chainguard-dev / **melange** Public

Code Issues 166 Pull requests 81 Actions Projects Security and qu

Commit 84f3b45

egibs and **antitree** authored on Feb 27 · 55 / 55 · Verified

```
Merge commit from fork

The saveLintResults function constructs output file paths using arch and
pkgName values from .PKGINFO fields without validation. A crafted arch
value like "../../etc" or pkgName with path separators could write files
outside the intended output directory.

This fix adds containsPathTraversal validation that rejects values
containing "..", "/" or filepath.Separator before they are used in
path construction.

Co-authored-by: Mark <mark.manning@chainguard.dev>
```

main · v0.50.3 ••• v0.43.4

1 parent 5829ca4 commit 84f3b45

1 file changed +18

Top

Filter files…

pkg/linter

results.go

Search within code

pkg/linter/results.go

```
@@ -20,13 +20,22 @@ import (
20  20      "fmt"
21  21      "os"
22  22      "path/filepath"
```

```
      23 +      "strings"
23    24
24    25        "github.com/chainguard-dev/clog"
25    26
26    27        "chainguard.dev/melange/pkg/config"
27    28        "chainguard.dev/melange/pkg/linter/types"
28    29    )
29    30
      31 + // containsPathTraversal checks if a string contains path traversal sequences
      32 + // or path separators that could be used to escape the intended directory.
      33 + func containsPathTraversal(s string) bool {
      34 +     return strings.Contains(s, "..") ||
      35 +         strings.Contains(s, string(filepath.Separator)) ||
      36 +         strings.Contains(s, "/")
      37 + }
      38 +
30    39    // saveLintResults saves the lint results to JSON files in the packages
            directory
31    40    func saveLintResults(ctx context.Context, cfg *config.Configuration, results
            map[string]*types.PackageLintResults, outputDir, arch string) error {
32    41        log := clog.FromContext(ctx)
          @@ -37,6 +46,11 @@ func saveLintResults(ctx context.Context, cfg
            *config.Configuration, results map
37    46            return nil
38    47        }
39    48
      49 +     // Validate arch to prevent path traversal
      50 +     if containsPathTraversal(arch) {
      51 +         return fmt.Errorf("invalid arch %q: contains path traversal sequence",
            arch)
      52 +     }
      53 +
40    54        // Ensure the package directory exists
41    55        packageDir := filepath.Join(outputDir, arch)
42    56        if err := os.MkdirAll(packageDir, 0o755); err != nil {
          @@ -45,6 +59,10 @@ func saveLintResults(ctx context.Context, cfg
            *config.Configuration, results map
45    59
46    60        // Save results for each package
47    61        for pkgName, pkgResults := range results {
```

```
      62  +         // Validate pkgName to prevent path traversal
      63  +         if containsPathTraversal(pkgName) {
      64  +             return fmt.Errorf("invalid package name %q: contains path traversal
          sequence", pkgName)
      65  +         }
48    66            // Generate the filename: lint-{packagename}-{version}-r{epoch}.json
49    67            filename := fmt.Sprintf("lint-%s-%s-r%d.json", pkgName,
          cfg.Package.Version, cfg.Package.Epoch)
50    68            filepath := filepath.Join(packageDir, filename)
```

Comments 0

Please sign in to comment.