abrignoni / **ALEAPP** Public

Code | Issues 24 | Pull requests 38 | Actions | Projects | Wiki | Secu

Commit `0cafd8f`

mobasi-team committed on Mar 2 · ✕ 0 / 1

```
security: block NQ Vault path traversal on decrypted file writes
```

`main` (#669)

1 parent 29fb64c commit 0cafd8f

2 files changed +114 -2 lines changed

Top

Filter files…

- admin/test/scripts
  - test_nq_vault_path_security.py
- scripts/artifacts
  - NQ_Vault.py

2 files changed +114 -2 lines changed

Search within code

...test/scripts/test_nq_vault_path_security.py

```
@@ -0,0 +1,77 @@
+ import os
+ import sys
+ import tempfile
+ import unittest
+ from pathlib import Path
+ import uuid
+
+
+ ROOT_DIR = os.path.dirname(
+     os.path.dirname(os.path.dirname(os.path.dirname(os.path.abspath(__file__)))))
```

```
+ )
+ if ROOT_DIR not in sys.path:
+     sys.path.insert(0, ROOT_DIR)
+
+ from scripts.artifacts import NQ_Vault
+ import scripts.ilapfuncs as ilapfuncs
+
+
+ class TestNQVaultPathSecurity(unittest.TestCase):
+     def test_sanitize_output_filename_removes_traversal(self):
+         sanitized =
  NQ_Vault._sanitize_output_filename('../../../Users/examiner/.zshrc')
+         self.assertEqual(sanitized, 'zshrc')
+
+     def test_build_safe_output_path_is_confined_to_report_folder(self):
+         with tempfile.TemporaryDirectory() as tmpdir:
+             report_folder = Path(tmpdir) / 'report'
+             report_folder.mkdir()
+             safe_path = NQ_Vault._build_safe_output_path(
+                 str(report_folder),
+                 '../../../../../../outside_written.bin',
+             )
+
  self.assertTrue(str(safe_path).startswith(str(report_folder.resolve())))
+             self.assertEqual(safe_path.name, 'outside_written.bin')
+
+     def test_file_decryption_blocks_path_traversal_write(self):
+         with tempfile.TemporaryDirectory() as tmpdir:
+             base = Path(tmpdir)
+             report_folder = base / 'report'
+             report_folder.mkdir()
+             ilapfuncs.OutputParameters.screen_output_file_path = str(base /
  'screen_output.html')
+
+             image_dir = base / 'payload' / '.image'
+             image_dir.mkdir(parents=True)
+             encrypted_file = image_dir / '1700000000.bin'
+             encrypted_file.write_bytes(b'A' * 200)
+
+             unique_name = f'outside_written_{uuid.uuid4().hex}.bin'
```

```
48  +            traversal_name = f'../../../{unique_name}'
49  +            file_info = {
50  +                '1700000000.bin': {
51  +                    'old_filepath': '/sdcard/DCIM/Camera/original.jpg',
52  +                    'old_filename': traversal_name,
53  +                    'vault_filepath':
      '/sdcard/SystemAndroid/Data/hash/.image/1700000000.bin',
54  +                    'timestamp': '2024-01-01 00:00:00',
55  +                    'vid_length': '',
56  +                    'resolution': '',
57  +                    'alb_name': 'Default',
58  +                    'prev_alb_name': '',
59  +                    'password_id': 'enc-pin-id',
60  +                }
61  +            }
62  +            pin_info = {
63  +                'enc-pin-id': {
64  +                    'xor_key': '0x00',
65  +                    'pin_for_XOR_key': '0000',
66  +                }
67  +            }
68  +
69  +            NQ_Vault.file_decryption([str(encrypted_file)], file_info, pin_info,
      str(report_folder))
70  +
71  +            outside_target = (report_folder / traversal_name).resolve()
72  +            self.assertFalse(outside_target.exists())
73  +            self.assertTrue((report_folder / unique_name).exists())
74  +
75  +
76  + if __name__ == '__main__':
77  +     unittest.main()
```

scripts/artifacts/NQ_Vault.py

```
    ...   @@ -1,4 +1,5 @@
1   1     import itertools
      2 + import re
2   3     import string
3   4     from pathlib import Path
4   5     from os.path import join
```

```
@@ -7,6 +8,39 @@
from scripts.ilapfuncs import logfunc, tsv, timeline, is_platform_windows, media_to_html, open_sqlite_db_readonly
 
 
+def _sanitize_output_filename(filename, default_name='recovered_file.bin'):
+    raw_name = str(filename or '')
+    normalized = raw_name.replace('\\', '/')
+    safe_name = Path(normalized).name
+    safe_name = re.sub(r'[\x00-\x1f<>:"/\\|?*]+', '_', safe_name)
+    safe_name = safe_name.strip().strip('.')
+    if not safe_name:
+        return default_name
+    return safe_name
+
+
+def _build_safe_output_path(report_folder, original_filename):
+    report_root = Path(report_folder).resolve()
+    sanitized_name = _sanitize_output_filename(original_filename)
+    output_path = (report_root / sanitized_name).resolve()
+    try:
+        output_path.relative_to(report_root)
+    except ValueError:
+        output_path = report_root / _sanitize_output_filename(None)
+
+    if output_path.exists():
+        stem = output_path.stem
+        suffix = output_path.suffix
+        index = 1
+        while True:
+            candidate = report_root / f'{stem}_{index}{suffix}'
+            if not candidate.exists():
+                output_path = candidate
+                break
+            index += 1
+    return output_path
+
+
 '''def extract_PIN_from_db(file_found):
     try:
```

```
12  46          connection = open_sqlite_db_readonly(file_found)
    @@ -187,12 +221,13 @@ def file_decryption(files_found, dict_of_file_info,
    dict_of_pin_dicts, report_fo
187 221                                 byte = file_to_decrypt.read(1)
188 222
189 223                             xord_bytes_decrypted = bytes(xor_list)
190     -                           with open(join(report_folder, decrypted_file_name),
    'wb') as decryptedFile:
    224 +                           decrypted_output_path =
    _build_safe_output_path(report_folder, decrypted_file_name)
    225 +                           with open(decrypted_output_path, 'wb') as
    decryptedFile:
191 226                                 decryptedFile.write(xord_bytes_decrypted)
192 227                                 decryptedFile.close()
193 228
194 229                                 tolink = []
195     -                               pathdec = join(report_folder,
    decrypted_file_name)
    230 +                               pathdec = str(decrypted_output_path)
196 231                                 tolink.append(pathdec)
197 232                                 thumb = media_to_html(pathdec, tolink,
    report_folder)
198 233
```

Comments 0

Please sign in to comment.