

Hoe u FCS en CRC-probleemoplossing voor ACI gebruikt

Inhoud

[Inleiding](#)

[Voorwaarden om het script handmatig uit te voeren](#)

[Voorwaarden om het script vanaf container uit te voeren](#)

[Stappen om scripts uit te voeren](#)

Inleiding

ACI volgt Cut-Through Switching waardoor het Packet al verzonden is voordat CRC kan worden berekend. Deze pakketten zijn doorgaans voorzien van een module en doorgestuurd als uitvoerfouten. Omdat ACI deze pakketten niet laat vallen, overschrijdt het pakket en worden de stroom-CRC-tellers op het pad verhoogd. Dit betekent niet dat alle interfaces die de CRC zien, defect zijn. Daarom is een goede triage nodig om de problematische poort/SFP/glasvezel te isoleren. Het proefproces is nu geautomatiseerd door Python-scripts die een eenvoudige oplossing kunnen bieden en handmatige taken kunnen voorkomen. De reikwijdte van dit document is om uit te leggen hoe de automatiseringsscripts moeten worden gebruikt (zie bijgevoegd).

Voorwaarden om het script handmatig uit te voeren

De clientmachine waar het script wordt uitgevoerd, moet aan de volgende vereisten voldoen

- a. Python3 moet worden geïnstalleerd
- b. Netwerktogang tot het ACI-domein
- c. ABBYY_CRC_requirements.txt (toegevoegd) om te installeren. Dit bestand staat [hier](#).

Download het bestand (ACI_CRC_requirements.txt) naar de client machine

Open terminal en voer het commando-pi3 installatie-r ACI_CRC_requirements.txt uit

```
ABCD-M-G24X:downloads abcd$ pip3 install -r ACI_CRC_requirements.txt
```

```
Collecting bcrypt==3.2.0 (from -r ACI_CRC_requirements.txt (line 1))
```

```
Downloading
```

```
https://files.pythonhosted.org/packages/bf/6a/0afb1e04aebd4c3ceae630a87a55fbfbbd94dea4eaf01e53d36743c85f02/bcrypt-3.2.0-cp36-abi3-macosx\_10\_9\_x86\_64.whl
```

```
Collecting cffi==1.14.6 (from -r ACI_CRC_requirements.txt (line 2))
```

```
Downloading
```

```
https://files.pythonhosted.org/packages/ca/e1/015e2ae23230d9de8597e9ad8c0b81d5ac181f08f2e6e75774b7f5301677/cffi-1.14.6-cp38-cp38-macosx\_10\_9\_x86\_64.whl (176kB)
```

```
|| 184kB 1.4MB/s
```

```
**snip**
```

```
Successfully installed DateTime-4.3 Pillow-8.3.2 bcrypt-3.2.0 cffi-1.14.6 cryptography-3.4.8  
cyclcr-0.10.0 kiwisolver-1.3.2
```


minuten (tot het eindtijd dat eerder door de gebruiker opgegeven is) en slaat het de gegevens op naar bestanden op het pad dat in eerdere invoer is gespecificeerd.

```
Enter the End Time until which the script runs(in the format of yyyy-mm-dd hh:mm, current time:2021-09-27 11:27.... maximum upto 2021-10-04 11:27): 2021-09-27 11:32 <<<<<
```

```
The script is executing .....
```

```
The script is executing .....
```

```
ABCD-M-G24X:downloads abcd$
```

4. Zodra het eerste script met succes is uitgevoerd, zal het rauwe gegevensbestanden op de locatie opslaan, die door de gebruiker in stap 2 zijn gespecificeerd.

Controleer hetzelfde als in het onderstaande voorbeeld.

```
ABCD-M-G24X:FCS_Checker kbosu$ pwd
```

```
/Users/abcd/Downloads/FCS_Checker
```

```
ls -l
```

```
total 16
```

```
-rw-r--r--@ 1 kbosu staff 1419 Sep 27 11:28 CRC_FCS_20210927_1128.txt
```

```
-rw-r--r--@ 1 kbosu staff 1419 Sep 27 11:33 CRC_FCS_20210927_1133.txt
```

```
ABCD-M-G24X:FCS_Checker abcd$
```

5. Nu is het tijd om het tweede script uit te voeren (ACI_CRC_Parser.py).

Script-2 gaat die bestanden gebruiken die gemaakt zijn door script-1 en verder werken.

Voer het OB IP-adres in voor een van de APIC's in de gegeven cluster en het is geloofsbrieven.

Voer ook dezelfde bestandsindeling in die u in stap 2 hebt ingevoerd tijdens het uitvoeren van het eerste script.

```
ABCD-M-G24X:downloads abcd$ python3 ACI_CRC_Parser.py
```

```
Enter the IP address or DNS Name of APIC: 10.197.204.184
```

```
Enter the username: admin
```

```
Enter the password: *****
```

```
Trying to connect to APIC
```

```
Connection established to the APIC
```

```
Please enter the folder where files are stored
```

```
Please make sure we have at least two files exists in the directory where you have saved data
```

```
Enter the absolute path of the folder where the files are stored:/Users/abcd/Downloads/FCS_Checker/
```

```
You have CRC and FCS for the below date range
```

```
1.2021-09-27
```

```
Fetching first and last file of the same date 20210927
```

```
CRC_FCS_20210927_1128.txt
```

```
CRC_FCS_20210927_1133.txt
```

```
The script is executing.....
```

```
The script execution has completed
```

6. Script-2 gaat de gegevens in tabelvorm afdrukken zoals in onderstaande voorbeeld wordt

getoond.

In de eerste plaats gaat het de knoopinterfaces met niet-nulfouten van CRC en FCS vermelden, samen met het verschil in hun CRC/FCS-tellers, gedurende het door de gebruiker gespecificeerde tijdsinterval. Met behulp van LLDP zal het script ook het buurapparaat bepalen dat aangesloten is op bepaalde interfaces. Nog belangrijker is dat het aangeeft welke knooppunt/interface de bron is van fouten uit weefselstandpoint en welke knoopinterfaces nu CRC's zien door Stomp.

Vanuit het perspectief van FCS-probleemoplossing wordt de nadruk gelegd op 'Rood' en aangeduid als 'Lokaal', waar verdere probleemoplossing moet worden gevonden.

Dit is waarschijnlijk de interface(s), waar slechte/gecorrumpeerde pakketten in het weefsel worden ingevoerd en ervoor zorgen dat de CRCs in het weefsel worden overstroomd.

```
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| POD_ID | NODE_ID | NODE_NAME | NODE_ROLE | INTERFACE | 20210927_1128 | 20210927_1133 | 20210927_1128 | 20210927_1133 |
| NEIGHBOR | | | | | ERROR SOURCE |
|-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| CRC | CRC Diff | FCS | FCS Diff |
|-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 302 | bgl-aci06-t2-leaf2 | leaf | eth1/44 | 5002806823759 | 127841888 | 5002806823759 | 127841888 | No
LLDP /CDP neighbours found please check physically where this interface connects | Local |
| 1 | 101 | bgl-aci06-spine1 | spine | eth1/1 | 2981200154 | 132103050 | 0 | 0 |
System:bgl-aci06-t1-leaf1.cisco.com,Interface:Eth1/49 | Stomp |
| 1 | 101 | bgl-aci06-spine1 | spine | eth1/2 | 968286 | 0 | 0 | 0 |
| Historic |
| 1 | 201 | bgl-aci06-t1-leaf1 | leaf | eth1/1 | 12 | 0 | 0 | 0 |
| Historic |
| 1 | 201 | bgl-aci06-t1-leaf1 | leaf | eth1/51 | 4999243774529 | 0 | 0 | 0 |
| Historic |
| 1 | 201 | bgl-aci06-t1-leaf1 | leaf | eth1/52 | 5002807353809 | 127841212 | 0 | 0 |
System:bgl-aci06-t2-leaf2.cisco.com,Interface:Eth1/49 | Stomp |
| 1 | 202 | bgl-aci06-t1-leaf2 | leaf | eth1/51 | 968286 | 0 | 0 | 0 |
| Historic |
| 1 | 301 | bgl-aci06-t2-leaf1 | leaf | eth1/44 | 4999245287405 | 0 | 4999245287405 | 0 |
| Historic |
| 1 | 301 | bgl-aci06-t2-leaf1 | leaf | eth1/49 | 4999823953891 | 0 | 0 | 0 |
| Historic |
| 1 | 302 | bgl-aci06-t2-leaf2 | leaf | eth1/49 | 4999243774529 | 0 | 0 | 0 |
| Historic |
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

7. Bovendien gaat het script de volgende opties aan de gebruikers aanbieden om korrelgegevens te sorteren en weer te geven, die zijn verzameld door script-1 en 2.

Gebruiker kan een optie tussen nummer 1-3 als invoer kiezen. Zie het voorbeeld hieronder.

- 1.Sort the data further
- 2.View the granular data of an interface
- 3.Exit

Input the number:

In het onderstaande voorbeeld gaan we naar optie 2 die ons helpt om korrelgegevens voor elke interface te bekijken.

Het script vraagt de gebruiker om het POD-nummer, Node-ID en interface-ID in de tabel die boven is afgedrukt (stap 6) in te voeren.

In dit voorbeeld gebruiken we 1-302-eth1/44, waar POD-ID 1 is, Node-ID 302 en interface-ID 1/44. Dit is de interface

waar lokale FCS door het script werd gemeld, zoals getoond in stap 6.

Input the number:2

Enter an interface for which you need granular data(POD_ID-NODE_ID-INTERFACE Example:1-101-eth1/5): 1-302-eth1/44

You have CRC and FCS data in the below date range
1.2021-09-27

Enter the date for which you need granular data(any number from the above list range(1-1)):

In ons voorbeeld verzamelden we de gegevens slechts voor een paar minuten van een dag, dus zien we slechts één optie voor gedateerd 27 sep.

Onze input zal dus "1" zijn.

Enter the date for which you need granular data(any number from the above list range(1-1)): 1

```
+-----+-----+-----+
| Time | CRC | FCS |
+-----+-----+-----+
| 11:28 | 5002806823759 | 5002806823759 |
| 11:33 | 5002934665647 | 5002934665647 |
+-----+-----+-----+
```

Do you want to continue viewing the granular data(0/1), 1=yes, 0=no:0

Please select any number below to sort the data further or to view granular data of an interface

- 1.Sort the data further
- 2.View the granular data of an interface
- 3.Exit

Input the number:3

ABCD-M-G24X:downloads abcd\$