

huggingface / lerobot Public[Code](#) [Issues 343](#) [Pull requests 305](#) [Actions](#) [Security and quality](#) [Ins](#)[New issue](#)

Security: Unsafe pickle deserialization in async inference enables Remote Code Execution (CWE-502) #3047

[Open](#)[#3048](#)

Labels

[configuration](#)[dependencies](#)[policies](#)[processor](#)[tests](#)

Chocapikk opened on Feb 27 · edited by Chocapikk

Edits ▾ ⋮

Summary

The async inference pipeline (`policy_server.py` and `robot_client.py`) uses `pickle.loads()` to deserialize data received over unauthenticated gRPC channels. This allows any network-reachable attacker to execute arbitrary code on the server (or client) by sending a crafted pickle payload.

Affected Code

All `pickle.loads()` calls in:

- `src/lerobot/async_inference/policy_server.py` (lines 127, 185, 238)
- `src/lerobot/async_inference/robot_client.py` (line 281)

The developers acknowledged the risk with `# nosec` comments but no mitigation was implemented.

Vulnerability Details

- Type:** CWE-502 - Deserialization of Untrusted Data
- Impact:** Unauthenticated Remote Code Execution
- Protocol:** gRPC (HTTP/2) over `add_insecure_port()` - no TLS, no authentication

Attack Vectors

1. **SendPolicyInstructions** (unary RPC): Attacker sends a malicious pickle as `PolicySetup.data`. The server calls `pickle.loads(request.data)` before any type validation.
2. **SendObservations** (streaming RPC): Attacker sends a malicious pickle as chunked `Observation.data`. The server reassembles and calls `pickle.loads(received_bytes)`.
3. **GetActions** (bidirectional): A compromised server sends a malicious pickle as `Actions.data`. The client calls `pickle.loads(actions_chunk.data)`.

Root Cause

The gRPC service uses raw `bytes` fields in protobuf messages to transport Python objects serialized with `pickle`. The `pickle.loads()` calls happen **before** any validation, and there is no authentication on the gRPC channel.

Proof of Concept

```
import os, pickle, grpc
from lerobot.transport import services_pb2, services_pb2_grpc

class RCE:
    def __reduce__(self):
        return (os.system, ("id > /tmp/pwned",))

channel = grpc.insecure_channel("TARGET:8080")
stub = services_pb2_grpc.AsyncInferenceStub(channel)
stub.Ready(services_pb2.Empty())
# RCE executes during pickle.loads(), before isinstance() check
stub.SendPolicyInstructions(services_pb2.PolicySetup(data=pickle.dumps(RCE())))
```



Tested against `lerobot` installed from the current main branch.

Suggested Fix

Replace `pickle` serialization with `safetensors` + JSON:

- **safetensors** (already a project dependency) for tensor data - designed specifically as a safe alternative to pickle for ML tensors
- **JSON** for scalar metadata and configuration (policy config, timestamps, etc.)

I have a PR ready with this fix: it introduces a `safe_serialization.py` module that handles all serialization without any use of pickle, with full roundtrip tests including a security regression test.

Prior Reports

A separate security report was submitted via [#2745](#) and through the Security tab, but its contents are not publicly visible so it is unclear whether it covers the same vulnerability. The maintainer acknowledged in [this comment](#) that the codebase "does pose a security risk" and needs refactoring, but no fix has been issued.

Timeline

- **2026-02-11:** Vulnerability independently discovered, confirmed via PoC, and reported to VulnCheck for CVE assignment
- **2026-02-27:** Public issue created with fix

Researcher

- **Name:** Valentin Lobstein
- **GitHub:** [@Chocapikk](#)



github-actions added **configuration** **dependencies** **policies** **processor** **tests** on Feb 27



Chocapikk linked a pull request that will close this issue on Feb 27

[Fix: Replace unsafe pickle with safetensors + JSON in async inference #3048](#)



Chocapikk on Feb 27

Author ...

The fix PR is [#3048](#) - currently WIP/draft while I validate the tests locally. Will mark it ready for review once everything passes.



imstevenpmwork on Feb 27

Collaborator ...

[#2745 \(comment\)](#)



Chocapikk on Feb 27 · edited by Chocapikk

Edits ▾ Author ...

My PR has already made everything public. I'm now focusing on getting the fix tested and ready for review.



Chocapikk on Mar 9

Author ...

Hi [@imstevenpmwork](#), friendly ping on this. PR [#3048](#) should be ready for review, it replaces all `pickle.loads()` calls with `safetensors` + JSON, with full roundtrip tests and a security regression test. Would be great to get some eyes on it when you get a chance.



[imstevenpmwork](#) mentioned this [3 weeks ago](#)

[Release 0.6.0 #3134](#)

[Sign up for free](#) to join this conversation on **GitHub**. Already have an account? [Sign in to comment](#)

Metadata

Assignees

No one assigned

Labels

- configuration
- dependencies
- policies
- processor
- tests

Type

No type

Fields

[Give feedback](#)

No fields configured for issues without a type.

Projects

No projects

Milestone

No milestone

Relationships

None yet

Development

[🔗](#) **Fix: Replace unsafe pickle with safetensors + JSON in async inference**

[huggingface/lerobot](#)

Participants

