# Algorithm for file updates in Python

## 

## **Project description**

## As a security professional at a healthcare company, I must regularly maintain an allow list of IP addresses with access to restricted systems. The allow list is stored in a text file named allow\_list.txt. I need to compare it against a remove list containing IP addresses of users who should no longer have access, and remove those IPs from the allow list file to keep access secure.

## **Open the file that contains the allow list**

## import\_file = "allow\_list.txt"

with open(import\_file, "r") as file:

ip\_addresses = file.read()

## **Read the file contents**

## I use the .read() method to read all the contents of the file as a single string. This string contains all the allowed IP addresses separated by spaces.

## **Convert the string into a list**

## ip\_addresses = ip\_addresses.split()

## **Iterate through the remove list**

I use a *for loop* to iterate over each IP address in the remove\_list. For each IP, I check if it exists in the ip\_addresses list.

**Remove IP addresses that are on the remove list**

for ip in remove\_list:

if ip in ip\_addresses:

ip\_addresses.remove(ip)

## **Update the file with the revised list of IP addresses**

ip\_addresses = " ".join(ip\_addresses)

with open(import\_file, "w") as file:

file.write(ip\_addresses)

## **Summary**

This algorithm ensures that the allow list for restricted access is always up to date by removing IP addresses that should no longer have access. It uses file handling (open, read, write), list operations (split, remove, join), and control flow (for, if) to accomplish the task in a secure, automated way.