



Open in app

Get started



Konstantin Burov

Follow

Sep 11 · 7 min read · Listen



Save



# WAPPLES Web Application Firewall Multiple Vulnerabilities



TL:DR

Find out how multiple vulnerabilities in WAPPLES WAF allow an attacker to run arbitrary commands on the device with root privileges as well as access the device with privileges via a backdoor account.

## Vulnerability Summary



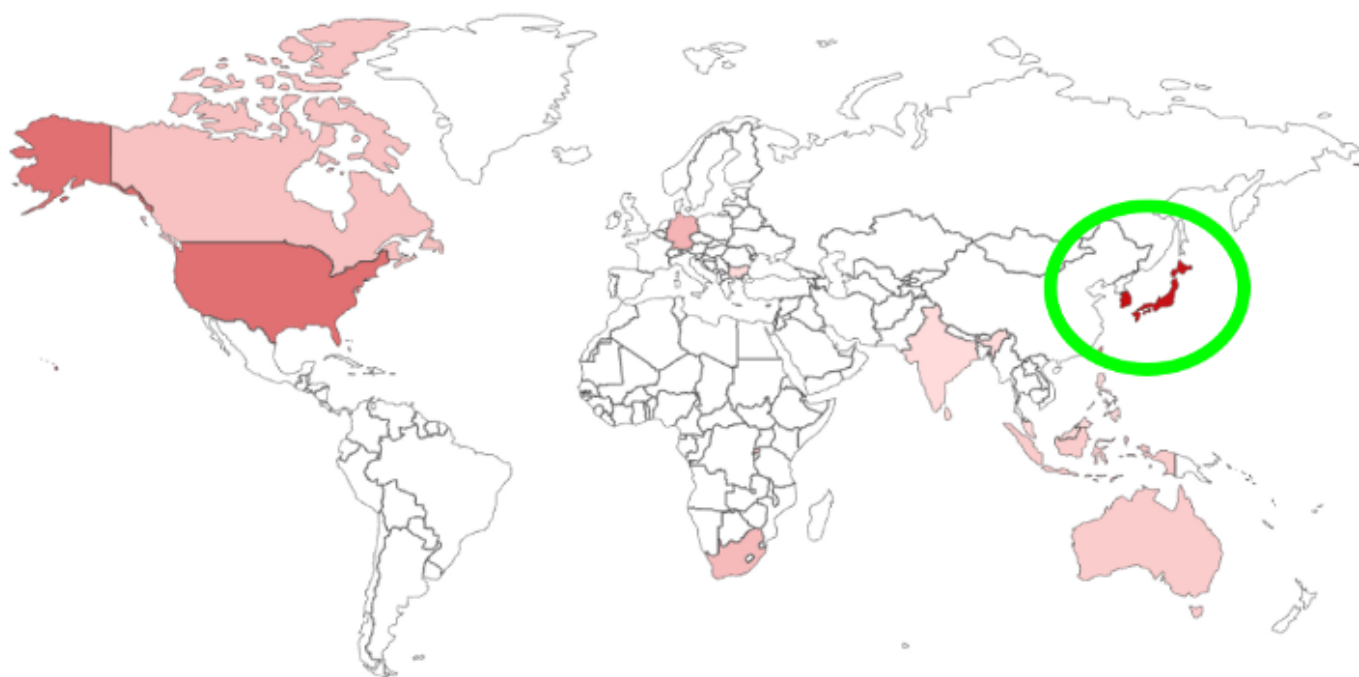
[Open in app](#)[Get started](#)

According to Shodan, it is most common in Korea and Japan.

In versions from 4.0 to 6.0, there are a number of vulnerabilities that allow a remote attacker to execute arbitrary code, obtain confidential information using predefined credentials that are not described in the user manual, as well as the use of vulnerable components. And also escalate user privileges to root in versions 5.0 and 6.0.

### CVE list

CVE-2022-24706, CVE-2022-31322, CVE-2022-35413, CVE-2022-31324, CVE-2022-35582



## Vulnerability Analysis

### Foreword

The vendor assured me that patches or mitigations are ready for all the problems found. And customers are already receiving updates.

I also want to thank Hyeon Jae Jang of Cloudbric Corp. for his professional approach



[Open in app](#)[Get started](#)

certificate is already a security risk, but the decision is up to the user.

```
ssl.cert.fingerprint:017b7408f0aa8da17c47c9fe326577df66ad7f78
```

Using the same certificate and private key on all installed copies carries additional risks for a Man-in-the-middle attack, since it costs nothing for an attacker to get a copy of the private key and the administrator will not notice the catch.

## SSL Certificate

Certificate:

Data:

Version: 3 (0x2)

Serial Number:

95:2a:89:0c:82:2d:08:6b

Signature Algorithm: sha1WithRSAEncryption

Issuer: C=KR, ST=Seoul, O=Penta Security Systems, CN=WAPPLES web access

Validity

Not Before: Jan 12 10:31:50 2012 GMT

Not After : Sep 20 10:31:50 2025 GMT

Subject: C=KR, ST=Seoul, O=Penta Security Systems, CN=WAPPLES web access

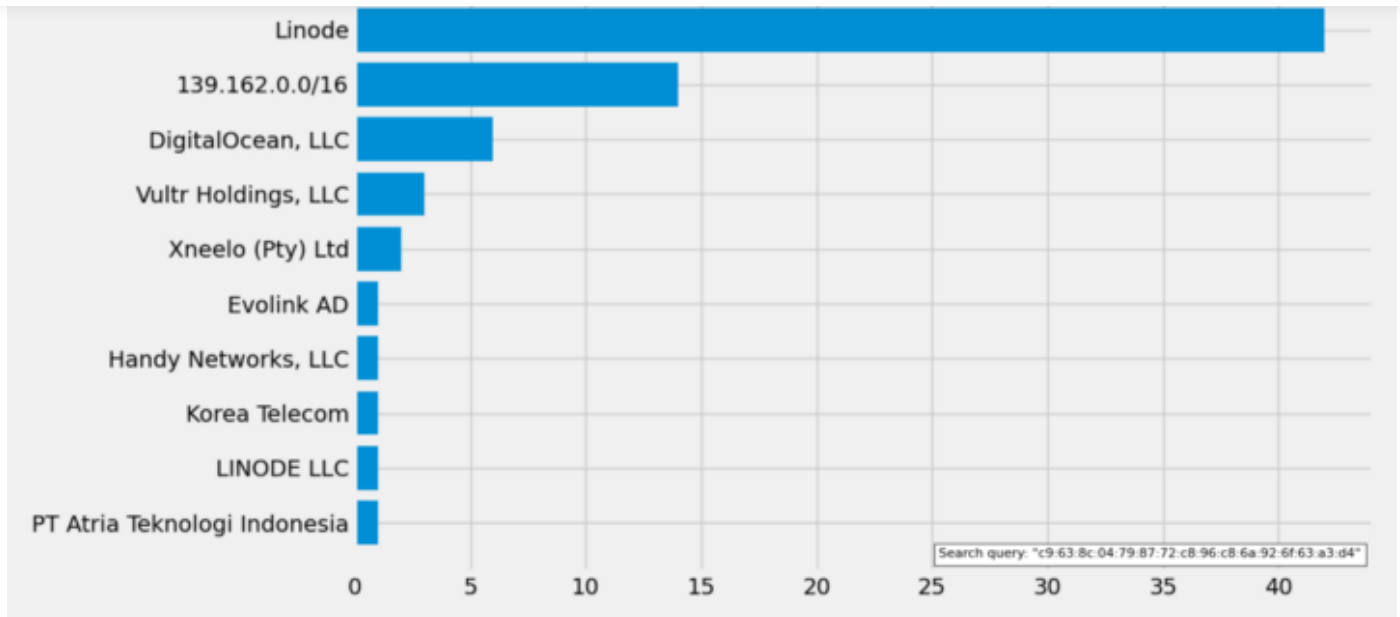
WAPPLES Web-UI certificate on Shodan

Duplicate ssh host keys are also a common problem among device and software manufacturers and WAPPLES is no exception. Digging in Shodan, I noticed the same SSH fingerprints in some instances belong to several cloud providers.

```
c9:63:8c:04:79:87:72:c8:96:c8:6a:92:6f:63:a3:d4
```

Most of the hosts are located in the Linode cloud.



[Open in app](#)[Get started](#)

## How to fix?

For a Web-UI certificate, simply generate a new self-signed certificate, or better yet, either buy a trusted certificate and install it through the WAPPLES management console.

For SSH, remove keys and reconfigure ssh-server if you have access to the root shell. If not, you might use the vulnerabilities below 😊

```
# rm /etc/ssh/ssh_host_*  
# dpkg-reconfigure openssh-server  
# service sshd restart
```

## Hidden Functionality (Backdoor)

The operating system that WAPPLES runs on has a built-in non-privileged user “penta” with a predefined password. The password for this user, as well as its existence, is not disclosed in the documentation. Knowing the credentials, attackers can use this feature to gain uncontrolled access to the device and therefore are considered an undocumented possibility for remote control.



[Open in app](#)[Get started](#)

copy of the program, it will not be difficult to get them. Since the vendor no longer supports these versions, I highly recommend upgrading.

For version 5.0.12.\* attacker can escalate privileges via vulnerability described below.

### How to fix it?

Update to the latest supported version.

*Vulnerable versions: 4.0.\*, 5.0.0.\*, 5.0.12.\**

*CVE-ID: CVE-2022-35582*

### Remote command execution

WAPPLES uses a vulnerable CouchDB version in default configuration that leads to remote OS command execution. To exploit this vulnerability the attacker must have access to the management interface. An attacker could gain unprivileged access to a system as a “couchdb” user, then escalate privileges using the other vulnerabilities described in this document.

You can read more about the exploitation and remediation of this vulnerability in my [write-up](#) about CouchDB vuln.

### How to fix it?

Contact the vendor for a patch and block all unnecessary ports using built-in firewall or look at the link above.



[Open in app](#)[Get started](#)

*Vulnerable versions: 6.0.4.\*, 6.0.6.\**

*CVE-2022-24706*

### **Local privilege escalation**

Built-in utilities for moving files with SUID flag allow an unprivileged user to overwrite any file on behalf of the superuser.

Local unprivileged users are able to overwrite any file on behalf of the superuser.

Which can lead to privilege escalation, such as adding permissions to use sudo without a password to the /etc/sudoers.d/ directory.

For version 6.0.\* this is:

```
/opt/penta/wapples/bin/conf_mover  
/opt/penta/data2/backup/wapples_integrity/bin/conf_mover
```

For version 5.0.12.\* this is:

```
/opt/penta/wapples/bin/confMover  
/opt/penta/backup/wapples_integrity/bin/confMover
```



[Open in app](#)[Get started](#)

First, let's try on an empty file. The example below shows that the unprivileged user *penta* copied the file as root:

It is also possible to copy files accessible only to the superuser:

### How to fix it?

Contact the vendor for a patch or mitigation.

*Vulnerable versions: 5.0.12.\*, 6.0.\**

*CVE-2022-31322*



[Open in app](#)[Get started](#)

issues below are a vulnerability. Nevertheless, I will describe them, because such use of the configurator shell was most likely not intended.

Wapples CLI provides access to a limited subset of standard linux system commands without the ability to launch a bash/sh shell, however the arguments to these commands are not filtered well enough for meta-character content like `&`, ```, `$` etc.

### Obtaining an interactive shell with superuser rights from the “wapples” CLI section

Login with administrator creds and enter the *enable* command to access privileged CLI mode.

Enter to the *wapples* section and run one of the predefined commands with adding `& bash` to the end as command argument. You can see the list of commands by typing `?` instead of command, e.g `command ?`

And now you are **root**:

### How to fix it?

Contact the vendor for a patch or mitigation.

*Vulnerable versions: 6.0.\**

*CVE-ID:none*





[Open in app](#)[Get started](#)

some system scripts.

The account can only be seen in the CouchDB database by connecting to it directly.

Credentials can be used only in API requests for the endpoint `https://<MGMT_IP>[:5001]/webapi/`. The list of API requests with examples can be viewed at `https://<MGMT_IP>[:5001]/docs/`.

There are many interesting things, such as user management, backup settings, SSL certificates and much more.

For example, with such a request, you can get the password from the SMTP account:





[Open in app](#)

Get started

### How to fix it?

Versions 4 and 5 need to be updated as they are no longer supported. For the version 6, you need to get a fix from the vendor.

*Vulnerable versions: 4.0.\* (>=4.0.54.1). 5.0.\*. 6.0.\**



[Open in app](#)[Get started](#)

can use the hard coded *systemi* account described above, so in fact, this is an unauthenticated arbitrary file download.

```
$ curl -k -b 'WP_SESSID=<SESSIONID>' --url \
'https://<MGT_IP>/webapi/file/transfer?
name=../../../../../../../../../../../../etc/passwd&type=db_backup'
```

### How to fix it?

And again, versions 4 and 5 need to be updated as they are no longer supported. For the version 6, you need to get a fix from the vendor.

*Vulnerable versions: 4.0.\* (>=4.0.54.1), 5.0.\*, 6.0.\**

*CVE-ID: none*

### Web-UI arbitrary file download (authenticated)

An arbitrary file download vulnerability in the `downloadAction()` function allows attackers to download arbitrary files via a crafted POST request.

An authenticated user can download an arbitrary file by sending a POST request with the file name in the `file_name` parameter via the link `https://MGMT_IP/report/download`.







Open in app

Get started

- <https://www.exploit-db.com/exploits/50914>
- [https://medium.com/@\\_sadshade/couchdb-erlang-and-cookies-rce-on-default-settings-b1e9173a4bcd](https://medium.com/@_sadshade/couchdb-erlang-and-cookies-rce-on-default-settings-b1e9173a4bcd)
- <https://en.wikipedia.org/wiki/Setuid>

[About](#) [Help](#) [Terms](#) [Privacy](#)

Get the Medium app

