

Commit 3186851



Hinotoi-agent authored 4 days ago · 4 / 4 · Verified

fix(ohmo): isolate shared-chat sessions by sender (#159)

main (#159) · v0.1.7

1 parent [c881c7d](#) commit 3186851

2 files changed +71 -12 lines changed

Top

Filter files...

- ohmo/gateway
 - router.py
- tests/test_ohmo
 - test_gateway.py

2 files changed +71 -12 lines changed

Search within code

```

ohmo/gateway/router.py
@@ -6,15 +6,16 @@
6 6
7 7
8 8     def session_key_for_message(message: InboundMessage) -> str:
9 -     """Route sessions by chat and thread when available."""
9 +     """Route sessions by sender plus chat/thread when available."""
10 10     if message.session_key_override:
11 11         return message.session_key_override
12 +     sender_id = str(message.sender_id).strip() or "anonymous"
12 13     thread_id = (
13 14         message.metadata.get("thread_id")

```

```

14 15         or message.metadata.get("thread_ts")
15 16         or message.metadata.get("message_thread_id")
16 17     )
17 18     if thread_id:
18 -         return f"{message.channel}:{message.chat_id}:{thread_id}"
19 -         return f"{message.channel}:{message.chat_id}"
19 +         return f"{message.channel}:{message.chat_id}:{thread_id}:{sender_id}"
20 +         return f"{message.channel}:{message.chat_id}:{sender_id}"
20 21

```

tests/test_ohmo/test_gateway.py

...



@@ -26,7 +26,7 @@

```

26 26     from ohmo.workspace import get_gateway_restart_notice_path,
        initialize_workspace

```

27 27

28 28

```

29 -     def test_gateway_router_uses_thread_when_present():

```

```

29 +     def test_gateway_router_uses_thread_and_sender_when_present():

```

```

30 30         message = InboundMessage(

```

```

31 31             channel="slack",

```

```

32 32             sender_id="u1",

```



@@ -35,18 +35,39 @@ def test_gateway_router_uses_thread_when_present():

```

35 35         timestamp=datetime.utcnow(),

```

```

36 36         metadata={"thread_ts": "t1"},

```

```

37 37     )

```

```

38 -     assert session_key_for_message(message) == "slack:c1:t1"

```

```

38 +     assert session_key_for_message(message) == "slack:c1:t1:u1"

```

39 39

40 40

```

41 -     def test_gateway_router_falls_back_to_chat_scope():

```

```

41 +     def test_gateway_router_falls_back_to_chat_and_sender_scope():

```

```

42 42         message = InboundMessage(

```

```

43 43             channel="telegram",

```

```

44 44             sender_id="u1",

```

```

45 45             chat_id="chat-1",

```

```

46 46             content="hello",

```

```

47 47             timestamp=datetime.utcnow(),

```

```

48 48     )

```

```

49 -     assert session_key_for_message(message) == "telegram:chat-1"

```

```

49 +     assert session_key_for_message(message) == "telegram:chat-1:u1"

```

```

50 +
51 +
52 + def test_gateway_router_separates_senders_in_same_chat_thread():
53 +     first = InboundMessage(
54 +         channel="slack",
55 +         sender_id="alice",
56 +         chat_id="shared-chat",
57 +         content="hello",
58 +         timestamp=datetime.utcnow(),
59 +         metadata={"thread_ts": "thread-1"},
60 +     )
61 +     second = InboundMessage(
62 +         channel="slack",
63 +         sender_id="bob",
64 +         chat_id="shared-chat",
65 +         content="hello",
66 +         timestamp=datetime.utcnow(),
67 +         metadata={"thread_ts": "thread-1"},
68 +     )
69 +     assert session_key_for_message(first) == "slack:shared-chat:thread-1:alice"
70 +     assert session_key_for_message(second) == "slack:shared-chat:thread-1:bob"

```

50 71

51 72

52 73 `def test_gateway_error_formats_claude_refresh_failure():`

@@ -122,18 +143,18 @@ class Result:

122 143

123 144

124 145 `@pytest.mark.asyncio`125 - `async def test_runtime_pool_restores_messages_for_session_key(tmp_path, monkeypatch):`146 + `async def test_runtime_pool_restores_messages_for_sender_scoped_session_key(tmp_path, monkeypatch):`126 147 `workspace = tmp_path / ".ohmo-home"`127 148 `initialize_workspace(workspace)`128 149 `save_session_snapshot(`129 150 `cwd=tmp_path,`130 151 `workspace=workspace,`131 152 `model="gpt-5.4",`

```

132 153         system_prompt="system",
133 -         messages=[ConversationMessage.from_user_text("remember me")],
154 +         messages=[ConversationMessage.from_user_text("remember alice only")],
134 155         usage=UsageSnapshot(),
135 156         session_id="sess123",
136 -         session_key="feishu:chat-1",
157 +         session_key="feishu:chat-1:alice",
137 158     )
138 159
139 160     captured: dict[str, object] = {}
@@ -152,12 +173,49 @@ async def fake_start_runtime(bundle):
152 173     monkeypatch.setattr("ohmo.gateway.runtime.start_runtime",
    fake_start_runtime)
153 174
154 175     pool = OhmoSessionRuntimePool(cwd=tmp_path, workspace=workspace,
    provider_profile="codex")
155 -     bundle = await pool.get_bundle("feishu:chat-1")
176 +     bundle = await pool.get_bundle("feishu:chat-1:alice")
156 177
157 178     assert captured["restore_messages"] is not None
158 179     assert bundle.session_id == "sess123"
159 180
160 181
182 + @pytest.mark.asyncio
183 + async def test_runtime_pool_does_not_restore_other_sender_session_key(tmp_path,
    monkeypatch):
184 +     workspace = tmp_path / ".ohmo-home"
185 +     initialize_workspace(workspace)
186 +     save_session_snapshot(
187 +         cwd=tmp_path,
188 +         workspace=workspace,
189 +         model="gpt-5.4",
190 +         system_prompt="system",
191 +         messages=[ConversationMessage.from_user_text("remember alice only")],
192 +         usage=UsageSnapshot(),
193 +         session_id="sess123",
194 +         session_key="feishu:chat-1:alice",
195 +     )
196 +
197 +     captured: dict[str, object] = {}

```

```

198 +
199 +     async def fake_build_runtime(**kwargs):
200 +         captured["restore_messages"] = kwargs.get("restore_messages")
201 +         return SimpleNamespace(
202 +             engine=SimpleNamespace(set_system_prompt=lambda prompt: None,
203 +             messages=[]),
204 +             session_id="newsession",
205 +         )
206 +     async def fake_start_runtime(bundle):
207 +         return None
208 +
209 +     monkeypatch.setattr("ohmo.gateway.runtime.build_runtime",
210 +     fake_build_runtime)
211 +     monkeypatch.setattr("ohmo.gateway.runtime.start_runtime",
212 +     fake_start_runtime)
213 +
214 +     pool = OhmoSessionRuntimePool(cwd=tmp_path, workspace=workspace,
215 +     provider_profile="codex")
216 +     bundle = await pool.get_bundle("feishu:chat-1:bob")
217 +
218 +     assert captured["restore_messages"] is None
219 +     assert bundle.session_id == "newsession"

```

```
161 219 @pytest.mark.asyncio
```

```
162 220 async def
test_runtime_pool_stream_message_emits_progress_and_tool_hint(tmp_path,
monkeypatch):
```

```
163 221 workspace = tmp_path / ".ohmo-home"
```



```
@@ -689,7 +747,7 @@ async def fake_restart(message, session_key: str) ->
None:
```

```
689 747 "🔄 正在重启 gateway , 马上回来。 \n"
```

```
690 748 "Restarting the gateway now. I'll be back in a moment."
```

```
691 749 )
```

```
692 -     assert restart_payloads == [("feishu", "c1", "feishu:c1")]
```

```
750 +     assert restart_payloads == [("feishu", "c1", "feishu:c1:u1")]
```

```
693 751
```

```
694 752
```

```
695 753 @pytest.mark.asyncio
```



Comments 0



Please [sign in](#) to comment.