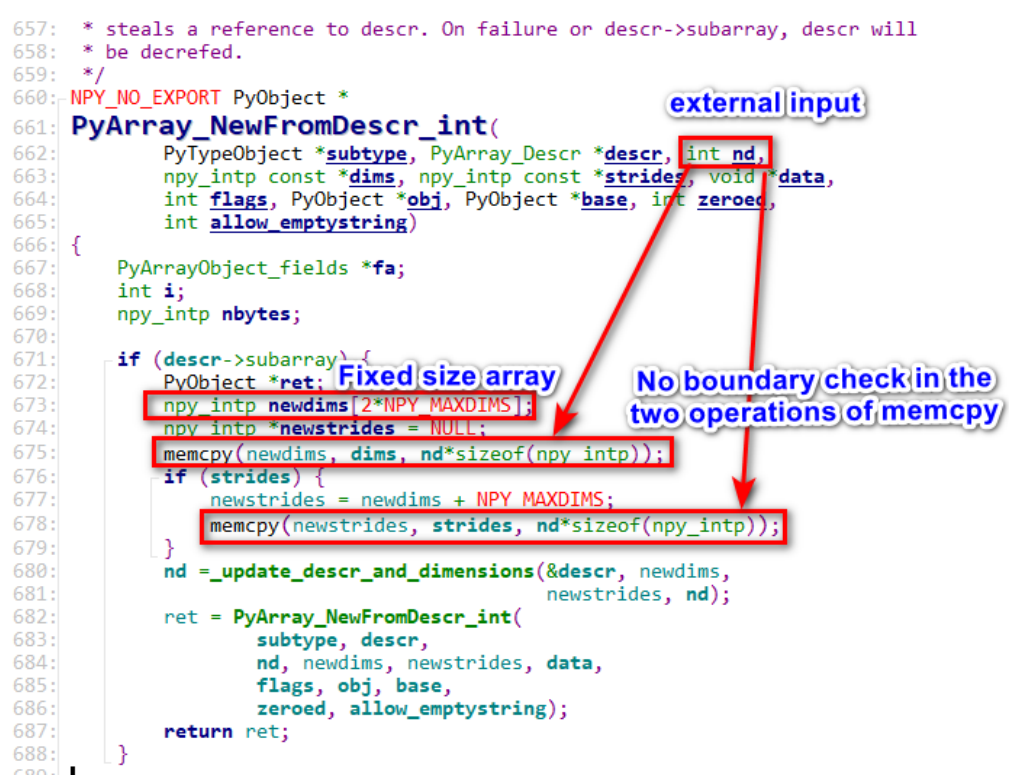
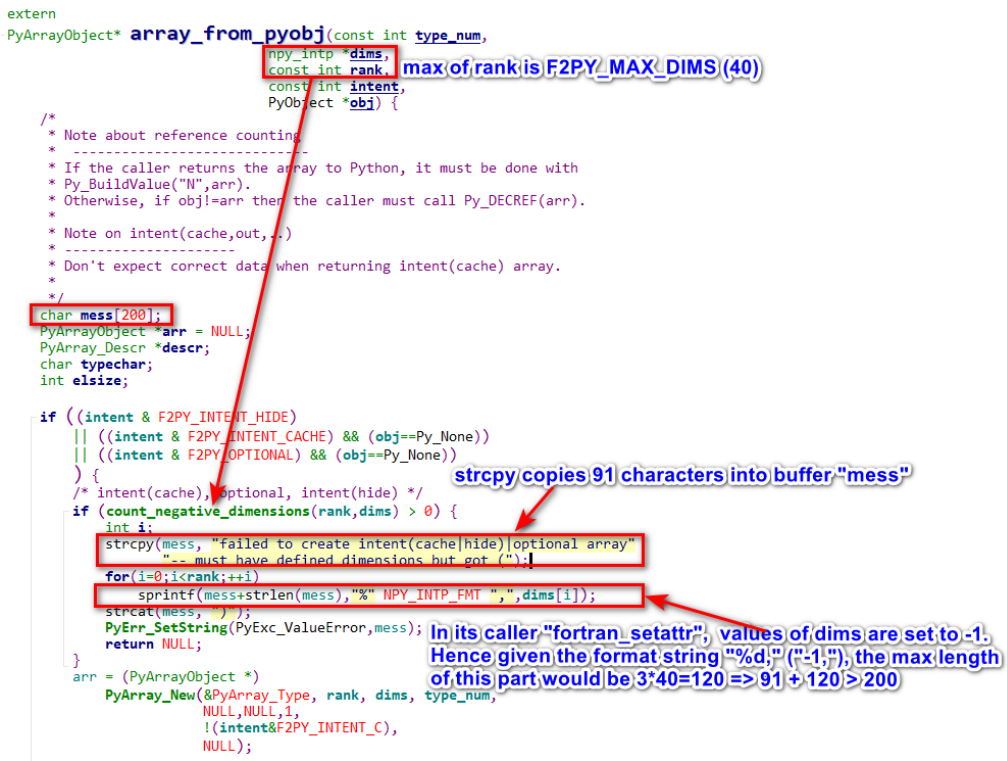
Benchmark	CVES	#CVE
Numpy	CVE-2021-33430, CVE-2021-41495, CVE-2021-41496, CVE-2021-34141	4
Bounter	CVE-2021-41497	1
Cvxopt	CVE-2021-41500	1
Pyo	CVE-2021-41498, CVE-2021-41499	2
	Total	8

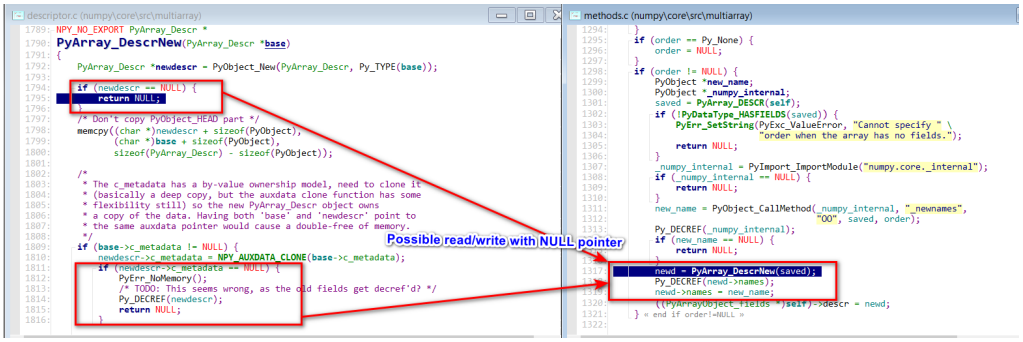
Subject	NumPy
Vulnerability Type	Buffer overflow
Input	Integration test
Description	 Affected component: Affect APIs for operations on the high-dimensional array

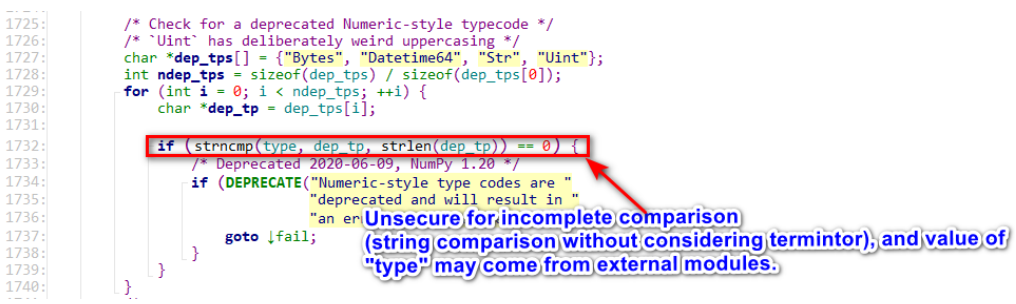
```
 * steals a reference to descr. On failure or descr->subarray, descr will
 * be decrefed.
 */
NPY_NO_EXPORT PyObject *
PyArray_NewFromDescr_int(
        PyTypeObject *subtype, PyArray_Descr *descr, int nd,
        npy_intp const *dims, npy_intp const *strides, void *data,
        int flags, PyObject *obj, PyObject *base, int zeroed,
        int allow_emptystring)
{
    PyArrayObject_fields *fa;
    int i;
    npy_intp nbytes;

    if (descr->subarray) {
        PyObject *ret;
        npy_intp newdims[2*NPY_MAXDIMS];
        npy_intp *newstrides = NULL;
        memcpy(newdims, dims, nd*sizeof(npy_intp));
        if (strides) {
            newstrides = newdims + NPY_MAXDIMS;
            memcpy(newstrides, strides, nd*sizeof(npy_intp));
        }
        nd = _update_descr_and_dimensions(&descr, newdims,
                                          newstrides, nd);
        ret = PyArray_NewFromDescr_int(
                subtype, descr,
                nd, newdims, newstrides, data,
                flags, obj, base,
                zeroed, allow_emptystring);
        return ret;
    }
```

	Attack vector: Construct a high-dimensional array file (npy); when the dimension is larger than 32, Python will crash when loading the array from the file. Buffer overflow in the operation of the high-dimensional array in NumPy (< 1.19) allows attackers to conduct DoS attacks by carefully constructing a npy file.
Taint flow path	numpy.load (Python) -> PyArray_Fromfile -> PyArray_NewFromDescr_int -> memcpy
PoC	https://github.com/baltsers/polycruise/tree/main/numpy/vulnerability-1
Status	Confirmed and fixed, **CVE assigned: CVE-2021-33430**
Issue	https://github.com/numpy/numpy/issues/18939

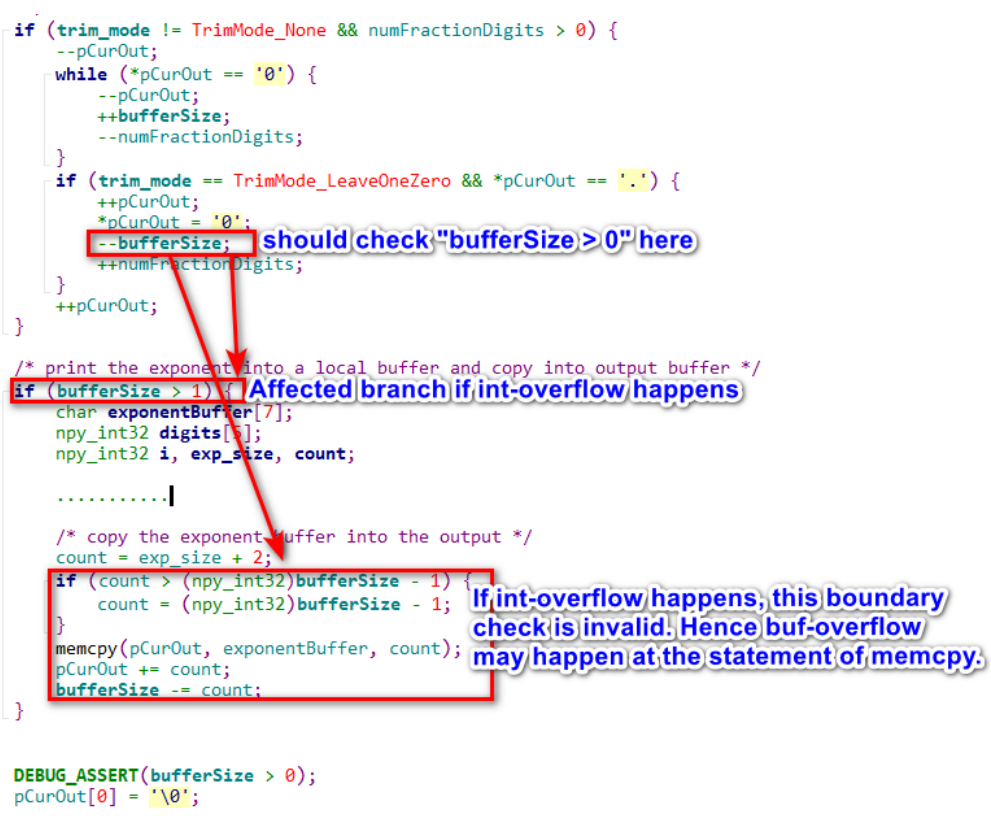
Subject	Numpy
Vulnerability Type	Buffer overflow
Input	Integration test
Description	 In the f2py module, when creating a high-dimension array through the **array API** (deliberately construct negative numbers in shape), an unexpected error

	occurs and causes Python to crash down. This allows attackers to conduct DoS attacks by carefully constructing an array with negative values in shape.
Taint flow path	numpy.setattr(Python) -> fortran_setattr -> array_from_pyobj
PoC	https://github.com/baltsers/polycruise/tree/main/numpy/vulnerability-3
Status	Confirmed, **CVE assigned: CVE-2021-41496**
Issue	https://github.com/numpy/numpy/issues/19000

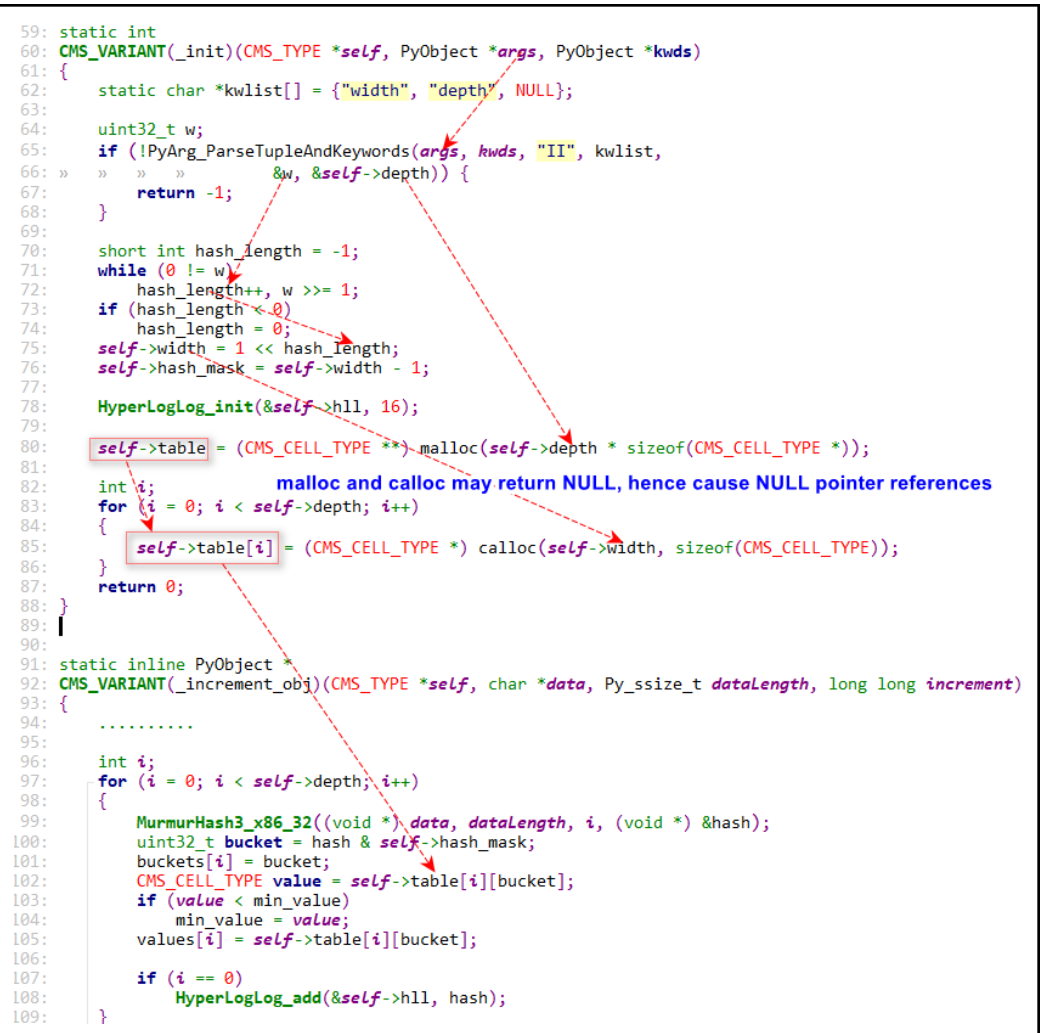
Subject	Numpy
Vulnerability Type	NULL pointer reference
Input	Integration test
Description	 When constructing a loop to create high-dimension arrays repetitively, and keeping the references of the arrays effective, an error of NULL pointer access happens hence causing the Python to crash down. This allows attackers to conduct DoS attacks by repetitively creating and sort arrays.
Taint flow path	numpy.sort (Python) -> array_sort -> PyArray_DescrNew
PoC	https://github.com/baltsers/polycruise/tree/main/numpy/vulnerability-2
Status	Confirmed, **CVE assigned: CVE-2021-41495**
Issue	https://github.com/numpy/numpy/issues/19038

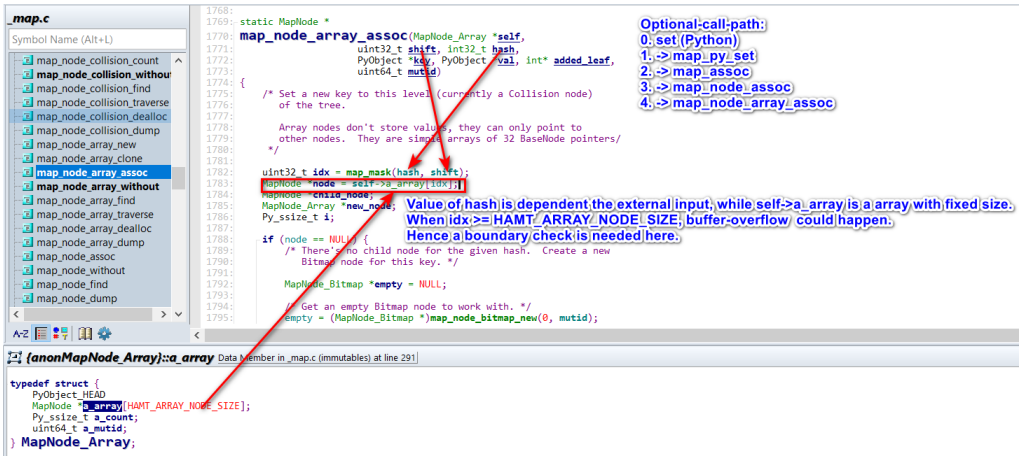
Subject	NumPy
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	
Taint flow path	numpy.empty(Python) -> PyArray_DescrAlignConverter -> _convert_from_any -> _convert_from_str -> strncmp
PoC	pending
Status	Confirmed, **CVE assigned: CVE-2021-34141**
Issue	https://github.com/numpy/numpy/issues/18993

Subject	Numpy
Vulnerability Type	Integer overflow
Input	Integration test

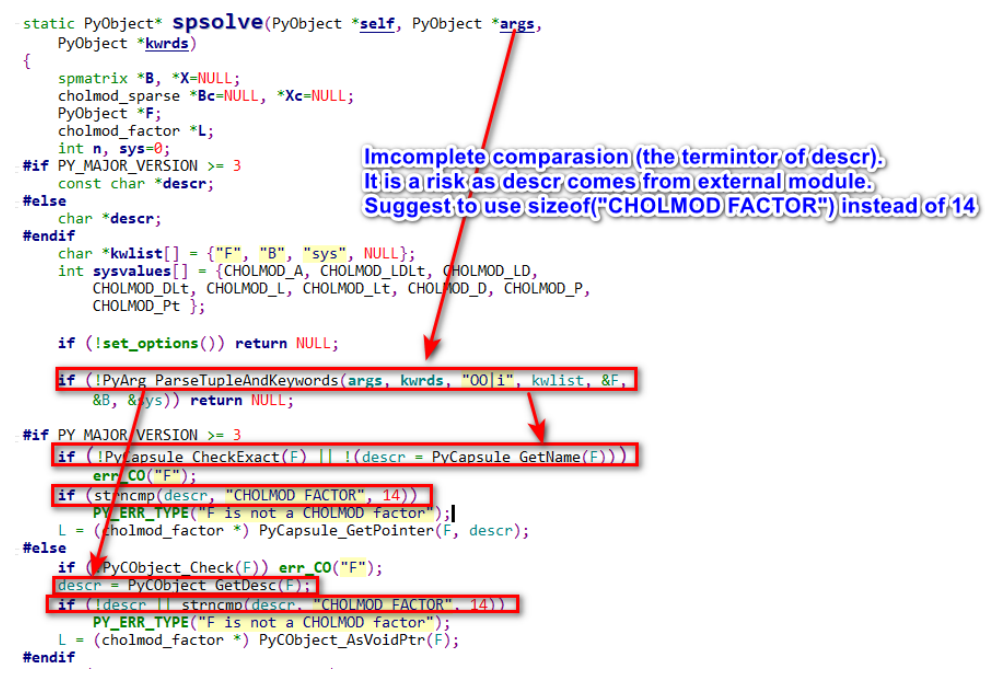
Description	
Taint flow path	numpy.format_float_scientific (Python) -> Dragon4_PrintFloat_IEEE_binary16 ->Format_floatbits-> FormatScientific -> bufferSize-1
PoC	pending
Status	pending
Issue	https://github.com/numpy/numpy/issues/18937

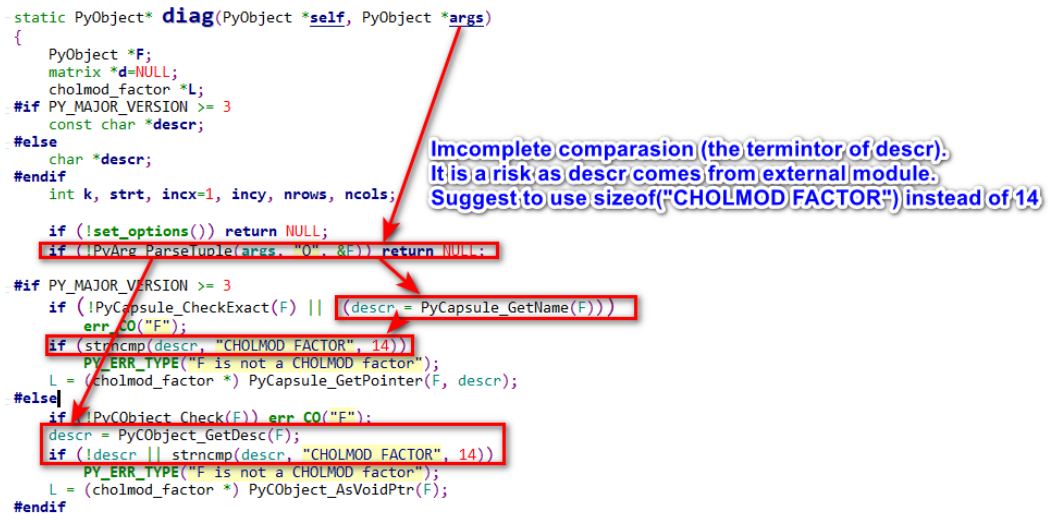
Subject	Bounter
Vulnerability Type	NULL pointer reference
Input	Integration test

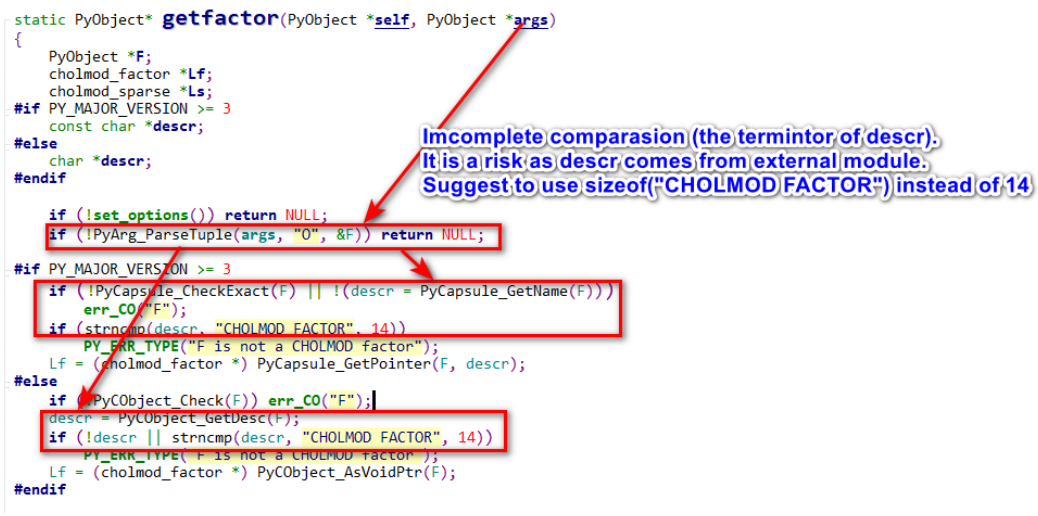
Description	 With carefully constructed inputs (when the width of the hash bucket is set large enough), NULL pointer access could happen hence causing the Python to crash down. This allows attackers to conduct DoS attacks by inputing a huge width of hash bucket.
Taint flow path	update (Python)-> CMS_Conservative_update -> CMS_Conservative_increment_obj increment (Python)-> CMS_Conservative_increment -> CMS_Conservative_increment_obj
PoC	https://github.com/baltsers/polycruise/tree/main/bounter/vulnerability-1
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41497**
Issue	https://github.com/RaRe-Technologies/bounter/issues/47
Subject	Immutables

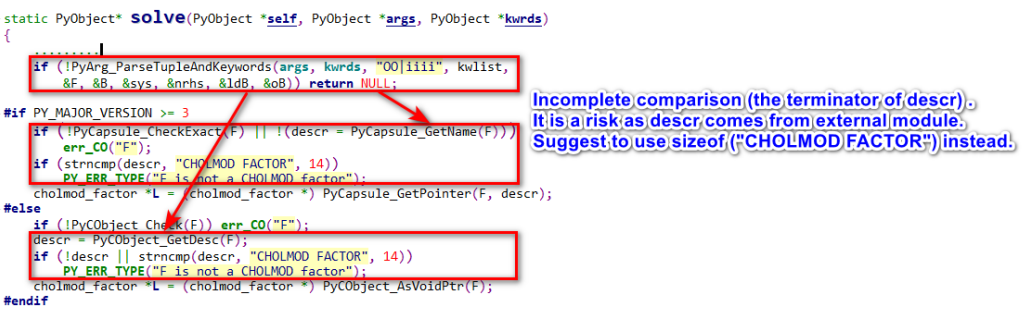
Vulnerability Type	Buffer overflow
Input	Integration test
Description	
Taint flow path	set (Python) -> map_py_set -> map_assoc -> map_node_assoc -> map_node_array_assoc
PoC	pending
Status	Pending
Issue	https://github.com/MagicStack/immutables/issues/67

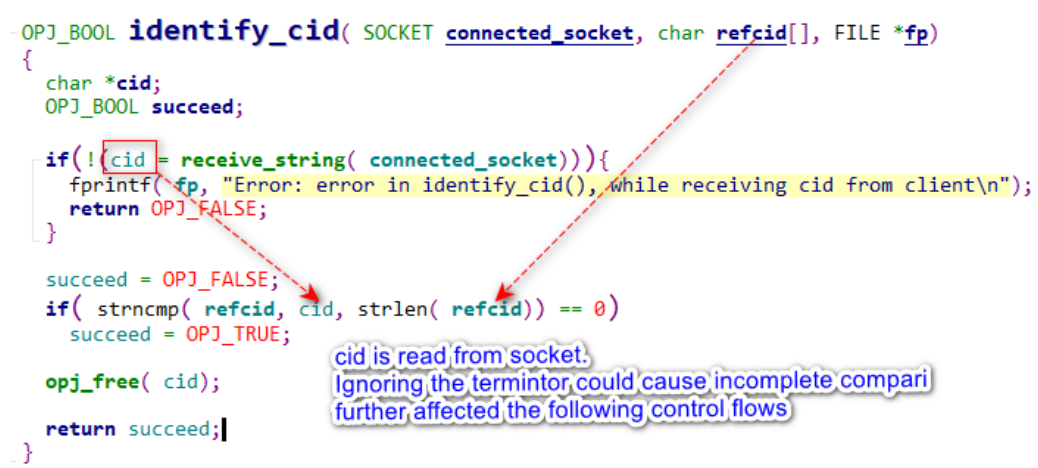
Subject	Japronto
Vulnerability Type	Unknown
Input	Integration test
Description	When passing byte stream context into the server API Response, the server runs abnormally, throws exceptions, and fails to deals with the following requests. This allows attackers to conduct DoS attacks.
Taint flow path	_
PoC	https://github.com/baltsers/polycruise/tree/main/japronto/vulnerability-1
Status	Pending
Issue	https://github.com/squeaky-pl/japronto/issues/183

Subject	Cvxopt
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	 Through carefully modify the name of a Capsule object, the PoC can easily bypass the validation in the **spcolve API** hence causing unexpected results (e.g., crash down during the execution of PoC). This allows attackers to conduct DoS attacks by construct fake Capsule objects.
Taint flow path	spsolve (Python) -> spsolve -> strncmp
PoC	https://github.com/baltsers/polycruise/tree/main/cvxopt/vulnerability-4
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41500**
Issue	https://github.com/cvxopt/cvxopt/issues/193

Subject	Cvxopt
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	 Through carefully modify the name of a Capsule object, the PoC can easily bypass the validation in the **diag API** hence causing unexpected results (e.g., crash down during the execution of PoC). This allows attackers to conduct DoS attacks by construct fake Capsule objects.
Taint flow path	diag(Python) -> diag -> strncmp
PoC	https://github.com/baltsers/polycruise/tree/main/cvxopt/vulnerability-1
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41500**
Issue	https://github.com/cvxopt/cvxopt/issues/193

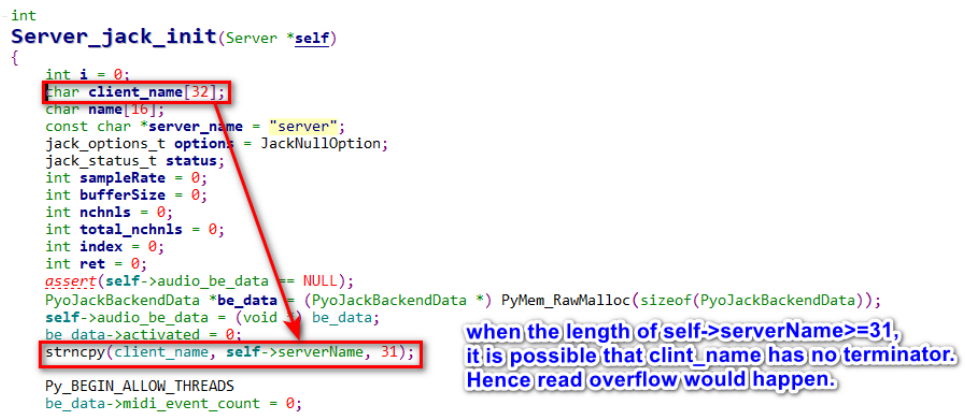
Subject	Cvxopt
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	 Through carefully modify the name of a Capsule object, the PoC can easily bypass the validation in the **getfactor API** hence causing unexpected results (e.g., crash down during the execution of PoC). This allows attackers to conduct DoS attacks by construct fake Capsule objects.
Taint flow path	getfactor(Python) -> getfactor-> strncmp
PoC	https://github.com/baltsers/polycruise/tree/main/cvxopt/vulnerability-2
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41500**
Issue	https://github.com/cvxopt/cvxopt/issues/193

Subject	Cvxopt
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	 Through carefully modify the name of a Capsule object, the PoC can easily bypass the validation in the **solve API** hence causing unexpected results (e.g., crash down during the execution of PoC). This allows attackers to conduct DoS attacks by construct fake Capsule objects.
Taint flow path	solve (Python) -> solve -> strncmp
PoC	https://github.com/baltsers/polycruise/tree/main/cvxopt/vulnerability-3
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41500**
Issue	https://github.com/cvxopt/cvxopt/issues/193

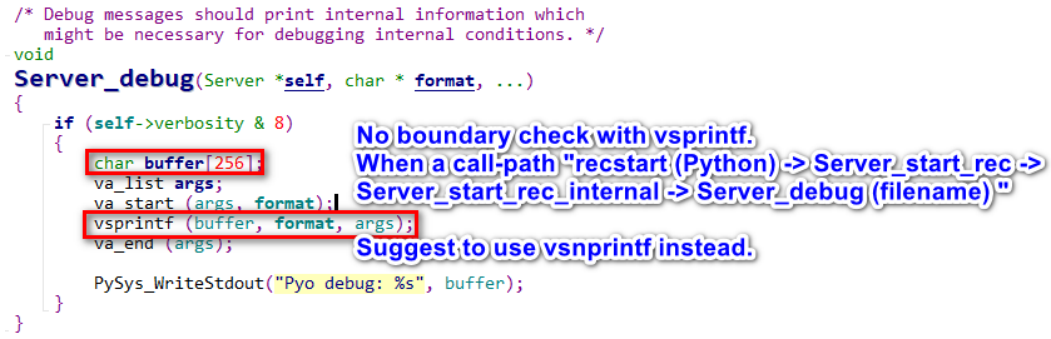
Subject	openjpeg2
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	
Taint flow path	recv -> identify_id -> strncmp
PoC	pending
Status	Pending
Issue	https://github.com/uclouvain/openjpeg/issues/1357

Subject	libarchive
Vulnerability Type	Incomplete string comparison
Input	Integration test
Description	
Taint flow path	1> read -> process_base_block -> process_head_main -> archive_read_open_filenames -> strncmp

	2> read -> process_base_block -> process_head_main -> archive_read_open_filenames_w -> strncmp
Status	Pending
PoC	pending
Issue	https://github.com/libarchive/libarchive/issues/1544

Subject	Pyo
Vulnerability Type	Buffer overflow
Input	Integration test
Description	 When using the Pyo library with audio type "jack", the server is initialized with an overlong (over 32) string, an error of buffer overflow happens and causes Python to crash down. This allows attackers to conduct DoS attacks by arbitrary constructing a overlong server name.
Taint flow path	boot (Python) -> Server_boot -> Server_jack_init
PoC	https://github.com/baltsers/polycruise/tree/main/pyo/vulnerability-1
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41498**
Issue	https://github.com/belangeo/pyo/issues/221

Subject	Pyo

Input	Integration test
Description	 After initializing a Pyo server, an arbitrary file name (length > 256) is passed to the **recstart API** and an error of segment fault happens hence causing Python to crash down. This allows attackers to conduct DoS attacks by deliberately passing on an overlong audio file name.
Taint flow path	recstart (Python) -> Server_start_rec -> Server_start_rec_internal -> Server_debug
PoC	https://github.com/baltsers/polycruise/tree/main/pyo/vulnerability-2
Status	Confirmed and fixed, **CVE Assigned: CVE-2021-41499**
Issue	https://github.com/belangeo/pyo/issues/222