Arbitrary File Upload

An example exploit for the vulnerability in the `Socket.io-file 2.0.31` npm package.

Basic Introduction to Socket.io-file

The `Socket.io-file` npm package is a library that supports file upload functionality. It can be used to create an endpoint for users of a website to upload their files. The library can customize the size of uploaded files, their names, their types, etc.

How to Start the Vulnerable Component

The vulnerabble component is provided as a docker image. You first need to build the image and then run it.

How to Build the Vulnerable Component Image

To build the image, you should go to the `vulnerable-component` directory and then build it as follows:

```
cd vulnerable-component
docker build -t socket-io-file-vul .
```

How to Run the Component on a Container

To run the vulnerable component propertly, you should expose port 3000 as follows:

```
docker run --rm -p 3000:3000 socket-io-file-vul
```

The Content of the Vulnerable Component

The vulnerable component is a node.js website. After running the component, the main page of the website is accessible at `http://localhost:3000`.

The website is supposed to belong to the army of the country CATS. The army is hiring new soldiers. The website uses `Socket.io-file` to allow youths upload their CVs and get recruited!

The website also has a nice gallery of its soldiers served as static files in the `vulnerable-component/public` directory. Once the component is running, you can access one of the images on `http://localhost:3000/cat.jpg`.

Background Theory of the Exploit

When using `Socket.io-file` to upload a file, you can also set the name of the file you are uploading. This makes a lot of sense. The big issue is that instead

of passing the file name, you pass a path! If you do so, the file will be saved in the path you have mentioned.

By default, the uploaded files are supposed to be saved in the `cv-applicants` directory (see this line). However, if you pass a name like `../public/filename.html`, it will be saved in the `public` directory. Note that files in this directory are served as static files. What if a malicious actor upploads a WAR ANNOUNCEMENT to that directory? Let's see what happes below.

Potential Fix for the Exploit

To fix this issue, the developers of `Socket.io-file` have to simply check the passed file name does not set the path. This can be simply done with a regex.

How to Conduct the Attack

After you run the vulnerable component on a container, you can simply exploit it using the `exploit_socket_io_file.py` script. As input, you must pass it the ip and port of the recruitment website as well as the name of a victim country you want to announce an invasion of (!!) as follows:

```
python exploit.py localhost 3000 DOGS
```

This script sends a file to the website and sets its name to `../public/announcements.html`. The content of the file is a declaration of war on the country DOGS! After the file is uploaded, anyone looking at the recruitment website of the country CATS' army at `http://localhost:3000/announcements.html` thinks the CATS have declared war on DOGS!

Exploit Requirement

To run the exploit script, you have to install the socketIO-client-nexus package. If you do not have it on your system, execute the following commands before executing `exploit_socket-io_file.py`:

```
python -m venv .venv
source .venv/bin/activate
pip install socketIO-client-nexus==0.7.6
```

Screencast Demo

You can download the screencast demo here.

References

- Socket.io-file 2.0.31 - Arbitrary File Upload
- Socket.io-file 2.0
- Example for using Socket.io-file