# To clone git

* Open MoboXterm
* Start local terminal
* Type “ssh [root@192.168.0.132](mailto:root@192.168.0.132)” <password: root123>
* Type “ssh [root@192.168.0.1](mailto:root@192.168.0.132)15” <password: root123>
* Type “ssh-keygen”
* Open the generated key from the file stored in default location <vim /root/.ssh/id\_rsa.pub>
* Copy the key
* Open Bitbucket in browser < <https://at.mavenir.com/bb/projects/VEPC/repos/vcm-gerrit/browse>>
* Go to My account->SSH Key-> Add Key-> paste the key copied
* Again in MoboXterm Type “git clone ssh://10.10.205.50:7999/at/~tk/vcm-gerrit-tkishor.git”
* It starts cloning ….

# To clone Repos in new VM

* First create ssh key using

ssh-keygen -t rsa -b 4096 -C "49server" (49server can be any string other than existing old repos ssh key name)

* Then add the ssh key in bitbucket (click on profileicon->ManageAccount->SSh Key->Add ssh key)
* Copy ssh key from VM in this path /home/tk/.ssh/id\_rsa.pub
* Update bashrc file with (for compilation issue)

export PATH=/opt/VCM/bin/:/<path>

<path> can be get from “echo $PATH”

* Then clone using “git clone ssh://10.10.205.50:7999/at/~tk/vcm-gerrit-tkishor.git”
* To add git username and email address to checkin in new VM

git config --global user.email “kishorkumar.t@mavenir.com”

git config --global user.name “Kishorkumar.T”

* Then git remote update
* Then git checkout <previous vms localbranch name> that you want to clone here eg: git checkout feature/amf-2

# To compile the code

* Go go to /home/vcm-gerrit-tkishor path
* Type “source compile-setup-no-saf.sh”
* Type “./clean.sh”
* Type “./build-parallel.sh”
* To access the code Type “run.cscope” search for the keywords/filename
* To compile a specific module/component “./build-parallel.sh <component>”

eg: “./build-parallel.sh amfcom”

# To create a local branch

* New local branch can be created in two ways (using terminal is easier)

1. Using bitbucket
   * Open bitbucket
   * Under repository option move to your vcm-gerrit <vcm-gerrit-tkishor>
   * Click “Create branch” to create a local branch
   * Check for Repository<vcm-gerrit-tkishor>, Branch from <5gc-master>
   * Change Branch type accordingly “Feature/ etc…”
   * Enter “Branch name”
   * Then click “create branch”
   * Copy branch name
   * Go to Mobaxterm/Terminal
   * Git branch
   * Git checkout 5gc-master
   * Git pull –rebase
   * Git remote update
   * Check for newly created local branch

Git branch

* + Git checkout <copied branch name>
  + New branch created!

1. Using terminal
   * Move to master branch from where you want to create new local branch

git checkout <master branch name>

* + git pull --rebase
  + git checkout -b <new local branch name>
  + check whether new branch is created

git branch

* + Clean and compile to check for branch code validity status

source compile-setup-no-saf.sh

./clean.sh

./build-parallel.sh

# To push/commit the code

1. First time
   1. “git status” <check for modified files>
   2. “git add <file name 1>  <filename 2>”
   3. “git commit”
   4. <write your description> ; save and quit the file
   5. “git push origin <branch name>” (for 1st patch set push)
   6. copy and paste the terminal link to browser
   7. add reviewer name
   8. click on create pull req.
2. Subsequent time
   1. “git status” <check for modified files>
   2. “git add <file name 1>  <filename 2>”
   3. “git commit --amend”
   4. <write your description> ; save and quit the file
   5. “git push -f origin <branch name>” (for 1st patch set push)

# To setup DTF automation testing framework

1. Overall step is maintained in the below attached doc



* 1. In Step1
     1. 1st create fork in browser for the url “opmt” and “pytest-dtfcommon”
     2. for now url given for “5g-dtf2” is not supported, so use Charu’s url for “5g-dtf2” which is “git clone ssh://10.10.205.50:7999/at/~aggarwalc/5g-dtf2-charu.git 5G” <ask charu to give permission>
  2. In Step2
     1. Then in MobXterm clone the 3 repos “opmt”, “pytest-dtfcommon”, “5g-dtf2”

Command to clone: git clone ssh://10.10.205.50:7999/at/~tk/opmt.git opmt

* + 1. Use Charu’s repo to clone “5g-dtf2”
  1. Do not do Step3 instead do Step4
     1. Install python3, then do Step3
     2. In step3, while executing the step “virtualenv -p /usr/local/bin/python3 My\_Env” if path is wrong, search for correct path where python is installed using the command “which python3” and use that path instead

eg: “virtualenv -p <path>/python3”

* + 1. After completing step3, then continue “pip installation part” in step-4
    2. Step4 is completed
  1. In Step5
     1. Create a folder in /home/<folder>
     2. Inside that folder copy the binaries manually as per Step5(c)
     3. To copy binaries
     4. Cd /home/vcm-gerrit-tkishor/bin.debug/
     5. Cp vcmAmfCom vcmEif vcmUas vcmUdsfFe /home/<folder>
     6. Then go inside wtf folder and do Step5(c)
  2. In Step6
     1. In 5G/product\_description.yml paste below lines and change ip address

1. systems:
2. - class: 5GMasarSystem
3. cluster\_id: 0
4. name: P0
5. nodes:
6. - name: node0
7. oam\_ip: 127.0.0.1
8. ipc\_ip: 127.0.0.1
9. ssh\_credentials: {root: root123}
10. gnode\_ip: 192.168.0.115
11. mme\_local\_ip: 192.168.0.115
12. mme\_remote\_ip: 192.168.0.115
13. ssh\_credentials: {root: root123}
15. * 1. Create a folder in /home/<folder>
      2. Inside /etc/hosts file replace the below lines

192.168.0.112 localhost

127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4

::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

#192.168.44.129 cnrfSvc

127.0.0.1 amfapsvc.webscale490.com

127.0.0.1 n2iwfsvc.webscale490.com

#192.168.0.36 n2iwfsvc.webscale490.com

127.0.0.1 gtpciwfsvc.webscale490.com

127.0.0.1 amfcomsvc.webscale490.com

127.0.0.1 amfeesvc.webscale490.com

127.0.0.1 amfcomsvc127.0.0.1.webscale490.com

127.0.0.1 amfmtsvc127.0.0.1.webscale490.com

127.0.0.1 amfpmsvc.webscale490.com

127.0.0.1 cnrfsvc.webscale490.com

127.0.0.1 cnrfsvc.5gc

127.0.0.1 ausf1.mavenir.com

127.0.0.1 udm1.mavenir.com

127.0.0.1 nssf1.mavenir.com

127.0.0.1 smf1.mavenir.com

127.0.0.1 nrf1.mavenir.com

127.0.0.1 n2iwf-deployment-1-789b7c87bb-njmb6

127.0.0.1 gtpiwf-deployment-1-789b7c87bb-njmb6

127.0.0.1 amfapsvc127.0.0.1.webscale490.com

* + 1. Inside /opt/VCM/config/eif/nps.conf file replace the below lines

compIpAddr=192.168.0.115 <should be your ip>

enableEvt=1

compInstance=1

enableStat=1

enableDao=1

evtDomainId=14 <should be unique; check no other user is using the no picked by you>

enableAmf=1

enableSig=1

nfType=amf

udsfIp=127.0.0.1

serviceType=cnrf

* + 1. In test case change only udsf present in “@pytest.mark.usefixtures(” to udsfcom
    2. Similarly in products/\_common/fixtures/processes.py change only “def usdf” to udsfcom
  1. In Step7
     1. Before executing the Test case need to enable #“if 0” in Amfcom for testing purpose
     2. After enabling; compile;copy and replace the Amfcom to /home/<folder> where binaries are copied in Step-5
     3. Then in another terminal capture PCAP logs by executing the below lines

tshark -f "sctp || tcp port 3000 || tcp port 3001 || tcp port 3002 || tcp port 3003 || tcp port 3004 || tcp port 3005||tcp port 5000 || tcp port 8080 || tcp port 8081 || tcp port 8082 || tcp port 8083 || tcp port 8084 || tcp port 8085 || udp port 2123 || udp port 38412 || sctp port 38412" -i any -P -w /tmp/DTF.pcap

* + 1. Go back to working terminal; run the test case
    2. Logs can be seen in /opt/VCM/logs/amfcom/vcm-amfcom1.log; /tmp/DTP.pcap

1. When you exit the virtual test environment; to go inside virtual environment
   * 1. Type “source 5G\_ENV/bin/activate” inside “wtf” folder
2. Each to re-execute the test case
   * 1. Go to another terminal
     2. Cd /root
     3. ./recreate-control-session.sh
     4. Back to DTF\_ENV terminal then “./stop\_amf.sh” and then execute the test case
     5. If you face any issue goto /root/ and run ./create-bucket\_control.sh
     6. Make sure to update AmfCom.json file present in wtf/5G/products/amf//config/mgmt/AmfCom.json
3. To skip a TC from running via Jenkins or ./run\_amf.sh add the following line before @pytest.mark.ds('DSAMF')

@pytest.mark.skip(reason='Testcase need to verify again')

# Debugging Assertion error at N2IWF/AMFCOM

1. In /opt/VCM/config/mgmt/AmfCom.json check and set the following values

+ "tPct": "30",

+ "disableAuthenticationCheck": true,

+ "disableIntegrityCheck": true,

+ "suciFormat15\_1\_0": false,

+ "numWorkerQueues": 1,

1. Make sure couchbase is running To check whether couchbase is running or not “/etc/init.d/couchbase-server status” and to start “/etc/init.d/couchbase-server start”

1. Make sure to run ./create-bucket\_control.sh in path /root/ if you face all binaries running manually but registration stopped after sending initial registration request to AMF
2. My couchbase console URL: [http://192.168.0.115:8091/ui/index.html#/overview](http://192.168.0.115:8091/ui/index.html" \l "/overview)
3. If printf statement is added in code instead of VCM\_LOG\_DEBG(), printf statements should be found at /opt/VCM/etc/stdouts/amfcom/amfcom\_\*.txt
4. Jenkins amfcom only build:

My git: ssh://git@at.mavenir.com:7999/~tk/vcm-gerrit-tkishor.git

Set in DOCKER\_LIST

Charu: amf-ap amf-comm amf-pathmgmt udsf udsf-util udsf-mgmt ts-man ts-exp ts-coordinator amf-n2-iwf amf-gtpc-iwf sp-disc

Palani: amf-ap amf-comm amf-ee amf-mt amf-pathmgmt amf-slicemanagement amf-n2-iwf amf-gtpc-iwf udsf udsf-util udsf-mgmt ts-man ts-exp ts-coordinator sp-disc

# PCAP log view in wireshark – Decrypt security protected nas msg

1. Go to Edit->Preferences->Protocols->NAS-5GS
2. Check the box “Try to detect and decode EEA0 ciphere msg”
3. All encrypted NAS msg will be decrypted!

# Start prometheus

1. Start the prometheus service and enable it to launch everytime at system startup.
   1. (no need to go here – status can be checked anywhere) cd /etc/systemd/system/
   2. systemctl start prometheus
   3. systemctl enable prometheus
2. to check status
   1. systemctl status prometheus
3. to restart
   1. systemctl restart prometheus
4. to run Prometheus
   1. change the ip address to 192.168.0.115 in /home/prometheus/prometheus/prometheus.yml
   2. change the port number next to ip address in /home/prometheus/prometheus/prometheus.yml same as the promPort number in /opt/VCM/config/\*.json file
5. to check the validity of Prometheus.yml file
   1. cd /home/Prometheus/prometheus
   2. ./promtool check config prometheus.yml <check for Success or no Error>

# AMF related compilation

./build-parallel.sh amfcom

./build-parallel.sh amfmt

./build-parallel.sh amfee

./build-parallel.sh uas

./build-parallel.sh rif

./build-parallel.sh amfpm

./build-parallel.sh amfslicemgmt

Or

./build-parallel.sh amfcom; mv build.log build\_amfcom.log;./build-parallel.sh amfmt; mv build.log build\_amfmt.log; ./build-parallel.sh amfee; mv build.log build\_amfee.log;./build-parallel.sh uas; mv build.log build\_uas.log;./build-parallel.sh rif; mv build.log build\_rif.log;./build-parallel.sh amfpm; mv build.log build\_amfpm.log;./build-parallel.sh amfslicemgmt; mv build.log build\_amfslicemgmt.log

# Json update& validation steps

1. Copy the existing vcm-base/3rd-party/json/config/amf/amfComConfig/data/Conf.json to a note pad.
2. Add the changes I have mentioned below.
3. Validate the json with <https://jsonlint.com/>
4. Copy the validated json to <https://jsonschema.net/>
5. Click on INFER SCHEMA
6. Click on the middle tab of RHS copy the same.
7. Login to 192.168.0.157 root/root123
8. Go to folder /root/rivu/**kishorT** parallel to /root/rivu/**Priyanka**
9. I have created a file named **Conf.schema.json**
10. Paste yours generated schema in this file.
11. Run this command replace the red highlighted part with your name

 js2model -o /root/rivu/**priyanka/yangConfig**/amfComConfig -l cpp --namespace Vcm5gc --prefix VcmAmfCom /root/rivu/**priyanka/yangConfig**/Conf.schema.json

Use below command

js2model -o /root/rivu/kishorT/amfComConfig -l cpp --namespace Vcm5gc --prefix VcmAmfCom /root/rivu/kishorT/Conf.schema.json

1. This command will generate \*.h and \*.cpp files.
2. Copy all \*.h to vcm-base/3rd-party/json/config/amf/amfComConfig/include
3. Copy all \*.cpp to vcm-base/3rd-party/json/config/amf/amfComConfig/src
4. Copy Conf.schema.json to vcm-base/3rd-party/json/config/amf/amfComConfig/data
5. Copy Conf.json to vcm-base/3rd-party/json/config/amf/amfComConfig/data.
6. Run ./clean.sh then build-parallel.sh amfcom
7. Place Updated  vcm-base/3rd-party/json/config/amf/amfComConfig/data/Conf.json in this path /opt/VCM/config/mgmt. And rename it as AmfCom.json
8. Go to VcmAmfConfigManager.cpp read and add the new parameters.
9. Run the amfcom binary manually to test if it is reading the newly added configuration.
10. To check the log go to /opt/VCM/log/
11. Then update Json file present in vcm-deployments/demo-charts/cnf-tpl-amf-demo/charts/namf-comm/config/mgmt/ same as vcm-base/3rd-party/json/config/amf/amfComConfig/data/Conf.json
12. Raise pull request from the config branch to 5gc-master.

# Yang update and validation steps

1. Refer supporting points



1. Once you are done Json updates, modify the yang file accordingl in below path

vcm-deployments/demo-charts/cnf-tpl-amf-demo/charts/namf-comm/config/mgmt.

1. copy “helm” from /tmp/ directory of 192.168.0.132 Machine ,And place it in /tmp/ of your vm
2. Then fire this command.
3. /tmp/helm  template  <path of cnf-tpl-amf-demo directory  on your VM>
4. Eg: /tmp/helm template /root/Repo2/bitbucket/vcm-gerrit-priyanka/vcm-deployments/demo-charts/cnf-tpl-amf-demo/
5. Then follow the steps in attached file

# Spec references

## Paging

1. Paging Message IE (38413 section:9.2.4)
2. Paging procedures (38413 section:8.5) & 24501 section: (5.6.2)
3. DRX for Paging (38304 section:7)
4. NW triggered service request (23502 section:4.2.3.3)

# Other commands

1. To search a word in vim editor as case insensitive “/\c<word>”
2. To get CPU details “lscpu”
3. To get IP address “ifconfig”
4. To get VM details “hostnamectl status”
5. To find os name and version in Linux “cat /etc/os-release” or “lsb\_release -a” or “hostnamectl”
6. To show current memory usage “free -m”
7. To delete all the files expect 1 or two files in a directory
   * shopt -s extglob
   * rm -v !(“filename1”|”filename2”)
   * shopt -u extglob
8. To get port number from processID “netstat -anlp|grep <pid>”
9. To get pid of running process “ps -ef|grep <processname>”
10. To check whether couchbase is running or not “/etc/init.d/couchbase-server status”
11. To start the couchbase “sudo systemctl start couchbase-server” or “/etc/init.d/couchbase-server start”
12. To extract single file from tar.gz “tar -xf etc.tar etc/apt/sources.list”
13. To list the contents of tar file
14. To setup Microsoft authenticator code for Mavenir type this URL in browser: aka.ms/MFAsetup
15. To compile avro filler source file

g++ amfAvroFiller.cc -lavrocpp -I /usr/include -std=c++11

1. To kill a screen: screen -X -S screen\_name quit
2. To detach and reattach a screen: screen -dR
3. To attach a screen: screen -x
4. rm -rf ~/.ccache/ to delete cache files
5. to kill a process by name: pkill “processname”
6. to replace a string in multiple files in multiple folder

Move to the main folder and enter this command

grep -rli 'old-word' \* | xargs -i@ sed -i 's/old-word/new-word/g' @

1. To extract a \*.zip file -> unzip filename.zip

If zip file is corrupted then try either of these methods

* 1. Try extracting with 7z
     1. Install using apt install p7zip-full
     2. 7z x filename.zip
  2. Try extracting with Jar
     1. Install using sudo apt-get install fastjar
     2. Jar xvf filename.zip
  3. Try extracting with gunzip -> gunzip -f filename.zip

# Git commands

1. Visit: <https://gist.github.com/jctosta/af918e1618682638aa82>
2. To check the changes made in a specific file in the current branch alone

git log <filename>

1. To check the changes made in a specific file in the all branches

git log --branches=\* <file name>

1. To check the changes made in current branch alone but details in only 1 line

git log --oneline

1. To list of all known branch names

git branch -a

1. To check all the changes made in a specific branch alone

git log <branch name>

1. To create a branch

git branch <new branch name>

1. To rename a branch (stay in old branch)

Git branch -m <new branch name>

1. To list all the banches

git branch

1. To move to a specific branch

git checkout <branch name>

1. To create a new branch base off current HEAD and to move to that branch

git checkout -b <new branch name>

1. To create a new branch base off current branch and to move to that branch

git checkout -b <new branch name> <existing branch>

1. To delete the branch

git branch -D <branch name>

1. To revert a specific file to a specific commit use the reset command and then commit the changes in the current branch

git reset <commit hash> <filename>

git commit

git push -f origin <current branch name>

1. To revert the branch to a specific commit

git log –oneline <find the specific change commit-hash>

git reset --hard <commit-hash>

1. To resolve merge conflict

git pull –rebase origin 5gc-master

git status

Check for ‘both modified’ files, open each, search for HEAD and resolve conflict and save

git add <modified file names>

git rebase –continue <to abort rebase inbetween “git rebase –abort”>

commit and raise pull request

1. To group/combine multiple commits and push request into a single commit

Do as many commits as need but don’t force push

git commit1;git push origin <local branch>

when ready to combine find the number of commits

git reset --soft HEAD~<number of commits>&&git commit

number of commits: Indicates the total number of commits starting from first commit, then

git push -f origin <local branch name>.

(or)

* + Git log //find the first commitid
  + Git reset --soft <commitid>
  + Git commit --amend
  + Git push -f origin <branch\_name>

1. To delete the branch “git branch -D <branch name>”
2. To check the updated file “git status”
3. To change the branch “git checkout <branch name>”
4. To revert back the deletes files after commit
   * git log --diff-filter=D –summary <to find the deleted files and its commit id>
   * git checkout <commit-id>~1 <file-name>
   * eg: git checkout 76fd34ef~1 three.txt
5. To patch and to retrive a patch to move to new branch
   * To patch: git format-patch -1
   * To update the branch details: git remote update
   * Move to new branch: git checkout -b <newbranch>
   * Apply patch again: git am -3 < <PatchFilename>
   * While applying patch if any merge conflict is found
     + Open each file and resolve conflict
     + git add <filename>
     + git am –resolved
     + if there is no merge conflict, patch is done
     + git commit –amend (it creates pull request using previous patch)
     + git push -f origin <newbranch>
6. When pushed directly from master branch – master branch is corrupted and not up-to-date
   * git pull --rebase origin 5gc-master
   * git remote -v
   * git remote add upstream ssh://git@bb.mavenir.com:7999/vepc/vcm-gerrit.git
   * git remote -v
   * git pull --rebase origin 5gc-master
   * git remote update
   * git pull --rebase upstream 5gc-master
   * git log (check if the last commit is not mine)
   * git push -f origin 5gc-master
   * git branch
   * git checkout feature/new-smsf-selection (move to local branch)
   * git pull --rebase origin 5gc-master
   * Resolve merge conflict, commit and push
7. To list the modified files in git using commit-id

git show --pretty="" --name-only <commit-id>

1. While doing rebase if faced this issue “error: The following untracked working tree files would be overwritten by checkout:”

Apply this command “git clean -d -f .” and then rebase

1. If there are multiple commits say head,a,b,c you want to remove commits a,b and do separate commit after c
   * git log //find the specific commitid previous to the commitid want to delete
   * git reset --soft <commitid>
   * git commit //add msg and commit
   * git log //find the same commitid used in git reset –soft
   * git checkout <commitid> -- filename1 filename2 //enter the filenames which you need to commit separately after commit c
   * git commit -m “enter your msg”
   * git push -f origin <currentbranch name>
2. To remove unwanted files

git clean -fdx

1. To remove a commit

Eg:

commit c <HEAD>

commit b

commit a

to remove b,

git reabse -i HEAD~2

pick xxxx a

pick xxxx b

pick xxxx c

<x git reset HEAD^>

git stash

git log

git commit –amend

git push -f origin <branch name>

1. Command to change URL of origin

git remote set-url origin ssh://git@bb.mavenir.com:7999/~tk/vcm-gerrit-tkishor.git

1. To remove last commit

git reset HEAD^

1. To remove a middle commit

git rebase --onto <branch name>~<first commit number to remove> <branch name>~<first commit to be kept> <branch name>

e.g 1,2,3 are the commits to remove commits 2 & 3

git rebase --onto sample~3 sample~1 sample

1. To create a patch locally without commit
   1. git diff > my\_patch.txt

git apply my\_patch.txt (or)

* 1. git stash

git stash show -p

1. git add -A stages **all changes**

git add . stages new files and modifications, **without deletions** (on the current directory and its subdirectories).

git add -u stages modifications and deletions, **without new files**

# To Update 5GC-Eir (response)

1. Check the file to get ip address of 5gc-eir siumulator running location

/home/wtf/5G/products/\_common/initial\_config/nfPeerConfig.json

1. Go to that file, currently it is running in (eir:192.168.0.159) screen -x 7\*.risha
2. Goto the tab2
3. Cut running 5gc-eir simulator (ctrl+c)
4. Open the file /root/TryNghttp2/nghttp2/examples/ asio-sv.cc
5. Update 5gc-eir response( fill\_eir\_rsp() ) and then save the file
6. Type command “make”
7. Use below command to run the 5gc-simulator

./asio-sv 192.168.0.159 3000 1

# Code hints

1. Place where http2 response messages are handled

File: vcm-amfcom/handler/src/VcmCpeAmfHttp2MessageHandler.cpp

Function: processHttp2ResponseMsg()

# Errors and Resolution

1. Compilation issue: “No module named jsonref”

Resolution:

* 1. pip3 install jsonref
  2. If pip3 command not found, then install python3
     1. python –version
     2. sudo yum install python36u
     3. sudo yum install python36u-pip
     4. python –version

1. compilation issue: cannot find -lcjose

Resolution:

1. compilation issue

Resolution:

1. rm -Rf cmake-build-debug

2. git clean -fdx

3. ./clean.sh

4. ./build-parellel.sh

1. When i try to manually run AmfCom binary, i'm getting below issue

trying connect to NATS Server nats://127.0.0.1:28005

NATS Connection to Server nats://127.0.0.1:28005 Failed

NATS failed to initialize. exiting...

set VCM\_NATS\_DISABLE if NATS is not required

Exiting amfcom

Resolution: export VCM\_NATS\_DISABLE=1

1. pyang issue

Resolution: pip3 install pyang

1. rebase issue “Permission denied”

Resolution: vim /root/.ssh/known\_hosts and delete everything with square bracket []

1. "AssertionError: dyn\_cfg\_cli: request failed, reason: ConnectionRefusedError" issue while running sanity on amf-sprint31-b1

Resolution: add ("VCM\_DISABLE\_MTCIL\_REGISTRATION","1") in

a) products/\_common/fixtures/processes.py {def udsfcommon section} and

b) products/amf/fixtures/processes.py {def spdisc and common\_env[] sections}

1. a) To change log level in dtf

In file tests/dtf2/products/amf/config/mgmt/mavenir-amf.jsonchange “log-level”: “DEBG” for namf-comm

b) To enable TCP dump in dtf

In file tests/dtf2/run\_amf.sh add tcp-dump in below command

run\_command="${my\_dir}/run.sh -s --alluredir=${allure\_dir} --log-stdout --kill-by-fixture --tcp-dump-by-agent --stdout-log-level DEBUG --deployment-schema module\_5g.ds.amf.DSAMF --prod products/amf/config/other/product\_description.yml --dtf-ip oam\_ip=127.0.0.1 --dtf-ip gnode\_ip=$eth0IP --dtf-ip mme\_remote\_ip=$eth0IP"

1. amf-sprint39-b1 branch dtf/Jenkins issue
   * In this file: vcm-sp/disc/sp\_disc.cc
   * In this function: ResultCode Service::initialize() noexcept
   * Comment this return line:

if (!m\_discoveryController->m\_tracingMgr.init()) {

SP\_LOG\_ERRO("Failed to initialize SLT");

return INIT\_FAILED;

}

1. To get port number of service netstat -anlp|grep <pid/service>”
2. nghttio library issue in dtf
3. For DTF2 tests, please copy the below 2 libraries to /usr/lib64/

a.

$ wget --no-check-certificate <https://jenkins-5gc.cz.intinfra.com/job/5GC/job/5GC_nghttp2/55/artifact/install/usr/local/lib64/libnghttp2.so>  
$ wget --no-check-certificate <https://jenkins-5gc.cz.intinfra.com/job/5GC/job/5GC_nghttp2/55/artifact/install/usr/local/lib64/libnghttp2_asio.so>

b. mv libnghttp\* /usr/lib64/

c. Then at the place where binary is running, try this command: ldconfig

d. try this command and see if these are present in library

nm -C /usr/lib64/libnghttp2\_asio.so | grep "nghttp2::asio\_http2::client::session::session(boost::asio::io\_context&, boost::asio::ssl::context&, std::string const&, std::string const&, boost::posix\_time::time\_duration const&, bool)"

# Curl command

1. curl -v --http2-prior-knowledge -X POST "http://192.168.0.115:5000/namf\_comm/v1/release-pdu-session/supi-405050000000490" -d @/root/pdu-rel.json -H "content-type: application/json"
   1. IP and port should be your Amfcom IP/Amfcom http2 port
   2. use release-pdu-session for AMF-COMM-RELEASE-PDU-SESSION-REQ
   3. use deregister-ue-request for AMF-COMM-DEREGISTER-UE-REQ
   4. According to the request you need to create your json and pass in this curl request.
   5. as I pass /root/pdu-rel.json for AMF-COMM-RELEASE-PDU-SESSION-REQ
   6. this file need to present in your system.

# Other VM

1. My DTF VM: [root@192.168.0.115](mailto:root@192.168.0.115) (root123) Path: /home/
2. My Dev VM: [tk@10.10.205.49](mailto:tk@10.10.205.49) (KPassword4$) Path:/home/tk/
   1. [tk@10.10.205.32](mailto:tk@10.10.205.32) (KPassword4$) Path:/home/tk
   2. [tk@10.10.205.2](mailto:tk@10.10.205.2)8 (KPassword4$) Path:/home/tk
3. Json VM: [root@192.168.0.157](mailto:root@192.168.0.157) (root123) Path: /root/rivu/kishort/
4. Curl VM: [root@192.168.0.159](mailto:root@192.168.0.159) (root123) Path: /root/kishor
5. Non DTF test VM: [root@10.10.10.120](mailto:root@10.10.10.120) (mavenir) Path: /home/DS\_Reference\_Scripts
6. Couchbase console:

<http://192.168.0.115:8091/ui/index.html#/overview>

Username: Administrator

Password: abc123

1. TRL avro decode VM: [root@10.69.2.229](mailto:root@10.69.2.229) (mavenir), then \_cdm3 (192.191.160.110), cd /root/ command

java -jar avro-tools-1.10.0.jar fragtojson --schema-file <schema\_file\*.json> <trl-avro-file\*.avro>

# Run Custom build in 5GC-CI-Custom

1. First as per wiki page, add repo permisson

<https://at.mavenir.com/wiki/display/5G/AMF+DTF+steps#AMFDTFsteps-AddingTestcasein5gc-masterandtestinJenkins>

1. Open jenkins

<https://jenkins-5gc.cz.intinfra.com/job/5GC/view/All/job/5GC-CI-CUSTOM/>

1. Goto-> Jenkins->5GC->All->5GC-CI-CUSTOM
2. Click “Build with Parameters”
3. In field “REPO\_5GC” fill your Clone URI (eg: ssh://git@at.mavenir.com:7999/~tk/vcm-gerrit-tkishor.git)
4. Make sure latest changes in local branches are checked-in, rebase with 5gc-master and push the changes(no need to add/commit if there is not conflict)
5. In field “BRANCH-5GC” fill your local branch name where UTs are checked-in (eg: ut-up-resto)
6. In field “TEST-TO-RUN” fill as amf and then Click “Build”
7. Build will start and apporx take 1 hr
8. If Build failed, see “Open Blue Ocean” for error report
9. If Build pass, see “Allure report” for success report

# Sed command

1. To replace a string in a file [sed -i ‘s/originalstr/replacestr/g’ filename1 filename2]
2. To delete a line in a file [sed -i ‘1d’ filename1 filename2] where 1 is the line number to delete
3. To delete last line in a file [sed -i ‘$d’ filename1 filename2]
4. To delete range of line in a file [sed -i ‘1,3d’ filename1 filename2] it deletes line 1,2
5. To delete all lines expect 1 particular line in a file [sed -i ‘5!d’ filename1 filename2] it deleted all lines except 5th line
6. To delete a line which a matches a string in a file [sed -i ‘/matchingstr/d’ filename1 filename2]
7. To insert lines after a matching string [sed ‘/^matchingstr\*/a newline1\nnewline2’ filename1 filename2]
8. To insert lines before a matching string [sed ‘/^matchingstr\*/i newline1\nnewline2’ filename1 filename2]
9. To delete all files with same prefix/suffix

To List the files to be deleted: find . -name "\*.bak" -type f

To Delete the files: find . -name "\*.bak" -type f -delete

Or

find . -name "\*.file" -exec rm -rf {} \;

1. To move all the files to a folder in the same path(eg: a b c d are folders, should move bcd to a)

mv -f \* foldername

# Steps to generate code from YAML file (We did this for Nsmsf)

Check whether the YAML file is present in the folder or not. If not present then add it.  
  
cp Npcf/AMPolicyControl/data/Makefile.codec Nsmf/data/  
cp Npcf/AMPolicyControl/data/Makefile.generate Nsmf/data/  
  
Change path to vcm-base/3rd-party/json/3gpp/Rel-16-Jun-20 in Nsmsf/data/Makefile.generate

/home/tk/vcm-gerrit-tkishor/vcm-base/3rd-party/json/Nsmf

export VCM\_SRC\_ROOT=/home/tk/vcm-gerrit-tkishor/   
export PREFIX=VcmSmf  
export OAPI=TS29502\_Nsmf\_PDUSession.yaml

cd /home/siddiquiz/vcm-gerrit-zaheer/vcm-base/3rd-party/json/Nsmsf/data

make -f Makefile.generate clean

make -f Makefile.generate (This will create yaml to json)

cd ..

make -f Makefile.generate clean

make -f Makefile.generate (This will generate code from json)

# Steps to add new binary in dtf

1. In test script, add “amfee” in @pytest.mark.usefixtures()
2. In module\_5g/libs/processes/processes\_manager.py, add the binary’s actual name and alias name
3. In products/amf/fixtures/processes.py,
   * add a new function for amfee
   * in amf\_tcpdump(), inside Ports.used\_by\_processes(), add entry for amfee
   * add the respective amfconfig.get\_amfee\_port(), amfconfig.get\_amfee\_config\_dir() in products/amf/libs/amf\_config.py
   * add entry for mgmt/amfee/nps.conf
   * add entry for /opt/VCM/config/amfee/
4. In module\_5g.egg-info/SOURCES.txt, add entry products/amf/config/mgmt/amfee/nps.conf
5. In products/amf/config/mgmt/amfStaticConfig.json, add entry "amfEeServiceAddr" :"amfeesvc.webscale490.com:8086",
6. In module\_5g/ports.py, add entry for amfee

# Steps to generate core dump on segmentation fault on DTF setup

..................................................................

1.> Check if core dump generation is enabled or not in linux machine:

ulimit -c // This returns 0 , if its not enabled

2.> Set ulimit: ulimit -c unlimited

3.> On linux , core dump file location is defined in below path

cat /proc/sys/kernel/core\_pattern

//Output in our system:

>/opt/VCM/etc/scripts/vcmGenAndCompressCoreDump.sh %e %s %t %p

4.> Check if this script is present or not. If not copy it to /opt/VCM/etc/scripts

# K8s Testing

Wikipage: <https://at.mavenir.com/wiki/display/5G/K8+Dev-Setup#K8DevSetup-PerformingUn-Installation&InstallationonK8ssetup>:

1. Login to [root@192.168.0.132](mailto:root@192.168.0.132)
   1. Type: sh k8.sh -> it will login to localhost
   2. Type: alias ->it will list the different servers

alias \_cdm2='sshpass -p mavenir ssh 192.168.122.10'

alias \_du='sshpass -p mavenir ssh 10.69.2.107'

alias \_master2='sshpass -p mavenir ssh 10.1.42.53'

alias \_worker2\_1='sshpass -p mavenir ssh 10.1.42.54'

alias \_worker2\_2='sshpass -p mavenir ssh 10.1.42.55'

alias cdm='sshpass -p mavenir ssh 192.168.122.148'

alias cp='cp -i'

alias egrep='egrep --color=auto'

alias fgrep='fgrep --color=auto'

alias grep='grep --color=auto'

alias l.='ls -d .\* --color=auto'

alias ll='ls -l --color=auto'

alias ls='ls --color=auto'

alias master='sshpass -p mavenir ssh 10.1.42.43'

alias mv='mv -i'

alias rm='rm -i'

alias which='alias | /usr/bin/which --tty-only --read-alias --show-dot --show-tilde'

alias worker='sshpass -p mavenir ssh 10.1.42.51'

alias worker1='sshpass -p mavenir ssh 10.1.42.52'

1. From localhost login to \_du
   1. cd /home/devamf/kishor/
   2. From Jenkins custom build, Click Build Artificats and download 5gc\_charts.tgz file

Or we can copy using wget <url of 5gc\_charts> --no-check-certificate

Eg: wget https://jenkins-5gc.cz.intinfra.com/job/5GC/view/All/job/5GC-CI-CUSTOM/25496/artifact/5gc\_charts.tgz --no-check-certificate

* 1. Copy it to \_du:/home/devamf/kishor
     1. tar -xvf 5gc\_charts.tgz -> you will get vcm-deployments folder
     2. In vcm-deployments/chartModifyScripts/amf-1-values-production.yaml
        + In all microservices change this

From: image: “docker-5gc.cz.intinfra.com/5gc/amf-eefat:2021789r34-centos7”

To: image: "docker-5gc.cz.intinfra.com/user\_tk/amf-eefat:latest"

* + 1. In vcm-deployments/chartModifyScripts/amf\_install.sh
       - Update AMF\_DIR="/home/devamf/kishor/vcm-deployments/demo-charts-2.7/amf"
       - **Note:** For ‘5gc\_Release\_3453’ branch charts-2.15 is not present, only charts-2.7 is present.
       - Update kubectl create -f ../cluster-role/amf-cluster-role.yaml
       - Comment amf-2 info such as

#NAMESPACE\_2="mvnr-mtcil1-appln-amf-amf2"

# Install AMF Instance = 'amf2'

#${HELM\_BIN} --kubeconfig ${KUBECONFIG} install amf2 ${AMF\_DIR} \

# --namespace $NAMESPACE\_2 \

# -f amf-2-values-production.yaml \

# -f global-values-production.yaml

* 1. Type: su devamf ->this will change root name to [devamf@- …]

1. From Localhost login to worker
   1. Open /root/pull.sh
   2. Change the username

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-slicemanagementfat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-pathmgmtfat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-n2-iwffat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-gtpc-iwffat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-mtfat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-eefat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-apfat:latest

docker pull docker-5gc.cz.intinfra.com/user\_tk/amf-commfat:latest

* 1. Delete old image using

docker images -a | grep "user\_tk" | awk '{print $3}' | xargs docker rmi

* 1. Type: sh /root/pull.sh -> this will pull the latest images from Jenkins build.

1. Repeat Step 3 in worker1 also
2. From Localhost login to master
   1. cd /root
   2. amf -> to get amf pods and its status (kubectl get pods -A |grep amf-)
   3. to delete all pods manually (kubectl delete pods -n mvnr-mtcil1-appln-amf-amf1 --all --force)
   4. amf -> check no amf pods are running which means status is not 3/3 or 2/3. It should be 0/3
   5. kubectl get pods -A -> to list all the pods and its status
   6. Goto \_du and run sh amf\_install.sh
   7. Goto master and check amf -> pod status is 2/3 ready for running
   8. alias cli='sshpass -p admin ssh -p 30164 [admin@10.1.42.43](mailto:admin@10.1.42.43)'
   9. To get cli port

-Cli server is same as master server only port is changed

-kubectl get svc -A

-after 1099: you will see one port that is cli port in our case it should be 1099:30164

* 1. cli
  2. configure
  3. Copy entire code in vcm-deployments/configuration-commands/amf.txt and paste it here -> it will automatically run each lines that are copied
  4. At last you ill get -complete
  5. exit
  6. exit
  7. amf -> get podname and run the below command with pod-name

kubectl describe pod -n mvnr-mtcil1-appln-amf-amf1 <pod-name>

eg: kubectl describe pod -n mvnr-mtcil1-appln-amf-amf1 amf-comm-7d97dc77f6-nbpqc

* 1. In \_du Type: sh amf\_uninstall.sh
  2. amf -> check pod status is 3/3

1. Check logs in worker or worker1
   1. cd /data/logs
   2. Type: ls -lrt |grep <pod-name> -> this will list the log files

Eg: ls -lrt |grep amf-comm-7d97dc77f6-nbpqc

1. To check if the image is correct

In master: kubectl get deploy -n <namespace of the pod> -o wide (get the image that pod is using).

In worker: docker inspect <image>|grep commit (get the commit details of that image)

eg:

in master: kubectl get deployment -n mvnr-mtcil1-appln-amf-amf1 -o wide | grep amf-comm

in worker: docker inspect docker-5gc.cz.intinfra.com/user\_tk/amf-commfat:latest | grep commit

Steps shared by risha

tab 1  
vim /root/pull.sh :[root@amf-cluster-worker1 logs]  
- do a docker pull from this  
  
2. Login to worker2 : pull the images  
tab 2 : [root@amf-cluster-worker0 logs]# vim /root/pull.sh -- > do same as worker 1

3. Copy chart to du server from build .(vcm-deployment directory will be created )  
tab3 :/home/devamf/stack/21.5/vcm-deployments/chartModifyScripts  
[devamf@her\_mwp2\_1\_pf9\_du chartModifyScripts]$

4. Go to chartModifyScripts:  
/home/devamf/risha  
/home/devamf/stack/21.5/vcm-deployments/chartModifyScripts

5. Open below file and change image name if needed.  
amf-1-values-production.yaml  
- search image and change images name

6. Install with 2.7 charts  
vim amf\_install.sh  
# AMF charts directory  
AMF\_DIR="../demo-charts-2.7/amf"

run this script

7. Go to master  
go to tab 4 - master

when pod comes in 2/3 state - (amf)  
type cli  
configure  
copy /amf/vcm-gerrit-risha/vcm-deployments/configuration-commands/amf1.txt commands  
exit  
exit

then check pod is in 3/3 or not

amf\_uninstall.sh

8. Uninstall using amf\_uninstall.sh .  
9. Repeat same steps from 6 to 8 for 2.15 charts also

# Weekly Task Accomplishments:

|  |  |
| --- | --- |
| **Weekly Accomplishments** | |
| **05-Aug-2022** | |
| **Area** | **Tasks** |
| **Feature development** |  |
| **Bug fixes** | 1)AMF-3342: Log analysis is done. Discussed the issue with Palani. Waiting for other NF logs to debug further.  2)AMF-3295: Discussed with Palani on the issue. Fix is implemented, verified and merged in amf-sprint61-b1 and 22.2  3)AMF-3333: Fix is implemented and shared for testing. Waiting for results.  4)AMF-3341: Fix is implemented in sprint&22.2 branches and merged.  5)AMF-3315,3317: Fix is verified and merged. |
| **Code Reviews** | 1)AMF-3349: Discussed with charu and Debashis on this issue. Reviewed the code changes done by Charu. |
| **Support & Remarks** |  |