

anthropics / anthropic-sdk-python Public

<> Code Issues 88 Pull requests 104 Discussions Actions Security and

# Commit 6599043



dtmeadows authored and stainless-app[bot] committed 4 days ago



```
fix(memory): return resolved path from async _validate_path
* fix(memory): return resolved path from async _validate_path to close TOCTOU window

The async _validate_path was returning the unresolved path while the sync
version correctly returned the resolved path, allowing a symlink swap between
validation and use.

* test(memory): add test for async _validate_path symlink TOCTOU fix

Verifies that the async _validate_path returns the resolved real path
rather than the unresolved symlink path, closing the TOCTOU window.

* fix: remove unused temp_directory parameter from test
```

main (#1264) · v0.89.0 ... v0.87.0

1 parent 715030c commit 6599043

2 files changed +34 -1 lines changed

↑ Top ⚙️

Filter files...

- src/anthropic/lib/tools
  - \_beta\_builtin\_memory\_tool.py
- tests/lib/tools/memory\_tools
  - test\_filesystem.py

2 files changed +34 -1 lines changed

Search within code ⚙️

...opic/lib/tools/\_beta\_builtin\_memory\_tool.py

```
@@ -653,7 +653,7 @@ async def _validate_path(self, path: str) -> AsyncPath:
653 653
```

```

654 654         await _async_validate_no_symlink_escape(full_path, self.memory_root)
655 655
656 -         return full_path
656 +         return AsyncPath(resolved_path)
657 657
658 658     @override
659 659     async def view(self, command: BetaMemoryTool20250818ViewCommand) -> str:

```



...s/lib/tools/memory\_tools/test\_filesystem.py



```
@@ -945,6 +945,39 @@ async def
```

```
test_path_validation_reject_paths_trying_to_escape_memories(
```

```

945 945         )
946 946     )
947 947
948 +     async def test_validate_path_returns_resolved_path_not_symlink_target(
949 +         self, async_local_filesystem_tool: BetaAsyncLocalFilesystemMemoryTool
950 +     ) -> None:
951 +         """_validate_path must return the resolved path so that subsequent file
952 +         operations hit the real location, not the (potentially swappable)
953 +         symlink.
954 +
955 +         Without this fix, an attacker could:
956 +         1. Create /memories/link -> /memories/legit (passes validation)
957 +         2. Swap /memories/link -> /etc between validation and the file
958 +         operation
959 +         3. The file operation would follow the new symlink target
960 +         """
961 +         memories_path = Path(str(async_local_filesystem_tool.memory_root))
962 +         memories_path.mkdir(parents=True, exist_ok=True)
963 +
964 +         # Create a real directory inside memories and a symlink pointing to it
965 +         legit_dir = memories_path / "legit"
966 +         legit_dir.mkdir()
967 +         (legit_dir / "file.txt").write_text("content", encoding="utf-8")
968 +
969 +         link_path = memories_path / "link"
970 +         os.symlink(legit_dir, link_path, target_is_directory=True)

```

```
970 +         # _validate_path should return the resolved real path, not the symlink
      path
971 +         result = await
      async_local_filesystem_tool._validate_path("/memories/link/file.txt")
972 +         result_str = str(result)
973 +
974 +         # The returned path should point to the resolved location (under
      legit/),
975 +         # not through the symlink
976 +         assert "link" not in result_str, (
977 +             f"_validate_path returned unresolved symlink path: {result_str}"
978 +         )
979 +         assert str(legit_dir.resolve()) in result_str
980 +
948 981     async def test_symlink_validation_reject_symlink_pointing_outside_memories(
949 982         self, async_local_filesystem_tool: BetaAsyncLocalFilesystemMemoryTool
950 983     ) -> None:
```



## Comments 0



Please [sign in](#) to comment.