

radareorg / radare2 Public[Code](#) [Issues 773](#) [Pull requests 46](#) [Discussions](#) [Actions](#) [Projects](#)

New issue



# [security] Command injection via PDB section header name in print\_gvars() #25752

✓ ClosedMilestone [6.1.4](#)

hungc opened last week



## Environment

```
Wed Apr 8 03:20:50 UTC 2026
radare2 6.1.3 +1 abi:82 @ linux-x86_64
birth: git.6.1.3 2026-04-08_02:44:23
commit: 4191e273095e1745d898f8b52ec63de414e663d7
options: gpl -0? cs:5 cl:2 make
Linux x86_64
```



## Description

A command injection vulnerability exists in the PDB parser's `print_gvars()` function, **distinct from the one fixed in #25731**. When PDB global symbols are emitted in RAD mode, the raw PE section header name (`sctn_header->name[8]`) from PDB binary data is interpolated into an `f` (flag) command via `%. *s` without sanitization. Because the standard `idp` command internally executes `.idpi*` (which runs RAD output as r2 commands), an attacker-crafted PDB file can achieve arbitrary command execution.

This vulnerability **survives the #25731 fix** because it uses a different injection point (section header name vs. symbol name) and a different output line (`f` flag command vs. `fN` rename command).

## Root cause

## Vulnerable code

libr/bin/format/pdb/pdb.c lines 1451-1460:

```

filtered_name = r_name_filter_dup (r_str_trim_head_ro (name));           // line 1451
pdb->cb_printf ("f pdb.%s = 0x%" PFMT64x " # %d %.*s\n",                // line 1452
    filtered_name,                                                    // line 1453
    (ut64) (img_base + omap_remap ((omap)? (omap->stream): 0,         // line 1454
        gdata->offset + sctn_header->virtual_address)),
    gdata->syntype,                                                    // line 1455
    PDB_SIZEOF_SECTION_NAME,                                          // line 1456
    sctn_header->name);        // <-- NOT sanitized, VULNERABILITY    // line 1457
char *b64name = r_base64_encode_dyn ((const ut8 *)name, strlen (name)); // line 1458
if (b64name) {                                                        // line 1459
    pdb->cb_printf ("fN pdb.%s base64:%s\n", filtered_name, b64name); // line 1460 -- SAFE (

```



`sctn_header->name` is an 8-byte array copied verbatim from the PDB's PE section header stream (`stream_pe.c:31`) with no sanitization. A `\n` (0x0A) byte in the section name terminates the `#` comment on the `f` command line and starts a new `r2` command line. Thus, by crafting a PDB with a section header name containing newlines, the `f` command can be escaped, thereby resulting in arbitrary command execution when the result of `idpi*` is executed.

Although each injected line is limited to 7 characters by the 8-byte section name field, arbitrary-length commands are achieved using a staged hex-encoding technique (inspired by HITCON CTF 2017 "BabyFirst Revenge") that writes hex-encoded command fragments to files, then decodes and executes them via `xxd`.

## Test

[poc\\_pdb\\_cmdinj\\_v2.py](#)

## Reproduction steps

1. Produce `payload.exe` which links to `payload.pdb`
2. `r2 payload.exe`
3. Execute `idp payload.pdb`

## Proposed Fix

Sanitize `sctn_header->name` before interpolation at lines 1457 and 1471, replacing non-printable bytes:

```

char sname[PDB_SIZEOF_SECTION_NAME + 1];
for (int j = 0; j < PDB_SIZEOF_SECTION_NAME; j++) {
    char ch = sctn_header->name[j];
    sname[j] = (ch >= 0x20 && ch < 0x7f) ? ch : '_';
}
sname[PDB_SIZEOF_SECTION_NAME] = '\0';

```



# Reporter

Hung Nguyen (mov) of Calif.io

**hhungc** changed the title ~~Command injection via PDB section header name in print\_gvars()~~ [security] Command injection via PDB section header name in print\_gvars() [last week](#)

**trufae** added a commit that references this issue [last week](#)

Fix [#25752](#) - Another command injection caused by the bad previous fix. d115425

**trufae** last week

Collaborator

yeah as said the previous fix was wrong and incomplete, i just fixed it in a PR. will be merged when the CI finishes to run. thanks

1

**trufae** added this to the [6.1.4](#) milestone [last week](#)

**trufae** added a commit that references this issue [last week](#)

Fix [#25752](#) - Another command injection caused by the bad previous fix. dd135fc

**trufae** closed this as [completed](#) in [5590c87](#) [last week](#)

**pull** added a commit that references this issue [last week](#)

Fix [radareorg#25752](#) - Another command injection caused by the bad pre. 5590c87

**wsparks-vc** mentioned this [10 hours ago](#)

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

[Sign up for free](#) to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

## Metadata

Assignees

No one assigned

---

### Labels

No labels

---

### Type

No type

---

### Projects

No projects

---

### Milestone

✔ **6.1.4**

No due date

---

### Relationships

None yet

---

### Development

No branches or pull requests

---

### Participants

