fix(security): close auth bypass + default-permit RBAC (issue #866) #950

Merged **viettranx** merged 1 commit into `nextlevelbuilder:dev` from `thieung:claude/sweet-solomon-474c…` 2 weeks ago

Conversation Commits 1 Checks Files changed

thieung commented 2 weeks ago • edited Contributor

Summary

Closes the 3-part vulnerability chain reported in issue #866 (auth bypass + default-permit RBAC). Flips the gateway from default-permit to **fail-closed**:

1. **Router Path 4 fallback no longer grants** `RoleViewer` — invalid token + no active pairing now returns `UNAUTHORIZED` instead of authenticating the client as a viewer.
2. **Policy engine is default-deny** — `MethodRole` returns a new `RoleNone` sentinel for unclassified methods; `CanAccess` denies it for every role (including owner). New RPCs are secure-by-default.
3. **Admin allowlist completed** — `heartbeat.set`, `heartbeat.toggle`, `heartbeat.test`, `heartbeat.checklist.set`, `logs.tail` (plus `api_keys.list`, hooks mutations, `tenants.*` writes, `voices.refresh`) are now admin-gated.

Root causes (pre-fix code paths)

#	File	Symptom
1	`internal/gateway/router.go:270-289` (old)	Path 4 set `client.role = RoleViewer; client.authenticated = true` whenever no valid token / no pairing → attacker with any string in `token` became an authenticated viewer.

#	File	Symptom
2	`internal/permissions/policy.go:140-146` (old)	`MethodRole` fell through to `return RoleViewer` for every method not in admin/write lists → any viewer could call anything not explicitly write/admin.
3	`isAdminMethod` list	Missing `heartbeat.set`, `heartbeat.toggle`, `heartbeat.test`, `heartbeat.checklist.set`, `logs.tail` → the three vulns combined let an unauth'd client mutate heartbeat + exfiltrate live server logs.

Reproduction scripts (inline — no repo artifacts)

Two deterministic, self-verifying scripts. Save them locally and run; both signal via exit code.

Script	Purpose	Exit before fix	Exit after fix
`repro-issue-866.mjs`	End-to-end WebSocket repro of the 3-vuln chain against a running gateway	vulnerable path confirmed	all 3 vulns blocked
`repro_approvals_misclass.go`	Standalone Go program exposing `writePrefixes` shadowing in `MethodRole`	`exit 1` (BUG)	`exit 0` (FIXED)

Script 1 — `repro-issue-866.mjs`

Run against a **vulnerable** gateway (pre-patch, upstream `ghcr.io/nextlevelbuilder/goclaw:latest` at time of filing):

```
node repro-issue-866.mjs ws://localhost:18790/ws
```

▶ Full source (Node + `ws`, 144 lines)

Before fix (observed live):

```
=== Step 1 — connect with invalid token ===
{"type":"res","ok":true,"payload":{"role":"viewer","tenant_slug":"master",...}}
[VULN] #1 confirmed: invalid token → connect.ok with role=viewer

=== Step 2 — heartbeat.set (mutation) ===
{"type":"res","ok":false,"error":{"code":"INVALID_REQUEST","message":"invalid agentId"}}
[VULN] #2 confirmed: RBAC let it through, only validation rejected (INVALID_REQUEST)
```

```
=== Step 3 — logs.tail (exfil) ===
{"type":"res","ok":true,"payload":{"status":"tailing"}}
[event] log {"level":"info","msg":"..."}   ← live server logs streaming to attacker
[VULN] #3 confirmed: viewer started logs.tail stream (exfil)

=== Step 4 — heartbeat.checklist.set (mutation) ===
[VULN] #4 confirmed: RBAC let it through, validation only (INVALID_REQUEST)
```

After fix (patched binary on port `18791`):

```
=== Step 1 — connect with invalid token ===
{"type":"res","ok":false,"error":{"code":"UNAUTHORIZED","message":"permission denied:
valid token or active pairing required"}}

=== Step 2 — heartbeat.set (mutation) ===
{"type":"res","ok":false,"error":{"code":"UNAUTHORIZED","message":"first request must be
'connect'"}}
[safe] heartbeat.set blocked by RBAC: UNAUTHORIZED

=== Step 3 — logs.tail (exfil) ===
{"type":"res","ok":false,"error":{"code":"UNAUTHORIZED","message":"first request must be
'connect'"}}
[safe] logs.tail blocked by RBAC: UNAUTHORIZED

=== Step 4 — heartbeat.checklist.set (mutation) ===
[safe] heartbeat.checklist.set blocked by RBAC: UNAUTHORIZED
```

Script 2 — `repro_approvals_misclass.go`

After review feedback, the narrow `writePrefixes` fallback (`"approvals."`, `"exec.approval."`) in `isWriteMethod` was found to short-circuit the public→admin→write→read ordering in `MethodRole`, misclassifying `exec.approval.list` as `RoleOperator` even though it is an explicit entry in `isReadMethod`. Viewers could not read their own approvals queue.

```
go run repro_approvals_misclass.go
```

▶ Full source (Go standalone, 86 lines)

Before removal of `writePrefixes`:

```
METHOD                       ACTUAL     EXPECTED   STATUS   NOTE
------------------------------------------------------------------------------------
----
exec.approval.list           operator   viewer     BUG      Listed in isReadMethod —
viewers must be able to read approvals
exec.approval.approve        operator   operator   OK       Listed in writeExact —
mutation, operators+admins only
exec.approval.deny           operator   operator   OK       Listed in writeExact —
```

```
mutation, operators+admins only

[BEFORE-FIX] 1/3 methods misclassified — prefix fallback is shadowing isReadMethod
exit status 1
```

After removal of `writePrefixes`:

```
METHOD                        ACTUAL     EXPECTED   STATUS   NOTE
-------------------------------------------------------------------------------------------
----
exec.approval.list            viewer     viewer     OK       Listed in isReadMethod —
viewers must be able to read approvals
exec.approval.approve         operator   operator   OK       Listed in writeExact —
mutation, operators+admins only
exec.approval.deny            operator   operator   OK       Listed in writeExact —
mutation, operators+admins only

[AFTER-FIX] All 3 methods classified correctly
exit status 0
```

The drift test (`TestMethodRole_DriftCoverage_AllProtocolMethodsClassified`) additionally flagged 12 Phase-3 methods (`tts.*`, `browser.act/snapshot/screenshot`, `zalo.personal.*`, `whatsapp.qr.start`) that were relying on the removed default-permit fallback. All now classified explicitly.

Happy-path regression check (valid gateway token)

```
connect.ok=true   role=admin   tenant=master
health.ok=true                                    # public method still reachable
heartbeat.get.ok=false code=INVALID_REQUEST       # RBAC passed; only validation
rejected nil agentId
```

Unit tests

```
go test ./internal/permissions/...   ok (0.542s)
go test ./internal/gateway/...       ok (0.621s / 1.201s)
go build ./...                       ok
go build -tags sqliteonly ./...      ok
go vet ./...                         ok
```

Tests locking in fail-closed behaviour:

- `TestCanAccess_UnknownMethod_DeniedForAll` — unclassified method denied for every role (inverts the removed default-permit test).

- `TestCanAccess_CVE866_HeartbeatAndLogs` — `heartbeat.set`, `heartbeat.toggle`, `heartbeat.test`, `heartbeat.checklist.set`, `logs.tail` require admin.
- `TestCanAccess_PublicMethods` — `connect`, `health`, `status`, `browser.pairing.status` remain reachable by all roles.
- `TestMethodRole_ApprovalsList_IsViewer` — locks in the `writePrefixes` removal fix.
- `TestMethodRole_DriftCoverage_AllProtocolMethodsClassified` — parses `pkg/protocol/methods.go` at test time and asserts every `Method*` constant resolves to a non-`RoleNone` role; catches future allowlist gaps before ship.

Changes

- `internal/gateway/router.go`: Path 4 fails closed with `UNAUTHORIZED`; dispatcher distinguishes "role too low" vs "unclassified method" for cleaner error + `security.permission_denied` structured log.
- `internal/permissions/policy.go`: add `RoleNone` sentinel; `CanAccess` denies it; `MethodRole` rewritten to fail-closed with explicit public / admin / write / read allowlists. `isWriteMethod` uses exact-match `writeExact` only (`writePrefixes` removed per review). `isReadMethod` enumerates viewer-reachable RPCs. 12 Phase-3 methods (TTS/browser/zalo/whatsapp) classified.
- `internal/permissions/policy_test.go`: fail-closed + drift-coverage + ApprovalsList lock-in tests.

Note: the two reproduction scripts are deliberately **not** committed — `scripts/` is reserved for install/setup tooling. Scripts live inline in this PR body for reviewer copy-paste; see `<details>` blocks above.

Backward compatibility

- **Path 2 preserved**: when `gateway.token` is unset (dev / onboard flow), unauthenticated connections still receive `RoleOperator` — same as before. No behaviour change for local dev setups.

Test plan

- [x] Reproduce all 3 vulns on live upstream image (pre-patch)
- [x] Verify all 3 vulns closed on patched binary (post-patch)
- [x] Reproduce `writePrefixes` shadowing bug with standalone Go script (exit 1)
- [x] Verify fix with same script post-removal (exit 0)
- [x] Verify admin happy-path (valid token → `heartbeat.get` reaches handler)
- [x] `go build ./...` + `go build -tags sqliteonly ./...`
- [x] `go vet ./...`
- [x] `go test ./internal/permissions/... ./internal/gateway/...`

Fixes #866.

`fix(security): close auth bypass + default-permit RBAC (issue nextlev…` ··· ✓ `42f598b`

thieung force-pushed the `claude/sweet-solomon-474c75` branch from **`72ba2ec`** to **`42f598b`** Compare
2 weeks ago

viettranx approved these changes 2 weeks ago

View reviewed changes

viettranx left a comment — Contributor

Review — APPROVE ✅

Đã audit bằng subagent + trust-but-verify. Build + test xanh, drift test parse 126 `Method*` constants, all classified. Path 2 (gateway.token unset → RoleOperator) vẫn nguyên → không break dev/onboard.

Verified

- Path 4 fail-closed đúng; `RoleNone` sentinel + `CanAccess` deny-all (kể cả owner).
- Role ordering Owner(4) > Admin(3) > Operator(2) > Viewer(1) — tests đúng.
- `writePrefixes` removal không regression — `MethodApprovalsList` giờ đúng RoleViewer (bonus fix).
- `go build ./...`, `go build -tags sqliteonly ./...`, `go vet`, `go test ./internal/permissions/... ./internal/gateway/...` all pass.

Follow-ups (non-blocking)

1. **Heartbeat admin-only có thể quá chặt (medium).** `internal/gateway/methods/heartbeat.go` `handleSet` nhận `agentId` nhưng **không có tenant/ownership check** trong handler. Đẩy lên admin là stopgap hợp lý đúng scope #866, nhưng fix gốc là thêm ownership check handler-level rồi nới lại về operator. Đề xuất mở follow-up issue "heartbeat per-agent ownership scoping" để không quên.

2. **Raw strings `"tenants.create"` / `"tenants.list"` … (low).** `pkg/protocol/methods.go` chưa có `MethodTenants*` constants. Drift test khớp string nên pass, nhưng rename sau này sẽ silent diverge. Thêm constants trong PR sau.

3. **Dead prefix trong `MethodScopes` (low).** `policy.go:196` còn `strings.HasPrefix(method, "pairing.")` nhưng constants thực là `device.pair.*` → nhánh không match gì. Xóa hoặc sửa.

Merge được. Sẽ xử lý các điểm follow-up sau khi về dev.

viettranx merged commit `406022e` into `nextlevelbuilder:dev` 2 weeks ago

3 checks passed

View details

duhd-vnpay added a commit to duhd-vnpay/goclaw that referenced this pull request 2 weeks ago

`chore(goclaw): merge upstream v3.9.0 — tenant-scoped web_search + sec…` ··· `2c19549`

duhd-vnpay added a commit to duhd-vnpay/goclaw that referenced this pull request 2 weeks ago

`fix(permissions): classify ardenn.* methods in RBAC policy` ··· `55562b8`

Reviewers

viettranx ✓

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

critical: unauthenticated log subscription and command injection in heartbeat execution flow

2 participants