Fixed 2026-09-30 — CVE-2026-105445 · Docker's security advisory

Fixed (2026-09-30)

Docker deployed a fix less than four hours after the report was received: converting an account to an organization no longer transfers personal access tokens. The flaw had existed since **August 26, 2024** — an exposure window of just over two years. Docker reviewed every account conversion since the flaw was introduced and examined the **~304,000 pushes** made with transferred tokens, finding **no evidence the vulnerability was exploited**. Exposed users were notified by email on 2026-10-02, and the transferred tokens were force-deactivated on 2026-10-08 (opt-out applies). CVE-2026-105445 — Docker's security advisory.

The bug

Any Docker Hub user can quietly mint a personal access token that logs in as *any chosen user*. No email. No notification. No trace — except a token sitting in someone else's account that they never created. We call it **GhostPat**: a ghost personal-access-token, haunting an account that isn't its own.

How it works

1. Attacker owns throwaway account **A** and creates a PAT with read/write/delete scope.
2. **A** is converted to an organization — the **receiver** is set to victim **B** (any existing username).
3. No confirmation email, no notification of any kind is sent to **B**.
4. A's PAT is attached to **B**'s account — and **authenticates as B**, inheriting B's org memberships.

Attacker cost: one throwaway account. The conversion is free. That's it.

The proof, in one line

```
$ docker login -u ch···
Username: ch···
Password: dckr_pat_········ (minted on a throwaway account the same morning)
Login Succeeded
```

Vector 2, 2026-09-29. The token was created on a different account, converted to an org with `ch···` as receiver — and then logged in *as* `ch···`, with zero notification sent.

What we proved

- ✓ **Reproduced twice, independently** — different IPs, different people, different days.
- ✓ `docker login` as the victim with the minted token: **Login Succeeded**.
- ✓ Pushed and created a **repository in the victim's own namespace** with the minted token.
- ✓ The token **inherits the victim's org memberships** — pushed to a repository in an org the victim belongs to.
- ✓ Read access, not just write: the minted token also **pulled the private repositories** of the targeted user.
- ✓ Registry API accepts the token for bearer exchange (`POST /v2/auth/token`) and repository listing.
- ✓ **Zero notification** — the victim's inbox was confirmed empty, repeatedly. The traces: an entry in the token list (the name is attacker-chosen — a familiar name proves nothing) and a new organization visible only when you click the account switcher.

Worst case (flagged, not confirmed)

If the Official Images (`library/` — `ubuntu` , `alpine` , `nginx` , …) are published under a plain account reachable by this same conversion path, a single conversion could push into the base images of the entire ecosystem: a **root-of-trust** compromise of the default registry, not merely a supply-chain one.

Why it matters

An image published under a trusted user's name — or their org's name — looks legitimate to every puller. That is the entire supply-chain trust model of the largest container registry in the world, and GhostPat lets a stranger publish into it silently. It is not only poisoning, either: the

minted token can pull the **private repositories** of the targeted user — full read access to whatever that user can read. And the attack requires zero technical knowledge to explain or execute — no exploit, no code, just a handful of clicks in the standard web UI. Anyone with a free Docker account can do it; even your grandma could.

The fix we recommended

Send a confirmation to the receiver *before* the conversion completes and require explicit approval. Carried-over tokens should at minimum be scoped to the org; even better, invalidate them and force the receiver to re-issue them explicitly — to avoid unforeseen consequences.

Q&A

Why "GhostPat"?

A **P**ersonal **A**ccess **T**oken (PAT) — a ghost twice over: its account was deleted in the conversion, yet the token lives on, now haunting an account that isn't its own. No one summoned it, no one was notified, and it behaves exactly like the owner's.

Is it a design flaw or an implementation bug?

A missing consent step in the account→org conversion: the receiver's account is the destination, yet no notification goes to the receiver — and the sender's tokens move with it, keeping full scope.

What should Docker Hub users check right now?

Two places. **1) Tokens** — click your avatar (top-left corner) → Account Settings → Personal Access Tokens. Don't rely on names: the minted token's name is attacker-chosen, so a familiar-looking name proves nothing. Revoke everything you don't actively need and re-issue. **2) Organizations** — check the account switcher for an org you don't remember joining. Those are the traces this bug leaves.

Was any victim data touched?

No, our PoC used consenting test accounts.

What's the CVE?

CVE-2026-105445 — assigned by Docker, the CVE Numbering Authority for its own products. Docker's security advisory.

Is it fixed?

Yes — fixed on 2026-09-30, the same day the report was received. The conversion no longer transfers PATs; the new owner must create their own tokens. The transferred tokens were force-deactivated on 2026-10-08 (unless the account holder opted out), and exposed users were notified on 2026-10-02.

Timeline

2026-09-28	First reproduction (vector 1)
2026-09-29	Second independent reproduction (vector 2), full impact testing
2026-09-30, 11:40	Report received at security@docker.com
2026-09-30, ~1:30	Docker independently reproduced the issue with internal test accounts
2026-09-30, ~3:10	Fix deployed to production and verified against the original PoC — conversion no longer transfers PATs
2026-09-30, ~3:23	Documentation updated: PATs do not transfer and must be recreated by the new owner
2026-09-30, 4:59	We were informed of the fix; both parties agreed to coordinate disclosure
2026-10-02, 5:30	Exposed users and their organization owners notified by email
2026-10-08, 3:45	All PATs transferred through conversion force-deactivated (except where the account holder opted out)
2026-10-12	Public disclosure — Docker's security advisory, CVE-2026-105445

Credit

Discovered, reproduced and reported by mptrdev.com — September 2026. Found by members of the IT security team: **Peter Molnar** and **Krisztian Hozak**.

Submitted to Docker under their Vulnerability Disclosure Policy (security@docker.com).

GhostPat logo is free to use, rights waived via CC0 — download SVG.

Support independent security research — `87H4L2kZxY2Jqind9qV7ZFC48QwrbxaKECgdguNcuoA59tDQefQdfqDaxJ9KP5oc2aepcNWKpHE8i2s4BkT4aPDh4duWwaM` (Monero) · `0x2e58907B29335081C0BF8556932877c9A88e7B29` (ETH, Base and EVM-compatible)

© 2026 mptrdev.com. All rights reserved. · ghostpat.com · contact: security@mptrdev.com · updated 2026-10-13