

Lab 1: Basic Lua Handling

Lab 1: Basic Lua Handling

Objective

Write a Lua script to read a JSON file containing ethernet port configurations and modify MAC addresses.

Implementation

data.json

```
{
  "ports": {
    "eth0": { "mac": "6C:10:8B:00:45:00", "name": "eth0", "link":
      false, "autoneg": false },
    "eth1": { "mac": "6C:10:8B:00:44:01", "name": "eth1", "link":
      false, "autoneg": true },
    "eth2": { "mac": "6C:10:8B:00:46:02", "name": "eth2", "link":
      false, "autoneg": true },
    "eth3": { "mac": "6C:10:8B:00:47:03", "name": "eth3", "link":
      false, "autoneg": true },
    "eth4": { "mac": "6C:10:8B:00:48:04", "name": "eth4", "link":
      false, "autoneg": true },
    "eth5": { "mac": "6C:10:8B:00:49:05", "name": "eth5", "link":
      true, "speed": 1000, "duplex": "full", "autoneg": true }
  }
}
```

script.lua

```
local cjson = require "cjson"

local function read_file(path)
```

```
    local file, err = io.open(path, "r")
    if not file then return nil, err end
    local content = file:read("*a")
    file:close()
    return content
end

local function increment_mac(mac)
    local prefix = mac:sub(1, 15)
    local last_byte = tonumber(mac:sub(16, 17), 16)
    last_byte = (last_byte + 1) % 256
    return string.format("%s%02X", prefix, last_byte)
end

local function main()
    local content, err = read_file("data.json")
    if not content then
        print("Error reading file: " .. err)
        return
    end

    local data = cJSON.decode(content)

    print("Processing Ports and Modifying MAC Addresses:")
    print("-----")

    for k, v in pairs(data.ports) do
        local old_mac = v.mac
        v.mac = increment_mac(old_mac)
        print(string.format("Port: %s | Old MAC: %s | New MAC: %s",
            v.name, old_mac, v.mac))
    end
    print("-----")
end

main()
```

Output

Processing Ports and Modifying MAC Addresses:

Port: eth3 | Old MAC: 6C:10:8B:00:47:03 | New MAC: 6C:10:8B:00:47:04

Port: eth0 | Old MAC: 6C:10:8B:00:45:00 | New MAC: 6C:10:8B:00:45:01

```
Port: eth5 | Old MAC: 6C:10:8B:00:49:05 | New MAC: 6C:10:8B:00:49:06
Port: eth4 | Old MAC: 6C:10:8B:00:48:04 | New MAC: 6C:10:8B:00:48:05
Port: eth1 | Old MAC: 6C:10:8B:00:44:01 | New MAC: 6C:10:8B:00:44:02
Port: eth2 | Old MAC: 6C:10:8B:00:46:02 | New MAC: 6C:10:8B:00:46:03
-----
```

Conclusion

The script successfully reads JSON data using the cJSON library and increments the last byte of each MAC address. This demonstrates basic Lua file I/O, JSON parsing, and string manipulation.