fix(expressions): avoid helper eval in literal checks #7321

Merged **ehsandeep** merged 2 commits into `dev` from `dwisiswant0/fix/expressions/avoid…` last week

dwisiswant0 commented 3 weeks ago • edited by coderabbitai bot — Member

Proposed changes

fix(expressions): avoid helper eval in literal checks

`hasLiteralsOnly()` currently evaluates helper expressions while deciding whether "{{...}}" contains unresolved variables, which makes validation paths run side-effectful helpers.

Just replace that runtime eval with a `Vars()` len check so unresolved-variable detection literally stays literal (am I writing it right?), and of course side-effect free.

Also make `Evaluate()` return template-authored expression compile/eval errors instead of logging and then keep continuing, so malformed helper calls still fail in the rendering path.

Fixes #7320

Proof

```
$ time ./bin/nuclei -t eval-while-validating.yaml -u "http://localhost:8088/" -var "fur
                     __     _
   ____  __  _____/ /__  (_)
  / __ \/ / / / ___/ / _ \/ /
```

```
 / / / / /_/ / /__/ /  __/ /
/_/ /_/\__,_/\___/_/\___/_/   v3.7.1

        projectdiscovery.io

[INF] Current nuclei version: v3.7.1 (unknown) - remove '-duc' flag to enable update checks
[INF] Current nuclei-templates version: v10.4.1 (unknown) - remove '-duc' flag to enable upda
[WRN] Scan results upload to cloud is disabled.
[INF] New templates added in latest release: 76
[INF] Templates loaded for current scan: 1
[WRN] Loading 1 unsigned templates for scan. Use with caution.
[INF] Targets loaded for current scan: 1
[INF] [eval-while-validating] Dumped HTTP request for http://localhost:8088/?func=wait_for&ar

GET /?func=wait_for&args=5 HTTP/1.1
Host: localhost:8088
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:72.0) Gecko/20100101 Firefox/72.0
Connection: close
Accept-Encoding: gzip

[DBG] [eval-while-validating] Dumped HTTP response http://localhost:8088/?func=wait_for&args=

HTTP/1.1 200 OK
Connection: close
Content-Length: 15
Content-Type: text/plain; charset=utf-8
Date: Thu, 02 Apr 2026 02:37:39 GMT

{{wait_for(5)}}
[INF] [eval-while-validating] Dumped HTTP request for http://localhost:8088/reuse

POST /reuse HTTP/1.1
Host: localhost:8088
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:78.0) Gecko/20100101 Firefox/78.
Connection: close
Content-Length: 20
Accept-Encoding: gzip

data={{wait_for(5)}}
[DBG] [eval-while-validating] Dumped HTTP response http://localhost:8088/reuse

HTTP/1.1 204 No Content
Connection: close
Content-Length: 0
Date: Thu, 02 Apr 2026 02:37:39 GMT

[INF] Scan completed in 3.026557ms. No results found.

real    0m0.816s
user    0m0.472s
sys 0m0.145s
```

Checklist

- [x] Pull request is created against the dev branch
- [x] All checks passed (lint, unit/integration/regression tests etc.) with my changes
- [x] I have added tests that prove my fix is effective or that my feature works
- [x] I have added necessary documentation (if appropriate)

Summary by CodeRabbit

- **Bug Fixes**
  - Expression evaluation now fails immediately with clear error messages instead of skipping, and avoids leaking resolved secret values.
  - Helper/template-like syntax embedded in resolved inputs is no longer executed.
  - Request argument processing aborts correctly on argument-evaluation errors.
- **Tests**
  - Added tests covering expression error handling, helper isolation, unresolved-variable behavior, and argument-evaluation failure handling.

auto-assign bot requested a review from **dogancanbakir** 3 weeks ago

coderabbitai bot commented 3 weeks ago • edited Contributor

Walkthrough

Expression evaluation now returns errors immediately on compile/eval failures and avoids executing expressions to determine "literals-only." Unresolved-marker replacement is gated by presence of unresolved markers and function tokens. Tests added to ensure helpers are not executed during unresolved-variable checks and to validate error messaging.

Changes

Cohort / File(s)	Summary
Evaluation & Replacement `pkg/protocols/common/expressions/expressions.go`	Evaluate now fails fast on compile/eval errors (returns wrapped error). Replacement may be set to unresolved-marker only if unresolved markers exist and the

Cohort / File(s)	Summary
	compiled expression contains govaluate function tokens; incremental placeholder replacement uses the computed `replacement`.
Validation (literals-only check) `pkg/protocols/common/expressions/variables.go`	`hasLiteralsOnly` treats `nil` as literals-only and determines literals-only by checking `len(expr.Vars()) == 0` instead of executing the expression.
Tests & Test Helpers `pkg/protocols/common/expressions/expressions_test.go`, `pkg/protocols/common/expressions/variables_test.go`	Added tests covering: helpers not executed during unresolved-variable checks, error messages for invalid template expressions (no leakage of resolved secret values), and behavior with marker-like substrings. Added `withTestHelperFunction` test helper.
JS Request error path & tests `pkg/protocols/javascript/js.go`, `pkg/protocols/javascript/js_test.go`	`Request.getArgsCopy` now returns immediately on arg-evaluation error (records failure and increments failed requests). New test ensures `ExecuteWithResults` returns an evaluation error (no panic) when args contain invalid expression like `{{base64()}}`.

Estimated code review effort

🎯 4 (Complex) | ⏱️ ~45 minutes

Poem

> 🐰
> I nibbled templates under moonlight bright,
> Kept helpers sleeping, avoided a fright.
> Errors call out, secrets tucked tight,
> Markers stay markers, functions take flight.
> Hooray for tidy burrows tonight! 🥕

▶ 🚥 Pre-merge checks | ✅ 4 | ❌ 1

▶ ✨ Finishing Touches

Comment `@coderabbitai help` to get the list of available commands and usage tips.

dwisiswant0 requested a review from **Ice3man543** 3 weeks ago

coderabbitai bot reviewed 3 weeks ago

View reviewed changes

coderabbitai bot left a comment — Contributor

Actionable comments posted: 2

▶ 🤖 Prompt for all review comments with AI agents

▶ 🪄 Autofix (Beta)

▶ ℹ️ Review info

`pkg/protocols/common/expressions/expressions.go` Outdated — Show resolved

`pkg/protocols/common/expressions/expressions.go` Outdated — Show resolved

`fix(expressions): avoid helper eval in literal checks` ··· ✕ `a3f3c99`

dwisiswant0 force-pushed the `dwisiswant0/fix/expressions/avoid-helper-eval-in-literal-checks` branch from **`0734d48`** to **`a3f3c99`** 3 weeks ago

Compare

coderabbitai bot reviewed 3 weeks ago

View reviewed changes

coderabbitai bot left a comment — Contributor

▶ 🧹 Nitpick comments (1)

▶ 🤖 Prompt for all review comments with AI agents

▶ ℹ️ Review info

 `fix(javascript): avoid nil map panic on arg eval failure` ··· ✓ `64721d8`

neo-by-projectdiscovery-dev bot commented 3 weeks ago • edited

Neo - PR Security Review

No security issues found

Comment `@pdneo help` for available commands. · Open in Neo

coderabbitai bot reviewed 3 weeks ago

View reviewed changes

coderabbitai bot left a comment — Contributor

▶ 🧹 Nitpick comments (1)

▶ 🤖 Prompt for all review comments with AI agents

▶ ℹ️ Review info

ehsandeep approved these changes last week

View reviewed changes

ehsandeep merged commit `6c803c7` into `dev` last week

32 of 33 checks passed

View details

ehsandeep deleted the `dwisiswant0/fix/expressions/avoid-helper-eval-in-literal-checks` branch last week

Sign up for free **to join this conversation on GitHub**. Already have an account? Sign in to comment

Reviewers

- **coderabbitai[bot]**
- **ehsandeep**
- **dogancanbakir**
- **Ice3man543**

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

[BUG] validation executes DSL helpers while checking unresolved variables

2 participants