radareorg / **radare2** Public

Code | Issues 767 | **Pull requests 41** | Discussions | Actions | Projects

Fix #25730 - Base64 encode PDB realnames before use in r2 command #25731

Merged **trufae** merged 1 commit into `radareorg:master` from `jro-calif:fix-pdb-cmdinj`

2 weeks ago

Conversation 4 | Commits 1 | Checks 48 | Files changed 2

jro-calif commented 2 weeks ago • edited — Contributor

- [x] Mark this if you consider it ready to merge
- [ ] I've added tests (optional)
- [ ] I wrote some lines in the book (optional)

Description

In order to fix the command injection caused by the use of unsanitized names obtained from untrusted PDBs in r2 commands, the `base64:` prefix for the `fN` command has been added.

This is in line with other commands such as `CCu` and `agn`, which also deal with arguments that potentially contain dangerous characters.

All realnames produced by the `idpi*` command are now emitted base64 encoded. This allows special characters to be used in function names without causing a security issue.

I have some test binaries that can be added. Let me know if you would like that.

Before:

```
[0x140001000]> idpi*
ERROR: There is no tpi stream in current pdb
f pdb.x____open__a_Calculator_ = 0x140001000 # 0 .text
"fN pdb.x____open__a_Calculator_ x" ;!open -a Calculator #"
```

After:

```
[0x140001000]> idpi*
ERROR: There is no tpi stream in current pdb
f pdb.x____open__a_Calculator_ = 0x140001000 # 0 .text
fN pdb.x____open__a_Calculator_ base64:eCIgOyFvcGVuIC1hIENhbGN1bGF0b3IgIw==
```

Fix radareorg#25730 - Base64 encode PDB realnames before use in r2 co… ··· ✓ 1549f96

jro-calif mentioned this pull request 2 weeks ago

[Security] Command injection caused by lack of sanitization of PDB symbol names #25730

Closed

trufae commented 2 weeks ago — Collaborator

A cleaner fix would be to use the single quote instead of the quoted command syntax, which was in process to be deprecated. Also, str_sanitize now filters newlines, but still, this is a real vulnerability and a proper fix

trufae reviewed 2 weeks ago

View reviewed changes

libr/bin/format/pdb/pdb.c

```
1458      -  pdb->cb_printf ("\"fN pdb.%s %s\"\n", filtered_name, name);
     1458 +  char *b64name = r_base64_encode_dyn ((const ut8 *)name, strl
     1459 +  if (b64name) {
     1460 +      pdb->cb_printf ("fN pdb.%s base64:%s\n", filtered_name,
```

trufae 2 weeks ago — Collaborator

this thing should work well too, without the need of base64 encoding

Suggested change

```
1460 -                                    pdb->cb_printf ("fN pdb.%s
       base64:%s\n", filtered_name, b64name);
1460 +                                    pdb->cb_printf ("'fN pdb.%s
       %s\n", filtered_name, name);
```

trufae 2 weeks ago Collaborator

also, single quote comands execute faster than non-quoted ones because the skip all the command parsing codepaths

trufae approved these changes 2 weeks ago

View reviewed changes

trufae left a comment Collaborator

i'm merging this because i think its worth having a base64 prefix support here and its a critical vulnerability. thanks for reporting and providing a fix all the way down!

1

trufae merged commit `0e38152` into `radareorg:master` 2 weeks ago

43 checks passed

View details

jro-calif deleted the `fix-pdb-cmdinj` branch 2 weeks ago

hhungc mentioned this pull request 2 weeks ago

[security] Command injection via PDB section header name in print_gvars() #25752

Closed

wsparks-vc mentioned this pull request last week

DIBS Request: Radare2 < 6.1.4 Command injection CVEProject/researcher-working-group#27

Closed

nixpkgs-security-tracker Bot mentioned this pull request last week

radare2 < 6.1.4 Command Injection via PDB Parser print_gvars() NixOS/nixpkgs#510500

Open

Reviewers

trufae ✓

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

[Security] Command injection caused by lack of sanitization of PDB symbol names

2 participants