**HTTP Send and Receive Manual Terminal Commands**

We are using **GoFile.io** and **0x0.st** services to upload files because **SCP cannot be used on the AM43x board**.

* No login required on GoFile.io and 0x0.st.
* Only individual **file uploads** are supported (no folder upload).
* Files are uploaded to the **public internet**, accessible via **generated URLs**.

1. **Source**

**cd ~/Downloads/myfiles**  # Navigate to the folder with files

**python3 -m http.server 8000** # Start a local HTTP server on port 8000

This starts a simple HTTP server accessible via:

http://<source-ip>:8000/

Files are available only to other devices on the **same local network**.

1. **Destination**

**wget http://192.168.0.153:8000/myfile.txt**

Replace 192.168.0.153 with the **source/server IP address**.

Replace myfile.txt with the **actual file name**.

**Example for same-machine testing:**

**wget http://127.0.0.1:8000/10thmay.txt**

#### Sample Server Logs (Sender Side):

Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...

192.168.0.130 - - [16/May/2025 10:34:10] "GET /10thmay.txt HTTP/1.1" 200 - # From another machine

127.0.0.1 - - [16/May/2025 10:36:29] "GET /10thmay.txt HTTP/1.1" 200 - # From same machine

1. **Upload File to the Internet (via 0x0.st)**

**curl -F 'file=@your\_file' https://0x0.st**

Replace your\_file with your actual file name.

The command uploads the file and returns a URL like:

https://0x0.st/8tH0.txt

This URL can be used from **anywhere on the internet** to download the file.

#### Retention Policy (0x0.st):

* File stays available for **around 30 days**.
* No guarantee of permanence.
* Anyone with the link can access the file.

**4) Download File from the Internet**

**wget https://0x0.st/8tH0.txt**

* Use this command on **any device with internet access**.
* Replace the URL with the one returned during upload.

**5) Upload File to GoFile.io**

**curl -F "file=@k2e\_nand-linux" https://store1.gofile.io/uploadFile**

Replace k2e\_nand-linux with the name of your file.

This uploads the file to **GoFile.io** and returns a JSON response.

**Example Output:**

{

"data": {

"downloadPage": "https://gofile.io/d/cTH4GA",

"name": "minicomlog.txt",

"size": 105334,

...

},

"status": "ok"

}

Download Link: <https://gofile.io/d/cTH4GA>

#### Retention Policy (GoFile.io):

* File size limit: **5 GB**
* Files are kept **indefinitely** if downloaded **at least once every 30 days**.
* If not accessed for 30 days, the file **may be deleted** automatically.

**Note:**  
After uploading a file to GoFile.io, **you must save the download link** (the downloadPage URL) provided in the response. This link is required to access or share your file later because GoFile.io does not require or use user accounts for managing uploaded files.

**6)Upload File to BashUpload.com**

You can use [https://bashupload.com](https://bashupload.com/) to upload files of any type (including .exe) anonymously from the terminal.

#### Command to Upload:

**curl --upload-file ./your\_file\_name https://bashupload.com/your\_file\_name**

Replace your\_file\_name with the actual filename (e.g., ConsoleApplication1.exe).

The second your\_file\_name part is optional and is only used for display in the URL.

#### Example:

**curl --upload-file ./ConsoleApplication1.exe https://bashupload.com/ConsoleApplication1.exe**

#### Output:

Uploaded 1 file, 110592 bytes

**wget https://bashupload.com/DSAsv/ConsoleApplication1.exe**

**Downloading the File:**

You (or anyone with the link) can download the file using:

wget <https://bashupload.com/DSAsv/ConsoleApplication1.exe>

#### Notes:

File size limit: unofficial, but tested with files up to ~2 GB.

Files are stored for approximately **7 days**, then auto-deleted.

No password protection or account required.

Public link — anyone with it can access the file.