Fix DNS rebinding TOCTOU bypass in `validate_restricted_url` #21591

Merged **desertaxle** merged 3 commits into `main` from `devin1/oss-7874-fix-dns-rebinding…`

2 weeks ago

devin-ai-integration Bot commented 2 weeks ago — Contributor

Closes OSS-7874.

`validate_restricted_url()` previously resolved a hostname once via `socket.gethostbyname()`, checked if the returned IP was private, then passed the original hostname to httpx. httpx re-resolves DNS at connection time, creating a TOCTOU window: an attacker-controlled DNS server can return a public IP during validation and a private IP during the actual connection, bypassing the SSRF check when `allow_private_urls=False`. `gethostbyname` also only considers the first A record, so a DNS response mixing public and private records could already bypass validation.

This PR closes both windows:

- Hardens `validate_restricted_url` to use `socket.getaddrinfo` so every resolved address is checked (defends against mixed public/private A/AAAA records).
- Adds `SSRFProtectedAsyncHTTPTransport` / `SSRFProtectedHTTPTransport` — httpx transports that wrap the httpcore network backend. On each TCP connect they resolve the hostname, reject private addresses, and pass the validated IP literal to the underlying backend so it cannot re-resolve DNS. TLS SNI is preserved because httpcore passes the original hostname to `start_tls` independently of the host used for the TCP connection.
- `Webhook` and `CustomWebhookNotificationBlock` now use the protected transports whenever `allow_private_urls=False`, in addition to the pre-flight `validate_restricted_url` check.

▶ Verification

Checklist

- [x] This pull request references any related issue by including "closes `<link to issue>`"

- If no issue exists and your change is not a small fix, please create an issue first.

- [x] If this pull request adds new functionality, it includes unit tests that cover the changes
- [ ] If this pull request removes docs files, it includes redirect settings in `mint.json`.
- [x] If this pull request adds functions or classes, it includes helpful docstrings.

Link to Devin session: https://app.devin.ai/sessions/0b65f4db71f04387be5945d92cce8ed5
Requested by: **@desertaxle**

`Fix DNS rebinding TOCTOU bypass in validate_restricted_url` ··· ✓ `f2bad07`

devin-ai-integration Bot assigned **desertaxle** 2 weeks ago

devin-ai-integration Bot commented 2 weeks ago — Contributor — Author

🤖 Devin AI Engineer

I'll be helping with this pull request! Here's what you should know:

✅ I will automatically:

- Address comments on this PR. Add '(aside)' to your comment to have me ignore it.
- Look at CI failures and help fix them

Note: I can only respond to comments from users who have write access to this repository.

⚙️ Control Options:

- [ ] Disable automatic comment and CI monitoring

desertaxle changed the title **Fix DNS rebinding TOCTOU bypass in validate_restricted_url** **Fix DNS rebinding TOCTOU bypass in `validate_restricted_url`** 2 weeks ago

desertaxle added the fix label 2 weeks ago

codspeed-hq Bot commented 2 weeks ago • edited

Merging this PR will not alter performance

✅ 2 untouched benchmarks

Comparing `devin1/oss-7874-fix-dns-rebinding-toctou-bypass-in-validate_restricted_url` (`e0a9ff2`) with `main` (`3458636`)

Open In CodSpeed

desertaxle marked this pull request as ready for review 2 weeks ago

desertaxle requested review from **chrisguidry**, **cicdw**, **desertaxle** and **zzstoatzz** as code owners 2 weeks ago

chatgpt-codex-connector Bot reviewed 2 weeks ago

View reviewed changes

chatgpt-codex-connector Bot left a comment

💡 Codex Review

Here are some automated review suggestions for this pull request.

Reviewed commit: `f2bad07afa`

▶ ℹ️ About Codex in GitHub

`src/prefect/utilities/urls.py` Outdated Show resolved

`Fall back across validated IPs in SSRF-protected backends` ··· ✓ `a080abd`

desertaxle commented 2 weeks ago Member

@**codex** review

👀 1

chatgpt-codex-connector Bot reviewed 2 weeks ago

View reviewed changes

chatgpt-codex-connector Bot left a comment

💡 Codex Review

Here are some automated review suggestions for this pull request.

Reviewed commit: `a080abda8e`

▶ ℹ️ About Codex in GitHub

src/prefect/utilities/urls.py Outdated

```
190 +         local_address: Optional[str] = None,
191 +         socket_options: Optional[Iterable[Any]] = None,
192 +     ) -> httpcore.AsyncNetworkStream:
193 +         validated_ips = _resolve_and_validate_for_connect(host)
```

chatgpt-codex-connector Bot 2 weeks ago

P2 Avoid blocking DNS lookup in async transport backend

`_SSRFProtectedAsyncBackend.connect_tcp()` calls `_resolve_and_validate_for_connect(host)`, which currently uses `socket.getaddrinfo()` synchronously; in async contexts this blocks the event loop during DNS resolution. When DNS is slow or intermittently failing, unrelated coroutines in the same loop can be stalled until resolution returns, which is a regression from the non-blocking behavior of the underlying AnyIO backend.

Useful? React with 👍 / 👎.

👀 1

devin-ai-integration Bot 2 weeks ago Contributor Author

Fixed in e0a9ff2. `_SSRFProtectedAsyncBackend.connect_tcp` now runs `_resolve_and_validate_for_connect` via `anyio.to_thread.run_sync`, so `getaddrinfo` happens on a worker thread and does not block the event loop. Added `test_async_backend_runs_getaddrinfo_off_the_event_loop` to assert the call is dispatched off the loop thread.

src/prefect/utilities/urls.py

Comment on lines +197 to +201

```
197 +                return await self._wrapped.connect_tcp(
198 +                    ip,
199 +                    port,
200 +                    timeout=timeout,
201 +                    local_address=local_address,
```

chatgpt-codex-connector Bot 2 weeks ago

P2 Keep connect timeout bounded across multi-IP retries

The retry loop applies the full `timeout` to each resolved IP attempt, so hosts with multiple addresses (for example, an unreachable AAAA followed by a reachable A record) can take `N * timeout` before succeeding or failing. This can significantly increase webhook/notification latency in dual-stack or partially broken network environments; the retry logic should preserve a single timeout budget across attempts (or emulate happy-eyeballs behavior) instead of resetting it per address.

Useful? React with 👍 / 👎.

👀 1

devin-ai-integration Bot 2 weeks ago Contributor Author

Fixed in `e0a9ff2`. Both backends now derive a shared deadline from the caller's `timeout` and pass the remaining budget to each per-IP attempt, so total connect time stays bounded by the caller's timeout rather than scaling with the number of resolved addresses. Added `test_{async,sync}_backend_shares_timeout_budget_across_retries` covering the shared-budget behavior.

`Run SSRF DNS lookup off the event loop and share retry timeout budget` ··· ✓ `e0a9ff2`

desertaxle approved these changes 2 weeks ago

View reviewed changes

desertaxle merged commit `7c70ac5` into `main` 2 weeks ago

View details

95 of 96 checks passed

desertaxle deleted the `devin1/oss-7874-fix-dns-rebinding-toctou-bypass-in-validate_restricted_url` branch 2 weeks ago

Reviewers

- **chatgpt-codex-connector[bot]**
- **desertaxle**
- **cicdw**
- **zzstoatzz**
- **chrisguidry**

Assignees

- **desertaxle**

Labels

fix

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

1 participant