Automate infrastructure updates in NIM environment

AIX patch management with Ansible

September 6, 2018

Updating AIX at a large scale infrastructure is easier than ever with the use of Ansible. One may either upgrade to latest or specific SP or TL and/or apply recommended security and hiper fixes to all its NIM clients.

Introduction

This article details how to use Ansible for **patch management automation** on IBM® AIX® systems. The hardware configuration, the installation process and the nominal use case are detailed.

Our development supports a NIM (Network Installation Management) environment in **PUSH mode**. Patch management playbooks are available on **AIXOSS GitHub repository**.
The **AIXOSS GitHub repository** contains Open Source Software ported to AIX. It also contains scripts to use with Open Source software to perform specific AIX tasks. Under “ansible-playbooks”, you will find a library including the Ansible scripts necessary for patch management with Ansible, and typical playbooks. These playbooks can be used as templates for your own purposes.

Ansible is agentless. The Ansible control machine connects to Ansible hosts (the NIM server/master in our case) thru ssh. Python is required on both the Ansible control machine and the Ansible hosts.

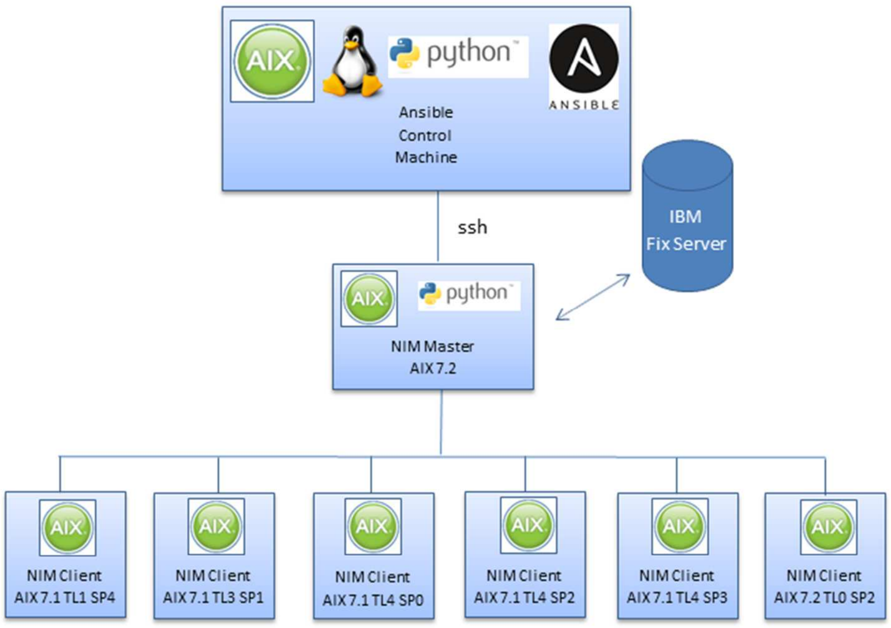
The NIM server must be able to connect to IBM Fix Server in order to download SPs or TLs through **SUMA** (Service Update Management Assistant). It can install those updates on its clients through **NIM** either by NFS or HTTP. The method is described in Managing AIX Updates Best Practices. It is also possible to patch NIM clients with recommended security and High Impact PERvasive (HIPER) fixes through **FLRTVC** (Fix Level Recommendation Tool Vulnerability Checker).

Configuration

Ansible can run on AIX or Linux machine. To install Ansible on the targeted machine, *Python, wget* and GIT tools must be installed as a prerequisite.

The NIM master must have access to the internet in order to download fixes and updates through HTTP and FTP protocol. It has several clients which are at different AIX releases and levels. It needs to be at a level at least as high as the highest level client.

The diagram below describes the hardware configuration for this use case.


AIX
python
ANSIBLE
Ansible Control Machine
ssh
IBM Fix Server
AIX
python
NIM Master AIX 7.2
NIM Client AIX 7.1 TL1 SP4
NIM Client AIX 7.1 TL3 SP1
NIM Client AIX 7.1 TL4 SP0
NIM Client AIX 7.1 TL4 SP2
NIM Client AIX 7.1 TL4 SP3
NIM Client AIX 7.2 TL0 SP2

We use fattony01.aus.stglabs.ibm.com (9.3.78.42) as NIM server in the examples below. Out NIM clients are: quimby01 to quimby012.

We assume that a "/home/users" directory exists on the Ansible Control Machine. This directory will be our home directory.

Ansible installation on the Control machine using pip

AIX system:

```
$ wget --no-check-certificate https://bootstrap.pypa.io/get-pip.py

$ python get-pip.py
…..

$ pip install ansible
…..
```

Linux system

```
$ sudo easy_install pip
…
$ sudo pip install ansible
```

Ansible installation on the Control machine using yum

AIX system:

```
$ wget --no-check-certificate https://ftp.software.ibm.com/aix/freeSoftware/aixtoolbox/INSTALLP/rpm.rte
…..
$ install -d. -acgXY rpm.rte
…..
$ wget --no-check-certificate
https://ftp.software.ibm.com/aix/freeSoftware/aixtoolbox/ezinstall/ppc/yum_bundle_v1.tar

…..
$ tar xf yum_bundle_v1.tar
…..
$ rpm -ivh *.rpm
…..
$ yum install ansible
…..
```

Linux system

```
$ sudo yum install ansible
```

Ansible installation on the Control machine using git

Using git to install Ansible allows you to benefit from the latest functionalities.

```
$ cd /home/users

$ git clone git://github.com/ansible/ansible.git --recursive
…..
$ cd ansible
…..
$ git pull --rebase
…..
$ git submodule update –init --recursive
…..
```

Ansible configuration

1) Set up the environment

```
$ source hacking/env-setup
…..
$ ansible --version
…..
```

2) Create the Ansible configuration file ansible.cfg under /etc/ansible to disable ssh interactive check for the known hosts.

```
$ mkdir /etc/ansible

$ cd /etc/ansible
$ cat > ansible.cfg
[default]
Host_key_checking = False
<Ctrl-D>
```

3) Create the /etc/ansible/hosts file to declare the hosts managed by Ansible.
We declare one host with ip address : 9.3.78.42. We call it “nimserver” for ease of use by Ansible and portability to execute with other NIM servers.

```
$ cat > /etc/ansible/hosts
[nimserver]
9.3.78.42
<Ctrl-D>
```

4) Create ssh keys for the Control machine and copy them to the NIM server.

```
$ ssh-keygen -t rsa
…
$ ssh-copy-id root@fattony01.aus.stglabs.ibm.com
…
```

5) Verify your configuration using ping through Ansible.

```
$ ansible all -m ping
9.3.78.42 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

6) Create a directory for playbooks.

```
$ mkdir playbooks
```

7) Create your first playbook in yaml format to automate tasks.

```
$ cd playbooks

$ cat > lsnim.yml
- name: "My Playbook"
  hosts: nimserver
  gather_facts: no

  tasks:
    - name: "List the NIM clients of your NIM server"
      command: "/usr/sbin/lsnim -t standalone"
      register: output

    - debug: var=output.stdout_lines
<Ctrl-D>
```

This task executes on « nimserver » defined before in /etc/ansible/hosts

Environment variables will not be set before execution of the task

The task runs a « lsnim » command

We record the task output in the « output » variable

The debug task displays the stdout registered in the « output » variable

8) Verify the syntax of your playbook.

```
$ ansible-playbook --syntax-check lsnim.yml

playbook: lsnim.yml
```

9) Execute your playbook.

```
$ ansible-playbook lsnim.yml
PLAY [My playbook] *************************************************************

TASK [List the NIM clients] ****************************************************
changed: [9.3.78.42]

TASK [debug] *******************************************************************
ok: [9.3.78.42] => {
    "output.stdout_lines": [
        "quimby11     machines      standalone",
        "quimby12     machines      standalone",
        "quimby07     machines      standalone",
        "quimby08     machines      standalone",
        "quimby06     machines      standalone",
        "quimby05     machines      standalone",
        "quimby01     machines      standalone",
        "quimby02     machines      standalone",
        "quimby03     machines      standalone",
        "quimby04     machines      standalone",
        "quimby09     machines      standalone",
        "quimby10     machines      standalone"
    ]
}

PLAY RECAP ********************************************************************
9.3.78.42                  : ok=2    changed=1    unreachable=0    failed=0
…
```

To learn more, watch the video:

https://youtu.be/-0PsBTDH7PM

Download AIX Patch Management Playbooks from AIXOSS GitHub

In order to take advantage of the **AIXOSS GitHub repository**:

```
$ /home/users/ansible

$ git clone git://github.com/aixoss/ansible-playbooks -recursive
….
$ cd ansible-playbooks

$ ls
library                           playbook_aix_suma_targets_list.yml
playbook_aix_flrtvc.yml           playbook_aix_suma_targets_range.yml
playbook_aix_nim_check.yml        playbook_aix_suma_targets_star.yml
playbook_aix_nim_reboot.yml       playbook_aix_suma.yml
playbook_aix_suma_nim.yml         README.md
playbook_aix_suma_targets_all.yml

$ ls library
aix_flrtvc.py  aix_nim.py  aix_suma.py

$
```

Update by SUMA example

1) Create a playbook to check the status of the NIM clients and run it.

```
$ cd /home/user/ansible/ansible-playbooks

$ cat > playbook_aix_nim_check.yml
- name: “NIM on AIX”
  hosts: all
  gather_facts: no

  tasks:
  - name: “AIX NIM”
    aix_nim:
      action: ‘check’
      description: ‘NIM check’
   register: nim_output

  - debug: var=nim_output
<Ctrl-D>

$ ansible-playbook playbook_aix_nim_check.yml
PLAY [NIM on AIX] **********************************************kkkkkkkkkkkkkkkkk***************

TASK [AIX NIM] ****************************************************************
ok: [9.3.78.42]

TASK [debug] ******************************************************************
ok: [9.3.78.42] => {
    "nim_output": {
        "changed": false,
        "debug_output": [
            "exec command:['lsnim', '-t', 'lpp_source', '-l']",
            "exec command Error:"
        ],
        "msg": "NIM check completed successfully",
        "nim_output": [
            "check update status:",
            [
                "",
                "",
                "+----------+-----------------+--------------------------+",
                "| machine  |     oslevel     |          Cstate          |",
                "+----------+-----------------+--------------------------+",
                "| quimby01 | 7100-03-01-1341 | ready for a NIM operation |",
                "| quimby10 | 7100-04-01-1543 | ready for a NIM operation |",
                "| quimby09 | 7100-03-04-1441 | ready for a NIM operation |",
```

Task to execute: « aix_nim »

« action » is a required argument for the «aix_nim» task. Possible values are: “check”, “update” and “reboot”

```
                "| quimby08 | 7100-03-01-1341 | ready for a NIM operation |",
                "| quimby02 | 7100-03-01-1341 | ready for a NIM operation |",
                "| quimby03 | 7100-04-01-1543 | ready for a NIM operation |",
                "| quimby11 | 7100-04-01-1543 | ready for a NIM operation |",
                "| quimby12 | 7200-01-00-0000 | ready for a NIM operation |",
                "| quimby07 | 7200-01-00-0000 | ready for a NIM operation |",
                "| quimby06 | 7200-00-02-1614 | ready for a NIM operation |",
                "| quimby05 | 7100-04-03-1642 | ready for a NIM operation |",
                "| quimby04 | 7100-04-02-1614 | ready for a NIM operation |",
                "|  master  | 7200-01-00-0000 | ready for a NIM operation |",
                "+----------+-----------------+---------------------------+"
                                                                                ]
        ]
    }
}

PLAY RECAP *********************************************************************
9.3.78.42                  : ok=2    changed=0    unreachable=0    failed=0
```

Note quimby03 level:
7100-04-01-1543

Note quimby04 level:
7100-04-02-1614

2) Create a playbook to build a NIM resource on the NIM Master using SUMA and run it.

The "aix_suma" task creates a resource on the NIM server from the fix repository. It gathers all the fixes necessary to update the targets from their current level to the level specified.

« action » is a required argument for the «aix_suma» task. Possible values are: "download" and "preview". These 2 actions have 3 required arguments: "oslevel", "location" and "targets".

Default value: the latest Service Pack (SP) available for the highest release. A Service Pack level (SP) or Technical Level (TL) can also be specified.

where the NIM resource will be created on the NIM.

NIM clients for which the NIM resource will be created.*

```
$ cat > playbook_aix_suma_demo.yml
- name: "SUMA on AIX"
  hosts: all
  gather_facts: no

  tasks:
  - name: "Downloading latest SP for latest release"
    aix_suma:
      oslevel: ‘Latest’
      location: '/export/extra'
      targets: “quimby0[3:4]”
      action: ‘download’
    ignore_errors: ‘True’

  register: output
- debug: var=output
<Ctrl-D>

$ ansible-playbook playbook_aix_suma_test.yml

PLAY [SUMA on AIX] ******************************************

TASK [Downloading latest SP for latest release] ********************************
ok: [9.3.78.42]

TASK [debug] ******************************************************************
ok: [9.3.78.42] => {
    "output": {
        "changed": false,
        "lpp_source_name": "",
        "msg": "Suma download completed successfully",
        "suma_output": [
            "SUMA - Target list: ['quimby03', 'quimby04']",
            "SUMA - Command:/usr/sbin/suma -x -a RqType=SP -a Action=Preview -a RqName=7100-
04-04-1717 -a FilterML=7100-04 -a DLTarget=/export/extra/7100-04-04-1717-lpp_source -a
DisplayName=\"download request for oslevel Latest\"",
            "Preview summary : 0 to download, 0 failed, 400 skipped"
            "NIM command:/usr/sbin/nim  -o define  -t lpp_source  -a server=master -a
location=/export/extra/7100-04-04-1717-lpp_source -a packages=all 7100-04-04-1717-lpp_source
",
            "NIM operation succeeded - output:Preparing to copy install images (this will
take several minutes)...\n\n\nNow checking for missing install images...\n\nAll required
install images have been found. This lpp_source is now ready.\n"

        ],
        "target_list": []
    }
}

PLAY RECAP ********************************************************************
9.3.78.42                  : ok=2    changed=0    unreachable=0    failed=0
```

* Star convention (e.g. quimby*) will apply the action to all the NIM clients of the NIM Master with a name starting by "quimby".
Range of machines (e.g. quimby0[3:4]) will apply the action to quimby03, and quimby04 if they are NIM clients of the NIM Master.
NIM clients can also be listed (e.g. "quimby01, quimby06").
Warning: SUMA will define the lowest SP and TL level of the targets and will build a NIM resource to bring these targets to the SP or TL specified in "oslevel". Trying then to use this resource on targets with a lower SP or TL will result in a failure.

3) Create a playbook to update the NIM clients and run it.

Required arguments for update: "lpp_source", and "targets".

Task to execute: "aix_nim"

NIM clients to update.

Resource to use to update the NIM clients

Optional argument: if set to 'False': ensure that a machine will be completely installed before installing the following one

```
$ cat > playbook_aix_nim_demo.yml
- name: "NIM on AIX"
  hosts: all
  gather_facts: no

  tasks:
    name: "AIX NIM"
    aix_nim:
      action: 'update'
      force: 'false'
      async: 'false'
      targets: "quimby0[3:4]"
      lpp_source: 'latest_sp'
      description: 'playbook_aix_nim_update'

  register: nim_output
- debug: var=nim_output
<Ctrl-D>

$ ansible-playbook playbook_aix_nim_demo.yml
PLAY [NIM on AIX]
*****************************************************************************************

TASK [AIX NIM]
*********************************************************************************************
changed: [9.3.78.42]
```

```
TASK [debug]
*********************************************************************************************
**
ok: [9.3.78.42] => {
    "nim_output": {
        "changed": true,
        "debug_output": [
            "exec command:['lsnim', '-t', 'lpp_source', '-l']",
            "exec command Error:"
        ],
        "msg": "NIM update completed successfully",
        "nim_output": [
            "NIM - Command:/usr/sbin/nim -o cust -a lpp_source=7100-04-04-1717-lpp_source -a 
fixes=update_all -a accept_licenses=yes -a async=no quimby03",
            "Start updating machine(s) quimby03 to 7100-04-04-1717-lpp_source",
            "",
            "+-----------------------------------------------------------------------------
+",
            "\t\t    Pre-installation Verification...",
            "+-----------------------------------------------------------------------------
+",
            "Verifying selections...done",
            "Verifying requisites...done",
            "Results...",
            "",
            "SUCCESSES",
            "---------",
            "  Filesets listed in this section passed pre-installation verification",
            "  and will be installed.",
            "",
            "  Selected Filesets",
            "  -----------------",
            "  Java5.sdk 5.0.0.620                         # Java SDK 32-bit",
            "  Java5_64.sdk 5.0.0.620                      # Java SDK 64-bit",
…

            "bos.rte.diag                7.1.4.30        USR         APPLY       SUCCESS    
",
            "bos.rte.diag                7.1.4.30        ROOT        APPLY       SUCCESS    
",
            "Java6.sdk                   6.0.0.641       USR         APPLY       SUCCESS    
",
            "Java6.sdk                   6.0.0.641       ROOT        APPLY       SUCCESS    
",
            "",
            "installp:  * * *  A T T E N T I O N ! ! ! ",
            "\tSoftware changes processed during this session require this system ",
            "\tand any of its diskless/dataless clients to be rebooted in order ",
            "\tfor the changes to be made effective.",
                "NIM - Finish updating quimby04 synchronously."
            ]
        }
    }

PLAY RECAP
*********************************************************************************************
****
9.3.78.42                  : ok=2    changed=1    unreachable=0    failed=0
```

4) Run the playbook_aix_nim_check to verify that quimbu03 and quimby04 have been updated.

```
$ ansible-playbook playbook_aix_nim_check.yml
PLAY [NIM on AIX] ***************************************************************************

TASK [AIX NIM] ******************************************************************************
ok: [9.3.78.42]

TASK [debug] ********************************************************************************
ok: [9.3.78.42] => {
    "nim_output": {
        "changed": false,
        "debug_output": [
            "exec command:['lsnim', '-t', 'lpp_source', '-l']",
            "exec command Error:"
        ],
        "msg": "NIM check completed successfully",
        "nim_output": [
            "check update status:",
            [
                "",
                "+----------+-----------------+--------------------------+",
                "| machine  |     oslevel     |          Cstate          |",
                "+----------+-----------------+--------------------------+",
                "| quimby01 | 7100-03-01-1341 | ready for a NIM operation |",
                "| quimby10 | 7100-04-01-1543 | ready for a NIM operation |",
                "| quimby09 | 7100-03-04-1441 | ready for a NIM operation |",
                "| quimby08 | 7100-03-01-1341 | ready for a NIM operation |",
                "| quimby02 | 7100-03-01-1341 | ready for a NIM operation |",
                "| quimby03 | 7100-04-04-1717 | ready for a NIM operation |",
                "| quimby11 | 7100-04-01-1543 | ready for a NIM operation |",
                "| quimby12 | 7200-01-00-0000 | ready for a NIM operation |",
                "| quimby07 | 7200-01-00-0000 | ready for a NIM operation |",
                "| quimby06 | 7200-00-02-1614 | ready for a NIM operation |",
                "| quimby05 | 7100-04-03-1642 | ready for a NIM operation |",
                "| quimby04 | 7100-04-04-1717 | ready for a NIM operation |",
                "|  master  | 7200-01-00-0000 | ready for a NIM operation |",
```

Note quimby03 level:
7100-04-04-1717

Note quimby04 level :
7100-04-04-1717

```
                "+----------+-----------------+--------------------------+"
            ]
        ]
    }
}

PLAY RECAP *************************************************************************************************
9.3.78.42                  : ok=2    changed=0    unreachable=0    failed=0
```

5) Once you have updated your machines, run playbook_aix_nim_reboot.yml to reboot the updated machines.

```
$ cat > playbook_aix_nim_reboot.yml
- name: "NIM reboot on AIX"
  hosts: all
  gather_facts: no

  tasks:
    - name: "AIX NIM"
      aix_nim:
        action: 'reboot'
        targets: "quimby0[3:4]"
        description: 'NIM reboot'

      register: nim_output
    - debug: var=nim_output
<Ctrl-D>

$ ansible-playbook playbook_aix_nim_reboot.yml
PLAY [NIM reboot on AIX] *******************************************************

TASK [AIX NIM] *****************************************************************
changed: [9.3.78.42]

TASK [debug] *******************************************************************
ok: [9.3.78.42] => {
    "nim_output": {
        "changed": true,
        "debug_output": [
            "exec command:['lsnim', '-t', 'lpp_source', '-l']",
            "exec command Error:"
        ],
        "msg": "NIM reboot completed successfully",
        "nim_output": [
            "NIM - Command:nim -o reboot quimby03 quimby04"
        ]
    }
}

PLAY RECAP ***********************************************************************
9.3.78.42                  : ok=2    changed=1    unreachable=0    failed=0
```

To learn more, watch the video:

https://youtu.be/J_mtUHzo5lo

Patch by FLRTVC example

1) Create a playbook and run it

```
$ cd /home/user/ansible/playbooks

$ cat > playbook_aix_flrtvc_demo.yml
- name: "FLRTVC on AIX playbook"
  hosts: fattony01
  gather_facts: no

  tasks:
    - name: "FLRTVC"
      aix_flrtvc:
        targets: quimby0[3:4]
        path: /tmp/flrtvc-ansible
        #apar: sec
        #csv: /tmp/apar.csv
        #check_only: False
        #download_only: False
        #force: False
        #clean: True


      register: result
    - debug: var=result
<Ctrl-D>
```

Task to execute: « aix_flrtvc »

Working directory for flrtvc ; will contain flrtvc reports

Path to the csv file describing the fixes if already transferred from the fix server

By default, the aix_flrtvc task will check the state of the target, download the missing fixes, and update the targets. “check_only: True” will only perform the check; “donwload_only: True” will check the targets and download the fixes (no update)

By default, do not remove currently installed fixes before running flrtvc

By default, do not remove the working directory at the end of execution

Apar type of the fixes to check or download.

“sec” : for corrections to security threats; “hiper” for corrections to High Impact PERvasive threats.

“all” specifies both types of apar and is the default (this is the value in this example since the line is commented)

```
$ ansible-playbook playbook_aix_flrtvc_demo.yml
PLAY [FLRTVC on AIX playbook] *************************************************

TASK [FLRTVC] *****************************************************************
ok: [fattony01]

TASK [debug] ******************************************************************
ok: [fattony01] => {
    "result": {
        "changed": false,
        "meta": {
               "quimby03": {
                   "0.report": [
                       "Fileset|Current Version|Type|EFix Installed|Abstract|Unsafe Versions|APARs|Bulletin
URL|Download URL|CVSS Base Score|Reboot Required|Last Update|Fixed In",
 …
 "1.parse": [
                       "ftp://aix.software.ibm.com/aix/efixes/security/bellmail_fix.tar",
                       "https://aix.software.ibm.com/aix/efixes/security/bind_fix11.tar",
 …
                   ],
                   "2.discover": [
                       "bellmail_fix/IV91006s8a.161125.epkg.Z",
                       "bellmail_fix/IV91007s7a.161125.epkg.Z",
                       "bellmail_fix/IV91008s3a.161125.epkg.Z",
                       "bellmail_fix/IV91010s2a.161125.epkg.Z",
                       "bellmail_fix/IV91011s1a.161125.epkg.Z",
 …
 …
                   ],
                   "3.download": [
                       "/flrtvc-ansible/work/tardir/bellmail_fix/IV91006s8a.161125.epkg.Z",
                       "/flrtvc-ansible/work/tardir/bellmail_fix/IV91007s7a.161125.epkg.Z",
                       "/flrtvc-ansible/work/tardir/bellmail_fix/IV91008s3a.161125.epkg.Z",
 …
 …
                   ],
                   "4.check": [
                       "/flrtvc-ansible/work/tardir/bellmail_fix/IV91008s3a.161125.epkg.Z",
                       "/flrtvc-ansible/work/tardir/bind_fix11/IV81281m0b.160331.epkg.Z",
                       "/flrtvc-ansible/work/tardir/bind_fix11/IV81281m1a.160315.epkg.Z",
                       "/flrtvc-ansible/work/tardir/tcpdump_fix2/IV94726s4c.170417.epkg.Z",
 …
 …
                   "5.install": [
                       "",
                       "Initializing log /var/adm/ras/emgr.log ...",
 …
                       "EPKG NUMBER       LABEL             OPERATION             RESULT           ",
                       "===========       ==============    =================     ==============   ",
                       "1                 IV94726s4c        INSTALL               SUCCESS          ",
                       "2                 IV93363m3a        INSTALL               SUCCESS          ",
                       "3                 IV92240m3a        INSTALL               SUCCESS          ",
                       "4                 IV91951m3a        INSTALL               SUCCESS          ",
                       "5                 IV91487s1a        INSTALL               SUCCESS          ",
                       "6                 IV91255m1b        INSTALL               FAILURE          ",
                       "7                 IV91008s3a        INSTALL               FAILURE          ",
                       "8                 IV90915s1a        INSTALL               SUCCESS          ",
                       "9                 IV90451s0a        INSTALL               SUCCESS          ",
                       "10                IV90451s0a        INSTALL               FAILURE          ",
                       "11               IV89829m1a          INSTALL               FAILURE           ",
                       "12               IV89737s2a          INSTALL               SUCCESS           ",
                       "13               IV88007s0a          INSTALL               SUCCESS           ",
                       "14               IV87640s2a          INSTALL               SUCCESS           ",
                       "15               IV87420m0a          INSTALL               FAILURE           ",
                       "16               IV84947m1a          INSTALL               FAILURE           ",
                       "17               IV84458s1a          INSTALL               FAILURE           ",
                       "18               IV83994m1a          INSTALL               FAILURE           ",
                       "19               IV81493s1a          INSTALL               SUCCESS           ",
                       "20               IV81493s1a          INSTALL               FAILURE           ",
                       "21               IV81459s1b          INSTALL               SUCCESS           ",
                       "22               IV81303s1a          INSTALL               FAILURE           ",
                       "23               IV81303s1a          INSTALL               FAILURE           ",
                       "24               IV81281m1a          INSTALL               FAILURE           ",
                       "25               IV81281m0b          INSTALL               FAILURE           ",
                       "26               IV80586s1a          INSTALL               SUCCESS           ",
                       "27               IV80191s1a          INSTALL               FAILURE           ",
                       "28               IV79944s1a          INSTALL               FAILURE           ",
                       "29               IV79262m1a          INSTALL               FAILURE           ",
                       "ATTENTION: system reboot is required. Please see the \"Reboot Processing\"",
                       "sections in the output above or in the /var/adm/ras/emgr.log file.",
                       "Return Status: FAILURE",
                       "0042-001 nim: processing error encountered on \"master\":",
                       "   0042-001 m_cust: processing error encountered on \"quimby03\":",
                       "   0042-175 c_script: An unexpected result was returned by the
   \"fattony01.aus.stglabs.ibm.com:/export/nim/scripts/quimby03.script\" command:",
                       "",
                       "0042-175 c_installp: An unexpected result was returned by the \"/usr/sbin/installp\"
   command:",
                       "See the log file:",
                       "\t/var/adm/ras/nim.installp",
                       "for details or use the \"showlog\" operation.",
   …
                   ],
                   "5.install": [
                       "",
                       "Initializing log /var/adm/ras/emgr.log ...",
   …
                       "EPKG NUMBER      LABEL               OPERATION             RESULT            ",
                       "===========      ==============      =================     ==============    ",
                       "1                IV94726s4c          INSTALL               SUCCESS           ",
                       "2                IV93845s3b          INSTALL               SUCCESS           ",
                       "3                IV93363m3a          INSTALL               SUCCESS           ",
                       "4                IV92240m3a          INSTALL               SUCCESS           ",
                       "5                IV91951m3a          INSTALL               SUCCESS           ",
                       "6                IV91487s3a          INSTALL               FAILURE           ",
                       "7                IV91431s3a          INSTALL               FAILURE           ",
                       "8                IV91255m3c          INSTALL               FAILURE           ",
                       "9                IV91019s3a          INSTALL               SUCCESS           ",
                       "10             IV91008s3a          INSTALL                FAILURE           ",
                       "11             IV90915s3a          INSTALL                SUCCESS           ",
                       "12             IV90451s0a          INSTALL                SUCCESS           ",
                       "13             IV90451s0a          INSTALL                FAILURE           ",
                       "ATTENTION: system reboot is required. Please see the \"Reboot Processing\"",
                       "sections in the output above or in the /var/adm/ras/emgr.log file.",
                       "Return Status: FAILURE",
                       "0042-001 nim: processing error encountered on \"master\":",
                       "   0042-001 m_cust: processing error encountered on \"quimby04\":",
                       "   0042-175 c_script: An unexpected result was returned by the
   \"fattony01.aus.stglabs.ibm.com:/export/nim/scripts/quimby04.script\" command:",
                       "",
                       "0042-175 c_installp: An unexpected result was returned by the \"/usr/sbin/installp\"
   command:",
                       "See the log file:",
                       "\t/var/adm/ras/nim.installp",
                       "for details or use the \"showlog\" operation.",
                       "",
                       "",
                       ""
                   ]
               }
           },
           "msg": "exit successfully"
       }
   }

   PLAY RECAP *********************************************************************
   fattony01                  : ok=2    changed=0    unreachable=0    failed=0
```

To learn more, watch the video:

https://youtu.be/95BvMB_U8A8