Algorithm for file updates in Python

Project description

This project focuses on updating an "allow list" in a file by removing specified IP addresses. The scenario involves managing access control to restricted information by maintaining an allow list that can be dynamically updated. The process begins by opening a file that contains the allow list, reading its contents, and converting the data from a string into a list of IP addresses. We then iterate through a "remove list" of IP addresses that are no longer authorized to access the information. If an IP address in the allow list matches one in the remove list, it is removed. Finally, the updated list of IP addresses is written back to the file, ensuring that only the allowed IPs remain.

Open the file that contains the allow list

```
# Assign `import_file` to the name of the file
import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.
remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# First line of `with` statement to open the file in read mode
with open(import_file, 'r') as file:
    # You can read or process the file content here
    content = file.read()
    print(content)
```

Read the file contents

In [2]:
```
# Assign `import_file` to the name of the file
import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.
remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file
with open(import_file, "r") as file:
    # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`
    ip_addresses = file.read()

# Display `ip_addresses`
print(ip_addresses)
```

```
ip_address
192.168.25.60
192.168.205.12
192.168.97.225
192.168.6.9
192.168.52.90
192.168.158.170
192.168.90.124
192.168.186.176
192.168.133.188
192.168.203.198
192.168.201.40
192.168.218.219
192.168.52.37
192.168.156.224
192.168.60.153
192.168.58.57
192.168.69.116
```

Convert the string into a list

In [3]:
```
# Assign `import_file` to the name of the file
import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.
remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file
with open(import_file, "r") as file:
    # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`
    ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list
ip_addresses = ip_addresses.splitlines()  # This splits the string into a list by newline characters

# Display `ip_addresses`
print(ip_addresses)
```

```
['ip_address', '192.168.25.60', '192.168.205.12', '192.168.97.225', '192.168.6.9', '192.168.52.90', '192.168.158.170', '192.16
8.90.124', '192.168.186.176', '192.168.133.188', '192.168.203.198', '192.168.201.40', '192.168.218.219', '192.168.52.37', '192.
168.156.224', '192.168.60.153', '192.168.58.57', '192.168.69.116']
```

Iterate through the remove list

In [6]:
```
# Assign `import_file` to the name of the file
import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.
remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file
with open(import_file, "r") as file:
    # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`
    ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list
ip_addresses = ip_addresses.split()

# Build iterative statement
# Name loop variable `element`
# Loop through `ip_addresses`
for element in ip_addresses:
    # Display `element` in every iteration
    print(element)
```

```
ip_address
192.168.25.60
192.168.205.12
192.168.6.9
192.168.52.90
192.168.90.124
192.168.186.176
192.168.133.188
192.168.203.198
192.168.218.219
192.168.52.37
192.168.156.224
192.168.60.153
192.168.69.116
```

Remove IP addresses that are on the remove list

In [7]:
```
# Assign `import_file` to the name of the file
import_file = "allow_list.txt"

# Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.
remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

# Build `with` statement to read in the initial contents of the file
with open(import_file, "r") as file:
    # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`
    ip_addresses = file.read()

# Use `.split()` to convert `ip_addresses` from a string to a list
ip_addresses = ip_addresses.split()

# Build iterative statement
# Name loop variable `element`
# Loop through `ip_addresses`
for element in ip_addresses:
    # Build conditional statement
    # If current element is in `remove_list`,
    if element in remove_list:
        # then current element should be removed from `ip_addresses`
        ip_addresses.remove(element)

# Display `ip_addresses`
print(ip_addresses)
```

```
['ip_address', '192.168.25.60', '192.168.205.12', '192.168.6.9', '192.168.52.90', '192.168.90.124', '192.168.186.176', '192.16
8.133.188', '192.168.203.198', '192.168.218.219', '192.168.52.37', '192.168.156.224', '192.168.60.153', '192.168.69.116']
```

Update the file with the revised list of IP addresses

```
In [9]: # Assign `import_file` to the name of the file
        import_file = "allow_list.txt"

        # Assign `remove_list` to a list of IP addresses that are no longer allowed to access restricted information.
        remove_list = ["192.168.97.225", "192.168.158.170", "192.168.201.40", "192.168.58.57"]

        # Build `with` statement to read in the initial contents of the file
        with open(import_file, "r") as file:
            # Use `.read()` to read the imported file and store it in a variable named `ip_addresses`
            ip_addresses = file.read()

        # Use `.split()` to convert `ip_addresses` from a string to a list
        ip_addresses = ip_addresses.split()

        # Build iterative statement
        # Name loop variable `element`
        # Loop through `ip_addresses`
        for element in ip_addresses:
            # Build conditional statement
            # If current element is in `remove_list`,
            if element in remove_list:
                # then current element should be removed from `ip_addresses`
                ip_addresses.remove(element)

        # Convert `ip_addresses` back to a string so that it can be written into the text file
        ip_addresses = " ".join(ip_addresses)

        # Build `with` statement to rewrite the original file
        with open(import_file, "w") as file:
            # Rewrite the file, replacing its contents with `ip_addresses`
            file.write(ip_addresses)
```

Summary

In this project, I developed a Python script that automates the process of managing an IP allow list. The script reads the contents of a file containing a list of authorized IP addresses, identifies and removes any IPs that should no longer have access (as defined in the remove list), and then updates the file with the revised list. This solution ensures that the file is kept up-to-date and prevents unauthorized IP addresses from accessing restricted data. The ability to automate such tasks enhances security and streamlines the management of access control lists in a network or server environment.