ANEXO B: Scripts de verificación

I. Documento readme.doc

El documento "readme.doc" nos permite conocer, entre otros, los tests que son realizados por cada uno de los scripts, es decir, como están distribuidos en los distintos programas.

Zip Label: Check_V5.zip

Total number of files: 34

List of the files included in the ZIP file:

Scripts:

- additional.sh
- capa.sh
- check_middleware.sh
- check_system.sh
- connexion.sh
- crontab.sh
- date.sh
- dynamic.sh
- fichier.sh
- firmware.sh
- general.sh
- instal.sh
- level004.sh
- level027.sh
- level-adden115.sh
- nitl9.sh
- oracle.sh
- servers.sh
- smp.sh
- ss7.sh
- stack.sh

Documents:

- Check_list_middleware_R22_ed1.pdf
- Check_list_system_R22_ed1.pdf
- MPI CHECK_LIST_Ed5.doc
- nitl225_09_checkup_after_installation_ed06.pdf
- readme.doc

Configuration file:

- conf_file (example file)

Addendum verification files:

- bep-adden115_4.inv
- fep-adden115_1.inv
- local-adden115_3.inv
- local-adden115_4.inv
- smp-adden115_3.inv
- smp-adden115_4.inv
- volatiles.inv

The current ZIP file includes all available scripts to check the platform installations.

The following table gives the main scripts:

SCRIPT NAME (alphabetical order)	FUNCTION	EDITION	EVOLUTION /Ed n-1
check_middleware.sh	- This script uses all commands detailed in the *Check Middleware* document located on the "nadmap" server. This document is included in the current zip file. The script calls the following scripts (also located in the current zip file): additional.sh, connexion.sh, general.sh, oraclme.sh, smp.sh, ss7.sh and also calls the configuration file. - The execution of this script generates the following files: "middle_verif", "filesystem" and "ss7_verif".	3	
check_system.sh	- This script uses all commands detailed in the *Check System* document located on the "nadmap" server. This document is included in the current zip file. The script calls the following scripts (also located in the current zip file): date.sh, dynamic.sh, crontab.sh, firmware.sh, servers.sh, instal.sh, fichier.sh, stack.sh. - The execution of this script generates the following files: "system_verif" , "filesystem" and "Licenses"	3	Test tool integration for level 004,027 and 115
nitl9.sh	- This script uses all commands detailed in the *nitl9* document located on the "nadmap" server. This document is included in the current zip file. - All file names to delete are copied in another script which will be run after the RTM agrees on this file listment - The execution of this script generates the "nitl9_file" file	2	

In the check_system.sh script, more tests have to be performed manually to check the /etc/hosts and /in/local/conf/db/listener.ora files.

The MPI documents describing the tests are also available in the current zip file.

The following table gives the scripts called by **check_system.sh** script:

Script Name (alphabetical order)	Tests Performed	Executed by script or manually (1)
capa.sh	- Subsystem Attributes (T01-A06) - Disk usage (T05-E02) - Total Physical Memory (T06-F01) - Swap Info (T06-F02) - Swap Mode (T06-F03)	script script script script script
crontab.sh	- Crontab file configuration (T01-A02)	script
date.sh	- Check of date and time zone (T01-A03)	script
dynamic.sh	- Dynamic Process information (T07-G01) - Dynamic memory, disk, network, swap information (T07-G02)	manuallly manuallly
fichier.sh	- Run Time configuration variable (T01-A07) - Hosts file (T03-C01) - Interface configuration (T03-C02) - Static route table (T03-C03) - Routing Table (T03-C04)	script manually manually manually manually
firmware.sh	- System Firmware console environment variables (T01-A01) - Information about Operating System (T01-A05) - Patch Kit reference (T02-B03) - Dna_base subsystem attributes (T04-D01) - Dna_dli subsystem attributes (T04-D02)	script script script script script
instal.sh	- Check the installed subsets (T02-B01) - License Checking (T02-B02) - CAM equipment device table (T05-E01) - Advfs domain structure (T05-C03) - Mount File System table (T04-E04) - LSM volumes and Plexes states (T05-E05) - Disk state (T05-E06)	script script script script script script script
level004.sh	- Check of the level 004	script
level027.sh	- Check of the level 027	script
level-adden115.sh	- Check of the level and addendum 115	script
servers.sh	- Hostname checking (T01-A04) - Synchronisation (T01-A04) - Ping commands (T03-C05)	script script script
stack.sh	- Check of the stack version (ex: 31JP10 or 31JP7)	script

(1): Indeed, the check_system.sh may display messages to specify that the test has to be performed manually.

The following table gives the scripts called by **check_middleware.sh** script:

Script Name (alphabetical order)	Tests Performed	Executed by script or manually (2)
additional.sh	- Platform and addendum level (T06-F01) - Monitoring alarm with comprov user (T07-G01)	script manually
connexion.sh	- Decnet connexion numbers (T01-A01) - Load sharing on dual ethernet (T01-A02) - Node addresses (T01-A03) - Dlogin Commands (T01-A04) - Platform processes (T06-F02) - BEP platform configuration (T06-F04)	script script script manually script manually
general.sh	- User linus rhosts file (T02-B02) - Harware agent output file (T07-G04) - pfman log file (T07-G05)	script manually manually
oracle.sh	- Netwatcher (T02-B01) - SMP and BEP links for ORACLE (T03-C01) - ORACLE listener configuration file (T03-C02) - Reachability of the SMP ORACLE database (T03-C03) - ORACLE utilities version (T03-C04)	manually script manually manually script
smp.sh	- ORBIX (T04-D01) - SMP platform configuration (T06-F03) - Ticket Buffering (T07-G01) - Dynamic update process alarm (T07-G03) - LMS open problem (T07-G06)	script manually script manually manually
ss7.sh	- User SS7 rhosts file (T02-B03) - SS7 Objects configuration file (T05-E01) - SS7 Release version (T05-E02) - E1 firmware version (T05-E03) - Process states (T05-E04) - SS7 links (T05-E05) - SS7 platform configuration (T05-E06) - SS7 rules (T05-E07)	script script script script script manually manually manually

(2): Indeed, the check_middleware.sh may display messages to specify that the test has to be performed manually.

A continuación voy a exponer el código de los 3 scripts principales y de algunas de las subrutinas o scripts de verificación, no el de todas ya que el texto se extendería demasiado:

II. Códigos de los programas de verificación del sistema

II.1 Código de check_system.sh :

```
#! /bin/sh
clear
echo " CHECK LIST SYSTEM---CHECK LIST SYSTEM---CHECK LIST SYSTEM "
echo " CHECK LIST SYSTEM---CHECK LIST SYSTEM---CHECK LIST SYSTEM" >
system_verif
echo ""
echo ""
inic=0
while [ $inic = "0" ]
do
      echo "Enter the server type where the script is run: smp, step, scp,
bep or fep"
      read machine
      if [ $machine = "smp" ] || [ $machine = "step" ] || [ $machine =
"scp" ] || [ $machine = "bep" ] || [ $machine = "fep" ]
      then
            inic=1
      else
            echo ""
            echo " The  entered server is not correct "
            echo " Try it again"
            echo ""
      fi
done
echo ""
yn=0
while [ $yn -eq 0 ]
do
      echo " Are you an installator : y/n (yes for an installator and not
for a RTM)?"
      read reponse
      if [ $reponse = "y" ] || [ $reponse = "yes" ]
      then
            touch /in/tmp/check/installator
            yn=1
      else
            if [ $reponse = "n" ] || [ $reponse = "not" ]
            then
                  if [ `ls -l /in/tmp/check |grep -c installator` -eq 1 ]
                  then
                        rm /in/tmp/check/installator
                  fi
                  yn=1
            else
                  echo ""
                  echo " The entered answer is not correct "
                  echo " Enter it again "

                  echo ""
            fi
```

```
      fi

done
if [ `ls -l |grep -c installator` -eq 1 ]
then
      pass="N"

      while [ $pass = "N" ]
      do
            echo ""
            echo "Choose the option corresponding to the test to perform: "
            echo ""
            echo "            0- Exit "
            echo "            1- Full tests"
            echo "            2- Firmware, Platform Version, patch kit, DNA
subsystem"
            echo "            3- Hostname, Synchronisation, Ping"
            echo "            4- Subsystem attributes, Disk usage, Total
physical memory, Swap info, Swap mode"
            echo "            5- Installed subsets, Licenses checking, Advfs
structure, Mount file system, LSM volumes and Plexes, Disk state"
            echo "            6- Files /etc/hosts, /etc/routes,
/etc/rc.config, Interface configuration, Routing table"
            echo "            7- Date and Time zone"
            echo "            8- Crontab file configuration"
            echo "            9- Check level 004"
            echo "            10-Check level 027"
            echo "            11-Check level and addendum 115"
            echo "            12-Dynamic process, memory, disk, network,
swap information"
            echo "            13-Check stack"
            echo ""
            echo ""
            echo "DON'T FORGET TO CHECK THE RESULT FILES: "
            echo "            -system_verif"
            echo "            -Licenses"
            echo "            -filesystem"
            echo ""
            echo ""
            read choix
            if [ $choix -eq 0 ]
            then
                  echo "Exit ... "
                  echo ""
                  pass="Y"
            fi
            if [ $choix -eq 1 ]
            then
                  echo "" >> system_verif
                  echo "CHECK LIST SYSTEM--CHECK LIST SYSTEM--CHECK LIST
SYSTEM" >system_verif
                  echo ""
                  echo "-----------------------------------"
                  echo "-- firmware.sh script eExecution --"
                  echo "" >> system_verif
                  echo "" >> system_verif
                  echo "-- firmware.sh script execution --" >>system_verif
                  echo "" >> system_verif

                  echo "-----------------------------------"
```

```
echo ""
sh firmware.sh $machine
echo ""

echo "---------------------------------"
echo "-- servers.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- servers.sh script execution --" >>system_verif

echo "" >> system_verif
echo "---------------------------------"
echo ""
sh servers.sh
echo ""
echo "-----------------------------"
echo "-- capa.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- capa.sh script execution --" >>system_verif
echo "" >> system_verif
echo "-----------------------------"
echo ""
sh capa.sh $machine
echo ""
echo "---------------------------------"
echo "-- instal.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- instal.sh script execution --" >>system_verif
echo "" >> system_verif
echo "---------------------------------"
echo ""
sh instal.sh $machine
echo ""
echo "---------------------------------"
echo "-- fichier.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- fichier.sh script execution --" >>system_verif
echo "" >> system_verif
echo "---------------------------------"
echo ""
sh fichier.sh $machine
echo ""
echo "-----------------------------"
echo "-- date.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- date.sh script execution --" >>system_verif
echo "" >> system_verif
echo "-----------------------------"
echo ""
sh date.sh
echo ""
echo "---------------------------------"
echo "-- crontab.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
```

```
        echo "-- crontab.sh script execution --" >>system_verif
        echo "" >> system_verif
        echo "---------------------------------"
        echo ""
        sh crontab.sh

        echo ""
        echo "---------------------------------"
        echo "-- dynamic.sh script execution --"
        echo "" >> system_verif
        echo "" >> system_verif
        echo "-- dynamic.sh script execution --" >>system_verif

        echo "" >> system_verif
        echo "---------------------------------"
        echo ""
        sh dynamic.sh
        echo ""
        echo "---------------------------------"
        echo "-- stack.sh script execution --"
        echo "" >> system_verif
        echo "" >> system_verif
        echo "-- stack.sh script execution --" >>system_verif
        echo "" >> system_verif
        echo "---------------------------------"
        echo ""
        sh stack.sh $machine
        echo ""
        echo "" >> system_verif
        pass="N"
fi
if [ $choix -eq 2 ]
then
        sh firmware.sh $machine
        echo ""
        pass="N"
fi
if [ $choix -eq 3 ]
then
        sh servers.sh
        echo ""
        pass="N"
fi
if [ $choix -eq 4 ]
then
        sh capa.sh $machine
        echo ""
        pass="N"
fi
if [ $choix -eq 5 ]
then
        sh instal.sh $machine
        echo ""
        pass="N"
fi
if [ $choix -eq 6 ]
then
        sh fichier.sh $machine
        echo ""
        pass="N"
```

```
            fi
            if [ $choix -eq 7 ]
            then
                  sh date.sh
                  echo ""
                  pass="N"

            fi
            if [ $choix -eq 8 ]
            then
                  sh crontab.sh
                  echo ""
                  pass="N"
            fi

            if [ $choix -eq 9 ]
            then
                  sh level004.sh $machine
                  echo ""
                  pass="N"
            fi
            if [ $choix -eq 10 ]
            then
                  sh level027.sh $machine
                  echo ""
                  pass="N"
            fi
            if [ $choix -eq 11 ]
            then
                  sh level-adden115.sh $machine
                  echo ""
                  pass="N"
            fi
            if [ $choix -eq 12 ]
            then
                  sh dynamic.sh
                  echo ""
                  pass="N"
            fi
            if [ $choix -eq 13 ]
            then
                  sh stack.sh $machine
                  echo ""
                  pass="N"
            fi
      done
else
      echo ""
      echo "-----------------------------------"
      echo "-- firmware.sh script execution --"
      echo "" >> system_verif
      echo "" >> system_verif
      echo "-- firmware.sh script execution --" >>system_verif
      echo "" >> system_verif
      echo "-----------------------------------"
      echo ""sh firmware.sh $machine
      echo ""
      echo "-----------------------------------"
      echo "-- servers.sh script execution --"
      echo "" >> system_verif
```

```
echo "" >> system_verif
echo "-- servers.sh script execution --" >>system_verif
echo "" >> system_verif
echo "--------------------------------"
echo ""
sh servers.sh
echo ""

echo "------------------------------"
echo "" >> system_verif
echo "" >> system_verif
echo "-- capa.sh script execution --"
echo "-- capa.sh script execution --" >>system_verif
echo "" >> system_verif
echo "------------------------------"
echo ""

sh capa.sh $machine
echo ""
echo "--------------------------------"
echo "-- instal.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- instal.sh script execution --" >>system_verif
echo "" >> system_verif
echo "--------------------------------"
echo ""
sh instal.sh $machine
echo ""
echo "--------------------------------"
echo "-- fichier.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- fichier.sh script execution --" >>system_verif
echo "" >> system_verif
echo "--------------------------------"
echo ""
sh fichier.sh $machine
echo ""
echo "------------------------------"
echo "-- date.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- date.sh script execution --" >>system_verif
echo "" >> system_verif
echo "------------------------------"
echo ""
sh date.sh
echo ""
echo "--------------------------------"
echo "-- crontab.sh script execution --"
echo "" >> system_verif
echo "" >> system_verif
echo "-- crontab.sh script execution --" >>system_verif
echo "" >> system_verif
echo "--------------------------------"
echo ""
echo "" >> system_verif
sh crontab.sh
echo "" >> system_verif
```

```
      sh dynamic.sh
      echo ""
      echo "---------------------------------"
      echo "-- stack.sh script execution --"
      echo "" >> system_verif
      echo "" >> system_verif
      echo "-- stack.sh script execution --" >>system_verif
      echo "" >> system_verif

      echo "---------------------------------"
      echo ""
      sh stack.sh $machine
      echo ""
      echo "DON'T FORGET TO CHECK THE RESULT FILES: "
      echo "            -system_verif"
      echo "            -Licenses"
      echo "            -filesystem"
      echo ""

fi
if [ `ls -l /in/tmp/check |grep -c installator` -eq 1 ]
then
      rm /in/tmp/check/installator
fi
```

II.2 Código de la subrutina firmware.sh:

```
#! /bin/sh
echo ""
echo ""
echo " This part is composed of several tests "
echo ""
echo ""
echo "" >> system_verif
echo "" >> system_verif
echo "    T01-A01 : FIRMWARE VERSION "
echo ""
echo "    T01-A01 : FIRMWARE VERSION " >> system_verif
echo "" >> system_verif
if [ `ls -l |grep -c installator` -eq 1 ]
then
      echo " Enter the firmware revision number (example: 6.2):  "
      read revfirm
      echo " The  entered firmware revision number is $revfirm " >>
      system_verif
      firmlub=`consvar -v -l | grep Firmware | cut -c 15-17`
else
      revfirm=`cat conf_file | grep "Version du Firmware" | cut -d " " -f7`
      echo " Firmware given : $revfirm " >> system_verif
      firmlub=`consvar -v -l | grep Firmware | cut -c 15-17`
fi
if [ $firmlub"A" = $revfirm"A" ]
then
      echo ""
      echo "" >> system_verif
      echo " OK: Firmware check "
      echo " OK: Firmware check " >> system_verif
else
```

```
      echo " NOK Firmware: because the current firmware release is
      $firmlub"   >> system_verif
      echo " NOK Firmware: because the current firmware release is
      $firmlub"
fi
echo ""
echo "" >> system_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else

            volonte="n"

      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then

            echo ""
            echo ""
            echo "" >> system_verif
            echo "    T01-A05 : PLATFORM VERSION "
            echo ""
            echo "    T01-A05 : PLATFORM VERSION " >> system_verif
            echo "" >> system_verif
            recom="Y"
            while [ $recom = "Y" ]
            do
                  if [ `ls -l |grep -c installator` -eq 1 ]
                  then
                        echo " Enter the platform version: 2.2.05 or 2.2.03
                        "
                        read verplat
                  else
                        verplat=`cat conf_file | grep plate-forme | cut -d
                        " " -f8`
                  fi
                  echo " Entered version: $verplat" >> system_verif
                  if [ $verplat = "2.2.05" ]
                  then
                        verun="1229"
                        recom="N"
                  else
                        if [ $verplat = "2.2.03" ]
                        then
                              verun="878"
                              recom="N"
                        else
                              echo " Enter again: "
                              recom="Y"
                        fi
                  fi
            done
            if [ `uname -a | grep $verun | wc -l ` =  "1" ]
            then
                  echo ""
```

```
                  echo "" >> system_verif
                  echo " OK: The Unix version is compatible with the
                  current platform "
                  echo " OK: The Unix version is compatible with the
                  current platform " >> system_verif
            else
                  echo ""
                  echo "" >> system_verif
                  echo " NOK: Incorrect platform version "
                  echo " NOK: Incorrect platform version " >> system_verif
            fi
            passe="Y"
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then

                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi

done
echo ""
echo "" >> system_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> system_verif
            echo "    T02-B03 : PATCH KIT REFERENCE "
            echo ""
            echo "    T02-B03 : PATCH KIT REFERENCE " >> system_verif
            echo "" >> system_verif

            #     Installed Patches Verification
            echo "  Patches installed on the system ... "
            echo "  Patches installed on the system ... " >> system_verif
            echo "" >> system_verif
            echo ""
            if [ $1 = "smp" ] || [ $1 = "step" ]
            then
                  nok=0
                  echo "    BL13 Patch Kit "
                  echo "    BL13 Patch Kit " >> system_verif
                  echo "" >> system_verif
                  echo ""
                  dupatch -track -type kit > temp1
```

```
if [ `cat temp1 | grep -c "DUV40FAS0002-19991116 OSF440"`
-eq 1 ]
then
      echo " OK: The patch DUV40FAS0002-19991116 OSF440
      is installed "
      echo " OK: The patch DUV40FAS0002-19991116 OSF440
      is installed " >> system_verif
else
      echo " NOK: The patch DUV40FAS0002-19991116 OSF440
      is not installed "
      echo " NOK: The patch DUV40FAS0002-19991116 OSF440
      is not installed " >> system_verif
      nok=1
fi

if [ `cat temp1 | grep -c "DUV40FAS0002-19991116 TCR160"`
-eq 1 ]

then
      echo " OK: The patch DUV40FAS0002-19991116 TCR160
      is installed "
      echo " OK: The patch DUV40FAS0002-19991116 TCR160
      is installed " >> system_verif
else

      echo " NOK: The patch DUV40FAS0002-19991116 TCR160
      is not installed "
      echo " NOK: The patch DUV40FAS0002-19991116 TCR160
      is not installed " >> system_verif
      nok=1
fi

echo ""
echo "" >> system_verif
echo "    BL18 Patch Kit "
echo "    BL18 Patch Kit " >> system_verif
echo "" >> system_verif
echo ""
lst=" DUV40FB18-C0074800-14099-20020423 OSF440"$lst

if [ `cat temp1 | grep -c "DUV40FB18-C0074203-14212-E-
20020503 OSF440"` -eq 1 ]
then
      echo " OK: The patch DUV40FB18-C0074203-14212-E-
      20020503 OSF440 is installed "
      echo " OK: The patch DUV40FB18-C0074203-14212-E-
      20020503 OSF440 is installed " >> system_verif
else
      echo " NOK: The patch DUV40FB18-C0074203-14212-E-
      20020503 OSF440 is not installed "
      echo " NOK: The patch DUV40FB18-C0074203-14212-E-
      20020503 OSF440 is not installed " >> system_verif
      nok=1
fi
if [ `cat temp1 | grep -c "DUV40FB18AS0007-20020102
OSF440"` -eq 1 ]
then
      echo " OK: The patch DUV40FB18AS0007-20020102
      OSF440 is installed "
```

```
      echo " OK: The patch DUV40FB18AS0007-20020102
      OSF440 is installed " >> system_verif
else
      echo " NOK: The patch DUV40FB18AS0007-20020102
      OSF440 is not installed "
      echo " NOK: The patch DUV40FB18AS0007-20020102
      OSF440 is not installed " >> system_verif
      nok=1
fi
if [ `cat temp1 | grep -c "DUV40FB18AS0007-20020102
TCR160"` -eq 1 ]
then
      echo " OK: The patch DUV40FB18AS0007-20020102
      TCR160 is installed "

      echo " OK: The patch DUV40FB18AS0007-20020102
      TCR160 " >> system_verif
else

      echo " NOK: The patch DUV40FB18AS0007-20020102
      TCR160 is not installed "
      echo " NOK: The patch DUV40FB18AS0007-20020102
      TCR160 is not installed " >> system_verif
      nok=1
fi
if [ `cat temp1 | grep -c "DUV40FB18-C0070700-13635-
20020313 OSF440"` -eq 1 ]
then

      echo " OK: The patch DUV40FB18-C0070700-13635-
      20020313 OSF440 is installed "
      echo " OK: The patch DUV40FB18-C0070700-13635-
      20020313 OSF440 is installed " >> system_verif
else
      echo " NOK: The patch DUV40FB18-C0070700-13635-
      20020313 OSF440 is not installed "
      echo " NOK: The patch DUV40FB18-C0070700-13635-
      20020313 OSF440 is not installed " >> system_verif
      nok=1
fi
if [ `cat temp1 | grep -c "DUV40FB18-C0090100-16323-
20021206 OSF440"` -eq 1 ]
then
      echo " OK: The patch DUV40FB18-C0090100-16323-
      20021206 OSF440 is installed "
      echo " OK: The patch DUV40FB18-C0090100-16323-
      20021206 OSF440 is installed " >> system_verif
else
      echo " NOK: The patch DUV40FB18-C0090100-16323-
      20021206 OSF440 is not installed "
      echo " NOK: The patch DUV40FB18-C0090100-16323-
      20021206 OSF440 is not installed " >> system_verif
      nok=1
fi
if [ `cat temp1 | grep -c "DUV40FB18-C0074800-14099-
20020423 OSF440"` -eq 1 ]
then
      echo " OK: The patch DUV40FB18-C0074800-14099-
      20020423 OSF440 is installed "
```

```
                echo " OK: The patch DUV40FB18-C0074800-14099-
                20020423 OSF440 is installed " >> system_verif
        else
                echo " NOK: The patch DUV40FB18-C0074800-14099-
                20020423 OSF440 is not installed "
                echo " NOK: The patch DUV40FB18-C0074800-14099-
                20020423 OSF440 is not installed " >> system_verif
                nok=1

        fi
else
        nok=0
        echo "    BL13 Patch Kit "
        echo "    BL13 Patch Kit " >> system_verif
        echo "" >> system_verif
        echo ""
        dupatch -track -type kit > temp1
        if [ `cat temp1 | grep -c "DUV40FAS0002-19991116 OSF440"`
        -eq 1 ]
        then

                echo " OK: The patch DUV40FAS0002-19991116 OSF440
                is installed "
                echo " OK: The patch DUV40FAS0002-19991116 OSF440
                is installed " >> system_verif
        else
                echo " NOK: The patch DUV40FAS0002-19991116 OSF440
                is not installed "
                echo " NOK: The patch DUV40FAS0002-19991116 OSF440
                is not installed " >> system_verif
                nok=1

        fi
        echo ""
        echo "" >> system_verif
        echo "    BL18 Patch Kit "
        echo "    BL18 Patch Kit " >> system_verif
        echo "" >> system_verif
        echo ""
        lst=" DUV40FB18-C0074800-14099-20020423 OSF440"$lst

        if [ `cat temp1 | grep -c "DUV40FB18-C0074203-14212-E-
        20020503 OSF440"` -eq 1 ]
        then
                echo " OK: The patch DUV40FB18-C0074203-14212-E-
                20020503 OSF440 is installed "
                echo " OK: The patch DUV40FB18-C0074203-14212-E-
                20020503 OSF440 is installed " >> system_verif
        else
                echo " NOK: The patch DUV40FB18-C0074203-14212-E-
                20020503 OSF440 is not installed "
                echo " NOK: The patch DUV40FB18-C0074203-14212-E-
                20020503 OSF440 is not installed " >> system_verif
                nok=1
        fi
        if [ `cat temp1 | grep -c "DUV40FB18AS0007-20020102
        OSF440"` -eq 1 ]
        then
                echo " OK: The patch DUV40FB18AS0007-20020102
                OSF440 is installed "
```

```
                        echo " OK: The patch DUV40FB18AS0007-20020102
                        OSF440 is installed " >> system_verif
                  else
                        echo " NOK: The patch DUV40FB18AS0007-20020102
                        OSF440 is not installed "
                        echo " NOK: The patch DUV40FB18AS0007-20020102
                        OSF440 is not installed " >> system_verif
                        nok=1
                  fi
                  if [ `cat temp1 | grep -c "DUV40FB18-C0070700-13635-
                  20020313 OSF440"` -eq 1 ]
                  then
                        echo " OK: The patch DUV40FB18-C0070700-13635-
                        20020313 OSF440 is installed "
                        echo " OK: The patch DUV40FB18-C0070700-13635-
                        20020313 OSF440 is installed " >> system_verif
                  else
                        echo " NOK: The patch DUV40FB18-C0070700-13635-
                        20020313 OSF440 is not installed "
                        echo " NOK: The patch DUV40FB18-C0070700-13635-
                        20020313 OSF440 is not installed " >> system_verif
                        nok=1

                  fi
            fi
            if [ $nok -eq 1 ]
            then
                  echo ""
                  echo "" >> system_verif
                  echo " NOK: Check the installed patch kit in the file
                  system_verif "
                  echo " NOK: The installed patch kit is (to be checked) :
                  " >> system_verif

                  dupatch -track -type kit >> system_verif
            fi
            echo ""
            echo "" >> system_verif
            passe="Y"
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

if [ $1 = "bep" ] || [ $1 = "fep" ] || [ $1 = "scp" ] || [ $1 = "step" ]
then
      echo ""
      echo "" >> system_verif
      passe="N"
      while [ $passe = "N" ]
      do
            if [ `ls -l |grep -c installator` -eq 1 ]
```

```
then
      echo " Do you want to perform the next check : y/n "

      read volonte
else
      volonte="n"
fi
if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]

then
      echo ""
      echo ""
      echo "    T04-D01 et D02 : DNA SUBSYSTEM ATTRIBUTES "
      echo ""
      echo "    T04-D01 et D02 : DNA SUBSYSTEM ATTRIBUTES " >>
      system_verif
      echo "" >> system_verif

      #variables a modifier si necessaire
      dat_dna_base="Feb 16 09:59"
      chaine=`sysconfig -q dna_base | grep eco-level | cut -d
      "(" -f2 | cut -c 1-12`
      if [ `echo $chaine|sum|cut -d " " -f1` -eq `echo
      $dat_dna_base|sum|cut -d " " -f1` ]
      then

            echo " OK: the dna_base.mod  date is correct "
            echo " OK: the dna_base.mod  date is correct " >>
            system_verif
      else
            echo " NOK: the dna_base.mod  date is not correct "
            echo " NOK: the dna_base.mod  date is not correct "
            >> system_verif
      fi

      #variables a modifier si necessaire
      dat_dna_dli="Nov 18 14:16"

      chaine=`sysconfig -q dna_dli | grep eco-level | cut -d
      "(" -f2 | cut -c 1-12`
      if [ `echo $chaine|sum|cut -d " " -f1` -eq `echo
      $dat_dna_dli|sum|cut -d " " -f1` ]
      then
            echo " OK: the dna_dli.mod date is correct "
            echo " OK: the dna_dli.mod date is correct " >>
            system_verif
      else
            echo " NOK: the dna_dli.mod date is not correct "
            echo " NOK: the dna_dli.mod date is not correct "
            >> system_verif
      fi
      passe="Y"
else
      if [ $volonte = "n" ]  && [ `ls -l |grep -c installator`
      -eq 1 ]
      then
            echo ""
            echo " It is mandatory to perform all checks ! "
            echo ""
            passe="N"
```

```
                fi
            fi
      done

else
      echo ""
      echo "" >> system_verif
      echo " OK: Verification T04-D01 only on bep and fep"
      echo " OK: Verification T04-D01 only on bep and fep" >> system_verif
      fi
```

II.3 Código del script: fichier.sh :

```
#! /bin/sh
echo ""
echo ""
echo " This part is composed of several tests "
echo ""
echo ""
echo "" >> system_verif
echo "" >> system_verif
echo "    T01-A07 : RUN TIME CONFIGURATION VARIABLE"
echo ""
echo "    T01-A07 : RUN TIME CONFIGURATION VARIABLE" >> system_verif
echo "" >> system_verif

if [ `ls -l |grep -c installator` -eq 1 ]
then
      echo " How many LANs are used ? "
      read lan
else
      lan=`cat conf_file | grep  "Number of lan" | cut -d " " -f5`
fi
res1=`cat /etc/rc.config | grep NUM_NETCONFIG= | tail -c3 | head -c1`
if [ $lan"A" = $res1"A" ]
then
      echo ""
      echo " OK: NUM_NETCONFIG=$lan "
      echo " OK: NUM_NETCONFIG=$lan " >> system_verif
else
      echo ""
      echo "" >> system_verif
      echo " NOK: NUM_NETCONFIG=$res1 in the file /etc/rc.config "
      echo " NOK: NUM_NETCONFIG=$res1 in the file /etc/rc.config " >>
      system_verif
      if [ $1 = "smp" ] || [ $1 = "step" ]
      then
            echo ""
            echo "" >> system_verif
            echo " If the smp is ES40, then NUM_NETCONFIG can be 4 (the E1
            board ports number) "
            echo " If the smp is ES40, then NUM_NETCONFIG can be 4 (the E1
            board ports number) " >> system_verif
      fi
fi
res2=`cat /etc/rc.config | grep IFCONFIG | grep speed| wc -l`
if [ $lan"A" = $res2"A" ]
then
      echo ""
```

```
      echo "" >> system_verif
      echo " OK: $lan IFCONFIG are declared "
      echo " OK: $lan IFCONFIG are declared " >> system_verif
      echo ""
else
      echo ""
      echo "" >> system_verif
      echo " NOK: $res2 IFCONFIG are declared in the file /etc/rc.config "
      echo " NOK: $res2 IFCONFIG are declared in the file /etc/rc.config "
      >> system_verif
      echo ""
fi
if [ $1 = "smp" ] || [ $1 = "step" ]
then
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo ""
            echo " Enter the SMP1 hostname:"
            read name1
            echo ""
            echo " Enter the SMP2 hostname:"
            read name2
      else
            name1=`cat conf_file | grep "Hostname of the SMP1"| cut -d" " -
            f6`
            name2=`cat conf_file | grep "Hostname of the SMP2"| cut -d" " -
            f6`
fi
host=`hostname`
res3=`cat /etc/rc.config | grep XNTP_SERV | grep $name1 | wc -l`
res4=`cat /etc/rc.config | grep XNTP_SERV | grep $name2 | wc -l`
if [ $name1 = $host ] && [ $res3 -gt 0 ]
then
      echo ""
      echo "" >> system_verif
      echo " NOK: error detected on XNTP_SERV in the file /etc/rc.config "
      echo " NOK: error detected on XNTP_SERV in the file /etc/rc.config "
      >> system_verif
else
      echo ""
      echo "" >> system_verif
      echo " OK: no error on XNTP_SERV "
      echo " OK: no error on XNTP_SERV " >> system_verif
fi
if [ $name2 = $host ] && [ $res4 -gt 0 ]
then
      echo ""
      echo "" >> system_verif
      echo " NOK: error detected on XNTP_SERV in the file /etc/rc.config "
      echo " NOK: error detected on XNTP_SERV in the file /etc/rc.config "
      >> system_verif
else
      echo ""
      echo "" >> system_verif
      echo " OK: no error on XNTP_SERV "
      echo " OK: no error on XNTP_SERV " >> system_verif
fi
fi
if [ $1 = "bep" ] || [ $1 = "fep" ] || [ $1 = "scp" ]
then
```

```
        if [ `ls -l |grep -c installator` -eq 1 ]
        then
                echo ""
                echo "Enter the SMP1 hostname:"
                read name1
                echo ""
                echo "Enter the SMP2 hostname:"
                echo ""
                read name2
        else
                name1=`cat conf_file | grep "Hostname of the SMP1"| cut -d" " -
                f6`
                name2=`cat conf_file | grep "Hostname of the SMP2"| cut -d" " -
                f6`
        fi
        res3=`cat /etc/rc.config | grep XNTP_SERV | grep $name1 | wc -l`
        res4=`cat /etc/rc.config | grep XNTP_SERV | grep $name2 | wc -l`
        if [ $res3 -eq 0 ]
        then
                echo ""
                echo "" >> system_verif
                echo " NOK: error detected on XNTP_SERV in the file
                /etc/rc.config "
                echo " NOK: error detected on XNTP_SERV in the file
                /etc/rc.config " >> system_verif
        else
                echo ""
                echo "" >> system_verif
                echo " OK: no error on XNTP_SERV "
                echo " OK: no error on XNTP_SERV " >> system_verif
        fi
        if [ $res4 -eq 0 ]
        then
                echo ""
                echo " NOK: error detected on XNTP_SERV in the file
                /etc/rc.config "
                echo " NOK: error detected on XNTP_SERV in the file
                /etc/rc.config " >> system_verif
                echo ""
        else
                echo ""
                echo "" >> system_verif
                echo " OK: no error on XNTP_SERV "
                echo " OK: no error on XNTP_SERV " >> system_verif
        fi
fi

if [ $1 = "smp" ] || [ $1 = "step" ]
then
        res=`cat /etc/rc.config | grep XNTPD_OPTS | grep -v export | cut -d=
        -f2`
        if [ $res -eq "-g" ]
        then
                echo ""
                echo "" >> system_verif
                echo " OK: the -g option is present in the file /etc/rc.config
                "
                echo " OK: the -g option is present in the file /etc/rc.config
                " >> system_verif
        else
```

```
            echo ""
            echo "" >> system_verif
            echo " NOK: the -g option not found in the file /etc/rc.config
            "
            echo " NOK: the -g option not found in the file /etc/rc.config
            " >> system_verif
      fi
fi

if [ $1 = "scp" ] || [ $1 = "bep" ] || [ $1 = "fep" ]
then
      res=`cat /etc/rc.config | grep XNTPD_OPTS | grep -v export | cut -d=
      -f2`
      if [ $res -eq "-g" ]
      then
            echo ""
            echo "" >> system_verif
            echo " OK: the -g option is present in the file /etc/rc.config
            "
            echo " OK: the -g option is present in the file /etc/rc.config
            " >> system_verif
      else
            echo ""
            echo "" >> system_verif
            echo " NOK: the -g option not found in the file /etc/rc.config
            "
            echo " NOK: the -g option not found in the file /etc/rc.config
            " >> system_verif
      fi
fi
echo ""
echo "" >> system_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo "" >> system_verif
            echo ""
            echo "    T03-C01 : HOSTS FILE "
            echo ""
            echo "    T03-C01 : HOSTS FILE " >> system_verif
            echo "" >> system_verif
            echo " The etc/hosts file is copied in the filesystem file (To
            be checked) "
            echo " The etc/hosts file is copied in the filesystem file (To
            be checked)">>system_verif
            echo "" > filesystem
            echo " TO3-C01 : FILE /etc/hosts " >> filesystem
            echo "" >> filesystem
            cat /etc/hosts >> filesystem
            echo ""
```

```
            echo ""
            sleep 2
            if [ `ls -l |grep -c installator` -eq 1 ]
            then
                  echo " Hosts file "
                  echo ""
                  more /etc/hosts
            fi
            passe="Y"
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
echo ""
echo "" >> system_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> system_verif
            echo "    T03-C02 : INTERFACE CONFIGURATION "
            echo ""
            echo "    T03-C02 : INTERFACE CONFIGURATION " >> system_verif
            echo "" >> system_verif
            echo " Check to be performed manually: enter the following
            command directly on the server \" ifconfig -a \" "
            echo "" >> system_verif
            echo " Check to be performed manually: enter the following
            command directly on the server \" ifconfig -a \" " >>
            system_verif
            sleep 2
            passe="Y"
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
```

```
echo ""
echo "" >> system_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> system_verif
            echo "    T03-C03 : ROUTES FILE "
            echo ""
            echo "    T03-C03 : ROUTES FILE " >> system_verif
            echo "" >> system_verif
            echo " The /etc/routes file is copied in the filesystem file
            (To be checked)"
            echo " The /etc/routes file is copied in the filesystem file
            (To be checked)" >> system_verif
            echo "" >> filesystem
            echo "" >> filesystem
            echo " TO3-C03 : FILE /etc/routes " >> filesystem
            echo "" >> filesystem
            cat /etc/routes >> filesystem
            echo ""
            echo ""
            sleep 2
            if [ `ls -l |grep -c installator` -eq 1 ]
            then
                  echo " Routes file "
                  echo ""
                  more /etc/routes
            fi
            passe="Y"
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
echo ""
echo "" >> system_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
```

```
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> system_verif
            echo "    T03-C04 : ROUTING TABLE "
            echo ""
            echo "    T03-C04 : ROUTING TABLE " >> system_verif
            echo "" >> system_verif
            echo " The Routing table is copied in the filesystem file (To
            be checked) "
            echo " The Routing table is copied in the filesystem file (To
            be checked) ">>system_verif
            echo ""
            echo "" >> filesystem
            echo "" >> filesystem
            echo " TO3-C04 : ROUTING TABLE " >> filesystem
            echo "" >> filesystem
            echo ""
            echo ""
            sleep 2
            if [ `ls -l |grep -c installator` -eq 1 ]
            then
                  echo " Routing table "
                  echo ""
                  netstat -nr
            fi
            netstat -nr >> filesystem
            passe="Y"
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
```

III. Códigos de los programas de verificación del middleware

III.1 Código de check_middleware.sh:

```
#! /bin/sh
clear
echo "CHECK-MIDDLEWARE--CHECK_MIDDLEWARE--CHECK_MIDDLEWARE"
echo " CHECK LIST MIDDLEWARE---CHECK LIST MIDDLEWARE---CHECK LIST
MIDDLEWARE" > middle_verif
echo ""
echo ""
echo "" >> middle_verif
echo "" >> middle_verif
```

```
inic=0
while [ $inic = "0" ]
do
      echo "Enter the server type where the script is run: smp, step, scp,
      bep or fep"
      read machine
      if [ $machine = "smp" ] || [ $machine = "step" ] || [ $machine =
      "scp" ] || [ $machine = "bep" ] || [ $machine = "fep" ]
      then
            inic=1
      else
            echo ""
            echo " The  entered server is not correct "
            echo " Try it again"
            echo ""
      fi
done
echo ""
yn=0
while [ $yn -eq 0 ]
do
      echo " Are you an installator : y/n (yes for an installator and not
      for a RTM)?"
      read reponse
      if [ $reponse = "y" ] || [ $reponse = "yes" ]
      then
            touch /in/tmp/check/installator
            yn=1
      else
            if [ $reponse = "n" ] || [ $reponse = "not" ]
            then
                  if [ `ls -l /in/tmp/check |grep -c installator` -eq 1 ]
                  then
                        rm /in/tmp/check/installator
                  fi
                  yn=1
            else
                  echo ""
                  echo " The entered answer is not correct "
                  echo " Enter it again "
                  echo ""
            fi
      fi
done

if [ `ls -l /in/tmp/check |grep -c installator` -eq 1 ]
then
      pass="N"
      while [ $pass = "N" ]
      do
            echo ""
            echo "Choose the option corresponding to the test to perform:"
            echo ""
            echo "            0-Exit"
            echo "            1-Oracle: links, listener file, utilities
            version, reachability of oracle database, netwatcher (Not
            performed on FEP)"
            echo "            2-SS7: .rhosts file, release version, objects,
            links, rules, platform configuration (Not on SMP)"
```

```
echo "            3-Decnet, Load sharing, Node address, Dlogin,
Platform process, bep platform configuration (Not on SMP)"
echo "            4-Orbix, Smp platform configuration, Tickets,
Dynamic update process, LSM open problem (SMP or STEP)"
echo "            5-Linus .rhosts file, Hardware agent file,
Pfman log file"
echo "            6-Platform and Addendum level, Monitoring
alarm"
echo "            7-Full tests"
echo ""
echo ""
echo "DON'T FORGET TO CHECK THE RESULT FILES ! : "
echo "            -middle_verif"
echo "            -filesystem"
echo "            -ss7_verif"
echo ""
echo ""
read choix

if [ $choix -eq 0 ]
then
      echo ""
      echo " Exit "
      echo ""
      pass="Y"
fi

if [ $choix -eq 6 ]
then
      echo "" >> middle_verif
      sh additional.sh $machine
      echo "" >> middle_verif
      echo ""
      pass="N"
fi

if [ $choix -eq 7 ]
then
      echo "" >> middle_verif
      echo "CHECK-MIDDLEWARE--CHECK_MIDDLEWARE--
      CHECK_MIDDLEWARE"
      echo " CHECK MIDDLEWARE---CHECK  MIDDLEWARE---CHECK
      MIDDLEWARE" > middle_verif
      echo ""
      echo "-----------------------------------"
      echo "-- oracle.sh script execution --"
      echo "" >> middle_verif
      echo "" >> middle_verif
      echo "-- oracle.sh script execution --" >>middle_verif
      echo "" >> middle_verif
      echo "-----------------------------------"
      echo ""
      sh oracle.sh $machine
      echo ""
      echo "-----------------------------------"
      echo "-- ss7.sh script execution --"
      echo "" >> middle_verif
      echo "" >> middle_verif
      echo "-- ss7.sh script execution --" >> middle_verif
      echo "" >> middle_verif
```

```
        echo "---------------------------------"
        echo ""
        sh ss7.sh $machine
        echo ""
        echo "-------------------------------"
        echo "-- connexion.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- connexion.sh script execution --" >>
        middle_verif
        echo "" >> middle_verif
        echo "-------------------------------"
        echo ""
        sh connexion.sh $machine
        echo ""
        echo "---------------------------------"
        echo "-- smp.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- smp.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "---------------------------------"
        echo ""
        sh smp.sh $machine
        echo ""
        echo "-- general.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- general.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "---------------------------------"
        echo ""
        sh general.sh
        echo ""
        echo "-------------------------------"
        echo "-- additional.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- additional.sh script execution --" >>
        middle_verif
        echo "" >> middle_verif
        echo "-------------------------------"
        echo ""
        echo "" >> middle_verif
        sh additional.sh $machine
        pass="N"
fi

if [ $choix -eq 1 ]
then
        echo "" >> middle_verif
        sh oracle.sh $machine
        echo "" >> middle_verif
        echo ""
        pass="N"
fi

if [ $choix -eq 2 ]
then
        echo "" >> middle_verif
```

```
                sh ss7.sh $machine
                echo "" >> middle_verif
                echo ""
                pass="N"
        fi

        if [ $choix -eq 3 ]
        then
                echo "" >> middle_verif
                sh connexion.sh $machine
                echo "" >> middle_verif
                echo ""
                pass="N"
        fi

        if [ $choix -eq 4 ]
        then
                echo "" >> middle_verif
                sh smp.sh $machine
                echo "" >> middle_verif
                echo ""
                pass="N"
        fi

        if [ $choix -eq 5 ]
        then
                echo "" >> middle_verif
                sh general.sh
                echo "" >> middle_verif
                echo ""
                pass="N"
        fi

done

else

        echo "" >> middle_verif
        echo "CHECK-MIDDLEWARE--CHECK_MIDDLEWARE--CHECK_MIDDLEWARE"
        echo " CHECK MIDDLEWARE---CHECK  MIDDLEWARE---CHECK MIDDLEWARE" >
        middle_verif
        echo ""
        echo "----------------------------------"
        echo "-- oracle.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- oracle.sh script execution --" >>middle_verif
        echo "" >> middle_verif
        echo "----------------------------------"
        echo ""
        sh oracle.sh $machine
        echo ""
        echo "----------------------------------"
        echo "-- ss7.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- ss7.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "----------------------------------"
        echo ""
```

```
        sh ss7.sh $machine
        echo ""
        echo "------------------------------"
        echo "-- connexion.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- connexion.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "------------------------------"
        echo ""
        sh connexion.sh $machine
        echo ""
        echo "--------------------------------"
        echo "-- smp.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- smp.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "--------------------------------"
        echo ""
        sh smp.sh $machine
        echo ""
        echo "---------------------------------"
        echo "-- general.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- general.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "---------------------------------"
        echo ""
        sh general.sh
        echo ""
        echo "------------------------------"
        echo "-- additional.sh script execution --"
        echo "" >> middle_verif
        echo "" >> middle_verif
        echo "-- additional.sh script execution --" >> middle_verif
        echo "" >> middle_verif
        echo "------------------------------"
        echo ""
        echo "" >> middle_verif
        sh additional.sh $machine
        echo "" >> middle_verif
        echo ""
        echo "DON'T FORGET TO CHECK THE RESULT FILES ! : "
        echo "            -middle_verif"
        echo "            -filesystem"
        echo "            -ss7_verif"
        echo ""
fi

if [ `ls -l /in/tmp/check |grep -c installator` -eq 1 ]
then
        rm /in/tmp/check/installator
fi
```

III.2 Código de la subrutina ss7.sh:

```
#! /bin/sh

echo ""
echo ""
echo " This part is composed of several tests "
echo ""
echo ""
echo "" >> middle_verif
echo "" >> middle_verif
echo "    T02-B03 : USER SS7 RHOSTS FILE "
echo "    T02-B03 : USER SS7 RHOSTS FILE " >> middle_verif
echo ""
echo "" >> middle_verif
if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "bep" ] || [ $1 = "fep" ]
then
      res=`su - ss7 -c "cat .rhosts|wc -l" | tail -1`
      res1=`su - ss7 -c "cat .rhosts|grep ss7| wc -l" | tail -1`
      su - ss7 -c "cat .rhosts" >> middle_verif
      if [ $res = "1" ] && [ $res1 = "1" ]
      then
            echo " OK: .rhosts file is correctly configured under ss7 "
            echo " OK: .rhosts file is correctly configured under ss7 " >>
            middle_verif
      else
            echo " NOK: .rhosts file is not correctly configured under ss7
            "
            echo " NOK: .rhosts file is not correctly configured under ss7
            " >> middle_verif
      fi
else
      echo " This check is performed only on scp, step, bep and fep "
      echo " This check is performed only on scp, step, bep and fep " >>
      middle_verif
fi
echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> middle_verif
            echo "    T05-E01 : SS7 OBJECTS CONFIGURATIONS FILE "
            echo "    T05-E01 : SS7 OBJECTS CONFIGURATIONS FILE " >>
            middle_verif
            echo ""
            echo "" >> middle_verif
            echo " Performed after N7 installation "
            echo " Performed after N7 installation " >> middle_verif
```

```
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> middle_verif
            echo "    T05-E02 : SS7 RELEASE VERSION "
            echo "    T05-E02 : SS7 RELEASE VERSION " >> middle_verif
            echo ""
            echo "" >> middle_verif
            if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "bep" ] || [ $1
            = "fep" ]
            then
                  if [ `ls -l |grep -c installator` -eq 1 ]
                  then
                        echo " Enter the stack version (311 or 31J) : "
                        read stv
                  else
                        stv=`cat conf_file |grep Stack | cut -d " " -f7`
                  fi
                  if [ $stv = "31J" ]
                  then
                        if [ `cat
                        /usr/var/ss7/platform_a/config/ss7_parameters |grep
                        -c "SS7_CS_POOL_SIZE=22000; export
                        SS7_CS_POOL_SIZE"` -eq 1 ]
                        then
                              echo ""
                              echo "" >> middle_verif
                              echo " OK: We are on 31J "
                              echo " OK: We are on 31J " >> middle_verif
                        else
                              echo ""
                              echo "" >> middle_verif
                              echo " NOK: The current Stack version is 31J
                              but the line "SS7_CS_POOL_SIZE=22000; export
```

```
                              SS7_CS_POOL_SIZE" is not found in the file
                              /usr/var/ss7/platform_a/config/ss7_parameters
                              "
                              echo " NOK: The current Stack version is 31J
                              but the line "SS7_CS_POOL_SIZE=22000; export
                              SS7_CS_POOL_SIZE" is not found in the file
                              /usr/var/ss7/platform_a/config/ss7_parameters
                              " >> middle_verif
                        fi
                  fi
                  ss7version=`cat /usr/opt/SS7*$stv/rel*/*.txt | head -30 |
                  tail -10 | grep P | cut -c 19-22`
                  if [ `ls -l |grep -c installator` -eq 1 ]
                  then
                        echo ""
                        echo " Enter the SS7 version: for example P731 or
                        P103 or P..."
                        read version
                  else
                        version=`cat conf_file |grep SS7 | cut -d: -f2`
                  fi
                  echo " Current version: $version" >> middle_verif
                  if [ $ss7version = $version ]
                  then
                        echo ""
                        echo "" >> middle_verif
                        echo " OK: The SS7 version is correct, $ss7version
                        found in the file "
                        echo " OK: The SS7 version is correct, $ss7version
                        found in the file " >> middle_verif
                  else
                        echo ""
                        echo " NOK: The SS7 version is not correct,
                        $ss7version found in the file "
                        echo " NOK: The SS7 version is not correct,
                        $ss7version found in the file " >> middle_verif
                  fi
            else
                  echo " This check is performed only on scp, step, bep and
                  fep "
                  echo " This check is performed only on scp, step, bep and
                  fep " >> middle_verif
            fi
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

echo ""
echo "" >> middle_verif
passe="N"
```

```
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "fep" ]
            then
                  echo ""
                  echo ""
                  echo "" >> middle_verif
                  echo "    T05-E03 : E1 FIRMWARE VERSION "
                  echo "    T05-E03 : E1 FIRMWARE VERSION " >> middle_verif
                  echo ""
                  echo ""
                  echo "" >> middle_verif
                  firmvers=`ss7show_pti -v ptpci0 |cut -c 19-22`
                  # Calcul of E1 firmware version
                  if [ $firmvers = "3145" ]
                  then
                        e1vers=4.5
                  elif [ $firmvers = "314F" ]
                  then
                        e1vers=4.F
                  elif [ $firmvers = "3150" ]
                  then
                        e1vers=5.0
                  elif [ $firmvers = "3153" ]
                  then
                        e1vers=5.3
                  else
                        echo " The found version number $firmvers is not
                        known"
                        echo " The found version number $firmvers is not
                        known" >> middle_verif
                  fi
                  recom="Y"
                  while [ $recom = "Y" ]
                  do
                        if [ `ls -l |grep -c installator` -eq 1 ]
                        then
                              echo " Enter the firmware version : example P7
                              "
                              read firma
                        else
                              firma=`cat conf_file | grep "Firmware version"
                              | cut -d: -f2`
                        fi
                        if [ $firma = "P14" ]
                        then
                              firmb="3145"
                              recom="N"
                        else
                              if [ $firma = "P15" ]
                              then
```

```
                        firmb="314F"
                        recom="N"
                  else
                        if [ $firma = "P7" ]
                        then
                              firmb="3150"
                              recom="N"
                        else
                              if [ $firma = "P10" ]
                              then
                                    firmb="3153"
                                    recom="N"
                              else
                                    if [ `ls -l |grep -c
                                    installator` -eq 1 ]
                                    then
                                          echo " Value not
                                          correct - Enter again:
                                          "
                                    else
                                          echo " Value not
                                          correct : restart"
                                    fi
                              fi
                        fi
                  fi
            fi
      done
      echo ""
      echo "" >> middle_verif
      echo " The found(current) version number is equal to
      $firmvers and corresponds to the E1 Firmware version =
      $e1vers "
      echo " The found(current) version number is equal to
      $firmvers and corresponds to the E1 Firmware version =
      $e1vers " >> middle_verif
      if [ $firmvers = $firmb ]
      then
            echo ""
            echo "" >> middle_verif
            echo " OK: The found version number $firmvers
            corresponding to the current firmware
            version($firma) is correct." >> middle_verif
            echo " OK: The found version number $firmvers
            corresponding to the current firmware
            version($firma) is correct."
      else
            echo ""
            echo " NOK: The entered(and desired) firmware
            version is $firma but the found(current) version
            number is $firmvers (3150 for P7 and 3153 for P10)"
            >> middle_verif
            echo " NOK: The entered(and desired) firmware
            version is $firma but the found(current) version
            number is $firmvers (3150 for P7 and 3153 for P10)"
      fi
fi
passe="Y"
sleep 2
```

```
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> middle_verif
            echo "    T05-E04 : PROCESS STATES "
            echo "    T05-E04 : PROCESS STATES  " >> middle_verif
            echo ""
            echo "" >> middle_verif
            echo " There is no process if SS7 is not configured "
            echo " There is no process if SS7 is not configured " >>
            middle_verif
            echo ""
            echo "" >> middle_verif
            echo " Check all processes are OK except if SS7 is not
            configured "
            echo " Check all processes are OK except if SS7 is not
            configured" >> middle_verif
            echo ""
            echo "" >> middle_verif
            if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "bep" ] || [ $1
            = "fep" ]
            then
                  su - ss7 -c "ss7man -l"
                  su - ss7 -c "ss7man -l" >> middle_verif
            else
                  echo " This check is performed only on scp, step, bep and
                  fep "
                  echo " This check is performed only on scp, step, bep and
                  fep " >> middle_verif
            fi
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
```

```
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n"
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo "" >> middle_verif
            echo ""
            echo "    T05-E05 : SS7 LINKS "
            echo "    T05-E05 : SS7 LINKS  " >> middle_verif
            echo ""
            echo "" >> middle_verif
            if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "bep" ]
            then
                  echo " If ss7 is configured, check the data_links are
                  IN_SERVICE "
                  echo " If ss7 is configured, check the data_links are
                  IN_SERVICE " >>middle_verif
                  echo ""
                  echo "" >> middle_verif
                  su - ss7 -c "ss7man -d"
                  su - ss7 -c "ss7man -d" >> middle_verif
                  echo ""
                  echo "" >> middle_verif
                  ndata=`su - ss7 -c "ss7man -d" | grep -c Data`
                  nin_ser=`su - ss7 -c "ss7man -d" | grep Data | grep -c
                  IN_SERVICE`
                  if [ $ndata = "0" ]
                  then
                        echo " There are no data_links because SS7 is not
                        configured  "
                        echo " There are no data_links because SS7 is not
                        configured  ">>middle_verif
                  else
                        if [ "A"$ndata = "A"$nin_ser ]
                        then
                              echo " OK: All the Data Links are running " >>
                              middle_verif
                              echo " OK: All the Data Links are running "
                        else
                              echo " NOK : There are some Data Links which
                              are not no running  "
```

```
                              echo " NOK : There are some Data Links which
                              are not running " >> middle_verif
                        fi
                  fi
            else
                  echo " This check is performed only on scp, step and bep
                  "
                  echo " This check is performed only on scp, step and bep
                  " >> middle_verif
            fi
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
echo ""
echo "" >>middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi

      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> middle_verif
            echo "    T05-E06 : SS7 PLATFORM CONFIGURATION "
            echo "    T05-E06 : SS7 PLATFORM CONFIGURATION  " >>
            middle_verif
            echo ""
            echo "" >> middle_verif
            if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "bep" ]
            then
                  echo " Check the applications list and the declaration of
                  each node by typing the line: "ss7showconfig a" (log as
                  ss7) "
                  echo " Check the applications list and the declaration of
                  each node by typing the line: "ss7showconfig a" (log as
                  ss7) " >> middle_verif
                  echo ""
                  sleep 1
                  echo "" >> middle_verif
                  echo ""
                  echo ""
                  echo "" >> middle_verif
```

```
                  echo " Check that the only one declared platform is a "
                  if [ `ls -l |grep -c installator` -eq 0 ]
                  then
                        echo " Check that the only one declared platform is
                        "a" with the order: "ss7configure" (log as ss7) "
                        >> middle_verif
                        su - ss7 -c "ss7configure" << EOF

                        EOF
                  else
                        echo " Check that the only one declared platform is
                        "a": " >> middle_verif
                        su - ss7 -c "ss7configure"
                        echo "" >> middle_verif
                        su - ss7 -c "ss7configure" >> middle_verif
                  fi
            else
                  echo " This check is performed only on scp, step and bep
                  "
                  echo " This check is performed only on scp, step and bep
                  " >> middle_verif
            fi
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo ""
            echo "" >> middle_verif
            echo "    T05-E07 : SS7 RULES "
            echo "    T05-E07 : SS7 RULES  " >> middle_verif
            echo ""
            echo "" >> middle_verif
            if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "fep" ]
            then
```

```
        if [ `ls -l /usr/var/ss7/platform_a/config | grep
        ss7_configure_fep_a | wc -l` -eq 1 ]
        then
              echo " The ss7_configure_fep_a file is present and
              is copied in the filesystem file (To be checked) "
              echo " The ss7_configure_fep_a file is present and
              is copied in the filesystem file (To be checked) "
              >> middle_verif
              echo "" >> filesystem
              echo "" >> filesystem
              echo "    T05-E07 : SS7 RULES  " >> filesystem
              echo "" >> filesystem
              echo " Check the ss7 configuration and the
              declaration of each node " >> filesystem
              echo "" >> filesystem
              echo " FILE:
              /usr/var/ss7/platform_a/config/ss7_configure_fep_a
              " >> filesystem
              echo "" >> filesystem
              su - ss7 -c " cat
              /usr/var/ss7/platform_a/config/ss7_configure_fep_a
              " >> filesystem
        else
              echo " The
              /usr/var/ss7/platform_a/config/ss7_configure_fep_a
              file is not present"
              echo " The
              /usr/var/ss7/platform_a/config/ss7_configure_fep_a
              file is not present" >> middle_verif
        fi
else
        echo " This check is performed only on scp, step and fep
        "
        echo " This check is performed only on scp, step and fep
        " >> middle_verif
fi
if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "bep" ]
then
        echo ""
        echo "" >> middle_verif
        if [ `ls -l /usr/var/ss7/platform_a/config | grep
        ss7_configure_bep_a | wc -l` -eq 1 ]
        then
              echo " The ss7_configure_bep_a file is present and
              is copied in the filesystem file (To be checked) "
              echo " The ss7_configure_bep_a file is present and
              is copied in the filesystem file (To be checked) "
              >>middle_verif
              echo "" >> filesystem
              echo " FILE:
              /usr/var/ss7/platform_a/config/ss7_configure_bep_a
              " >> filesystem
              echo "" >> filesystem
              echo "" >> filesystem
              su - ss7 -c " cat
              /usr/var/ss7/platform_a/config/ss7_configure_bep_a
              " >> filesystem
              echo "" >> filesystem
        else
```

```
                        echo " The
                        /usr/var/ss7/platform_a/config/ss7_configure_bep_a
                        file is not present "
                        echo " The
                        /usr/var/ss7/platform_a/config/ss7_configure_bep_a
                        file is not present " >> middle_verif
                        echo ""
                        echo ""
                  fi
            else
                  echo " This check is performed only on scp, step and bep
                  "
                  echo " This check is performed only on scp, step and bep
                  " >> middle_verif
            fi
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "fep" ] || [ $1 = "bep" ]
then
passe="N"
      while [ $passe = "N" ]
      do
            if [ `ls -l |grep -c installator` -eq 1 ]
            then
                  echo ""
                  echo ""
                  echo " Do you want to perform the next check : y/n "
                  read volonte
            else
                  volonte="n"
            fi
            if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
            then
                  echo ""
                  echo ""
                  echo "" >> middle_verif
                  echo "" >> middle_verif
                  echo "    TEST: COCs test "
                  echo "    TEST: COCs test " >> middle_verif
                  echo ""
                  echo "" >> middle_verif
                  if [ `ls -l |grep -c installator` -eq 1 ]
                  then
                        echo " Enter the number of COC's on each FEP: "
                        read ncoc
                  fi
                  server=`hostname`
                  echo $server > /in/tmp/check/temporary
```

```
serverLC=`tr '[a-z]' '[A-Z]' < /in/tmp/check/temporary`
ncocrl=`su - ss7 -c "ss7man -s linkset | grep -c
fep=$serverLC"`
statec=`su - ss7 -c "ss7man -s linkset | grep -c
SS7_K_LIN_AVAILABLE"`
nfeps=`su - ss7 -c "ss7man -s linkset | grep -c fep="`
echo ""
echo " The COCs list is: "
echo " The COCs list is: " >> middle_verif
echo "" >> middle_verif
echo ""
su - ss7 -c "ss7man -s linkset"
su - ss7 -c "ss7man -s linkset" >> middle_verif
if [ `ls -l |grep -c installator` -eq 1 ]
then
      if [ "a"$ncoc = "a"$ncocrl ]
      then
            echo ""
            echo "" >> middle_verif
            echo " OK: the number of COCs on each FEP is
            correct "
            echo " OK: the number of COCs on each FEP is
            correct " >> middle_verif
      else
            echo ""
            echo "" >> middle_verif
            echo " NOK: the entered number of COCs is not
            correct, the real value is: $ncocrl "
            echo " NOK: the entered number of COCs is not
            correct, the real value is: $ncocrl " >>
            middle_verif
      fi
else
      echo ""
      echo "" >> middle_verif
      echo " The number of COCs on each FEP is: $ncocrl "
      echo " The number of COCs on each FEP is: $ncocrl "
      >> middle_verif
fi
if [ $nfeps -eq 0 ]
then
      echo ""
      echo "" >> middle_verif
      echo " There are not any COC on the FEP "
      echo " There are not any COC on the FEP " >>
      middle_verif
else
      if [ "a"$nfeps = "a"$statec ]
      then
            echo ""
            echo "" >> middle_verif
            echo " OK: All the COCs are in correct state "
            echo " OK: All the COCs are in correct state "
            >> middle_verif
      else
            echo ""
            echo "" >> middle_verif
            echo " NOK: There are some COCs which are not
            AVAILABLE "
```

```
                              echo " NOK: There are some COCs which are not
                              AVAILABLE " >> middle_verif
                        fi
                  fi
                  passe="Y"
                  sleep 2
            else
                  if [ $volonte = "n" ]  && [ `ls -l |grep -c installator`
-eq 1 ]
                  then
                        echo ""
                        echo " It is mandatory to perform all checks ! "
                        echo ""
                        passe="N"
                  fi
            fi
      done
fi

if [ $1 = "scp" ] || [ $1 = "step" ] || [ $1 = "fep" ] || [ $1 = "bep" ]
then
      passe="N"
      while [ $passe = "N" ]
      do
            if [ `ls -l |grep -c installator` -eq 1 ]
            then
                  echo ""
                  echo ""
                  echo " Do you want to perform the next check : y/n "
                  read volonte
            else
                  volonte="n"
            fi
            if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
            then
                  echo ""
                  echo ""
                  echo "" > ss7_verif
                  echo "" >> ss7_verif
                  echo "    TEST: The ss7man orders "
                  echo "    TEST: The ss7man orders " >> ss7_verif
                  echo ""
                  echo "" >> ss7_verif
                  echo ""
                  echo "   RESULT OF THE ORDER: ss7man -sa: "
                  echo "   RESULT OF THE ORDER: ss7man -sa: " >> ss7_verif
                  echo ""
                  echo "" >> ss7_verif
                  echo ""
                  echo "" >> ss7_verif
                  su - ss7 -c "ss7man -sa"
                  su - ss7 -c "ss7man -sa" >> ss7_verif
                  echo ""
                  echo "" >> ss7_verif
                  echo ""
                  echo "" >> ss7_verif
                  echo "   RESULT OF THE ORDER: ss7man -slinkset: "
                  echo "   RESULT OF THE ORDER: ss7man -slinkset: " >>
                  ss7_verif
                  echo ""
```

```
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
su - ss7 -c "ss7man -slinkset"
su - ss7 -c "ss7man -slinkset" >> ss7_verif

echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo "   RESULT OF THE ORDER: ss7man -strunk: "
echo "   RESULT OF THE ORDER: ss7man -strunk: " >>
ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
su - ss7 -c "ss7man -strunk"
su - ss7 -c "ss7man -strunk" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo "   RESULT OF THE ORDER: ss7man -sdatalink: "
echo "   RESULT OF THE ORDER: ss7man -sdatalink: " >>
ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
su - ss7 -c "ss7man -sdatalink"
su - ss7 -c "ss7man -sdatalink" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo "   RESULT OF THE ORDER: ss7man -sroute: "
echo "   RESULT OF THE ORDER: ss7man -sroute: " >>
ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
su - ss7 -c "ss7man -sroute"
su - ss7 -c "ss7man -sroute" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
echo "   RESULT OF THE ORDER: ss7man -sn7dest: "
echo "   RESULT OF THE ORDER: ss7man -sn7dest: " >>
ss7_verif
echo ""
echo "" >> ss7_verif
echo ""
echo "" >> ss7_verif
su - ss7 -c "ss7man -sn7dest"
su - ss7 -c "ss7man -sn7dest" >> ss7_verif
echo ""
echo "" >> ss7_verif
```

```
                  echo ""
                  echo "" >> ss7_verif
                  echo "   DATA OF THE FILE
                  /usr/var/ss7/ccitt/ss7_rules_ccitt.dat: "
                  echo "   DATA OF THE FILE
                  /usr/var/ss7/ccitt/ss7_rules_ccitt.dat: " >> ss7_verif
                  echo ""
                  echo "" >> ss7_verif
                  cat /usr/var/ss7/ccitt/ss7_rules_ccitt.dat | more
                  cat /usr/var/ss7/ccitt/ss7_rules_ccitt.dat >> ss7_verif

                  passe="Y"
                  sleep 2
            else
                  if [ $volonte = "n" ]  && [ `ls -l |grep -c installator`
                  -eq 1 ]
                  then
                        echo ""
                        echo " It is mandatory to perform all checks ! "
                        echo ""
                        passe="N"
                  fi
            fi
      done
else
      echo "" >> ss7_verif
      echo " This file is empty because the current server is smp " >>
      ss7_verif
fi
echo ""
```

III.3 Código del script: general.sh

```
#! /bin/sh

echo ""
echo ""
echo " This part is composed of several tests "
echo ""
echo ""
echo "" >> middle_verif
echo "" >> middle_verif
echo "    T02-B02 : USER LINUS RHOSTS FILE "
echo "    T02-B02 : USER LINUS RHOSTS FILE " >> middle_verif
echo ""
echo "" >> middle_verif
su - linus -c "cat .rhosts" >> middle_verif
res=`su - linus -c "cat .rhosts | wc -l" | tail -1`
res1=`su - linus -c "cat .rhosts | grep linus | wc -l" | tail -1`
res2=`su - linus -c "cat .rhosts | grep comprov | wc -l" | tail -1`
if [ $res = "2" ] && [ $res1 = "1" ] && [ $res2 = "1" ]
then
      echo ""
      echo "" >> middle_verif
      echo " OK: the .rhosts file is correctly configured under linus "
      echo " OK: the .rhosts file is correctly configured under linus " >>
      middle_verif
else
      echo ""
```

```
      echo "" >> middle_verif
      echo " NOK: the .rhosts file is not correctly configured under linus
      "
      echo " NOK: the .rhosts file is not correctly configured under linus
      " >> middle_verif
fi
echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo "" >> middle_verif
            echo ""
            echo "    Verifications T07-G04 : HARDWARE AGENT OUTPUT FILE "
            echo "    Verifications T07-G04 : HARDWARE AGENT OUTPUT FILE "
            >> middle_verif
            echo ""
            echo "" >> middle_verif
            echo " Check there are no alarms in the hwagent.out file (the
            content of the hwagent.log is copied in the result file) "
            echo " Check there are no alarms in the hwagent.out file " >>
            middle_verif
            echo ""
            if [ `ls -l |grep -c installator` -eq 1 ]
            then
                  echo " FILE: /in/local/var/log/hwagent.out "
                  more /in/local/var/log/hwagent.out
            fi
            echo "" >> middle_verif
            echo " FILE: /in/local/var/log/hwagent.out " >> middle_verif
            echo "" >> middle_verif
            cat /in/local/var/log/hwagent.out >> middle_verif
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done

echo ""
echo "" >> middle_verif
passe="N"
while [ $passe = "N" ]
```

```
do
      if [ `ls -l |grep -c installator` -eq 1 ]
      then
            echo " Do you want to perform the next check : y/n "
            read volonte
      else
            volonte="n"
      fi
      if [ $volonte = "y" ] || [ `ls -l |grep -c installator` -eq 0 ]
      then
            echo ""
            echo "" >> middle_verif
            echo ""
            echo "    Verifications T07-G05 : PFMAN LOG FILE "
            echo "    Verifications T07-G05 : PFMAN LOG FILE " >>
            middle_verif
            echo ""
            echo "" >> middle_verif
            echo " Check there are neither alarms nor warnings in the
            pfman.log file (the content of the pfman.log is copied in the
            result file) "
            echo " Check there are neither alarms nor warnings in the
            pfman.log file " >> middle_verif
            echo ""
            echo "" >> middle_verif
            echo " FILE: /in/local/var/log/pfman.log " >> middle_verif
            echo "" >> middle_verif
            cat /in/local/var/log/pfman.log >> middle_verif
            passe="Y"
            sleep 2
      else
            if [ $volonte = "n" ]  && [ `ls -l |grep -c installator` -eq 1
            ]
            then
                  echo ""
                  echo " It is mandatory to perform all checks ! "
                  echo ""
                  passe="N"
            fi
      fi
done
```

IV. Código del script principal nitl9.sh:

```
#! /bin/sh
clear
echo ""
echo "" >> nitl9_file
echo "NITL 2.2.05-009 -- NITL 2.2.05-009 -- NITL 2.2.05-009 -- NITL 2.2.05-
009"
echo "NITL 2.2.05-009 -- NITL 2.2.05-009 -- NITL 2.2.05-009 -- NITL 2.2.05-
009" > nitl9_file
echo ""
echo "Enter the server type where the script is run: smp, step, scp, bep or
fep"
echo ""
read machine
if [ `ls -l | grep -c rem_file.sh` -eq 1 ]
then
```

```
      echo "" > rem_file.sh
fi
echo ""
echo "" >> nitl9_file
echo "    SENDMAIL"
echo "    SENDMAIL" >> nitl9_file
echo ""
echo "" >> nitl9_file
/sbin/init.d/sendmail stop
/sbin/init.d/sendmail select old
echo ""
/sbin/init.d/sendmail start
echo ""
chown root /usr/sbin/sendmail.old
chown root /usr/sbin/sendmail
chmod a-rw /usr/sbin/sendmail.old
chmod a-rw /usr/sbin/sendmail
echo " Check the owner is root and the access rights are a-rw "
echo " for the sendmail and sendmail.old files"
echo ""
echo " Check the owner is root and the access rights are a-rw " >>
nitl9_file
echo " for the sendmail and sendmail.old files" >> nitl9_file
echo "" >> nitl9_file
ls -l /usr/sbin | grep sendmail
ls -l /usr/sbin | grep sendmail >> nitl9_file
echo ""
echo "" >> nitl9_file
sleep 2
echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            if [ $machine = "bep" ] || [ $machine = "fep" ] || [ $machine =
            "step" ] || [ $machine = "scp" ]
            then
                  echo ""
                  echo "    TEST: /in/local/etc/bep_backup and
                  /in/local/etc/fep_backup FILES "
                  echo ""
                  echo "    TEST: /in/local/etc/bep_backup and
                  /in/local/etc/fep_backup FILES " >> nitl9_file
                  echo "" >> nitl9_file
                  if [ $machine = "bep" ] || [ $machine = "step" ] || [
                  $machine = "scp" ]
                  then
                        if [ `cat  /in/local/etc/bep_backup | grep -c
                        /in/local/bin` -eq 0 ]
                        then
                              sed "s/gtar/\/in\/local\/bin\/gtar/g"
                              /in/local/etc/bep_backup >
                              /in/tmp/check/bep_backup.tmp
                              mv /in/tmp/check/bep_backup.tmp
                              /in/local/etc/bep_backup
```

```
                        echo " The file /in/local/etc/bep_backup has
                        been modified "
                        echo " by adding: /in/local/bin/ before gtar.
                        Check it in the result file: nitl9_file "
                        echo " The file /in/local/etc/bep_backup has
                        been modified " >> nitl9_file
                        echo " by adding: /in/local/bin/ before gtar.
                        Check it: " >> nitl9_file
                        more /in/local/etc/bep_backup >> nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                        echo ""
                  else
                        echo " OK: The file /in/local/etc/bep_backup
                        is correct "
                        echo " OK: The file /in/local/etc/bep_backup
                        is correct " >> nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                        echo ""
                  fi
            fi
            if [ $machine = "fep" ] || [ $machine = "step" ] || [
            $machine = "scp" ]
            then
                  if [ `cat  /in/local/etc/fep_backup | grep -c
                  /in/local/bin` -eq 0 ]
                  then
                        sed "s/gtar/\/in\/local\/bin\/gtar/g"
                        /in/local/etc/fep_backup >
                        /in/local/etc/fep_backup.tmp
                        mv /in/local/etc/fep_backup.tmp
                        /in/local/etc/fep_backup
                        echo " The file /in/local/etc/fep_backup has
                        been modified "
                        echo " by adding: /in/local/bin/ before gtar.
                        Check it in the result file: nitl9_file "
                        echo " The file /in/local/etc/fep_backup has
                        been modified " >> nitl9_file
                        echo " by adding: /in/local/bin/ before gtar.
                        Check it: " >> nitl9_file
                        echo "" >> nitl9_file
                        more /in/local/etc/fep_backup >> nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                        echo ""
                  else
                        echo " OK: The file /in/local/etc/fep_backup
                        is correct "
                        echo " OK: The file /in/local/etc/fep_backup
                        is correct " >> nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                        echo ""
                  fi
            fi
      else
            echo ""
            echo " This check is performed only on step,scp, bep and
            fep "
```

```
                echo ""
            fi
            passe="Y"
            sleep 2
      else
            echo " It is mandatory to perform all checks !"
            passe="N"
      fi
done

echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            if [ $machine = "bep" ] || [ $machine = "fep" ] || [ $machine =
            "step" ] || [ $machine = "scp" ]
            then
            echo ""
            echo "    SS7_FEP_CCITT.EXE FILE "
            echo "    SS7_FEP_CCITT.EXE FILE " >> nitl9_file
            echo ""
            echo "" >> nitl9_file
            echo " Enter the stack version (311 or 31J) : "
            read stv
            if [ $stv = "311" ]
            then
                  chmod 511 /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe
                  chgrp adm /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe
                  chown ss7 /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe
                  echo " Check the access rights of the file:
                  /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe  are 511, the
                  owner is ss7 and the group is adm "
                  echo " Check the access rights of the file:
                  /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe are 511, the
                  owner is ss7 and the group is adm " >> nitl9_file
                  echo ""
                  echo "" >> nitl9_file
                  ls -l /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe
                  ls -l /usr/opt/SS7CCITT311/bin/ss7_fep_ccitt.exe >>
                  nitl9_file
                  echo ""
                  echo "" >> nitl9_file
            else
                  if [ $stv = "31J" ]
                  then
                        chmod 511
                        /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe
                        chgrp adm
                        /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe
                        chown ss7
                        /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe
                        echo " Check the access rights of the file:
                        /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe are 511,
                        the owner is ss7 and the group is adm "
```

```
                        echo " Check the access rights of the file:
                        /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe are 511,
                        the owner is ss7 and the group is adm " >>
                        nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                        ls -l /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe
                        ls -l /usr/opt/SS7CCITT31J/bin/ss7_fep_ccitt.exe >>
                        nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                  fi
                  fi
            else
                  echo ""
                  echo " This check is performed only on step,scp, bep and
                  fep "
                  echo ""
            fi
            passe="Y"
            sleep 2
      else
            echo " It is mandatory to perform all checks !"
            passe="N"
      fi
done

echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            echo ""
            echo "    CHECK DISK USAGE"
            echo ""
            echo "    CHECK DISK USAGE" >> nitl9_file
            echo "" >> nitl9_file
            echo ""
            echo " Check that / <80%, /proc =100%, /usr <80%, /oracle <75%,
            /in <5%, /indelivery <50%, /in/smp <5% "
            echo ""
            echo ""
            df -k
            sleep 5
            echo ""
            df -k >> nitl9_file
            echo "" >> nitl9_file
            echo ""
            echo "Are these values correct : y/n "
            echo ""
            read reponse
            if [ $reponse = "y" ]
            then
                  echo " You assert that the values are correct: no file
                  will be deleted "
                  echo " OK: The values are correct " >> nitl9_file
```

```
        echo "" >> nitl9_file
else
        echo " NOK: As the values are not correct, some files
        will be selected to be deleted, but after agreement of
        the RTM "
        echo ""
        if [ $machine = "bep" ] || [ $machine = "fep" ] || [
        $machine = "scp" ] || [ $machine = "step" ]
        then
                echo " Files of platforms Y and Z in
                /usr/local/tmp "
                echo " Files of platforms Y and Z in
                /usr/local/tmp " >> nitl9_file
                echo " echo \" Files of platforms Y and Z in
                /usr/local/tmp \" " >> rem_file.sh
                echo ""
                echo "" >> nitl9_file
                echo "echo \"\" " >> rem_file.sh
                if [ `ls -l /usr/local/tmp|head -1|cut -d " "
                -f2` = "0" ]
                then
                        echo " This directory is empty, no files
                        to delete "
                        echo " This directory is empty, no files
                        to delete " >> nitl9_file
                else
                        echo " The files to delete are listed in
                        the nitl9_file file, request
                        confirmation from the RTM "
                        echo ""
                        echo " The files to delete are listed in
                        the nitl9_file file, request
                        confirmation from the RTM " >>
                        nitl9_file
                        echo "" >> nitl9_file
                        echo "/usr/local/tmp/SS7\$BEP_GSEC_Y" >>
                        nitl9_file
                        echo "/usr/local/tmp/SS7\$BEP_GSEC_Z" >>
                        nitl9_file
                        echo "/usr/local/tmp/SS7\$BEP_EFN_Y" >>
                        nitl9_file
                        echo "/usr/local/tmp/SS7\$BEP_EFN_Z" >>
                        nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GS_RES_SMPCP2_S_Y"
                        >> nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GS_RES_SMPCP2_S_Z"
                        >> nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GS_RES_SMPCP2_Y" >>
                        nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GS_RES_SMPCP2_Z" >>
                        nitl9_file
                        echo "/usr/local/tmp/SS7_GTT_STB_GS_Y"
                        >> nitl9_file
                        echo "/usr/local/tmp/SS7_GTT_STB_GS_Z"
                        >> nitl9_file
```

```
                        echo
                        "/usr/local/tmp/SS7_GTT_STB_GS_Y_BEP" >>
                        nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GTT_STB_GS_Z_BEP" >>
                        nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GTT_STB_GS_Y_FEP" >>
                        nitl9_file
                        echo
                        "/usr/local/tmp/SS7_GTT_STB_GS_Z_FEP" >>
                        nitl9_file
                        echo "" >> nitl9_file
                        echo "rm /usr/local/tmp/SS7\$BEP_GSEC_Y"
                        > rem_file.sh
                        echo "rm /usr/local/tmp/SS7\$BEP_GSEC_Z"
                        >> rem_file.sh
                        echo "rm /usr/local/tmp/SS7\$BEP_EFN_Y"
                        >> rem_file.sh
                        echo "rm /usr/local/tmp/SS7\$BEP_EFN_Z"
                        >> rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GS_RES_SMPCP2_S_Y" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GS_RES_SMPCP2_S_Z" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GS_RES_SMPCP2_Y" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GS_RES_SMPCP2_Z" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GTT_STB_GS_Y" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GTT_STB_GS_Z" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GTT_STB_GS_Y_BEP" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GTT_STB_GS_Z_BEP" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GTT_STB_GS_Y_FEP" >>
                        rem_file.sh
                        echo "rm
                        /usr/local/tmp/SS7_GTT_STB_GS_Z_FEP" >>
                        rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                fi
        fi
        echo ""
        echo " Deletion of core files"
        echo " Deletion of core files " >> nitl9_file
        echo "echo \" Deletion of core files \" " >> rem_file.sh
        echo "echo \"\" " >> rem_file.sh
        echo ""
```

```
echo "" >> nitl9_file
if [ `find / -name core | wc -l` = "0" ]
then
            echo " There are no core files"
            echo " There are no core files " >> nitl9_file
else
            echo "find / -name core" >> rem_file.sh
            echo " echo \"These files must be deleted only
            after RTM agreement \" " >> rem_file.sh
            echo " echo \"\" " >> rem_file.sh
fi
echo ""
echo ""
echo ""
echo " Files to remove under /usr"
echo " echo \"files to remove under /usr\" " >>
rem_file.sh
if [ `ls -l /usr/var/spool |grep mqueue|wc -l` -ge 1 ]
then
            echo "rm /usr/var/spool/mqueue" >> rem_file.sh
else
            echo " There is no file neither mqueue
            directory in the /usr/var/spool directory "
            echo " There is no file neither mqueue
            directory in the /usr/var/spool directory " >>
            nitl9_file
fi
if [ `ls -l /usr/var/adm |grep syslog.dated|wc -l` -ge 1
]
then
            echo "echo \"Delete all files except those of
            current date\" " >> rem_file.sh
            echo "rm /usr/var/adm/syslog.dated" >>
            rem_file.sh
else
            echo " There is no file neither syslog.dated
            directory in the /usr/var/adm directory "
            echo " There is no file neither
            syslog.dated directory in the /usr/var/adm
            directory " >> nitl9_file
fi
if [ `ls -l /usr/var/ss7 |grep core|wc -l` -ge 1 ]
then
            echo "rm /usr/var/ss7/core" >> rem_file.sh
else
            echo " There is no file neither core directory
            in the /usr/var/ss7 directory "
            echo " There is no file neither core directory
            in the /usr/var/ss7 directory " >> nitl9_file
fi
echo ""
echo ""
echo " Files to empty under /usr"
echo "echo \"files to empty under /usr\" " >> rem_file.sh
if [ `ls -l /usr/var/adm |grep messages|wc -l` -ge 1 ]
then
      echo "> /usr/var/adm/messages" >> rem_file.sh
else
      echo " There is no file neither messages directory
      in the /usr/var/adm directory "
```

```
                echo " There is no file neither messages directory
                in the /usr/var/adm directory " >> nitl9_file
        fi
        if [ `ls -l /usr/var/adm |grep wtmp |wc -l` -ge 1 ]
        then
                echo "> /usr/var/adm/wtmp" >> rem_file.sh
        else
                echo " There is no file neither wtmp directory in
                the /usr/var/adm directory "
                echo " There is no file neither wtmp directory in
                the /usr/var/adm directory " >> nitl9_file
        fi
        if [ `ls -l /usr/var/adm |grep binary.errlog |wc -l` -ge
        1 ]
        then
                        echo "> /usr/var/adm/binary.errlog" >>
                        rem_file.sh
        else
                        echo " There is no file neither binary.errlog
                        directory in the /usr/var/adm directory "
                        echo " There is no file neither binary.errlog
                        directory in the /usr/var/adm directory " >>
                        nitl9_file
        fi
        echo ""
        echo ""
        echo " Files to empty under /oracle "
        echo "echo \"files to empty under /oracle\" " >>
        rem_file.sh
        if [ `ls -l /oracle/network/log |grep sqlnet.log|wc -l` -
        ge 1 ]
        then
                echo "> /oracle/network/log/sqlnet.log" >>
                rem_file.sh
        else
                        echo " There is no file neither sqlnet.log
                        directory in the /oracle/network/log directory
                        "
                        echo " There is no file neither sqlnet.log
                        directory in the /oracle/network/log directory
                        " >> nitl9_file
        fi
        if [ `ls -l  /oracle/network/admin|grep sqlnet.log |wc -
        l` -ge 1 ]
        then
                        echo "> /oracle/network/admin/sqlnet.log" >>
                        rem_file.sh
        else
                        echo " There is no file neither sqlnet.log
                        directory in the /oracle/network/admin
                        directory "
                        echo " There is no file neither sqlnet.log
                        directory in the /oracle/network/admin
                        directory " >> nitl9_file
        fi
    fi
    passe="Y"
    sleep 2
else
        echo " It is mandatory to perform all checks ! "
```

```
                passe="N"
        fi
done

echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
        echo " Do you want to perform the next check : y/n "
        read volonte
        if [ $volonte = "y" ]
        then
                if [ $machine = "smp" ] || [ $machine = "bep" ] || [ $machine =
                "step" ] || [ $machine = "scp" ]
                then
                        echo ""
                        echo ""
                        echo "" >> nitl9_file
                        echo "" >> nitl9_file
                        echo "    ORBIX INSTALLATION "
                        echo "    ORBIX INSTALLATION " >> nitl9_file
                        echo "echo \"ORBIX INSTALLATION \" " >> rem_file.sh
                        echo ""
                        echo "" >> nitl9_file
                        echo "echo \"\" " >> rem_file.sh
                        echo " WARNING: ORBIX has to be removed if it is
                        installed "
                        echo " WARNING: ORBIX has to be removed if it is
                        installed" >> nitl9_file
                        echo "echo \" ORBIX has to be removed if it is
                        installed\" " >> rem_file.sh
                        echo ""
                        echo "" >> nitl9_file
                        echo "echo \"\" " >> rem_file.sh
                        echo "umount local_domain#orbix_fs" >> rem_file.sh
                        echo "rmfset local_domain orbix_fs" >> rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                        echo "echo \"Specific commands has to be entered on the
                        servers: Don't forget them \" " >> rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                        echo "echo \"Remove the line with orbix in the file
                        /etc/fstab\" " >> rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                        echo "echo \" In /in/local/conf/local.ini, in the orbix
                        part, set proc launch to 0 \" " >> rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                        echo "rmdir /orbix"  >> rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                        echo "echo \"\" " >> rem_file.sh
                else
                        echo ""
                        echo " This check is performed only on smp,step, bep and
                        scp "
                        echo ""
                fi
                passe="Y"
                sleep 2
        else
                echo " It is mandatory to perform all checks !"
```

```
            passe="N"
      fi
done

echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            if [ $machine = "smp" ] || [ $machine = "step" ]
            then
            echo ""
            echo ""
            echo "" >> nitl9_file
            echo "" >> nitl9_file
            echo "    CHECK OF THE SHARED DOMAIN "
            echo "    CHECK OF THE SHARED DOMAIN " >> nitl9_file
            echo ""
            echo "" >> nitl9_file
            echo " A double switchover has to be performed between the two
            SMPs after this check "
            echo " A double switchover has to be performed between the two
            SMPs after this check " >> nitl9_file
            echo ""
            echo "" >> nitl9_file
            echo " Caracteristics of the /in/smp directory:"
            echo " Caracteristics of the /in/smp directory:" >> nitl9_file
            echo ""
            echo "" >> nitl9_file
            ls -la /in/smp | more
            ls -la /in/smp >> nitl9_file
            echo ""
            echo " Check, on the stand-by server, that the directory is
            empty. "
            echo " Check also, on the active server, that the same
            directory is not empty. "
            echo "" >> nitl9_file
            echo " Check, on the stand-by server, that the directory is
            empty. " >> nitl9_file
            echo "" >> nitl9_file
            echo " Check also, onthe active server, that the same directory
            is not empty " >> nitl9_file
            echo ""
            echo ""
            echo "" >> nitl9_file
            else
                  echo ""
                  echo " This check is performed only on step and smp "
                  echo ""
            fi
            passe="Y"
            sleep 2
      else
            echo " It is mandatory to perform all checks !"
            passe="N"
      fi
```

```
done

echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            echo ""
            echo ""
            echo ""
            echo "" >> nitl9_file
            echo "" >> nitl9_file
            echo "    SYSCONFIGTAB VERIFICATION "
            echo "    SYSCONFIGTAB VERIFICATION " >> nitl9_file
            echo ""
            echo "" >> nitl9_file
            echo "Enter the RAM value: example 256M or 2G"
            echo ""
            read ram
            if [ `echo $ram | grep M|wc -l` -eq 1 ]
            then
                  val=`echo $ram |cut -d "M" -f1`
                  echo $val > cal.dc
                  echo 1024 >> cal.dc
                  echo 1024 >> cal.dc
                  echo 0.8 >> cal.dc
                  echo "*" >> cal.dc
                  echo "*" >> cal.dc
                  echo "*" >> cal.dc
                  echo p >> cal.dc
                  echo q >> cal.dc
                  ram_val=`dc cal.dc`
                  echo $ram_val > cal.dc
                  echo 1 >> cal.dc
                  echo / >> cal.dc
                  echo p >> cal.dc
                  echo q >> cal.dc
                  ram_val=`dc cal.dc`
                  echo " The 80% of the RAM size is : $ram_val "
                  echo ""
            fi
            if [ `echo $ram | grep G | wc -l` -eq 1 ]
            then
                  val=`echo $ram | cut -d "G" -f1`
                  echo $val > cal.dc
                  echo 1024 >> cal.dc
                  echo 1024 >> cal.dc
                  echo 1024 >> cal.dc
                  echo 0.8 >> cal.dc
                  echo "*" >> cal.dc
                  echo "*" >> cal.dc
                  echo "*" >> cal.dc
                  echo "*" >> cal.dc
                  echo p >> cal.dc
                  echo q >> cal.dc
                  ram_val=`dc cal.dc`
```

```
        echo $ram_val > cal.dc
        echo 1 >> cal.dc
        echo / >> cal.dc
        echo p >> cal.dc
        echo q >> cal.dc
        ram_val=`dc cal.dc`
        echo " The 80% of the RAM size is: $ram_val"
        echo ""
fi
if [ $ram_val -le 2147475456 ]
then
        if [ `sysconfig -q ipc | grep shm-max | cut -c 11-` -eq
        $ram_val ]
        then
                echo " OK: for shm-max"
                echo " OK: for shm-max" >> nitl9_file
        else
                echo " NOK: for shm-max. His value must be:
                $ram_val "
                echo " NOK: for shm-max. His value must be:
                $ram_val " >> nitl9_file
        fi
else
        if [ `sysconfig -q ipc | grep shm-max | cut -c 11-` -eq
        2147475456 ]
        then
                echo " OK: for shm-max"
                echo " OK: for shm-max" >> nitl9_file
        else
                echo " NOK: for shm-max. His value must be equal or
                lower than: 2147475456 "
                echo " NOK: for shm-max. His value must be equal or
                lower than: 2147475456 " >> nitl9_file
        fi
fi
if [ $ram_val -ge 1536000000 ]
then
        if [ `sysconfig -q proc | grep max-per-proc-address-space
        |cut -c 30-` -eq $ram_val ] ; then
        echo " OK for max-per-proc-address-space"
        echo " OK for max-per-proc-address-space" >> nitl9_file
        else
                echo " NOK for max-per-proc-address-space. His
                value must be: $ram_val "
                echo " NOK for max-per-proc-address-space. His
                value must be: $ram_val " >> nitl9_file
        fi
else
        if [ `sysconfig -q proc | grep max-per-proc-address-space
        |cut -c 30-` -eq 1536000000 ]
        then
                echo ""
                echo "" >> system_verif
                echo " OK for max-per-proc-address-space "
                echo " OK for max-per-proc-address-space " >>
                nitl9_file
        else
                echo " NOK for max-per-proc-address-space. His
                value must be: 1536000000 "
```

```
                echo " NOK for max-per-proc-address-space. His
                value must be: 1536000000 " >> nitl9_file
        fi
fi
if [ $ram_val -ge 1073741824 ]
then
        if [ `sysconfig -q proc | grep max-per-proc-data-size|cut
        -c 26-` -eq $ram_val ] ; then
        echo " OK for max-per-proc-data-size"
        echo " OK for max-per-proc-data-size" >> nitl9_file
        else
        echo " NOK for max-per-proc-data-size. His value must be:
        $ram_val "
        echo " NOK for max-per-proc-data-size. His value must be:
        $ram_val " >> nitl9_file
        fi
else
        if [ `sysconfig -q proc | grep max-per-proc-data-size|cut
        -c 26-` -eq 1073741824 ]
        then
                echo ""
                echo "" >> system_verif
                echo " OK for max-per-proc-data-size "
                echo " OK for max-per-proc-data-size " >>
                nitl9_file
        else
                echo " NOK for max-per-proc-data-size. His value
                must be: 1073741824 "
                echo " NOK for max-per-proc-data-size. His value
                must be: 1073741824 " >> nitl9_file
        fi
fi
if [ $ram_val -ge 1073741824 ]
then
        if [ `sysconfig -q proc | grep per-proc-address-space
        |cut -c 26- | tail -1` -eq $ram_val ] ; then
        echo " OK for per-proc-address-space "
        echo " OK for per-proc-address-space " >> nitl9_file
        else
        echo " NOK for per-proc-address-space. His value must be:
        $ram_val "
        echo " NOK for per-proc-address-space. His value must be:
        $ram_val " >> nitl9_file
        fi
else
        if [ `sysconfig -q proc | grep per-proc-address-space
        |cut -c 26- | tail -1` -eq 1073741824 ]
        then
                echo ""
                echo "" >> system_verif
                echo " OK for per-proc-address-space "
                echo " OK for per-proc-address-space " >>
                nitl9_file
        else
                echo " NOK for per-proc-address-space. His value
                must be: 1073741824 "
                echo " NOK for per-proc-address-space. His value
                must be: 1073741824 " >> nitl9_file
        fi
fi
```

```
            if [ `sysconfig -q proc | grep per-proc-data-size |cut -c 22- |
            head -1` -eq 1073741824 ] ; then
            echo " OK for per-proc-data-size "
            echo " OK for per-proc-data-size " >> nitl9_file
            else
            echo " NOK for per-proc-data-size. His value must be:
            1073741824 "
            echo " NOK for per-proc-data-size. His value must be:
            1073741824 " >> nitl9_file
            fi
            if [ $ram_val -ge 1073741824 ]
            then
                  if [ `sysconfig -q vm| grep vm-maxvas|cut -c 13-` -eq
                  $ram_val ]
                  then
                        echo " OK for vm-maxvas "
                        echo " OK for vm-maxvas " >> nitl9_file
                  else
                        echo " NOK for vm-maxvas. His value must be:
                        $ram_val "
                        echo " NOK for vm-maxvas. His value must be:
                        $ram_val " >> nitl9_file
                  fi
            else
                  if [ `sysconfig -q vm| grep vm-maxvas|cut -c 13-` -eq
                  1073741824 ]
                  then
                        echo " OK for vm-maxvas "
                        echo " OK for vm-maxvas " >> nitl9_file
                  else
                        echo " NOK for vm-maxvas. His value must be:
                        1073741824 "
                        echo " NOK for vm-maxvas. His value must be:
                        1073741824 " >> nitl9_file
                  fi
            fi
            passe="Y"
            sleep 2
      else
            echo " It is mandatory to perform all checks ! "
            passe="N"
      fi
done

echo ""
echo ""
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            if [ $machine = "smp" ] || [ $machine = "step" ]
            then
                  if [ `df -k | grep shared | wc -l` -eq 1 ]
                  then
                        echo ""
                        echo ""
                        echo "" >> nitl9_file
```

```
                        echo "" >> nitl9_file
                        echo "    FILE: /in/smp/platform/smp.ini "
                        echo "    FILE: /in/smp/platform/smp.ini" >>
                        nitl9_file
                        echo ""
                        echo "" >> nitl9_file
                        res=`cat /in/smp/platform/smp.ini | grep
                        Charching|wc -l`
                        rez=`cat /in/smp/platform/smp.ini | grep
                        charching|wc -l`
                        rez2=`cat /in/smp/platform/smp.ini | grep
                        CHARCHING|wc -l`
                        if [ $res -ge 1 ] || [ $rez -ge 1 ] || [ $rez2 -ge
                        1 ]
                        then
                              echo " NOK: The /in/smp/platform/smp.ini file
                              is not correct, it contains CHARCHING instead
                              of CHARGING"
                              echo " NOK: The /in/smp/platform/smp.ini file
                              is not correct, it contains CHARCHING instead
                              of CHARGING " >> nitl9_file
                        else
                              echo " OK: The /in/smp/platform/smp.ini file
                              is correct"
                              echo " OK: The /in/smp/platform/smp.ini file
                              is correct" >> nitl9_file
                        fi
                        echo ""
                        echo "" >> nitl9_file
                  else
                        echo ""
                        echo " This check is performed only on active smp "
                        echo ""
                  fi
            else
                  echo ""
                  echo " This check is performed only on step and smp "
                  echo ""
                  echo ""
            fi
            passe="Y"
            sleep 2
      else
            echo " It is mandatory to perform all checks !"
            passe="N"
      fi
done

echo ""
echo "" >> nitl9_file
passe="N"
while [ $passe = "N" ]
do
      echo " Do you want to perform the next check : y/n "
      read volonte
      if [ $volonte = "y" ]
      then
            echo ""
            echo ""
            echo "    FILE: /indelivery/etc/ipd.log "
```

```
            echo ""
            echo "" >> nitl9_file
            echo "    FILE /indelivery/etc/ipd.log " >> nitl9_file
            echo "" >> nitl9_file
            echo "" >> nitl9_file
            echo ""
            echo " The files found: "
            echo " The files found: " >> nitl9_file
            echo "" >> nitl9_file
            echo ""
            ls -l /indelivery/etc | grep ipd.log
            ls -l /indelivery/etc | grep ipd.log >> nitl9_file
            res1=`cat /indelivery/etc/ipd.log* | grep aborted | wc -l`
            res2=`cat /indelivery/etc/ipd.log* | grep Aborted | wc -l`
            res3=`cat /indelivery/etc/ipd.log* | grep ABORTED | wc -l`
            if [ $res1 -ge 1 ] || [ $res2 -ge 1 ] || [ $res3 -ge 1 ]
            then
                  echo "" >> nitl9_file
                  echo ""
                  echo " NOK: The file ipd.log is not correct. There are
                  some deliveries "aborted" "
                  echo " NOK: The file ipd.log is not correct. There are
                  some deliveries "aborted" " >> nitl9_file
                  echo ""
                  echo "" >> nitl9_file
            else
                  echo "" >> nitl9_file
                  echo ""
                  echo " OK: The file ipd.log is correct "
                  echo " OK: The file ipd.log is correct " >> nitl9_file
                  echo ""
                  echo "" >> nitl9_file
            fi
            passe="Y"
            sleep 2
      else
            echo " It is mandatory to perform all checks ! "
            passe="N"
      fi
done
```