

Lua + Smart Card make it easy

1. Copy the binary (LuaSmartCardLibrary.dll) to the ZeroBrane directory to the same location where Lua53.dll is located
2. Open the zerobrane and run this script:

```
--load all the module from the dll
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_card")()
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_sam")()
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_log")()
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_bytes")()
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_asn1")()
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_crypto")()
package.loadlib("LuaSmartCardLibrary.dll", "luaopen_luasql_odbc")()

--the place to save or log
local msg = log.open_logfile("log\\log_dump.txt", 5)
print("msg = " .. msg)

card = pcsc_card
sam = pcsc_sam

-----
-- FUNCTIONS
-----

function verify_sw(recv_sw, ref_sw)
    if (recv_sw ~= ref_sw) then
        card.disconnect_reader()
        error("Received " .. string.format("sw = %x", recv_sw) .. " Expect " ..
string.format("sw = %x", ref_sw))
        return false    --not verified
    end

    return true
end

-----
-- START MAIN
-----

local readers = card.list_readers()
log.print(log.DEBUG, "Available readers = " .. readers)

local isInitialized = card.init_reader()
if (isInitialized) then
    log.print(log.DEBUG, "Reader Initialized")
else
    log.print(log.ERROR, "Reader Not Initialized")
    error("Fail to initialize the reader/encoding station")
    return
end

--connect to card
```

```

-- put the name of your card reader
local reader_name = "ACS ACR122 0"
local isconnected = card.connect_reader(reader_name)
if (isconnected == false) then
    error("Cannot connect to reader")
    return
end

local atr = card.get_atr()
log.print(log.INFO, "Card ATR = " .. atr)

--connect to sam, put the name of your card reader
reader_name = "O2Micro CCID SC Reader 0"
isconnected = sam.connect_reader(reader_name)
if (isconnected == false) then
    error("Cannot connect to reader")
    return
end
atr = card.get_atr()
log.print(log.INFO, "Card ATR = " .. atr)

--select applet
log.print(log.DEBUG, "Select DF ")
sw, response = card.send_auto("00A404000BF7534D4152442E4D61696E")
verify_sw(sw, 0x9000)

--select sam
log.print(log.DEBUG, "Select DF VSDC")
sw, response = sam.send_auto("00A4040007A0000000031010")
verify_sw(sw, 0x9000)

--close log file
log.close_logfile()

```