freedomofpress / **securedrop-client** Public

Code Issues 111 Pull requests 25 Actions Projects Wiki Sec

Commit e518ada

deeplow authored and **legoktm** committed last week Partially verified

```
Security fix: path traversal for abs path in gzip

Assuming a compromised server, a maliciously crafted gzip can be sent
to the client application. The file name after decompression is checked
against path traversals before actually creating a physical path.
However, absolute paths would not be filtered out.

This vulnerability was discovered by cookiejack15 on Bugcrowd.
```

release/0.17.5 · 0.17.5

1 parent 5840215 commit e518ada

2 files changed +26 -3 lines changed

Top

Filter files…

- client
  - securedrop_client
    - crypto.py
  - tests
    - test_crypto_gzip.py

2 files changed +26 -3 lines changed

Search within code

client/securedrop_client/crypto.py

```
@@ -84,6 +84,12 @@ def read_gzip_header_filename(filename: str) -> str:
84  84              original_filename = str(fb, "utf-8")
85  85
86  86          check_path_traversal(original_filename)
    87  +
```

```
88  +     # Ensure this is a single file and not a path
89  +     if original_filename != Path(original_filename).name:
90  +         # Otherwise, treat it as an unrecoverable fault for this file:
91  +         raise ValueError("Unsafe file name; aborting further processing")
92  +
87  93        return original_filename
88  94
89  95
```

client/tests/test_crypto_gzip.py

```
@@ -153,7 +153,24 @@ def test_path_traversal(tmp_path):
153 153           f.write(original_name.encode("utf-8") + b"\000")
154 154           f.write(gzip.compress(b"test content")[10:])
155 155
156     -     with pytest.raises(
157     -         ValueError, match="Unsafe file or directory name: '../../../../../etc/passwd'"
158     -     ):
    156 +     with pytest.raises(ValueError, match=f"Unsafe file or directory name: '{original_name}'"):
157 +         read_gzip_header_filename(str(tmp_file))
158 +
159 +
160 + def test_overwrite_via_abs_path(tmp_path):
161 +     """Test that an absolute path in the gzip header filename is rejected."""
162 +     tmp_file = tmp_path / "test.gz"
163 +     original_name = "/home/user/.securedrop_client/svs.sqlite"
164 +
165 +     with open(tmp_file, "wb") as f:
166 +         f.write(GZIP_FILE_IDENTIFICATION)  # ID
167 +         f.write(GZIP_COMPRESSION_BYTES)  # Compression method
168 +         f.write(bytes([GZIP_FLAG_FILENAME]))  # Flags
169 +         f.write(b"\000\000\000\000")  # mtime
170 +         f.write(b"\000")  # XFL
171 +         f.write(b"\377")  # OS
172 +         f.write(original_name.encode("utf-8") + b"\000")
173 +         f.write(gzip.compress(b"test content")[10:])
174 +
```

```
      175 +     with pytest.raises(ValueError, match="Unsafe file name; aborting further
          processing"):
159   176           read_gzip_header_filename(str(tmp_file))
```

Comments 0

Please sign in to comment.