

# Portfolio piece #6. Update a file with a python algorithm

## Project description

In this project, I developed a Python algorithm to manage access control in a healthcare setting. My task was to update a file containing the list of allowed IP addresses for employees accessing restricted subnet areas. The changes were based on a list of IPs that needed to be removed. The algorithm reads the current allow list, removes unauthorized IPs from the remove list, and updates the file accordingly. This script automates a common cybersecurity task—managing file-based access control.

## Open the file that contains the allow list

```
Python
import_file = "allow_list.txt"
with open(import_file, "r") as file:
```

This uses the with statement for safe file handling, automatically closing the file after reading. The open() function is used with mode "r" to open the file in read mode. file is the variable that stores the file object.

## Read the file contents

```
Python
ip_addresses = file.read()
```

The .read() method reads the entire file content into a string and stores it in the ip\_addresses variable. This is necessary before splitting the content into a list of IP addresses.

## Convert the string into a list

Python

```
ip_addresses = ip_addresses.split()
```

This uses the `.split()` method to break the string into a list using whitespace as the default delimiter. Each IP address becomes an individual list element.

## Iterate through the remove list

Python

```
remove_list = ["192.168.1.10", "10.0.0.2"]
for element in remove_list:
```

This for loop iterates through each item in `remove_list`, assigning each value to `element` for use inside the loop.

## Remove IP addresses that are on the remove list

Python

```
if element in ip_addresses:
    ip_addresses.remove(element)
```

The if statement checks whether the IP in `element` exists in `ip_addresses`. If it does, the `.remove()` method deletes it. This method is safe to use here because there are no duplicate entries in the allow list.

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

```
Python
updated_ips = "\n".join(ip_addresses)

with open(import_file, "w") as file:
    file.write(updated_ips)
```

The `.join()` method converts the list back into a string, separating each IP with a newline character. Then another `with` statement is used to open the file in write mode ("w"), and `.write()` updates the file with the revised content.

## Summary

This algorithm automates the task of updating a restricted-access IP allow list. It opens and reads the original file, parses the content into a list, and uses a loop to remove unauthorized IPs listed in a separate remove list. The updated list is then written back to the original file. Key Python tools used include the `with` statement for safe file access, the `open()` function with "r" and "w" modes, the `.read()` and `.write()` methods for file manipulation, the `.split()` and `.join()` methods for string/list conversion, and a `for` loop with `.remove()` for filtering.