The bridge to possible

TCL & EEM Scripts and Guest Shell

Scripting the network

Stanley Fofano, Technical Consulting Engineer

TACENT-2015

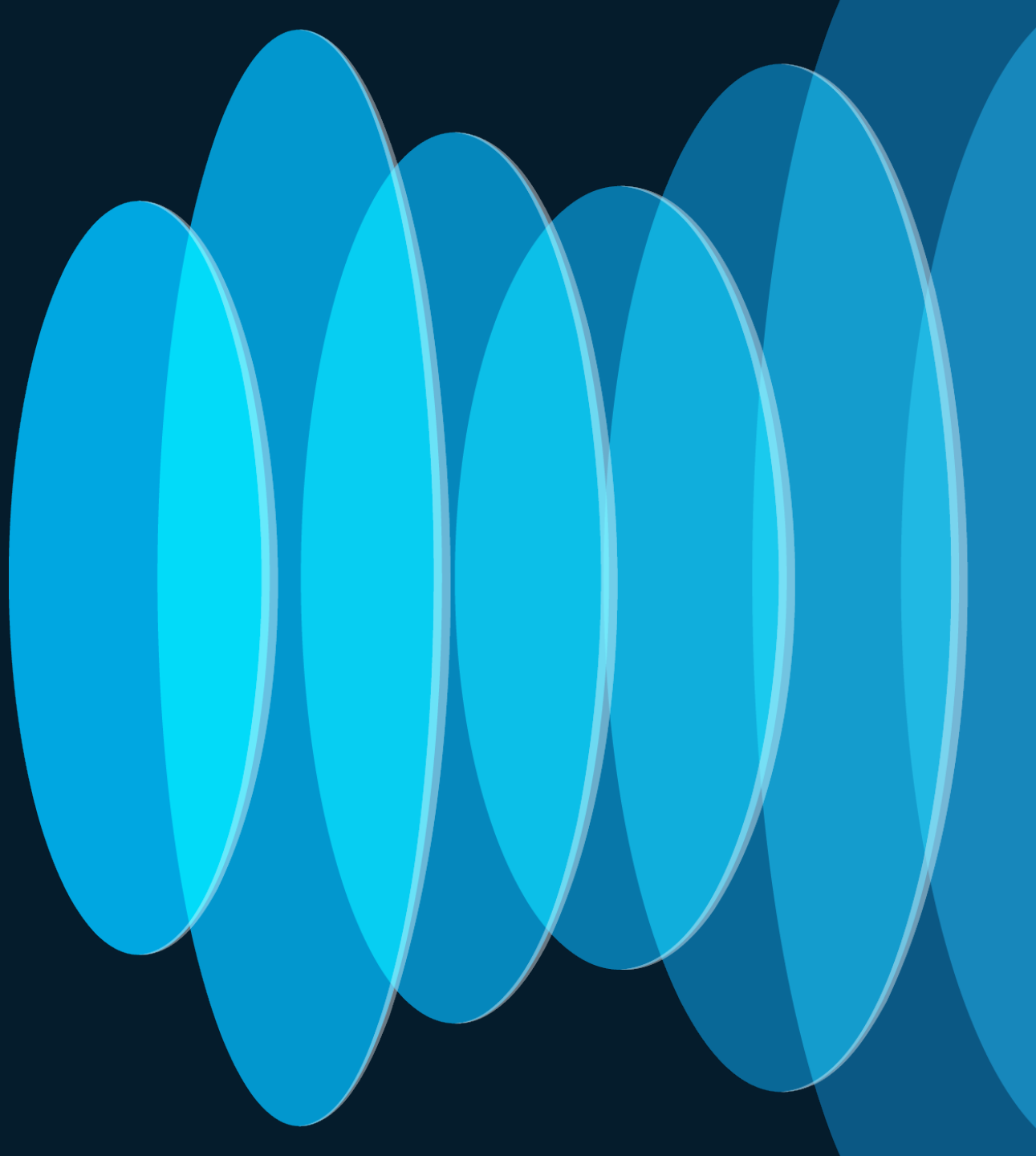
cisco Live!

#CiscoLive

Agenda

- Introduction
- TCL
- EEM
- Guest Shell
- Conclusion

Introduction



Acing Chemistry

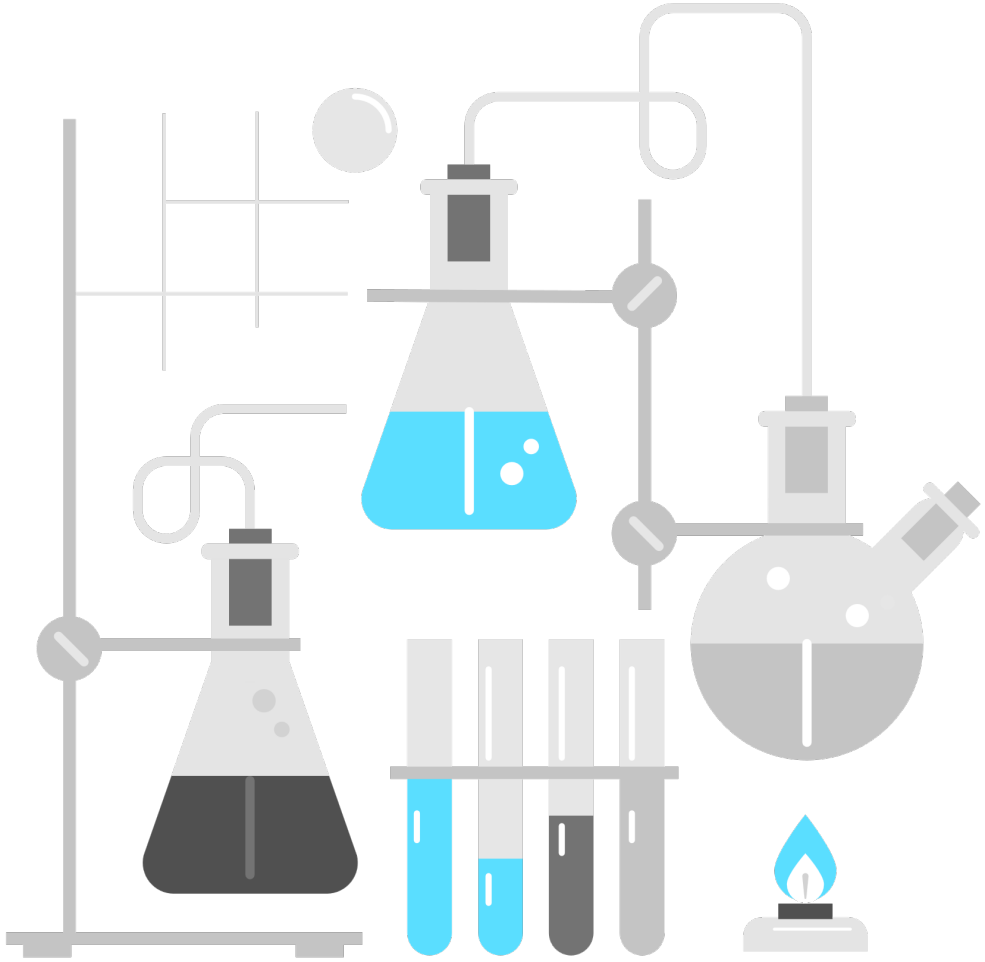
Scripting to the rescue

Challenges

- Complex equations
- Large constants
  - 1 mole = $6.02214076 \times 10^{23}$
- Strict order of operation
- Time

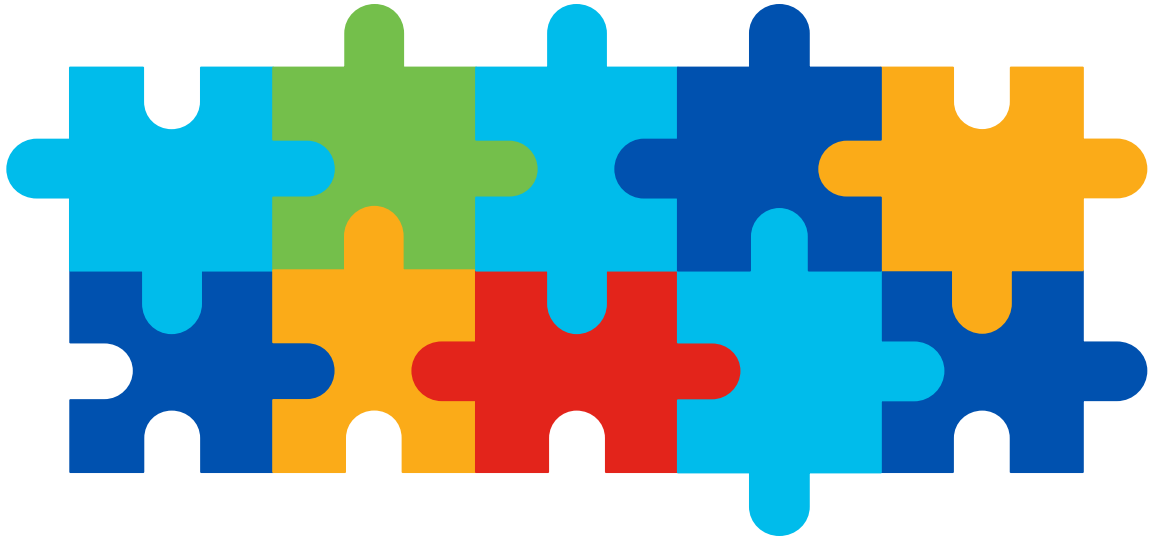
Solution

- Programmability
- Creativity
- Availability



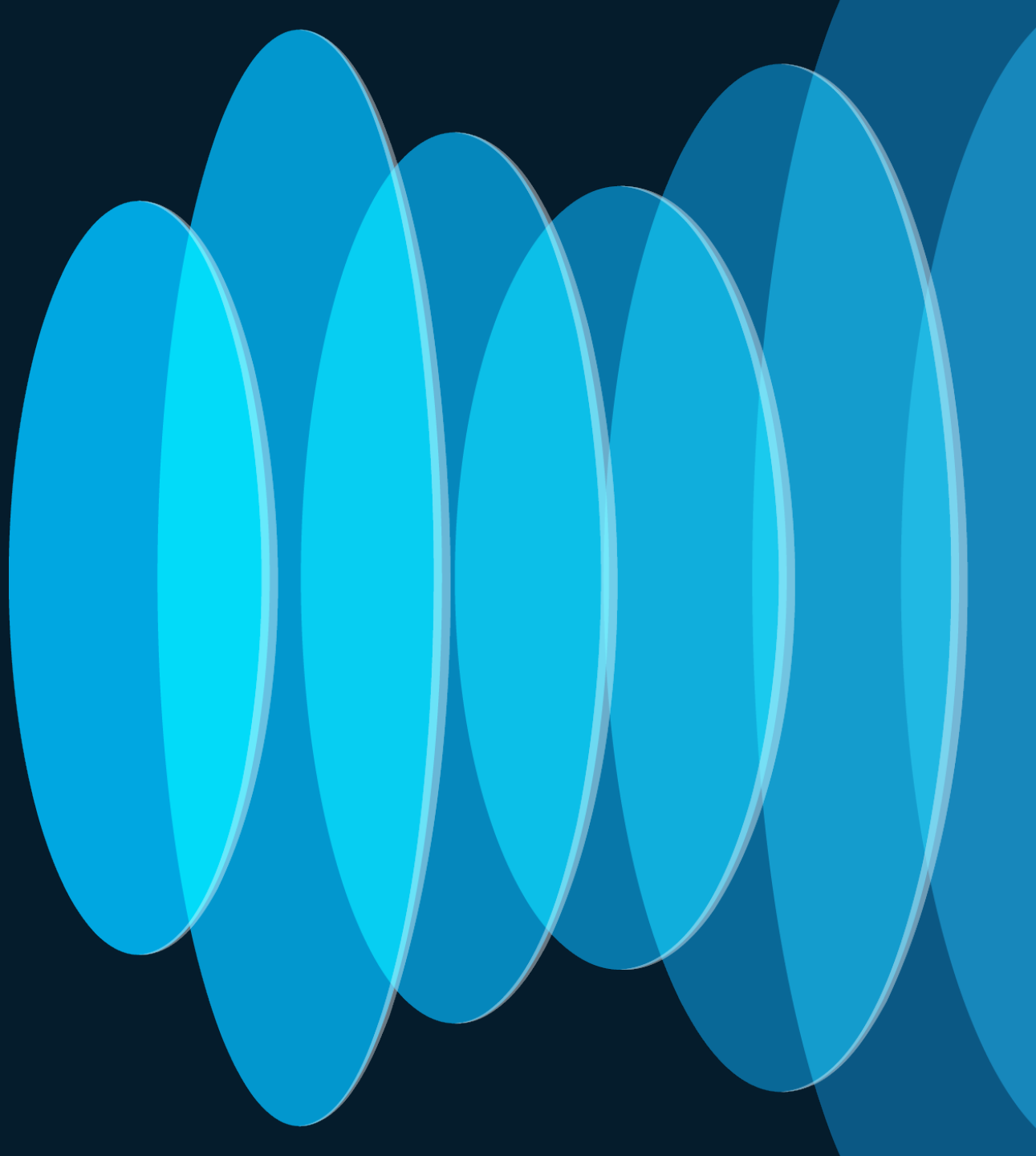
Network Challenges

Why the need for scripting tools?



- Complexity
  - Multiple protocols
  - Multiple devices
- Troubleshooting
  - Intermittent problems
  - Lack of data for Root Cause Analysis (RCA)
- Automation
  - Need to increase network responsiveness
  - Need to decrease repetition and human error

TCL (“tickle”)



Tool Command Language (TCL)

Scripting language before Python was cool

- Created in 1988 by John Ousterhout
- Built into Cisco IOS XE

```
C9300#tclsh
C9300(tcl)#info patchlevel
8.3.4
C9300(tcl)#pwd
flash:
C9300(tcl)#show clock
*02:21:08.379 UTC Wed May 15 2024
C9300(tcl)#tclquit
C9300#
```

TCL Shell

Useful Commands

- Run script from shell
  - `C9300(tcl)#source flash:lab.tcl`
- Display available built-in commands
  - `C9300(tcl)#info commands`

open	close	socket
regexp	regsub	snmp_getnext
snmp_getone	snmp_getbulk	snmp_setany
expr	after	proc

- Run EXEC mode commands
  - `C9300(tcl)#set time [exec "show clock"]`
  - `C9300(tcl)#puts $time`
- Run Configuration mode commands
  - `C9300(tcl)#ios_config "hostname Switch"`
  - `Switch(tcl)#`

Conditionals and Flow Control

```
set x 3
set x 4
set x 5

if {$x == 3} {
 puts "x is $x"
} elseif {$x == 4} {
 puts "x is $x"
} else {
 puts "x is $x"
}

x is 3
x is 4
x is 5
```

```
for {set x 1} {$x < 3} {incr x} {
  puts "x is $x"
}

foreach x [list 1 2] {
 puts "x is $x"
}

set x 1
while {$x < 3} {
  puts "x is $x"
  incr x
}

x is 1
x is 2
```

Regular Expressions

- A pattern used to match text:

Welcome to Cisco Live 2024 at Las Vegas Live

Special Character	Meaning
.	Match any element
^	Match beginning of text Within [], negate match
$	Match end of text
*	Match zero or more elements
+	Match one or more elements
?	Match zero or one elements
[ ]	Character set
{ x }	Match x repetitions

Regex	Match
Cisco	Cisco
[0-9]+	2024
[0-9]{2}	20
Live$	Live
^W.*[^0-9]	Welcome to Cisco Live
welcome	
C.*ve	Cisco Live

TCL #1

Interface Monitoring

- Problem
  - Run script to identify router interfaces actively sending traffic.
- Constraint
  - No access to file server.

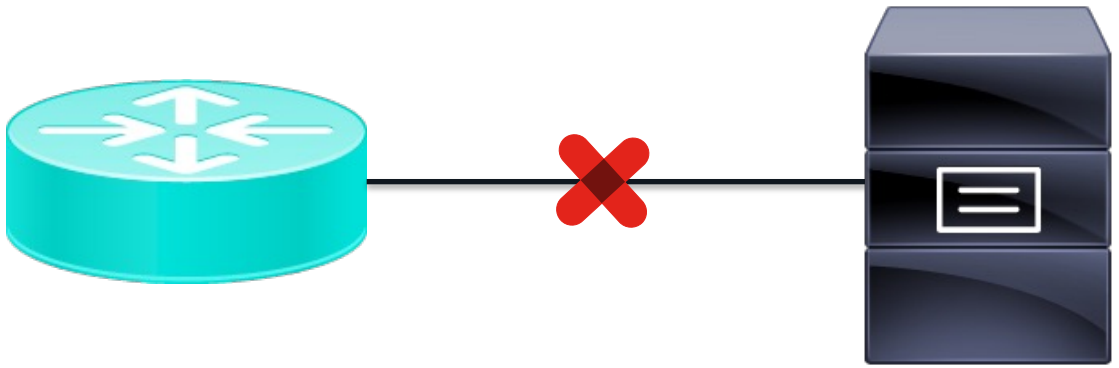
```
tclsh
puts [open "flash:traffic_out.txt" w+] {
    set showInterfaces [exec "show interfaces"]
    set interfaces [regexp -lineanchor -all -inline {^[^ ]+} $showInterfaces]
    foreach interface $interfaces {
        set showOneInterface [exec "show interfaces $interface"]
        regexp {output rate [0-9]+} $showOneInterface output_rate_line
        regexp {[0-9]+} $output_rate_line output_rate
        if {$output_rate > 0} {
            puts "$interface has output rate of $output_rate bps."
        }
    }
}
tclquit
```

```
C9300#tclsh flash:traffic_out.txt
GigabitEthernet1/0/1 has output rate of 1000 bps.
GigabitEthernet1/0/2 has output rate of 1000 bps.
GigabitEthernet1/0/3 has output rate of 1000 bps.
GigabitEthernet1/0/4 has output rate of 1000 bps.
AppGigabitEthernet1/0/1 has output rate of 2000 bps.
```

TCL #2

SNMP Monitoring

- Problem
  - Identify SNMP OIDs available on router.
- Constraint
  - No access to NMS server.



SNMP Walk

```
set oid 1.3.6
for {set i 0} {$i < 5} {incr i} {
  set var [snmp_getnext public $oid]
  puts $var
  regexp {oid='.*' val} $var oid
  set oid [string range $oid 5 end-5]
}
```

SNMP Get

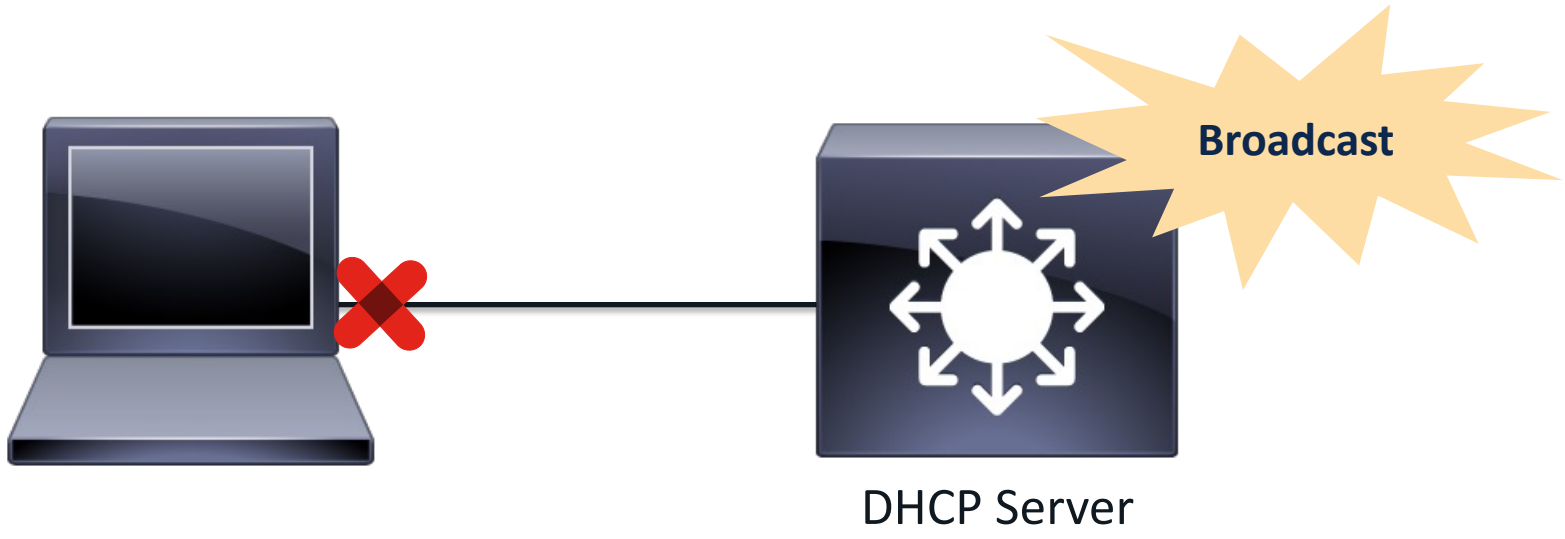
```
C9300(config)#snmp-server community public ro

C9300#show snmp mib ifmib ifindex | i Tw.*1/1/1
TwentyFiveGigE1/1/1: Ifindex = 71

C9300(tcl)#snmp_getone public
1.3.6.1.2.1.2.2.1.4.71
{<obj oid='ifMtu.71' val='1500'/>}
```

```
{<obj oid='system.1.0' val='Cisco IOS Software [Cupertino], Catalyst L3 Switch Software
(CAT9K_IOSXE), Version 17.9.3, RELEASE SOFTWARE (fc6)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2023 by Cisco Systems, Inc.
Compiled Tue 14-Mar-23 18:26 by mcpre'/>}
{<obj oid='system.2.0' val='products.2494'/>}
{<obj oid='sysUpTime.0' val='37543242'/>}
{<obj oid='system.4.0' val=''/>}
{<obj oid='system.5.0' val='C9300'/>}
```

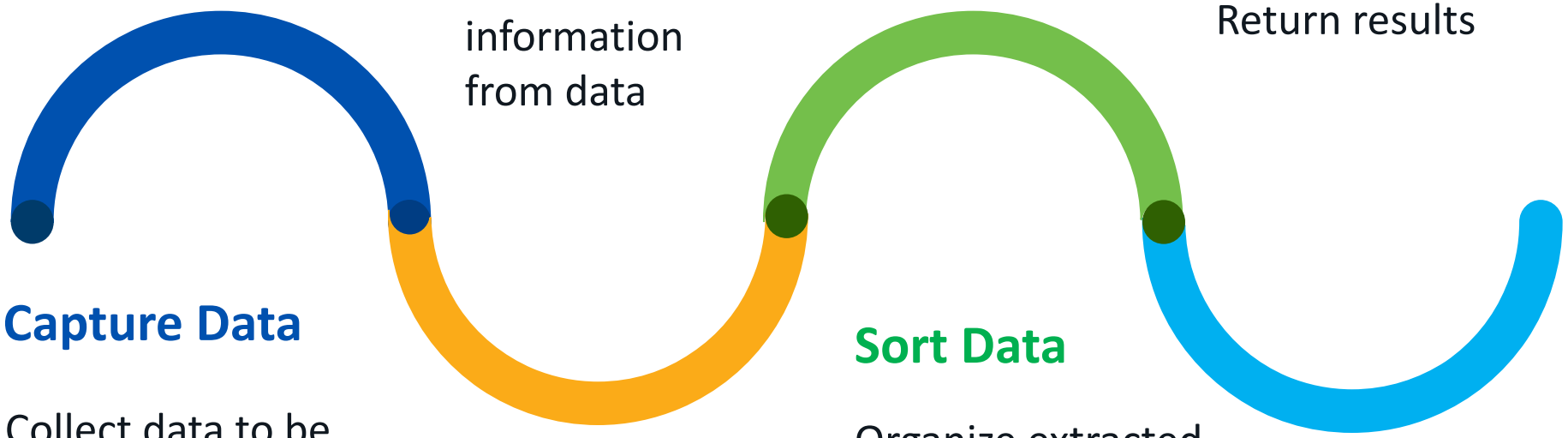
TCL #3

Control-plane Packet Capture

- Problem
  - Users do not get an IP address from DHCP server configured on L3 switch.
- Constraints
  - No access to file server to upload packet capture for analysis.



Interactive Scripting



Capture Data

Collect data to be analyzed

Parse Data

Extract needed information from data

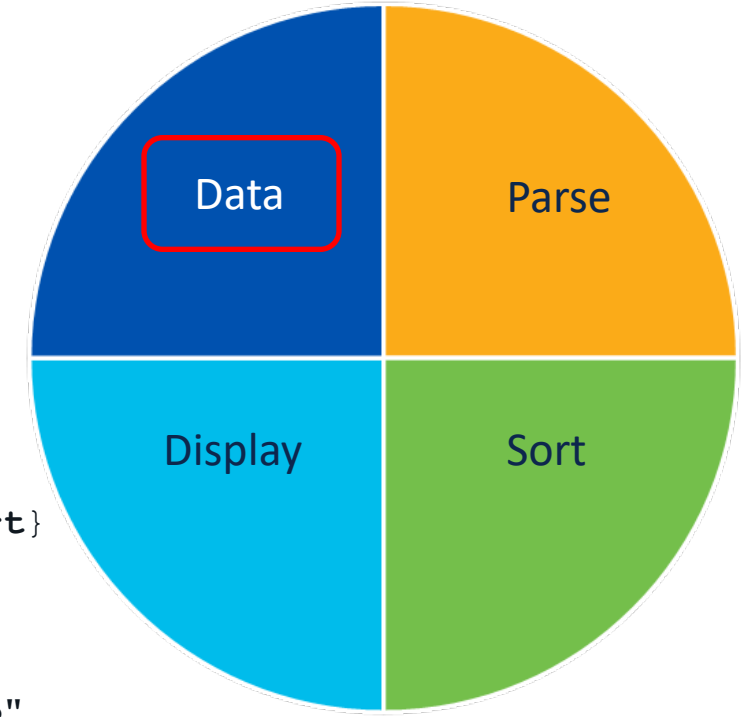
Sort Data

Organize extracted information

Display Data

Return results

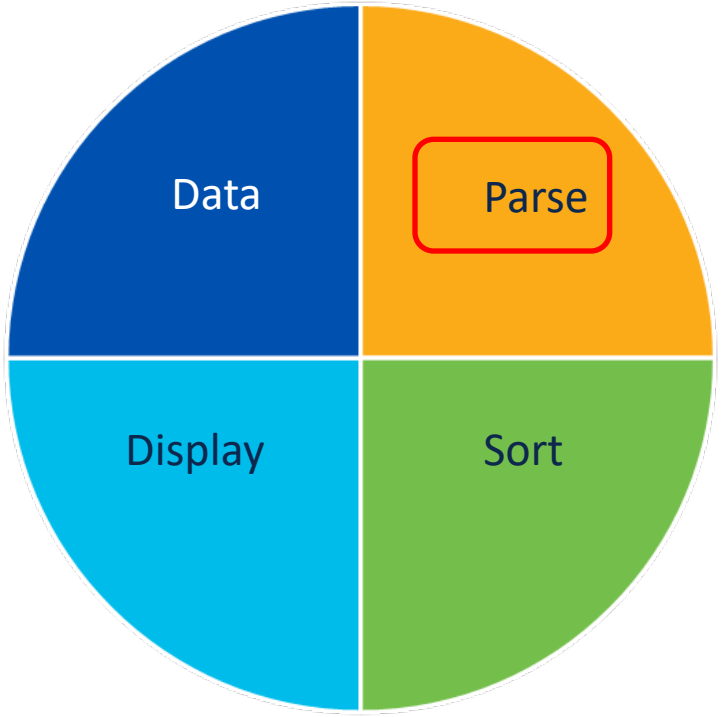
fedCap




```
proc fedCap { {top 5} {wait 30} } {
    # Collect fed punt packet capture
    puts "\n\nStarting punt CPU packet capture"
    exec {debug platform software fed switch active punt packet-capture start}
    after [expr 1000 * $wait]
    puts "Stopping punt CPU packet capture"
    exec "debug platform software fed switch active punt packet-capture stop"
    puts "Reading packets into variable for processing"
    set capture [exec "show platform software fed switch active punt packet-capture detailed"]
```

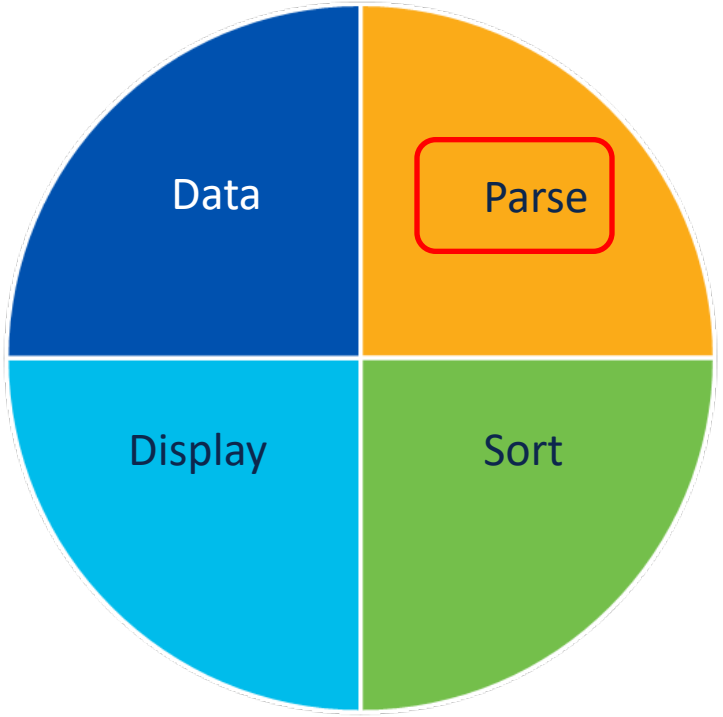
fedCap

```
# Basic Regex, MAC, IP, Vlan/Interface
set mac_rgx {[a-z0-9]{4}\.[a-z0-9]{4}\.[a-z0-9]{4}}
set ip_rgx {[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}}
set pal_rgx {\w*/\w/\w|Vlan\d*|Port-channel\d*}

# Regex to match capture output
set src_mac_regex "src mac: $mac_rgx"
set dest_mac_regex "dest mac: $mac_rgx"
set src_ip_regex "src ip: $ip_rgx"
set dest_ip_regex "dest ip: $ip_rgx"
set interface_regex {interface : [^\[]*}
set pal_regex {pal: [^\[]*}
set cause_regex {\scause: [^,]*}
set queue_regex {q-no: [^,]*}
```



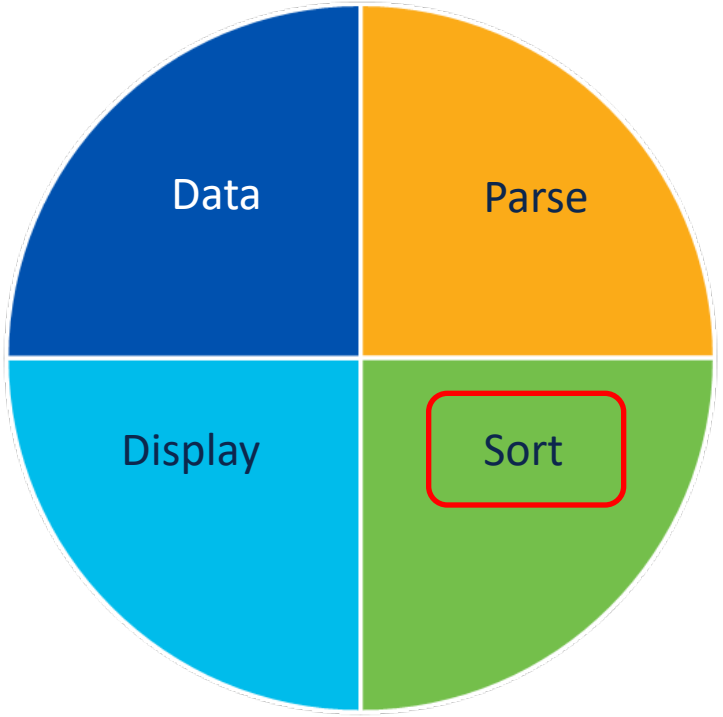
fedCap

```
# Parsing for SRC/DST MAC/IP, Vlan, Interface, Cause and CoPP Queue
 set src_mac_lst [regexp -all -inline $src_mac_regex $capture]
 set src_mac_lst [regexp -all -inline $mac_rgx $src_mac_lst]
 set dest_mac_lst [regexp -all -inline $dest_mac_regex $capture]
 set dest_mac_lst [regexp -all -inline $mac_rgx $dest_mac_lst]
 set src_ip_lst [regexp -all -inline $src_ip_regex $capture]
 set src_ip_lst [regexp -all -inline $ip_rgx $src_ip_lst]
 set dest_ip_lst [regexp -all -inline $dest_ip_regex $capture]
 set dest_ip_lst [regexp -all -inline $ip_rgx $dest_ip_lst]
```



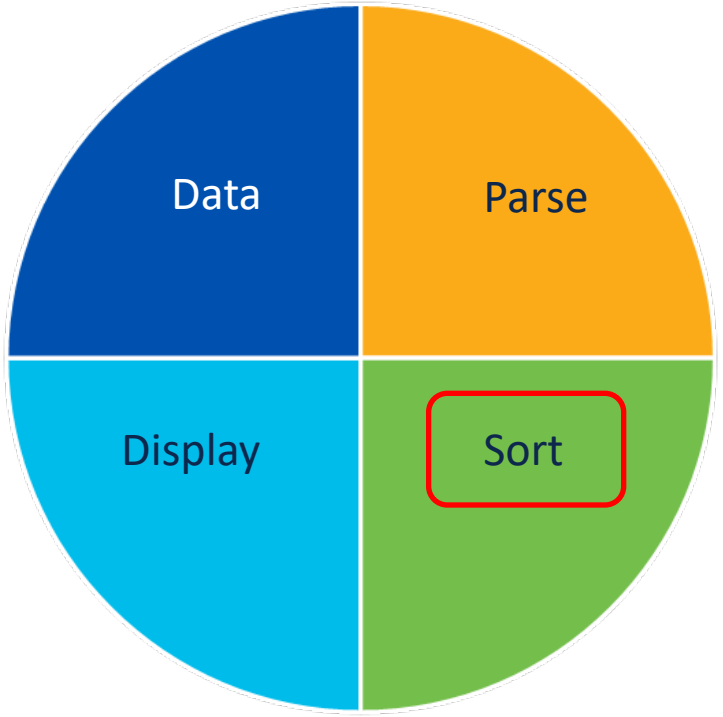
fedCap

```
proc counter {parsed_lst} {
        # Declare array to store element and count
        array set Count {}
        # Declare list to store sorted counts
        set lcount [list]
        foreach value $parsed_lst {
            # Check if element was already counted
            if {![info exists Count($value)]} {
                # Clear element from list
                set count [regsub -all $value $parsed_lst "" parsed_lst]
                # Add value to array count
                set Count($value) $count
            }
        }
```

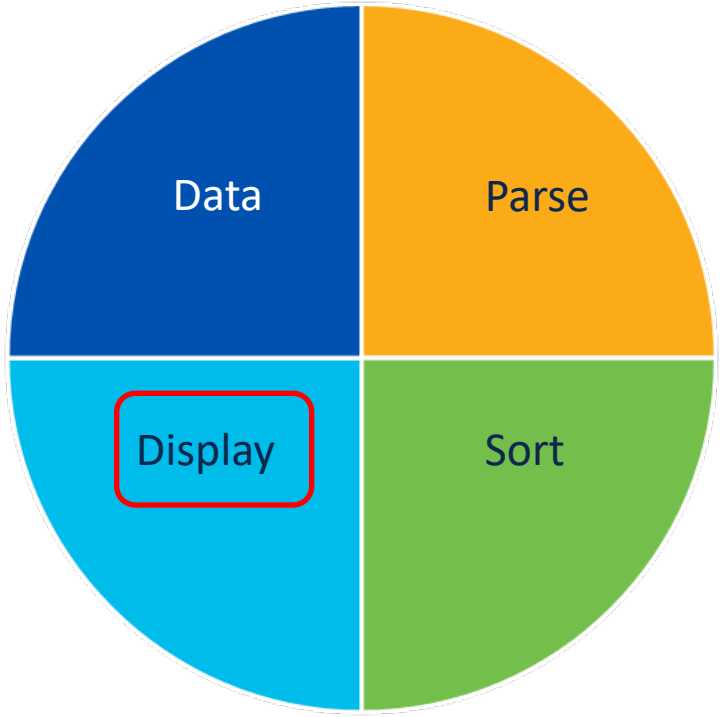


fedCap

```
    # Convert array to double list
    foreach {v c} [array get Count] {
        lappend lcount [list $v $c]
    }
    # Sorted list needed to get top talkers
    return [lsort -integer -decreasing -index 1 $lcount]
}
```



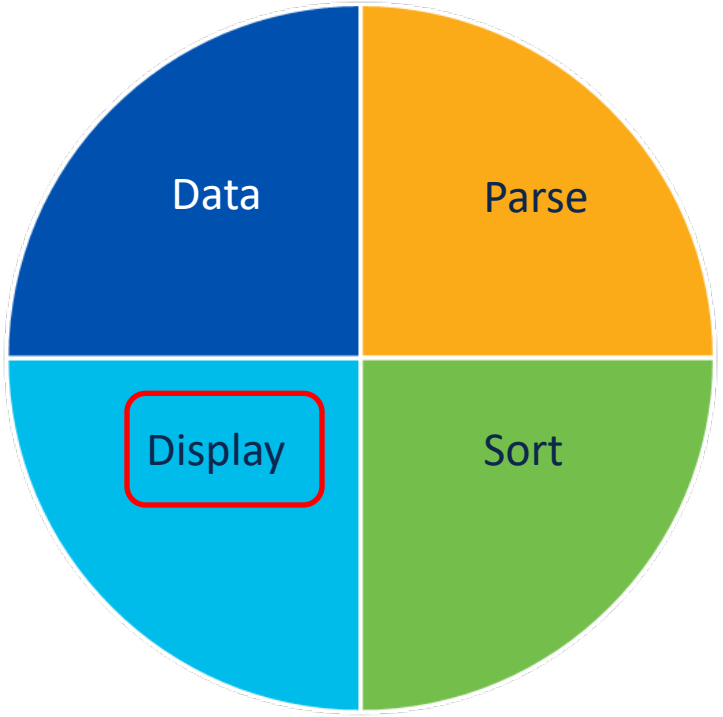
fedCap




```
# Format and display top talkers
 proc to_print {cat_lst show_num title} {
     # Do not display anything if regex did not match
     if {![llength $cat_lst]} {return}
     puts "\n\t$title\n"
     foreach categ [lrange [counter $cat_lst] 0 $show_num] {
         puts -nonewline [format "%.2f" [expr 100 * [expr double([lindex $categ 1])] / [expr
double([llength $cat_lst])]] ]
         puts -nonewline "%\t---->  "
         puts [lindex $categ 0]
     }
     return
 }
```

fedCap

```
puts [
        to_print $cause_lst $top "REASON"
        to_print $queue_lst $top "Queue ID"
        to_print $dest_ip_lst $top "DEST IP"
        to_print $dest_mac_lst $top "DEST MAC"
        to_print $src_ip_lst $top "SRC IP"
        to_print $src_mac_lst $top "SRC MAC"
        to_print $pal_lst $top "Platform Interface"
        to_print $interface_lst $top "Physical Interface"
    ]
    # End of fedCap
    return
```



Results of fedCap on CLI

REASON

```
89.55%  ---->  ARP request or response
7.46%   ---->  Layer2 control protocols
2.99%   ---->  IP subnet or broadcast packet
```

Physical Interface

```
46.27%  ---->  GigabitEthernet1/0/3
44.78%  ---->  GigabitEthernet1/0/2
2.99%   ---->  GigabitEthernet1/0/8
2.99%   ---->  GigabitEthernet1/0/10
1.49%   ---->  GigabitEthernet1/0/1
1.49%   ---->  GigabitEthernet1/0/46
```

SRC IP

```
100.00% ---->  0.0.0.0
```

SRC MAC

```
46.27%  ---->  780c.f090.b7c2
44.78%  ---->  848a.8dba.e612
2.99%   ---->  a00f.379c.385c
```

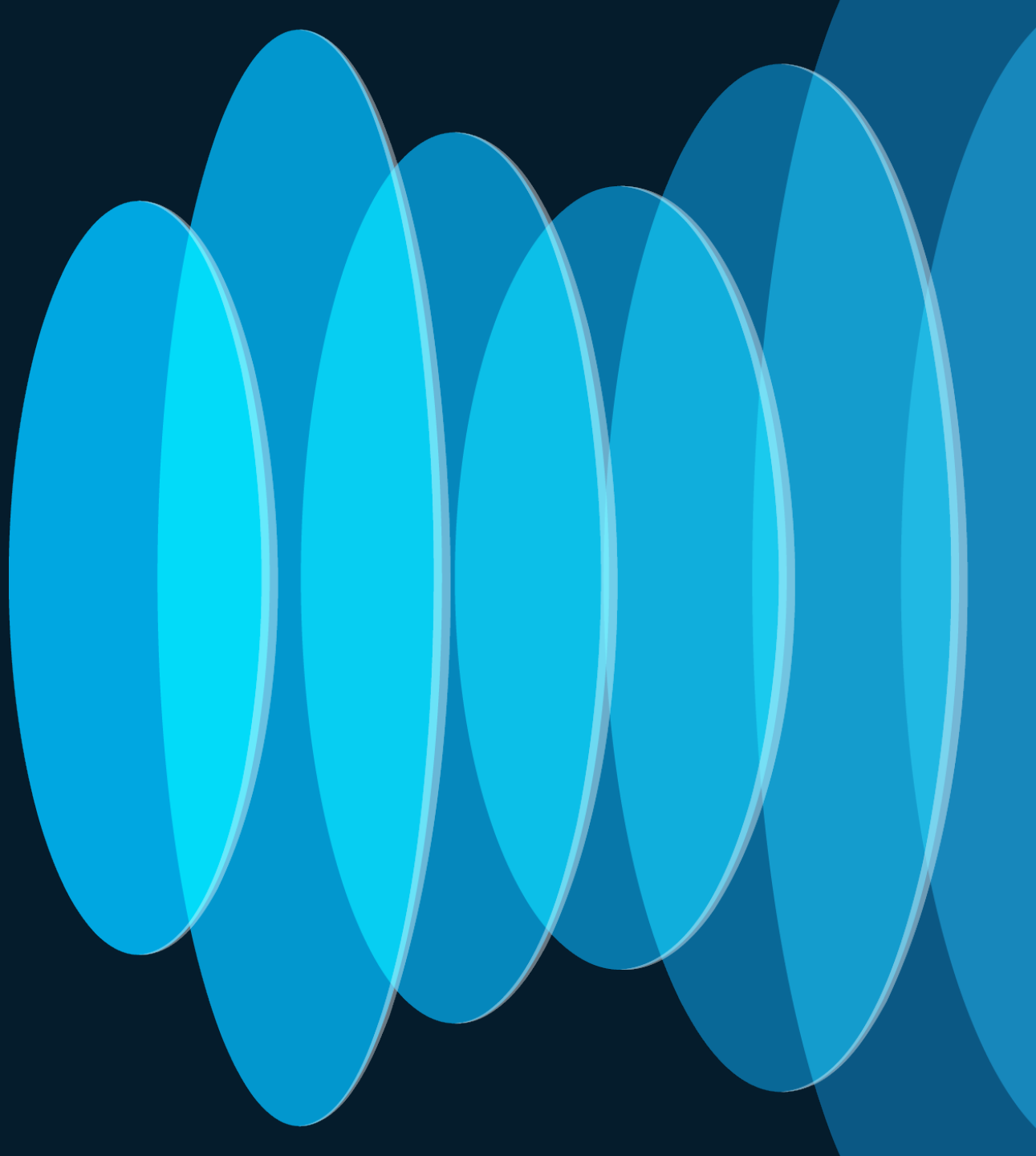
DEST IP

```
100.00% ---->  255.255.255.255
```

DEST MAC

```
92.54%  ---->  ffff.ffff.ffff
4.48%   ---->  0100.0ccc.cccc
2.99%   ---->  0180.c200.000e
```

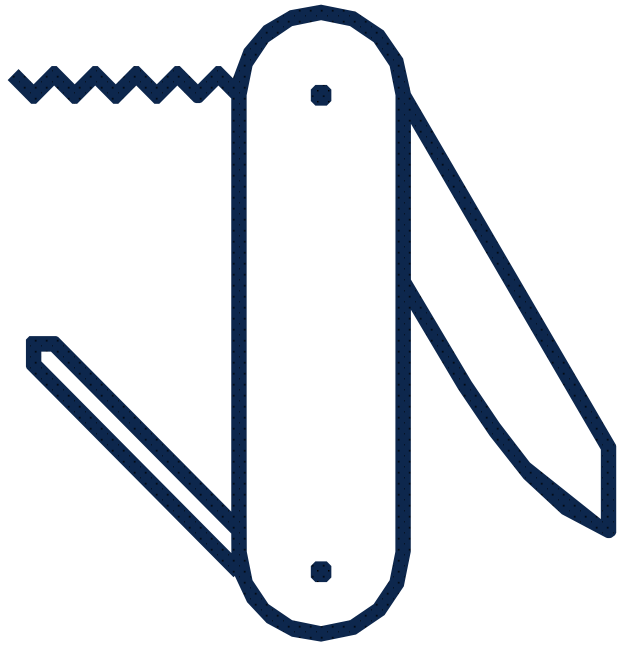
EEM



Embedded Event Manager (EEM)

TAC Engineer's Swiss Army Knife

- Versatility in trigger
- Availability
- Data collection
- Recovery
- Monitoring
- Regular Expression
- Programmability



EEM Breakdown

```
event manager applet clock authorization bypass
 event none maxrun 30
 action 010 cli command "enable"
 action 020 cli command "show clock"
 action 030 cli command "show clock | append flash:clock.txt"
 action 040 syslog msg "Ran clock EEM"
```

```
C9300#event manager run clock
C9300#
*May 15 19:49:40.915: %HA_EM-6-LOG: clock: Ran clock EEM

C9300#more flash:clock.txt
*19:49:40.862 UTC Wed May 15 2024
```

EEM #1

DHCP Conflict

Problem - Users are unable to get an IP address from DHCP server on L3 switch.

```
ip dhcp excluded-address 192.168.10.0 192.168.10.30
!
ip dhcp pool vlan2
 network 192.168.10.0 255.255.255.0

Switch#sh ip dhcp pool
Pool vlan2 :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 114
 Excluded addresses             : 49
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index        IP address range                    Leased/Excluded/Total
 192.168.10.187       192.168.10.1     - 192.168.10.254    114   / 49    / 254
```

EEM #1

Clear DHCP Conflict

```
event manager applet CLEAR_DHCPCONFLICT authorization bypass
 event timer cron cron-entry "0 0 1 * *"
 action 010 cli command "enable"
 action 020 cli command "clear ip dhcp conflict *"
 action 030 syslog msg "Cleared DHCP conflict."
```

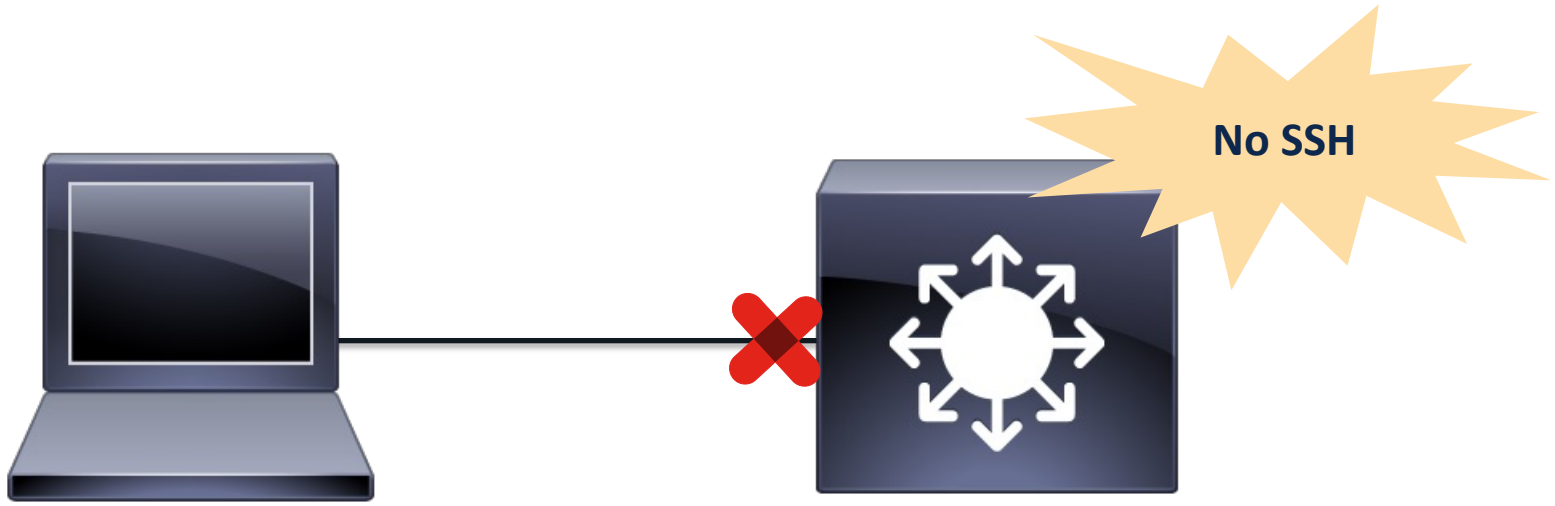
- Cron-entry
  - Minute – 0 to 59
  - Hour – 0 to 23
  - Day-of-month – 1 to 31
  - Month – 1 to 12
  - Day-of-week 0 to 6 (Sunday is 0)

```
C9300(config-applet)# event timer cron cron-entry "* * * * *"

C9300#
*May 15 20:14:26.737: %SYS-5-CONFIG_I: Configured from console by
console
*May 15 20:15:00.225: %HA_EM-6-LOG: CLEAR_DHCPCONFLICT: Cleared DHCP
conflict.
*May 15 20:16:00.225: %HA_EM-6-LOG: CLEAR_DHCPCONFLICT: Cleared DHCP
conflict.
*May 15 20:17:00.222: %HA_EM-6-LOG: CLEAR_DHCPCONFLICT: Cleared DHCP
conflict.
```

EEM #2

Forus CoPP Queue Drops

- Problem
  - Users are intermittently unable to SSH into L3 switch.
- Constraint
  - Issue is currently not present but happens a few times during the week.



EEM #2

Embedded Packet Capture (EPC)

```
clear platform hardware fed switch active qos statistics internal cpu policer
monitor capture tac control-plane in match any buffer size 50 circular
monitor capture tac limit pps 1000
monitor capture tac start
configure terminal
```

```
C9300#show platform hardware fed switch active qos queue stats internal cpu policer

                        CPU Queue Statistics
============================================================================================

                                                (default)  (set)      Queue          Queue
QId PlcIdx  Queue Name                  Enabled   Rate      Rate       Drop(Bytes)   Drop(Frames)
--------------------------------------------------------------------------------------------
0    11     DOT1X Auth                   Yes     1000       1000      0             0
1    1      L2 Control                   Yes     2000       2000      0             0
2    14     Forus traffic                Yes     4000       4000      5819488       34809
<SNIP>
```

EEM #2

Monitor Forus CoPP Queue

```
event manager applet copp_drops authorization bypass
event timer watchdog time 60 maxrun 60
action 010   cli command "enable"
action 015   info type routername
action 020   cli command "show platform hardware fed active qos queue stats internal cpu policer | i Forus traffic"
action 025   regexp "2[ ]+14[ ]+Forus traffic[ ]+Yes[ ]+[0-9]+[ ]+[0-9]+[ ]+([0-9]+)" "$_cli_result" match drops1
action 030   if $drops1 gt 0
action 040      cli command "monitor capture tac stop"
action 045      cli command "monitor capture tac export location flash:copp_drops.pcap"
action 050      cli command "no monitor capture tac"
action 060      cli command "configure terminal"
action 065      syslog msg "EEM copp_drops:  Auto-removing EEM script."
action 070      cli command "no event manager applet copp_drops authorization bypass"
action 080   end
action 090   cli command "clear platform hardware fed active qos statistics internal cpu policer"
```

EEM #3

BFD Drops

bfd interval 250 min_rx 250 multiplier 3

- Problem - BFD drops randomly every few hours between IPv6 neighbor L3 switches.

No.	Time	Delta	Sour	Dest	Protocol	Info
81692	*REF*	*REF*	fe…	fe…	BFD Control	Diag: No Diagnostic, State: Up, Flags: 0x00
81693	0.111691	0.111691	fe…	fe…	BFD Control	Diag: No Diagnostic, State: Up, Flags: 0x00
81694	0.329039	0.217348	fe…	fe…	BFD Control	Diag: No Diagnostic, State: Up, Flags: 0x00
81695	0.574227	0.245188	fe…	fe…	BFD Control	Diag: No Diagnostic, State: Up, Flags: 0x00
81696	0.750325	0.176098	fe…	fe…	BFD Control	Diag: Control Detection Time Expired, State: Down, Flags: 0x00
81697	1.649253	0.898928	fe…	fe…	BFD Control	Diag: Control Detection Time Expired, State: Down, Flags: 0x00
81698	2.602270	0.953017	fe…	fe…	BFD Control	Diag: Control Detection Time Expired, State: Down, Flags: 0x00
81699	3.552505	0.950235	fe…	fe…	BFD Control	Diag: Control Detection Time Expired, State: Down, Flags: 0x00
81700	3.970745	0.418240	fe…	ff…	PIMv2	Hello
81701	4.536271	0.565526	fe…	fe…	BFD Control	Diag: Control Detection Time Expired, State: Down, Flags: 0x00
81702	4.933548	0.397277	fe…	ff…	PIMv2	Hello
81703	5.123541	0.189993	fe…	ff…	ICMPv6	Neighbor Solicitation for [redacted]
81704	5.124940	0.001399	fe…	fe…	ICMPv6	Neighbor Advertisement [redacted]
81705	5.484220	0.359280	fe…	fe…	BFD Control	Diag: Control Detection Time Expired, State: Down, Flags: 0x00
81706	5.570474	0.086254	fe…	ff…	PIMv2	Hello
81707	6.019626	0.449152	fe…	fe…	BFD Control	Diag: Neighbor Signaled Session Down, State: Init, Flags: 0x00
81708	6.019825	0.000199	fe…	fe…	BFD Control	Diag: No Diagnostic, State: Up, Flags: 0x20

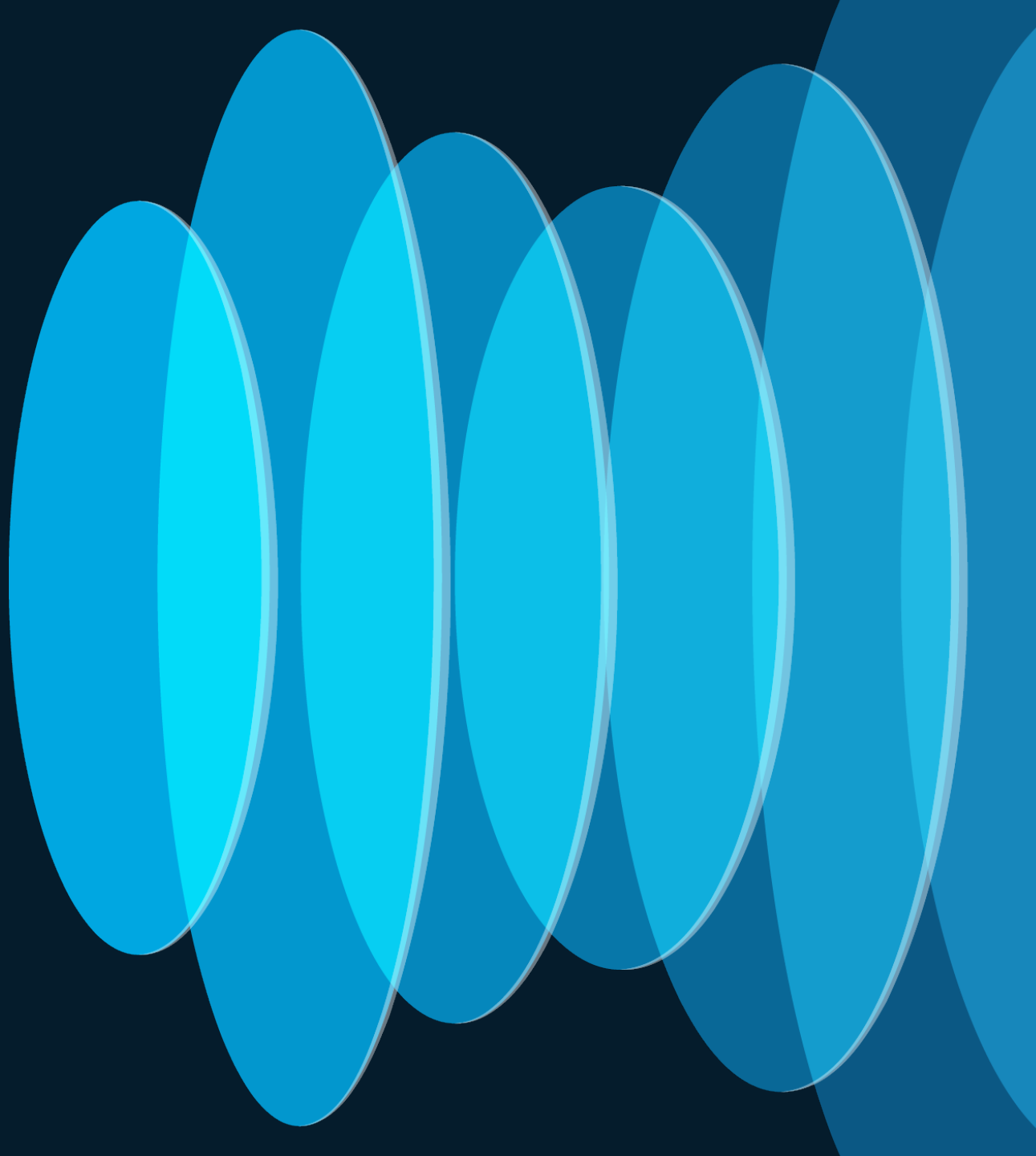
EEM #3

BFD Drops

```
event manager applet BFD_DOWN authorization bypass
event syslog pattern "%BFDFSM-6-BFD_SESS_DOWN: BFD-SYSLOG: BFD session ld:11 handle:11"
action 010 syslog msg "Log collection for BFD_DOWN EEM script."
action 020 cli command "enable"
action 030 cli command "terminal length 0"
action 040 cli command "terminal exec prompt timestamp"
action 045 cli command "terminal exec prompt expand"
action 047 cli command "show clock | append flash:BFD.txt"
action 050 cli command "show ipv6 neighbors detail | append flash:BFD.txt"
action 065 cli command "show ipv6 interface Te1/0/14 | append flash:BFD.txt"
action 067 cli command "show bfd neighbors details | append flash:BFD.txt"
action 080 cli command "show interface Te1/0/14 | append flash:BFD.txt"
action 190 cli command "configure terminal"
action 200 cli command "no event manager applet BFD_DOWN"
action 210 cli command "end"
```

```
ipv6 nd cache expire 120 refresh
```

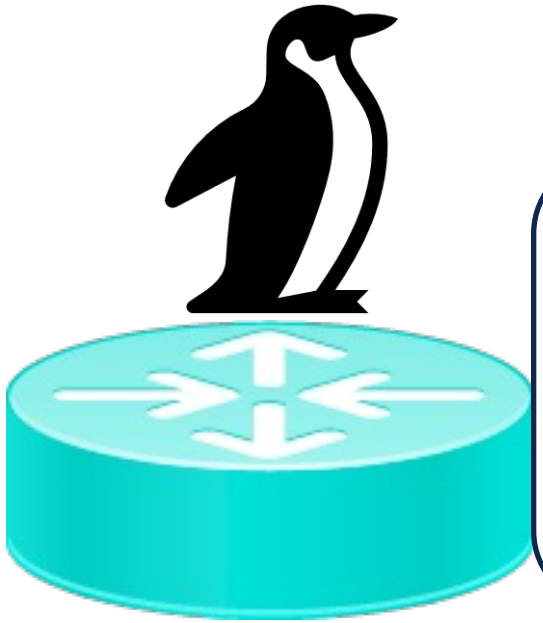
Guest Shell



Guest Shell

Virtualized Linux-based Environment

- Application Hosting - IOx
- Isolated from host kernel
- CentOS 8
- Python 3.6



Port
- Management
- AppGigabitEthernet
- **VirtualPortGroup**

Enabling Guest Shell

IOx

```
C8kv1(config)#iox

C8kv1#show iox-service

IOx Infrastructure Summary:
---------------------------
IOx service (CAF)              : Running
IOx service (HA)               : Not Supported
IOx service (IOxman)           : Running
IOx service (Sec storage)      : Not Supported
Libvirtd 5.5.0                 : Running
```

VirtualPortGroup

```
C8kv1(config)#interface VirtualPortGroup 0
C8kv1(config-if)#ip address 192.168.1.1 255.255.255.0
C8kv1(config-if)#end
C8kv1#show run interface VirtualPortGroup0
Building configuration...

Current configuration : 73 bytes
!
interface VirtualPortGroup0
 ip address 192.168.1.1 255.255.255.0
end
```

Enabling Guest Shell

```
C8kv1#show run | section guestshell
app-hosting appid guestshell
 app-vnic gateway1 virtualportgroup 0 guest-interface 0
  guest-ipaddress 192.168.1.10 netmask 255.255.255.0
 app-default-gateway 192.168.1.1 guest-interface 0

C8kv1#guestshell enable
Interface will be selected if configured in app-hosting
Please wait for completion
guestshell installed successfully
Current state is: DEPLOYED
guestshell activated successfully
Current state is: ACTIVATED
guestshell started successfully
Current state is: RUNNING
Guestshell enabled successfully

C8kv1#show app-hosting list
App id                                   State
---------------------------------------------------------
guestshell                               RUNNING
```

Running Guest Shell

```
C8kv1#guestshell run pwd
/home/guestshell

C8kv1#guestshell run bash
[guestshell@guestshell ~]$ exit
exit

C8kv1#guestshell run python3
Python 3.6.8 (default, Dec 22 2020, 19:04:08)
[GCC 8.4.1 20200928 (Red Hat 8.4.1-1)] on linux
Type "help", "copyright", "credits" or "license"
for more information.
>>> exit()
```

Deleting Guest Shell

```
C8kv1#guestshell disable
Guestshell disabled
successfully

C8kv1#guestshell destroy
Guestshell destroyed
successfully

C8kv1#show app-hosting list
No App found
```

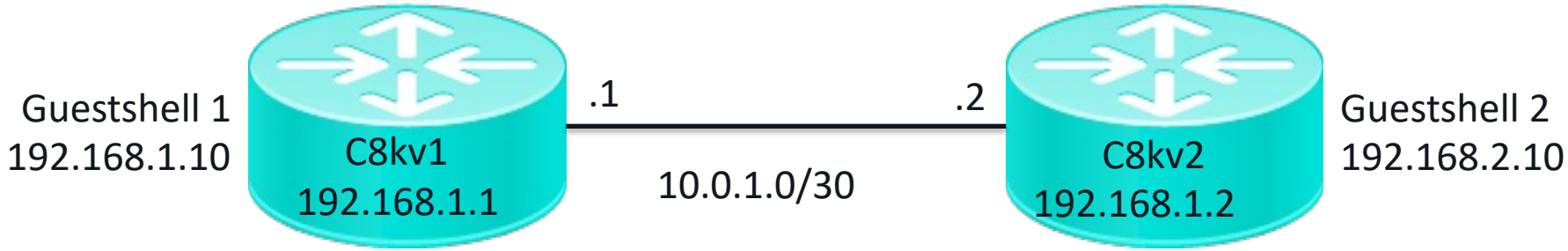
Exec and Config from Guest Shell

```
C8kv1#guestshell run bash
[guestshell@guestshell ~]$ python3
Python 3.6.8 (default, Dec 22 2020, 19:04:08)
[GCC 8.4.1 20200928 (Red Hat 8.4.1-1)] on linux
Type "help", "copyright", "credits" or "license" for more
information.
>>> import cli
>>> output = cli.execute("show clock")
>>> print(output)
*04:10:24.337 UTC Thu May 16 2024

>>> cli.configure("hostname RR")
'Line 1 SUCCESS:  hostname RR\n'
>>> exit()
[guestshell@guestshell ~]$ exit
exit

RR#
```

Guest Shell #1

iPerf

- Problem
  - Need to run iPerf between two routers in the network.




```
C8kv1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.2.1       1   FULL/DR         00:00:31    10.0.1.2        GigabitEthernet1

[guestshell@guestshell ~]$ ping 192.168.2.10
PING 192.168.2.10 (192.168.2.10) 56(84) bytes of data.
64 bytes from 192.168.2.10: icmp_seq=2 ttl=63 time=2.21 ms
64 bytes from 192.168.2.10: icmp_seq=3 ttl=63 time=4.38 ms
```

Guest Shell #1

iPerf

Download binaries from iPerf website:

```
https://iperf.fr/download/ubuntu/libiperf.so.0_3.1.3
https://iperf.fr/download/ubuntu/iperf3_3.1.3
```

```
C8kv1#copy tftp: bootflash:/guest-share

C8kv1#guestshell run bash
[guestshell@guestshell ~]$ cd /bootflash/guest-share
[guestshell@guestshell guest-share]$ ls -lthr
total 476K
-rw-rw-r--. 1 nobody network-admin 438K May 16 04:36 libiperf.so.0
-rw-rw-r--. 1 nobody network-admin  28K May 16 04:36 iperf3
```

Guest Shell #1

iPerf

```
[guestshell@guestshell guest-share]$ sudo cp libiperf.so.0 /usr/lib64/libiperf.so.0
[guestshell@guestshell guest-share]$ sudo cp iperf3 /usr/bin/iperf3

[guestshell@guestshell guest-share]$ sudo chmod +x /usr/lib64/libiperf.so.0
[guestshell@guestshell guest-share]$ sudo chmod +x /usr/bin/iperf3

[guestshell@guestshell guest-share]$ sudo /usr/bin/iperf3 -v
iperf 3.1.3
Linux guestshell 5.4.108 #1 SMP Tue Mar 30 16:43:30 UTC 2021 x86_64
Optional features available: CPU affinity setting, IPv6 flow label, TCP congestion
algorithm setting, sendfile / zerocopy, socket pacing
```

Guest Shell #1

iPerf

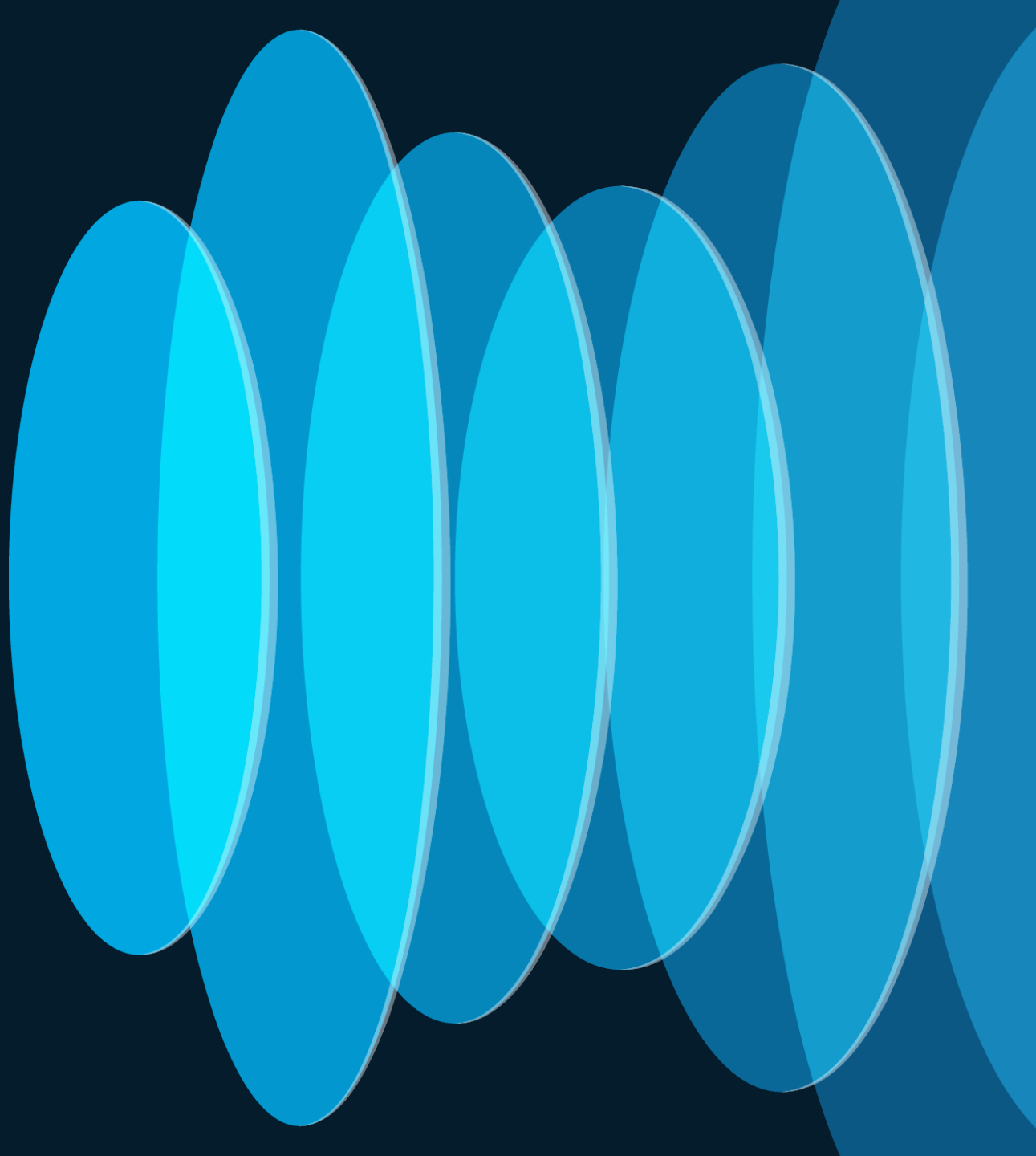
```
C8kv1#guestshell run bash
[guestshell@guestshell ~]$ /usr/bin/iperf3 -c 192.168.2.10
Connecting to host 192.168.2.10, port 5201
[  4] local 192.168.1.10 port 44486 connected to 192.168.2.10 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.18   sec  2.99 MBytes  21.3 Mbits/sec    0    139 KBytes
[  4]   1.18-2.18   sec  1.24 MBytes  10.4 Mbits/sec   10    117 KBytes
[  4]   2.18-3.00   sec  1.12 MBytes  11.4 Mbits/sec    0    136 KBytes
[  4]   3.00-4.00   sec  1.18 MBytes  9.90 Mbits/sec    5    107 KBytes
[  4]   4.00-5.00   sec  1.12 MBytes  9.38 Mbits/sec    0    116 KBytes
[  4]   5.00-6.38   sec  1.06 MBytes  6.44 Mbits/sec    0    119 KBytes
[  4]   6.38-7.18   sec  1.12 MBytes  11.7 Mbits/sec    0    123 KBytes
[  4]   7.18-8.00   sec  1.24 MBytes  12.6 Mbits/sec   10   90.5 KBytes
[  4]   8.00-9.18   sec   954 KBytes  6.65 Mbits/sec    0    107 KBytes
[  4]   9.18-10.00  sec  1.37 MBytes  13.9 Mbits/sec    0    123 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec  13.4 MBytes  11.2 Mbits/sec   25             sender
[  4]   0.00-10.00  sec  12.6 MBytes  10.6 Mbits/sec                  receiver

iperf Done.
```

Guest Shell #1

iPerf

```
C8kv2#guestshell run bash
[guestshell@guestshell ~]$ /usr/bin/iperf3 -s
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
Accepted connection from 192.168.1.10, port 44484
[  5] local 192.168.2.10 port 5201 connected to 192.168.1.10 port 44486
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-1.18   sec  2.18 MBytes  15.5 Mbits/sec
[  5]   1.18-2.18   sec  1.20 MBytes  10.1 Mbits/sec
[  5]   2.18-3.00   sec  1.16 MBytes  11.9 Mbits/sec
[  5]   3.00-4.00   sec  1.13 MBytes  9.51 Mbits/sec
[  5]   4.00-5.00   sec  1.14 MBytes  9.56 Mbits/sec
[  5]   5.00-6.18   sec  1.06 MBytes  7.53 Mbits/sec
[  5]   6.18-7.18   sec  1.08 MBytes  9.09 Mbits/sec
[  5]   7.18-8.00   sec  1.27 MBytes  12.9 Mbits/sec
[  5]   8.00-9.18   sec   950 KBytes  6.63 Mbits/sec
[  5]   9.18-10.00  sec  1.35 MBytes  13.8 Mbits/sec
[  5]  10.00-10.11  sec   123 KBytes  9.47 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-10.11  sec  0.00 Bytes  0.00 bits/sec                  sender
[  5]   0.00-10.11  sec  12.6 MBytes  10.5 Mbits/sec                receiver
```

Conclusion



Use Case

	TCL	EEM	Guest Shell
Interpreter Shell	X		X
Requires Configuration		X	X
Event Trigger		X	
Regular Expression	X	X	X
Linux Environment			X
Interactive within IOS-XE	X		
Interactive within CLI	X		X
Max Runtime / Authorization bypass		X	

Network Challenges

How TCL, EEM and Guest Shell can help?

- Complexity
  - Isolate devices or protocols related to problem
  - Parse output to simplify data analysis
- Troubleshooting
  - Trigger packet capture during problem
  - Trigger commands during problem
- Automation
  - Monitor events or counters to detect errors and remediate
  - Implement flow control and programmability to increase efficiency

References

- **TCL**
  - IOS Scripting with TCL Configuration Guide

  https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ios_tcl/configuration/xe-16-6/ios-tcl-xe-16-6-book/Cisco_IOS_XE_Scripting_with_Tcl.html
  - Copy Files to a Router without File Transfer Access

  https://www.cisco.com/c/en/us/support/docs/ip/telnet/116214-technote-technology-00.html
- **EEM**
  - Configuration Guide: Embedded Event Manager Overview

  https://www.cisco.com/c/en/us/td/docs/routers/ios/config/17-x/syst-mgmt/b-system-management/m_eem-overview.html
  - Understand Best Practices and Useful Scripts for EEM

  https://www.cisco.com/c/en/us/support/docs/ios-nx-os-software/ios-xe-16/216091-best-practices-and-useful-scripts-for-ee.html#anc23
  - Application Hosting on the Cisco Catalyst 9000 Series Switches White paper

  https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-9300-series-switches/white-paper-c11-742415.html
- **Guest Shell**
  - Configuration Guide: Guest Shell

  https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/179/b_179_programmability_cg/m_179_prog_guestshell.html
  - Configuration Guide: CLI Python Module

  https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/1612/b_1612_programmability_cg/cli_python_module.html

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

Contact me at: sfofano@cisco.com


The bridge to possible
Thank you
cisco Live!
#CiscoLive

Q & A