

Pseudo-random hex character generator

Use Case:

Pseudo-random hex characters can be used in the authentication and validation flow across the ADC and Client/Server. The hex characters can be used for different purposes as well.

F5 iRule:

```
# Generate 32 random hex characters
when RULE_INIT {

    for { set i 0 } {$i < 32 } { incr i } {
        append rand_hex [format {%x} [expr {int(rand()*16)}]]
    }
    log local0. "\$rand_hex: $rand_hex"
}
```

Netscaler Solution:

We are using Policy Extensions (custom written LUA function) to achieve this

LUA Script:

LUA Script:

```
function NSNUM:hex_generator() : NSTEXT
    local input = self
    local out = ""
    if input%2 ~= 0 then
        ns.logger:info("Input should be an Even number")
        return(out)
    end
    local s = 0 -- Start string
    for i = 1, input do
        s = math.random(0, input/2+1)
        out = out .. string.format('%X',s)
    end
    return(out) -- Return string
end
```

The above script should be named as rand_hex.lua and placed in /var/tmp directory in Netscaler for the below configs to work.

Policy and Action configuration

```
add responder action act1 respondwith "\"HTTP/1.1 200 OK\r\nHEX:\r\n"+http.req.header(\"Hex_Input\").typecast_num_at.hex_gen+\"\\r\\n\\r\\n\""  
add responder policy pol1 true act1
```

Bind the above policy to response side Global or vserver bind point. LUA scripts are powerful tool on NetScaler which allow you to do a specific task which cannot be done in the policy flow but the scripts can be called from any policy flow.