

Script1.txt

```
return {
  on = {
    devices = {'E16Conf'} -- The name of this switch is to be adapted.
  },
  execute = function(domoticz, Switch)
    if (Switch.state == 'On') then

      -- Fetch the switch that started this script
      local MySwitch=domoticz.devices(Switch.name)
      -- print(MySwitch.name)

      -- Retrieve the IP address and the Devices list, stored in the
      description of the Switch
      -- and store them in table MyVars
      local MyDesc=MySwitch.description
      local MyVars = {}
      for MyVar in string.gmatch(MyDesc, "[^|]+") do
        MyVars[#MyVars + 1] = MyVar
        -- print(#MyVars .. '-' .. MyVar)
      end

      -- Retrieve the IP address of the ESPEasy with NEXTION, stored in a
      uservariable pointed to
      local NextionIP = MyVars[1]
      NextionIP = domoticz.variables(NextionIP).value
      -- Retrieve the list of devices
      local MyDevices = MyVars[2]
      MyDevices = domoticz.variables(MyDevices).value

      -- catch all devices in a table
      local DeviceNames = {}
      for DeviceName in string.gmatch(MyDevices, "[^|]+") do
        DeviceNames[#DeviceNames + 1] = DeviceName
        -- print(#DeviceNames .. '-' .. DeviceName)
      end

      -- Loop round each of the devices and get their descriptions.
      for i,DeviceName in ipairs(DeviceNames) do
        local MyDevice=domoticz.devices(DeviceName)
        MyDesc=MyDevice.description
        local MyStateList = MyDevice.state

        -- display some interesting details
        print("-----")
        print('MyName = ' .. MyDevice.name)
        print('MyDescription = ' .. MyDevice.description)
        print('MyStatelist = ' .. MyStateList)
        print('MyType = ' .. MyDevice.deviceType)
      end
    end
  end
}
```

```

Script1.txt
print('MySubType = ' .. MyDevice.deviceSubType)

-- destilate the commandsets from the description and put them in a
table
local MyCommands = {}
for MyCommand in string.gmatch(MyDesc, "[^|]+") do
    MyCommands[#MyCommands + 1] = MyCommand
--    print(#MyCommands .. '-' .. MyCommand)
end

-- a commandset consists of the name of the device attribute to
retrieve and the IP string
-- that will be send to ESPEasy. Example nValue|page0.t1.val=
local SubAtt = 1

-- for each commandset, retrieve the information from the domoticz
device
while (SubAtt < #MyCommands) do
    local MyIndex=MyCommands[SubAtt]
    local MyIPString=MyCommands[SubAtt+1]
    local MyAttrib=MyDevice[MyIndex]

    -- put some quotes around the value, when it has to be treated
as a string
    if(string.find(MyIPString,"txt")) then
        MyIPString = MyIPString .. '"'
        MyAttrib = MyAttrib .. '"'
    end

    print(MyIPString .. MyAttrib)

    -- create a complete URL and send it
    local MyURL ='http://' .. NextionIP .. '/control?cmd=NEXTION,'
.. MyIPString .. MyAttrib
    domoticz.openURL({
        url = MyURL
    })
    print('URL =' .. MyURL)
    -- do that for every device attribute you want to update
    SubAtt=SubAtt+2
end
-- loop for every device in the uservariable
end

print("-----")
MySwitch.switchOff()

end

```

Script1.txt

} end