CONFIGURATION GUIDE

Debug Tools

Introduction

Debug Tools is a suite of instruments that can help in the debug process by bringing up additional information based on test results.

KEYSIGHT TECHNOLOGIES

Table of Contents

UE Filter

Description

Per UE PCAP Filter is a tool that filters the packets generated by 4G and 5G architectures for every User Equipment and creates individual captures based on different fields that identifies an UE.

The filtration brings together all the packets that are related to an UE by using multiple layers of filtration and using directly the content of the packets.

The protocols that can be filtered:

- NG Application Protocol (NGAP), by:
    - Making a connection between Ngap and imsi, using:
        - Ngap Id
        - Mcc
        - Mnc
        - Msin
        - Ip address
- S1 Application Protocol (S1AP), by:
    - Making a connection between S1ap id and imsi, using:
        - S1ap id
        - Imsi
- GPRS Tunneling Protocol (GTP), by:
    - Making a connection between the ip addresses of the packets that are found in NGAP packets
    - Making a connection between the ip addresses of the packets that are found in S1AP packets
    - Making a connection between the ip addresses of the packets that are found in HTTP/2 packets
- GPRS Tunnelling Protocol version 2 (GTPv2), by:
    - Making a connection between imsi and teid (Tunnel Endpoint Identifier)
- Packet Forwarding Control Protocol (PFCP), by:
    - Making a connection between imsi and seid (Session Endpoint Identifier)

- Diameter, by:
    - Making a connection between session id and multiple Attribute–Value Pairs (AVPs)
- HTTP/2
    - The specific paths that can be filtered are shown below under the *SBI Validator - The packets that can be validated,* they are the same

OS Systems that are supported:

- Ubuntu 20.04.3 LTS
- Windows 10 20H2

Requirements

Requirements for Ubuntu

The following packages are required to run:

Name	Installation	Version check	Working version
Python	• sudo add-apt-repository ppa:deadsnakes/ppa • sudo apt-get update • sudo apt-get install python3.8	• python3 -V • python -V	> Python 3.8.* (3.8.10)
Pip	sudo apt install python3-pip	pip -V	> pip 20.0.2
Pyshark	pip install pyshark	pip show pyshark | grep Version	> Pyshark 0.4.3
Tshark	sudo apt install tshark	tshark -v	TShark 3.2.3
Getmac	pip install getmac	pip show getmac | grep Version	Getmac 0.8.2
Progress	pip install progress	pip show progress | grep Version	Progress 1.6
Requests	pip install requests	pip show requests | grep Version	Requests 2.22.0

Requirements for Windows

The following packages are required to run:

Name	Installation	Version check	Working version
Python	Python should be installed from here, not from Microsoft store	• python3 -V • python -V	> Python 3.8.* (3.8.10)
Pip	pip is already installed with Python in newer versions	pip -V	> pip 20.0.2
Pyshark	pip install pyshark	pip show pyshark | findstr Version	> Pyshark 0.4.3
Tshark	• Install Wireshark for Windows 10 • Add in Path the path of the folder that contains **tshark.exe**	See Wireshark version in app	Wireshark 3.6.0
Getmac	pip install getmac	pip show getmac | findstr Version	Getmac 0.8.2
Progress	pip install progress	pip show progress | findstr Version	Progress 1.6
Requests	pip install requests	pip show requests | findstr Version	Requests 2.22.0
Pypiwin32	pip install pypiwin32	pip show pypiwin32 | findstr Version	Pypiwin32 223

Notes

- In order to install a specific package version with pip, the following command can be used: **pip install YourPackage==YourVersion**
- The executable tshark.exe should be in **C:\Program Files\Wireshark** (for Windows).
- There were problems with older versions of the python packages, but newer versions should work
- There were problems with newer versions of Wireshark, the testing worked for up to Wireshark 3.6.2

- In order to add a path into Path environment variable, the steps from here should be followed.
- The builds are OS specific, so the build for Ubuntu 20.04 will not work on another versions (eg: 18). Cross-compiling errors may appear.

How to run

The following arguments can be used to run:

Argument meaning	Argument	Type
Print help message	-h, --help	Optional
Path of the capture file	-f FILE, --file FILE	Mandatory
Release version	-r {15,16}, --release	Mandatory
Debug on	-d, --debug	Optional
The ip of the licensing server	-ip SERVER_IP, --server_ip SERVER_IP	Mandatory

Output

When the program is running, on the standard output will be shown some progress bars about the stage of the filtration and then the number of captures that have been generated.

The captures for every UE can be found near the capture itself in a folder with a relevant name, containing the name of the capture and the timestamp.

Another file that is created is a log file containing important logs about every layer of filtration and the packets that have been skipped during the process, but also the status of the connection with the licensing server. The output of this file depends on the debugging mode that can be set as a command-line argument.

Some reasons why some packets have been skipped can be either the packet is not directly related to an UE, or the type of packet is not supported yet. The packets that have been skipped are included in the log file along with a wireshark filter.

At the end of the log file is a list with every UE's wireshark filter that can be used in the initial capture.

SBI Validator

Description

The main purpose of the tool is to validate the YAML schema in the payload of both requests and responses HTTP/2 packets used in 5G procedures from a Wireshark capture.

The packets that can be validated

Procedures provided by different nodes that can be validated:

- Nausf
    - nausf-auth
        - /ue-authentications
        - /ue-authentications/{authCtxId}/5g-aka-confirmation
- Nudm
    - nudm-ueau
        - /{supiOrSuci}/security-information/generate-auth-data
        - /{supi}/auth-events
    - nudm-uecm
        - /{ueId}/registrations/amf-3gpp-access
        - /{ueId}/registrations/smf-registrations/{pduSessionId}
    - nudm-sdm
        - /{supi}/nssai
        - /{supi}/am-data
        - /{supi}/smf-select-data
        - /{supi}/sdm-subscriptions
        - /{ueId}/id-translation-result
        - /{supi}/sdm-subscriptions/{subscriptionId}
        - /{supi}/sm-data
- Npcf
    - npcf-am-policy-control
        - /policies
        - /policies/{polAssoId}/update
        - /policies/{polAssoId}
    - npcf-smpolicycontrol
        - /sm-policies
        - /sm-policies/{smPolicyId}/delete

        - /sm-policies/{smPolicyId}/update
        - /sm-policies/{smPolicyId}
        - /sm-policies, SmPolicyUpdateNotification
        - /sm-policies, SmPolicyControlTerminationRequestNotification
    - npcf-policyauthorization
        - /app-sessions
        - /app-sessions/{appSessionId}/delete
        - /app-sessions, eventNotification
        - /app-sessions/{appSessionId}/events-subscription
        - /app-sessions/{appSessionId}
    - npcf-ue-policy-control
        - /policies
        - /policies/{polAssoId}

- Nudr
    - nudr-dr
        - /policy-data/ues/{ueId}/am-data
        - /policy-data/ues/{ueId}/sm-data
        - /policy-data/subs-to-notify
        - /policy-data/subs-to-notify/{subsId}
        - /application-data/influenceData/subs-to-notify
        - /application-data/influenceData/subs-to-notify/{subscriptionId}
        - /application-data/subs-to-notify
        - /application-data/serviceParamData/{serviceParamId}

- Nsmf
    - nsmf-pdusession
        - /sm-contexts
        - /sm-contexts/{smContextRef}/modify
        - /sm-contexts/{smContextRef}/release
        - /sm-contexts/{smContextRef}/retrieve

- Namf
    - namf-comm
        - /ue-contexts/{ueContextId}/n1-n2-messages/subscriptions/{subscriptionId}
        - /ue-contexts/{ueContextId}/n1-n2-messages/subscriptions
        - /ue-contexts/{ueContextId}/n1-n2-messages

        - /ue-contexts/{ueContextId}/assign-ebi
        - /ue-contexts/{ueContextId}
    - namf-evts
        - /subscriptions
        - /subscriptions/{subscriptionId}

- Nchf
    - nchf-spendinglimitcontrol
        - /subscriptions
        - /subscriptions/{subscriptionId}
        - /subscriptions, statusNotification
        - /subscriptions, subscriptionTermination
    - nchf-convergedcharging
        - /chargingdata
        - /chargingdata/{ChargingDataRef}/update
        - /chargingdata/{ChargingDataRef}/release
        - /chargingdata, myNotification

- Nnrf
    - nnrf-nfm
        - /nf-instances/{nfInstanceID}
    - nnrf-disc
        - /nf-instances

- N5g-eir
    - n5g-eir-eic
        - /equipment-status

- Nsmsf
    - nsmsf-sms
        - /ue-contexts/{supi}/sendsms
        - /ue-contexts/{supi}

- Nnssf
    - nnssf-nsselection
        - /network-slice-information

- nnssf-nssaiavailability
  - /nssai-availability/{nfId}
  - /nssai-availability/subscriptions
  - /nssai-availability/subscriptions/{subscriptionId}

- 3gpp

  - 3gpp-as-session-with-qos
    - /{scsAsId}/subscriptions
    - /{scsAsId}/subscriptions/{subscriptionId}
  - 3gpp-service-parameter
    - /{afId}/subscriptions
    - /{afId}/subscriptions/{subscriptionId}

OS Systems that are supported:

- Ubuntu 20.04.3 LTS
- Windows 10 20H2

Requirements

Requirements for Ubuntu

The following packages are required to run:

Name	Installation	Version check	Working version
Python	Python should be installed from here, not from Microsoft store	• python3 -V • python -V	> Python 3.8.* (3.8.10)
Pip	sudo apt install python3-pip	pip -V	> pip 20.0.2
Pyshark	pip install pyshark	pip show pyshark | grep Version	> Pyshark 0.4.3
Tshark	sudo apt install tshark	tshark -v	TShark 3.2.3
Getmac	pip install getmac	pip show getmac | grep Version	Getmac 0.8.2
Progress	pip install progress	pip show progress | grep Version	Progress 1.6
Jsonschema	pip install jsonschema	pip show jsonschema | grep Version	Jsonschema 3.2.0
Json-ref-dict	pip install json-ref-dict	pip show json_ref_dict | grep Version	Json-ref-dict 0.7.1
Pyyaml	pip install pyyaml	pip show pyyaml | grep Version	Pyyaml 5.3.1
Requests	pip install requests	pip show requests | grep Version	Requests 2.22.0

Requirements for Windows

The following packages are required to run:

Name	Installation	Version check	Working version
Python	• Python should be installed from here, not from Microsoft store • Add in Path the path of Python Scripts. See in notes below more information	• python3 -V • python -V	Python 3.8.* (3.8.10)
Pip	pip is already installed with Python in newer versions	pip -V	pip 20.0.2
Pyshark	pip install pyshark	pip show pyshark | findstr Version	Pyshark 0.4.3
Tshark	• Install Wireshark for Windows 10 • Add in Path the path of the folder that contains **tshark.exe**	See Wireshark version in app	> Wireshark 3.6.0
Getmac	pip install getmac	pip show getmac | findstr Version	Getmac 0.8.2
Progress	pip install progress	pip show progress | findstr Version	Progress 1.6
Jsonschema	pip install jsonschema	pip show jsonschema | findstr Version	Jsonschema 3.2.0
Json-ref-dict	pip install json-ref-dict	pip show json_ref_dict | findstr Version	Json-ref-dict 0.7.1
Pyyaml	pip install pyyaml	pip show pyyaml | findstr Version	Pyyaml 5.3.1

Notes

- In order to install a specific package version with pip, the following command can be used: **pip install YourPackage==YourVersion**

- The Python Scripts folder path should look like this: **C:\Users\USER_NAME\AppData\Local\Programs\Python\Python*(your python version)\Scripts.** This path should be added in Path environment variable in order for the

python packages to be found. Modify it to comply with your path. The packages need this folder in Path because they have to be recognised as commands (by their names).

- There were problems with older versions of the python packages, but newer versions should work
- In order to add a path into Path environment variable, the steps from here should be followed.
- The builds are OS specific, so the build for Ubuntu 20.04 will not work on another versions (eg: 18). Cross-compiling errors may appear.
- If the path of the input file contains spaces or other characters that should be escaped, put the path between quotes, like **-f "path to capture"**.

How to run

The following arguments can be used to run:

Argument meaning	Argument	Type
Print help message	-h, --help	Optional
Path of the capture file	-f FILE, --file FILE	Mandatory
Release version	-r {15,16}, --release	Mandatory
Path to the output directory. If missing, the path will be a folder near the capture	-d DIR, --dir DIR	Optional
Ip addresses for optional filtration	-ips IP IP, --ips IP IP	Optional
The ip of the licensing server	-ip SERVER_IP, --server_ip SERVER_IP	Mandatory

Output

The output of the tool is a folder near the capture or at the provided directory path from command line. The name of the folder contains the name of the capture and the timestamp.

Inside the output folder there are 3 files:

- FAIL.txt

- Contains for every wrong packet:
  - the TS file that the procedure is defined in (example: TS29503_Nudm_SDM.yaml)
  - the release version (15 or 16)
  - the type of packet (request or response)
  - the URI used in HTTP/2 layer
  - status code if response
  - filter for Wireshark
  - the cause of the failure (that can be caused by a missing required field, wrong schema, wrong regex and others)
  - the expected schema
  - the found data
- The number of failed validations

- SUCCESS.txt
  - Contains for every correct packet:
    - the TS file that the procedure is defined in (example: TS29503_Nudm_SDM.yaml)
    - the release version (15 or 16)
    - the type of packet (request or response)
    - the URI used in HTTP/2 layer
    - status code if response
    - filter for Wireshark
  - The number of succeeded validations

- LOG.txt
  - Error messages/stack traces
  - Status of the connection with the licensing server

Csv Analysis

Description

This tool is intended to parse and analyze all csv files (and logs, optionally) that are generated by a test executed with LoadCore. The output will be a report of all stats (and logs) that can be considered as errors or warnings for the test.

OS Systems that are supported

- Ubuntu 20.04.3 LTS
- Windows 10 20H2

Requirements

Requirements for Ubuntu

Name	Installation	Version check	Working version
Python	• sudo add-apt-repository ppa:deadsnakes/ppa • sudo apt-get update sudo apt-get install python3.8	• python3 -V • python -V	> Python 3.8.* (3.8.10)
Pip	sudo apt install python3-pip	pip -V	> pip 20.0.2
Getmac	pip install getmac	pip show getmac | grep Version	Getmac 0.8.2
Requests	pip install requests	pip show requests | grep Version	Requests 2.22.0
Pypiwin32	pip install pypiwin32	pip show pypiwin32 | grep Version	Pypiwin32 223
Matplotlib	pip install matplotlib	pip show matplotlib | grep Version	Matplotlib 3.5.1
Pylatex	pip install pylatex	pip show pylatex | grep Version	PyLaTeX 1.4.1
texlive-latex-base	sudo apt-get install -y texlive-latex-base	apt list --installed | grep texlive	2019.20200218-1
texlive-latex-extra	sudo apt-get install -y texlive-latex-extra	apt list --installed | grep texlive	2019.202000218-1

Name	Installation	Version check	Working version
latexmk	sudo apt-get install -y latexmk	latexmk --v	4.67

Requirements for Windows

The following packages are required to run:

Name	Installation	Version check	Working version
Python	Python should be installed from here, not from Microsoft store	• python3 -V • python -V	> Python 3.8.* (3.8.10)
Pip	pip is already installed with Python in newer versions	pip -V	> pip 20.0.2
Getmac	pip install getmac	pip show getmac | findstr Version	Getmac 0.8.2
Requests	pip install requests	pip show requests | findstr Version	Requests 2.22.0
Pypiwin32	pip install pypiwin32	pip show pypiwin32 | findstr Version	Pypiwin32 223
Matplotlib	pip install matplotlib	pip show matplotlib | findstr Version	Matplotlib 3.5.1
Pylatex	• pip install pylatex • Install Miktex Console for Windows in order to have access to **latexmk** compiler • Add in Path the path of the folder that contains **latexmk.exe.**	pip show pylatex | findstr Version	PyLaTeX 1.4.1 Miktex Console 4.6
Perl	See below in Notes	N/A	N/A

Notes

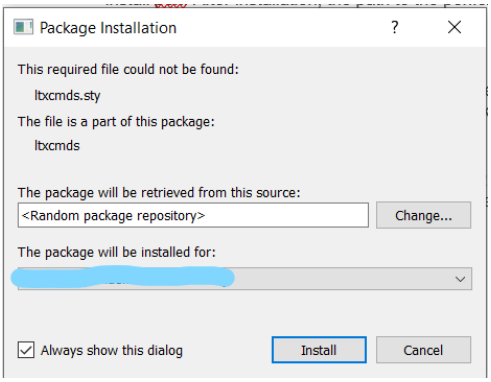
- **Latexmk** compiler can be installed from Miktex Console, following the next steps:
    - Start the Miktex Console app
    - Go to packages tab on the left

    - In the field under the “Install in” path enter **latexmk**
    - Right click on **latexmk**
    - Install package
- The folder path containing latexmk.exe executable should look like this: **C:\Users\USER_NAME\AppData\Local\Programs\MiKTeX\miktex\bin\x64** This path should be added in Path environment variable in order to generate the pdf. The generation uses latexmk compiler
- In order to add a path into Path environment variable, the steps from here should be followed.

- **Miktex Console** can be installed from here.

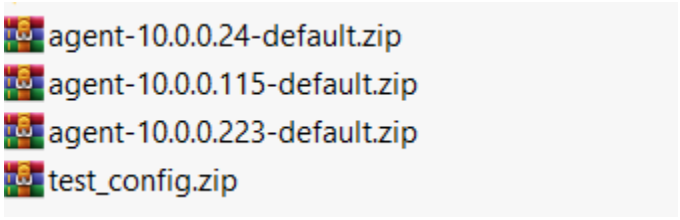
- **Perl** can be installed from here -> DOWNLOAD STRAWBERRY PERL -> select 64bit installer, then follow the steps of the installation. After installation, the path to **perl.exe** should be automatically placed in Path. The path to perl.exe should look like this: **C:\Strawberry\perl\bin.**

- While running the first time on Windows, **pop-ups** like below should appear. Press Install on all of them. The packages are required to generate the pdf using latex.



- The builds are OS specific, so the build for Ubuntu 20.04 will not work on another versions (eg: 18). Cross-compiling errors may appear.

- As shown below under **“How to run”**, there can be used an optional argument while running, *-log / --enable-logs.* The archive containing the logs should be placed inside the input folder containing csv files. In this way, 2 arguments for 2 input folders have been avoided. The contents of the archive should look like below.



How to run

The following arguments can be used to run:

Argument meaning	Argument	Type
Print help message	-h, --help	Optional
Path to the input folder containing csv statistics files	-dir DIRECTORY, --directory DIRECTORY	Mandatory
Enable logs parser	-log, --enable-logs	Optional
The ip of the licensing server	-ip SERVER_IP, --server_ip SERVER_IP	Mandatory

Output

The output consists in:

- Inside the input folder, near the csv files, there will be output folders for every run, including timestamp, like: **csv_analysis_output_2022-03-09_16-59-09**
- Inside the output folder, there will be:
  - **csv_analysis.pdf**, containing a summary for errors and warnings from csv statistics.
  - **log_analysis.pdf** if log parser is enabled. The pdf contains a table for every log file, from every agent. And for every error/warning, the following fields are showed:
    - thread id
    - log_type
    - date
    - timestamp
    - details (details of the error or a hyper reference to the stack trace)
  - **info.log** file containing information about the license server.

Known limitations

SBI Validate

- YAML schemes that have the property **nullable = True** set on true in the packet itself are not supported in the validation. For example, if **PpSubsRegTimer** from **TS29503_NUDM_PP.yaml** comes null, it will not be validated. If it is not null, it will be validated against its schema.

UE Filter

- in terms of NGAP/NAS packets, only plain NAS messages (null scheme) are supported for now, not encrypted

- HTTPS encrypted captures are not supported

Learn more at: www.keysight.com

For more information on Keysight Technologies' products, applications or services, please contact your local Keysight office. The complete list is available at: www.keysight.com/find/contactus