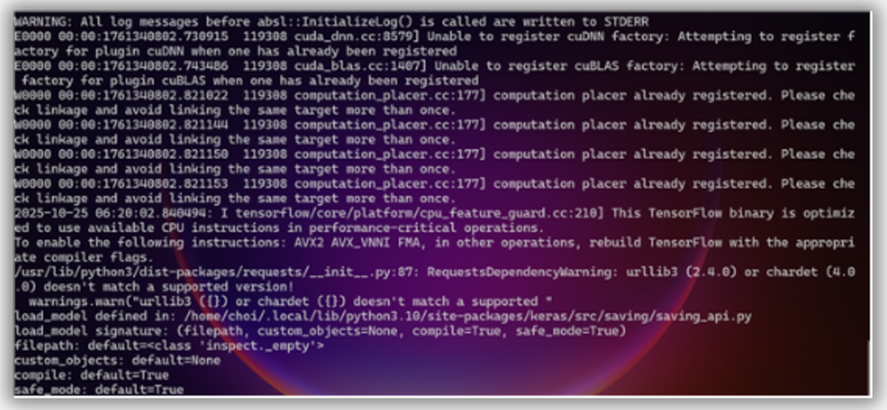


04 결과

Safe_mode 체크

```
choi@WIN-0P1HTNQPPKH:~/keras_Cve$ python3 - <<'PY'
import inspect
from importlib import import_module

try:
    mod = import_module('keras.models')
    load_model = getattr(mod, 'load_model')
    print('load_model defined in:', inspect.getsourcefile(load_model))
    sig = inspect.signature(load_model)
    print('load_model signature:', sig)
    for name, param in sig.parameters.items():
        print(f'{name}: default={param.default!r}')
except Exception as e:
    print('Error:', e)
PY
```

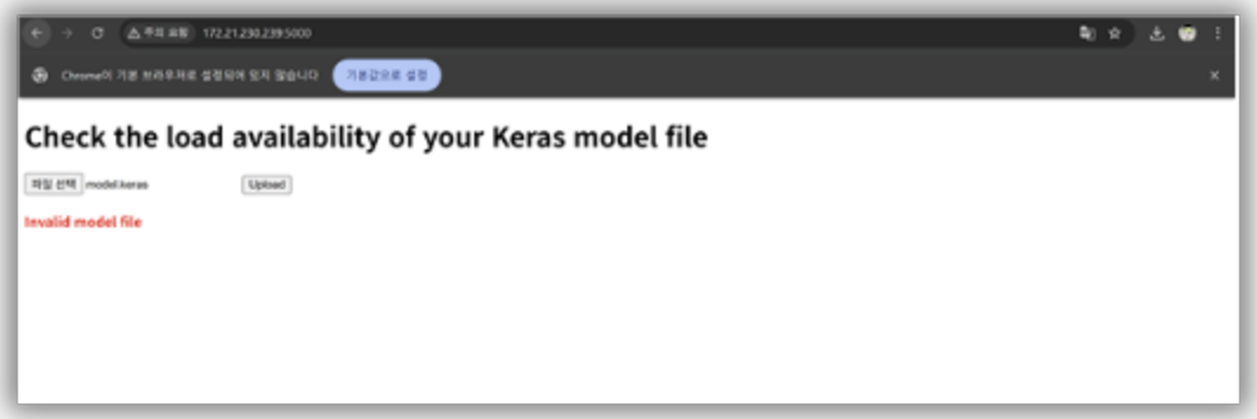


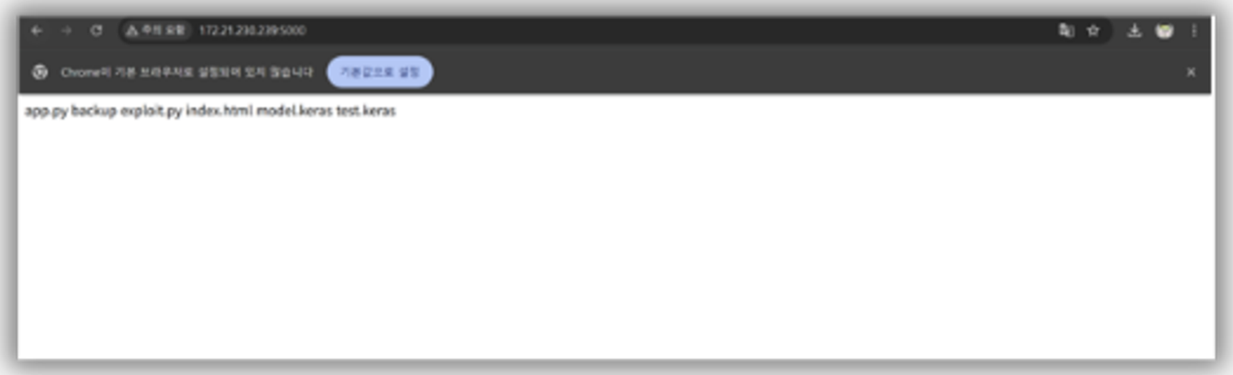
```
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1761340802.730915  119308 cuda_dnn.cc:8579] Unable to register cuDNN factory: Attempting to register f
actory for plugin cuDNN when one has already been registered
E0000 00:00:1761340802.743486  119308 cuda_blas.cc:1407] Unable to register cuBLAS factory: Attempting to register
 factory for plugin cuBLAS when one has already been registered
W0000 00:00:1761340802.821022  119308 computation_placer.cc:177] computation placer already registered. Please che
ck linkage and avoid linking the same target more than once.
W0000 00:00:1761340802.821144  119308 computation_placer.cc:177] computation placer already registered. Please che
ck linkage and avoid linking the same target more than once.
W0000 00:00:1761340802.821150  119308 computation_placer.cc:177] computation placer already registered. Please che
ck linkage and avoid linking the same target more than once.
W0000 00:00:1761340802.821153  119308 computation_placer.cc:177] computation placer already registered. Please che
ck linkage and avoid linking the same target more than once.
2025-10-25 06:20:02.840494: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimiz
ed to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 AVX_VNNI FMA, in other operations, rebuild TensorFlow with the appropri
ate compiler flags.
/usr/lib/python3/dist-packages/requests/__init__.py:87: RequestsDependencyWarning: urllib3 (2.4.0) or chardet (4.0
.0) doesn't match a supported version!
  warnings.warn("urllib3 ({}) or chardet ({}) doesn't match a supported "
load_model defined in: /home/choi/.local/lib/python3.10/site-packages/keras/src/saving/saving_api.py
load_model signature: (filepath, custom_objects=None, compile=True, safe_mode=True)
filepath: default=<class 'inspect._empty'>
custom_objects: default=None
compile: default=True
safe_mode: default=True
```

04 결과

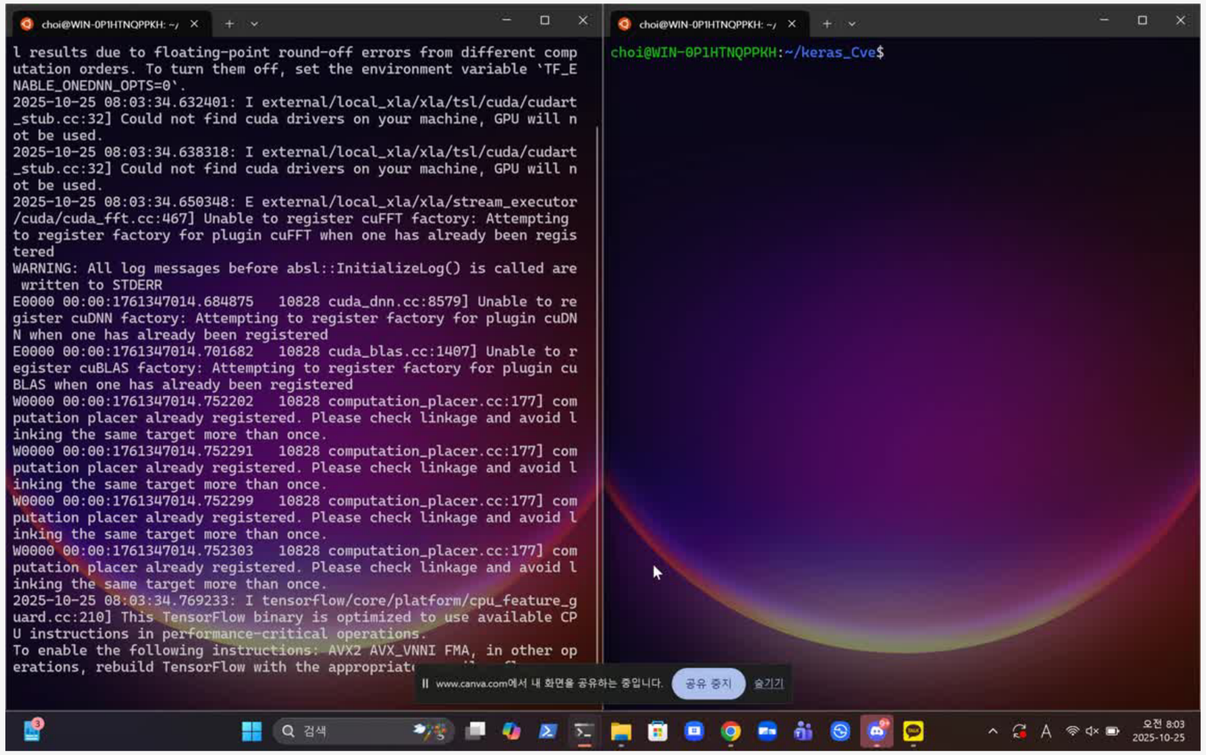
악성 모델 업로드

악성 모델을 서버에 업로드





04 결과


choi@WIN-0P1HTNQPPKH: ~/
l results due to floating-point round-off errors from different comp
utation orders. To turn them off, set the environment variable 'TF_E
NABLE_ONEDNN_OPTS=0'.
2025-10-25 08:03:34.632401: I external/local_xla/xla/tsl/cuda/cudart
_stub.cc:32] Could not find cuda drivers on your machine, GPU will n
ot be used.
2025-10-25 08:03:34.638318: I external/local_xla/xla/tsl/cuda/cudart
_stub.cc:32] Could not find cuda drivers on your machine, GPU will n
ot be used.
2025-10-25 08:03:34.650348: E external/local_xla/xla/stream_executor
/cuda/cuda_fft.cc:467] Unable to register cuFFT factory: Attempting
to register factory for plugin cuFFT when one has already been regis
tered
WARNING: All log messages before absl::InitializeLog() is called are
written to STDERR
E0000 00:00:1761347014.684875 10828 cuda_dnn.cc:8579] Unable to re
gister cuDNN factory: Attempting to register factory for plugin cuDN
N when one has already been registered
E0000 00:00:1761347014.701682 10828 cuda_blas.cc:1407] Unable to r
egister cuBLAS factory: Attempting to register factory for plugin cu
BLAS when one has already been registered
W0000 00:00:1761347014.752202 10828 computation_placer.cc:177] com
putation placer already registered. Please check linkage and avoid l
inking the same target more than once.
W0000 00:00:1761347014.752291 10828 computation_placer.cc:177] com
putation placer already registered. Please check linkage and avoid l
inking the same target more than once.
W0000 00:00:1761347014.752299 10828 computation_placer.cc:177] com
putation placer already registered. Please check linkage and avoid l
inking the same target more than once.
W0000 00:00:1761347014.752303 10828 computation_placer.cc:177] com
putation placer already registered. Please check linkage and avoid l
inking the same target more than once.
2025-10-25 08:03:34.769233: I tensorflow/core/platform/cpu_feature_g
uard.cc:210] This TensorFlow binary is optimized to use available CP
U instructions in performance-critical operations.
To enable the following instructions: AVX2 AVX_VNNI FMA, in other op
erations, rebuild TensorFlow with the appropriat
choi@WIN-0P1HTNQPPKH: ~/
choi@WIN-0P1HTNQPPKH:~/keras_Cve$
www.canva.com에서 내 화면을 공유하는 중입니다.
공유 중지
숨기기
검색
오전 8:03
2025-10-25

05 대응방안

외부에서 업로드되는 ML모델 파일은 잠재적 역직렬화 코드 실행 위협을 포함할 수 있으므로, 자동 로딩을 중지하고 서명,정적검사,격리 실행을 결합한 다중 방어를 적용해야 한다. 또한 업계 권고에 따른 Keras/Tensorflow 보안 패치를 신속히 적용하고, 업로드 및 프로세스 생성 이벤트에 대한 탐지 룰을 운영하여 유사 사건을 조기에 포착하도록 권고된다.

```python
import zipfile, json

BLACK_MODULES = {"subprocess","os","pickle","importlib","builtins","ctypes"}
def is_safe_keras(path):
    with zipfile.ZipFile(path, 'r') as z:
        if "config.json" not in z.namelist():
            return False
        cfg = json.loads(z.read("config.json").decode())
    def scan(obj):
        if isinstance(obj, dict):
            for k,v in obj.items():
                if k == "module" and isinstance(v, str) and any(b in v for b in BLACK_MODULES):
                    return False, f"disallowed module {v}"
                ok = scan(v)
                if ok is not True:
                    return ok
        elif isinstance(obj, list):
            for item in obj:
                ok = scan(item)
                if ok is not True:
                    return ok
        return True
    return scan(cfg)
```

Thank you

감사합니다.