Centralized Backup & Restore of NETWITNESS Version 11.2+
A Wrapper for Netwitness Recovery Tool (NRT)

SCENARIO -

You need to remotely backup your NetWitness hosts to a central location, to satisfy Disaster Recovery Requirements, perform a Tech Refreshes, or to be prepared for RMA replacement of a device.

SOLUTION – A WRAPPER FOR NRT

Building off the framework of the original nw-backup scripts written for 10.x backup/restore and migration to 11.x, a new set of version 11 scripts has been written as a "wrapper" to the built in NRT functionality of NetWitness Version 11.2+.

OVERVIEW –

The solution consists of 3 scripts (all run from the NW Admin server (node-zero)) and on example nw-base.nrt file:

- get-all-systems11.sh – generates an inventory file of all hosts in the NW server deployment.
- ssh-propagate11.sh – Generates an ECDSA key pair on the NW Server and propagates the public key to all hosts.
- nw-base.nrt – An NRT script to add any custom files to the backup directory as part of the backup process.
- nw-backup11.sh – The main backup script.

PROCESS –

Copy the attached zip file (containing the above files) to the NW Admin Server host (node-zero) and unzip:

```
mkdir /root/scripts
unzip nw-backup11.zip -d /root/scripts
cd /root/scripts/nw-backup11
chmod +x *.sh
```

Step 1: Run get-all-systems11.sh

```
./get-all-systems11.sh
```

This runs on the NW Admin Server, no options necessary. It will create the /var/netwitness/nw-backup directory on the NW Admin Server. Using a combination of mongo and salt commands, it creates the "all-systems" file in the created directory. The all-systems file is used by the other scripts (and for many other scripts I've developed), with entries that contain the following in comma separated format:

```
DeviceType,Hostname,IPAddress,MinionID,SerialNumber
```

DeviceType = NRT Category Type (i.e. AdminServer,Broker,Concentrator,Decoder,ESAPrimary, etc.)
Hostname = Short hostname (not FQDN)
IPAddress = Management Interface IP address of host
MinionID = Unique Salt MinionID of host
SerialNumber = Device Serial Number for Reference and Support

Example:

```
AdminServer,nw-admin,192.168.1.129,70f95dc0-3cb6-4fd4-b9f2-ac923d0ba594,PK10T51
ESAPrimary,nw-esa,192.168.1.131,a598cb6b-4bd2-4ba2-af6a-79df3dab35e6,R9L8LNM
LogHybrid,nw-loghyb,192.168.1.133,87fc872c-68e3-45e3-9108-e30f847dc14e,PK10T0A
Malware,nw-malware,192.168.1.132,2c98e425-57a0-47d2-82d7-15795a6165f5,R90BCFWP
```

```
NetworkHybrid,nw-nethyb,192.168.1.134,9a99294e-3889-48b0-9555-11d3c21e2018,R90218K6
```

The script is designed to run from the cron on a regular basis (if you are in a dynamic environment were systems are added/removed on a regular basis), it has a 30 second timeout on the one question it asks.

One new feature is the generation of "new-systems" and "old-systems" files if it finds new hosts or missing hosts compared to the last time run. If new systems have been added to the environment, the new-systems file can be used by other scripts for running specific targeted actions against the newer hosts. If a system is "offline" the old-systems file will have the entries for those hosts, so they are not lost and can be added back into the main "all-systems" file if needed.

Step 2: Run ssh-propagate11.sh

```
./ssh-propagate11.sh <options>
```

Propagate the Nw Admin Server root user ecdsa-521 bit public ssh key to target hosts

- Run with no options, uses the /var/netwitness/nw-backup/all-systems file to run against all hosts in the all-systems file (note: if key already exists on host, it is just verified).
- Run with argument of "new-systems" will just run against the hosts found in /var/netwitness/nw-backup/new-systems.
- Run with argument of <hostname> (a specific system hostname, or part of a hostname) it will grep the matching host(s) out of the all-systems file and run against those hosts.

Step 3: Modify the nw-base.nrt file

```
vi /root/scripts/nw-backup11/nw-base.nrt
```

The default /etc/netwitness/recoverytool/nw-base.nrt file, distributed file with 11.2 and 11.3 systems, only contains the following entries:

```
name nw-base
directory /etc/netwitness/platform/nodeinfo
file /etc/machine-id

# unmanaged files
stash /etc/fstab
stash /etc/hosts
stash /etc/sysconfig/iptables
stash /root/.ssh

# for azure
stash /etc/krb5.conf
stash /etc/logrotate.d/waagent.logrotate
stash /etc/mdadm.conf
stash /etc/waagent.conf
```

The "STASH" entries are NOT restored during an NRT import action, but are available for reference in the /var/netwitness/backup/unmanaged folder, after the restore. I have included an expanded list of stash files and directories to include modifications I have seen at several customers, that would be needed to restore complete functionality after a restore. Edit the file to include any additional locations (files or directories) that contain customizations in your deployment, and save the file in the same directory as the nw-backup11.sh script is located (if you move it), the backup script will verify the file on each system matches your modified file and if not, will automatically copy the modified file to each host before running NRT on that host.

Step 4: Run nw-backup11.sh

```
./nw-backup11.sh <options>

Usage:
      ./nw-backup11.sh -b <NRTBackupPath> -m <RemoteTransferMode>
      -l <NFSMountPoint> -p <NWBackupPath> -s <RemoteServerIP> -d <RemotePath>
      -t <DeviceType>

Configuration Options:
       Note: All command-line options are optional. With no options selected, the
script will use options set within the file itself and backup ALL devices listed in
the all-systems file (except for any commented(#) out).

General Options:
      -b <NRTBackupPath>: Path to the location for write NRT backup files on each
                          server.  Default:(/var/netwitness/backup)
      -m <RemoteTransferMode>: Remote transfer mode (scp or nfs).  Default:(scp)
      -p <NWBackupPath>: NW Admin server path for logs and all-systems file(s).
                         Default:(/var/netwitnes/nw-backup)
      -d <RemotePath>: Path on remote server to store backup files, via nfs or scp.
                       Default:(/var/netwitnes/nw-backup)
      -s <RemoteServerIP>: Server IP address to store backup files, via nfs or scp.
                           Default:(IP of NW Admin Server)
      -l <NFSMountPoint>: Local mount point for NFS share. Default:(/mnt/backup)
      -t <DeviceType>: Backup ONLY targeted device type or device(s). Default:(all)
                       special devTypes:
                          core (Broker, Concentrator, (Log)Decoder, Archiver,
                                LogCollector(vlc), NetworkHybrid, LogHybrid)
                          nonw (all devices excluding AdminServer)
                        nwonly (AdminServer only)
                       esaonly (all ESAPrimary & ESASecondary devices only)
                      endpoint (Gateway, EndpointHybrid, EndpointLogHybrid)

        Example1:  ./nw-backup11.sh
                   Would run the backup with options as set in the Header of the
                   script itself.

        Example2:  ./nw-backup11.sh -b /var/netwitness/backup -m nfs -l /mnt/backup
                   -d /var/netwitness/nw-backup -s 192.168.1.129 -t NetworkHybrid

                   Would run a backup with the following options:
                   -b: store NRT output in /var/netwitness/backup on each server
                   -m: use NFS to move the backup files to the remote backup store
                   -l: mount the remote nfs share to /mnt/backup.
                   -s: use 192.168.1.129 as the nfs server IP address.
                   -d: nfs mount /var/netwitness/nw-backup from the remote server.
                   -t: target only Network Hybrids on this backup run
```

Notes:

- If using SCP to a server other than the NW Admin Server, the copy uses the "-3" option and makes the transfer of the backup file via the NW Admin Server, so ONLY the NW Admin server needs to have SSH key authentication configured to all hosts.

- By Default the /etc/netwitness/recovery-tool/category.sequence file does not include the mongo databases in the backup of the ESAPrimary server. This script checks that file and ADDS the line for backing up the mongo databases automatically.
- Make sure the modified nw-base.nrt file is in the same directory you are running the nw-backup11.sh script from, the script will hash that file, and verify that hash against the file on each server, if they do not match, it will copy the file in the script directory (where you ran the nw-backup11.sh script from) to the remote host, before triggering the NRT backup on that host. If you do not want to make any modifications to the default file, don't include the file in the script run directory.
- Deploy Admin password is programmatically called, so no exposure of password in the scripts
- The /var/netwitness/nw-backup/all-systems file can be used for a myriad of other scripting calls, or for targeting a specific type of host, especially when using "salt" commands.