CVE-2019-15813

Score: 8.8+

Multiple file upload restriction
bypass vulnerabilities in Sentrifugo 3.2 could allow
authenticated users to execute arbitrary code via a webshell.

Manish Singh
wearemanishsingh@gmail.com

Introduction

Requirement

Sentrifugo 3.2
Employee login

What is LFI?

An attacker can use Local File Inclusion (LFI) to trick the web application into exposing or running files on the web server. An LFI attack may lead to information disclosure, remote code execution, or even Cross-site Scripting (XSS). Typically, LFI occurs when an application uses the path to a file as input. If the application treats this input as trusted, a local file may be used in the include statement.
Local File Inclusion is very similar to Remote File Inclusion (RFI). However, an attacker using LFI may only include local files (not remote files like in the case of RFI).

Manish Singh
wearemanishsingh@gmail.com

Getting started

Running vulnerable sentrifugo

Sentrifugo has been pre-configured with database seeded in. Just run

This will create, run and install all the things that are required to run The docker container with pre-Seeded values in database.

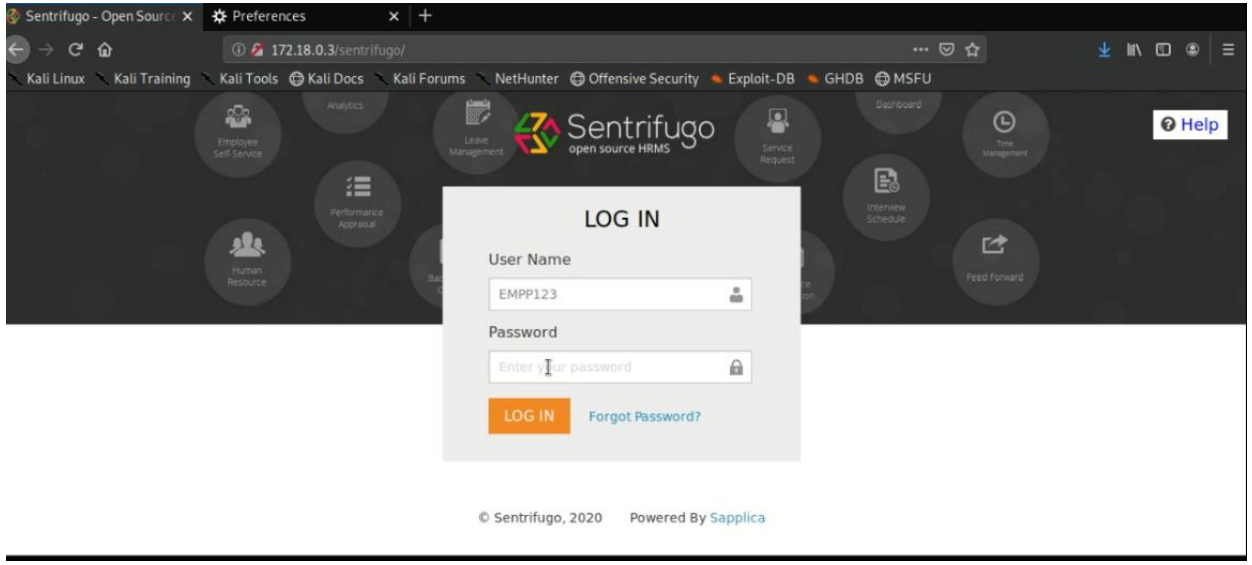
Sentrifugo.zip – Contains 3.2 version files to sentrifugo
Docker-compose.yml – Creational to db and additional services
Dockerfile – Pre-requite actions required
Db-data – Roll back data to db.

Manish Singh
wearemanishsingh@gmail.com

Getting started

Running vulnerable sentrifugo

Run sentrifugo service locally in browser



Use the pre-seeded values to login.

{{This container requires no Configuration and is pre-configured}}

----Super Admin----
Username : empp0001
Password : 5faa7bdf3d7af

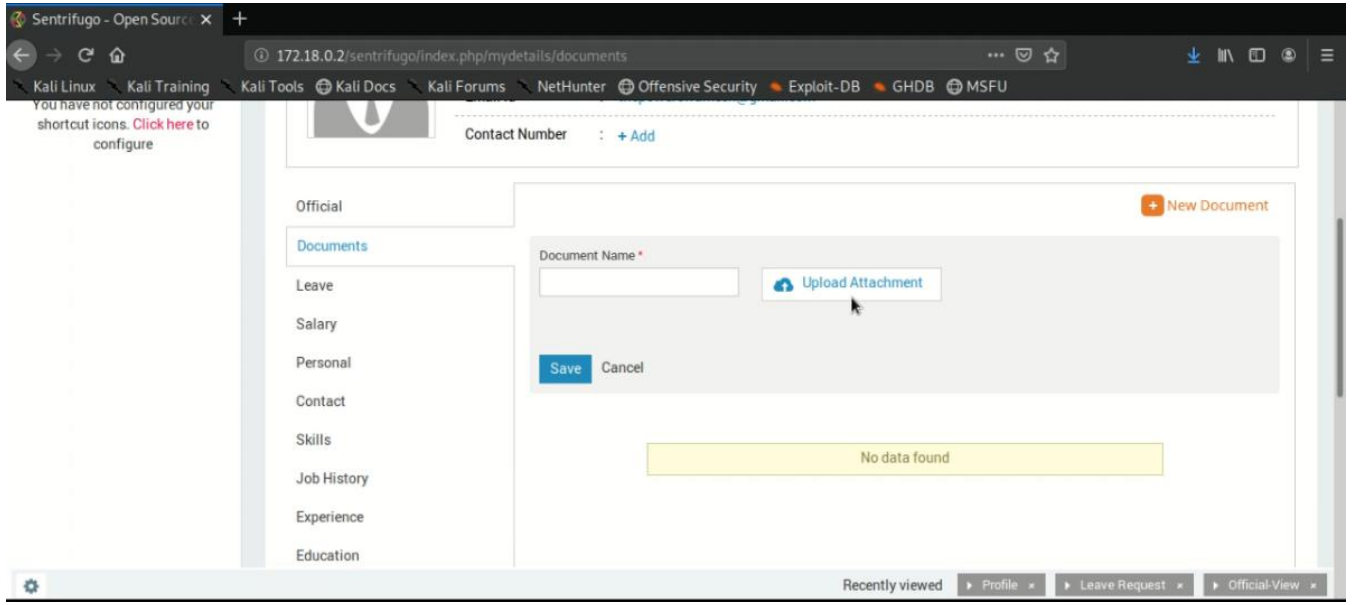
---Employee----
Username: EMPP123
Password: bygedupub

Manish Singh
wearemanishsingh@gmail.com

Exploiting CVE 15813

Sentrifugo dashboard

Multiple File Upload Restriction Bypass vulnerabilities were found in Sentrifugo 3.2. This allows for an authenticated user to potentially obtain RCE via webshell



POC(USING)

/sentrifugo/index.php/mydetails/documents
Self Service >> My Details >> Documents

Upload

Since sentrifugo runs on php, we will simply upload a reverse_shell to our network.

How it works?

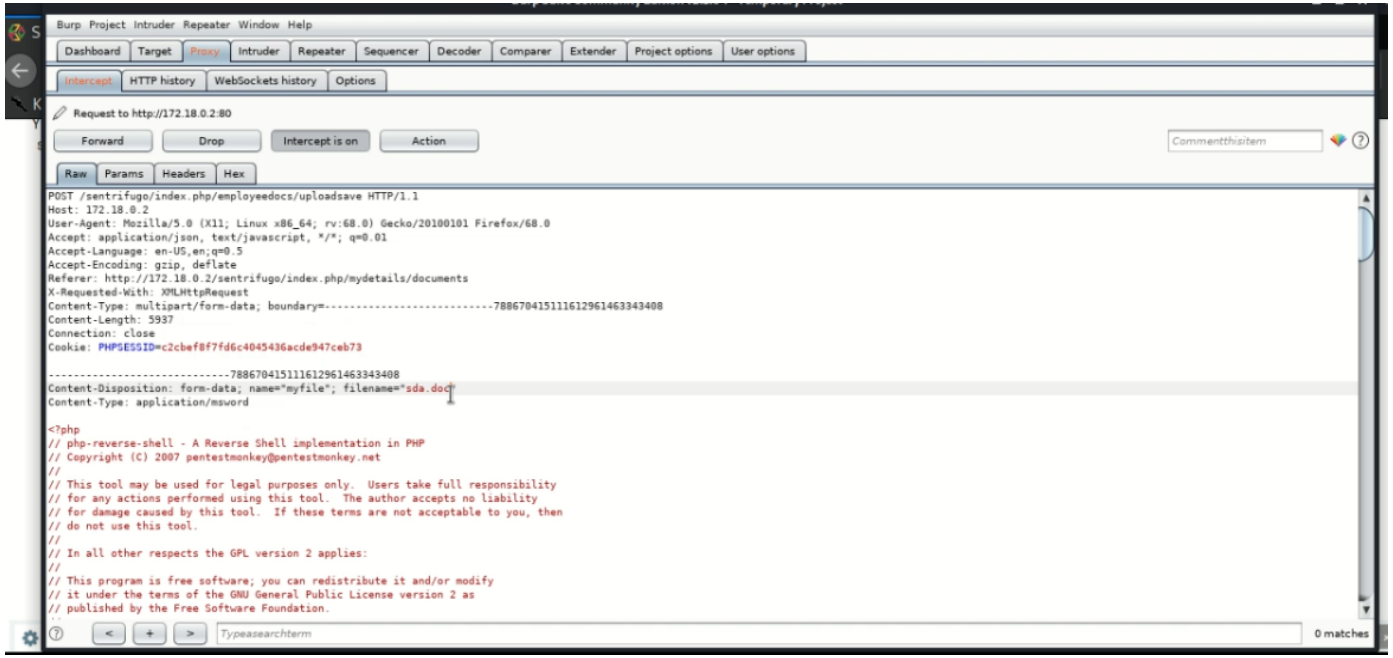
Sentrifugo allows user to upload their documents on webserver, the way it protects against RCE it restricts user from uploading only images and doc files. The verification is done with a extension check

Manish Singh
wearemanishsingh@gmail.com

Exploiting CVE 15813

Using burpsuite

Now that we have uploaded the shell with doc extension to it, we use burpsuite to intercept our traffic and make changes on client side



Making changes

We intercept the ongoing traffic and make changes In extension, so from .doc we change it to .php

Application type

Now that we have made changes in file extension We will now change the application type in http Header. So the server knows what type of file is it.

How it works?

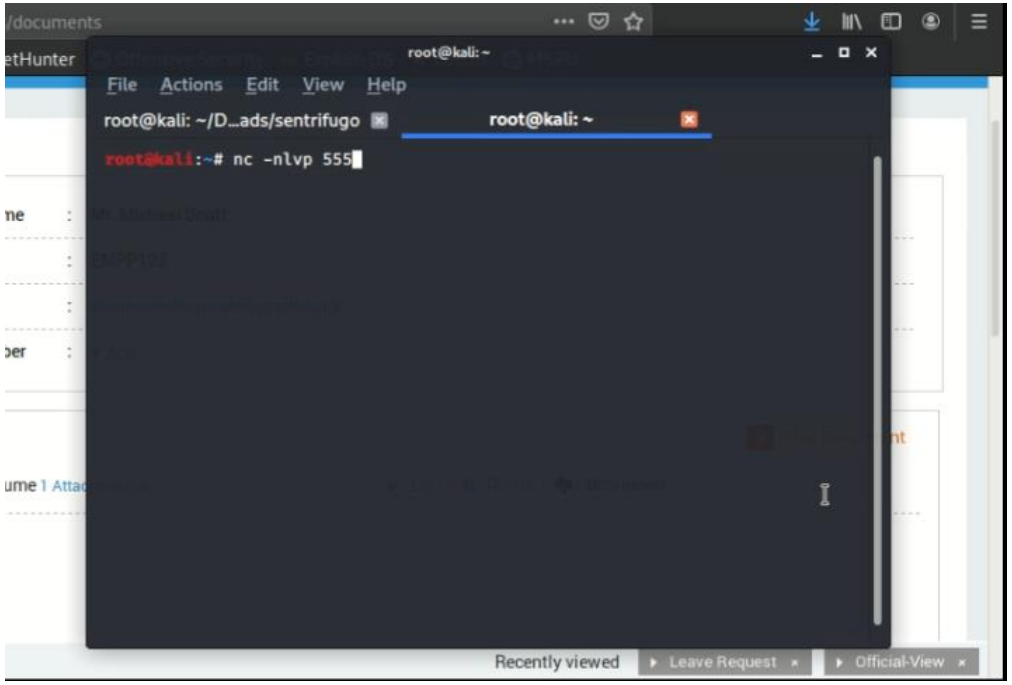
Now that we have tricked the js that does verification of file type on client side, we can intercept the traffic and make changes to file type and the application type to trick the server into uploading a vulnerable php file.

Manish Singh
wearemanishsingh@gmail.com

Exploiting CVE 15813

Using netcat

Now that we have successfully uploaded the shell we create a ongoing listening action on selected port.



NC

-nlvp specifies that we are listening on all the active Network on our system

How it works?

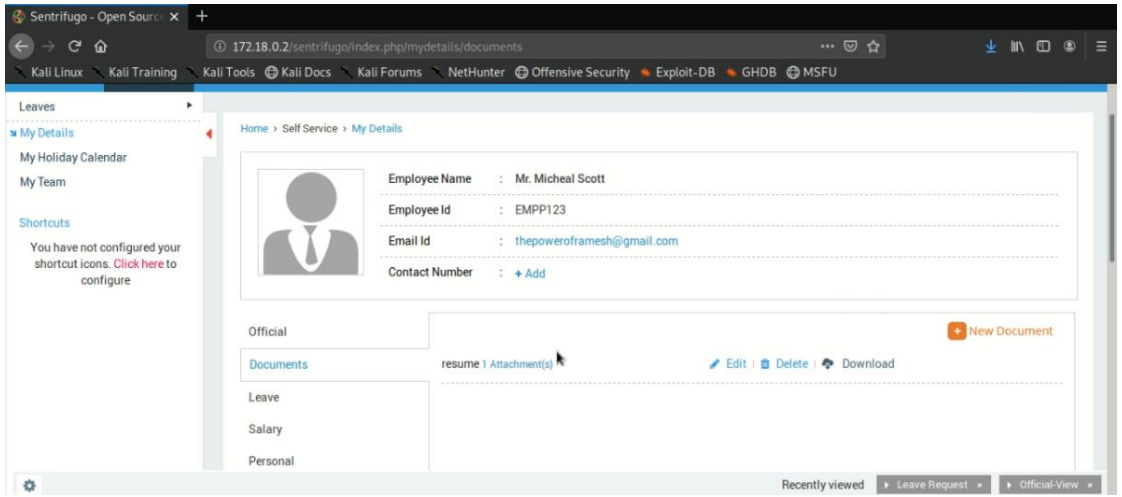
Now as we have started listening, it's time to execute a basic payload at the target so that we could get a reverse shell.

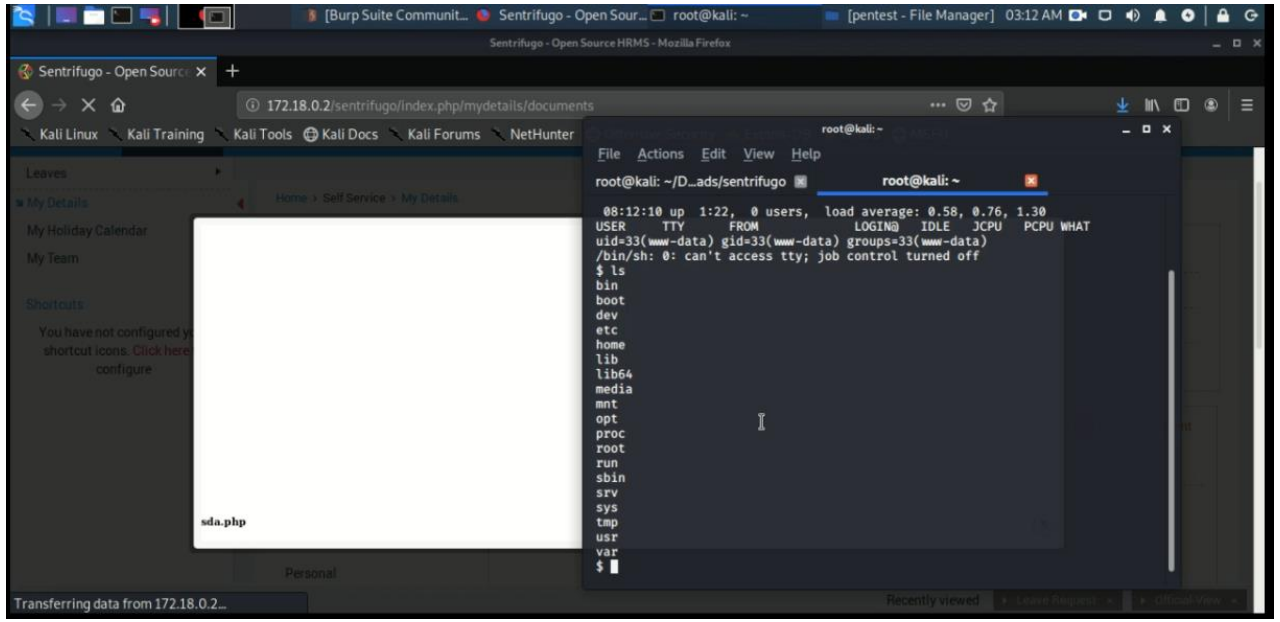
Manish Singh
wearemanishsingh@gmail.com

Exploiting CVE 15813

Execution

We simply open the php file now and we end up getting a reverse shell to out connection.





How it works?

Reverse shell can be to make changes in and get user as well as root shell using privilege escalation.

Manish Singh
wearemanishsingh@gmail.com

This report was generated for applying for internship at Pentester academy

Manish Singh
wearemanishsingh@gmail.com