What is HDF5 external storage file read?

HDF5 external raw storage file read is an arbitrary-file-read bug that abuses a legitimate HDF5 feature: a dataset's raw bytes can be stored in an external file referenced by path. An attacker crafts an .h5 or .hdf5 file whose dataset points at a local path such as `/etc/passwd` or a traversal path. When a victim application opens the file and reads the dataset, the HDF5 library reads that external file and hands back its contents. It matters most for ML pipelines that load untrusted HDF5 models or datasets. The fix is to reject or restrict external storage on untrusted files.

HOW IT WORKS

01 How it works and example

The attacker builds an HDF5 file with a dataset configured for external storage (the `H5Pset_external` property, exposed in h5py as `Dataset.external`) whose path is an absolute path like `/etc/passwd` or a traversal path such as `../../../../etc/passwd`. They send that file to any surface that parses HDF5: a model upload, a dataset import, a conversion API. When the app reads the dataset, for example `f['data'][:]` in h5py or a model load, the HDF5 library opens the referenced local file and returns its bytes through whatever exposes the dataset, an API response, an error message, or model weights. This is a file read, not code execution, but it leaks secrets, source, keys, and configuration. Examples are shown for defensive testing.

SOURCES

[1] OWASP Web Security Testing Guide

[2] MITRE CWE-610: Externally Controlled Reference to a Resource

[3] HDF5 external storage (H5Pset_external)

[4] MITRE CWE-73: External Control of File Name or Path

Test your file parsers and ML pipeline for this and 30 or more other classes.

`securelayer7.net/learn/application-security/hdf5-external-storage-file-read`

Open online