

NCAR-Developed Tools

Bill Anderson and Marc Genty
National Center for Atmospheric Research
HUF 2017

Introduction

- Over the years, we've benefited from tools that others have developed
- In this talk, we'll share information about tools we've developed

Implementation Goals

- simplicity
- portability
- scalability

Tools

- tapeinfo
- checkForMigration
- Nagios

tapeinfo

- Need for tape info in an easy-to-use tabular form
- dump_sspvs, etc. help, but not all info
- hpssadm.pl “Cartridges and Volumes” output not tabular
- Also, helpful to have library location info

tapeinfo

- Combines info from hpssadm.pl and ACSLS
- Two components:
 - script that gathers and merges data once a day via cron and stores output in a file
 - command line tool that displays that data as tabular output

tapeinfo

medium	type	host-status	GB	timeLastRd	timeLastWt
400000	T10KD	hpss01-EOM	9301.88	01-Jan-1969	01-Sep-2016
400001	T10KD	hpss01-EOM	9708.78	01-Jan-1969	12-Apr-2016
400002	T10KD	hpss01-EOM	9270.82	01-Jan-1969	20-Apr-2016
400003	T10KD	hpss01-EOM	11773.24	01-Jan-1969	21-Oct-2016
400004	T10KD	hpss01-EOM	10278.83	01-Jan-1969	05-May-2016
400005	T10KD	hpss01-EOM	8830.21	01-Jan-1969	21-Oct-2016
400006	T10KD	hpss01-EOM	10192.85	01-Jan-1969	04-May-2016

- Estimate compression ratio

tapeinfo

- Tapes associated with a file family
- Cold tapes

medium	type	nsecs	GB	timeLastRd	timeLastWt	family
400000	T10KD	60929	9301.88	01-Jan-1969	01-Sep-2016	"Disaster Recovery"
400001	T10KD	55618	9708.78	01-Jan-1969	12-Apr-2016	"Disaster Recovery"
400002	T10KD	10514	9270.82	01-Jan-1969	20-Apr-2016	"Disaster Recovery"
400003	T10KD	22746	11773.24	01-Jan-1969	21-Oct-2016	"Disaster Recovery"
400004	T10KD	15009	10278.83	01-Jan-1969	05-May-2016	"Disaster Recovery"

tapeinfo

- Tape distribution across libraries

medium	library	type	family
400120	Lib-5	T10KD	"Disaster Recovery"
400121	Lib-5	T10KD	"COSMIC"
400122	Lib-4	T10KD	"COSMIC"
400123	Lib-6	T10KD	"None"
400124	Lib-3	T10KD	"None"
400125	Lib-1	T10KD	"None"

tapeinfo

- **Simple:** A couple of hundred lines of python code
- **Portable:** standard interfaces (hpssadm.pl and ACSLS cmd)
- **Scalable:** Runs with thousands of tapes

checkForMigration

- A need to find out which files have not yet been migrated from disk to tape
- When upgrading Linux on movers, wanted to ensure all files had a tape copy
- When something goes wrong with a RAID logical volume, need to know which files and how many are unavailable

checkForMigration

- Example run:

```
# checkForMigration 12345600  
/home/smith/file1 not on tape  
/home/smith/file2 not on tape  
/home/smith/file3 not on tape
```

checkForMigration

- script first runs 'lsvol' to get a listing of files
- script then invokes a C client API program that checks if files have a copy on tape

checkForMigration

- Client API program is 25 lines (including comments):

```
rc = hpss_FileGetXAttributes(path, API_GET_STATS_FOR_LEVEL, 1,
&AttrOut);
if (rc == 0) {
    if (AttrOut.SCAattrib[1].VVAttrib[0].PVList == 0) {
        printf("%s not on tape\n", path);
    }
}
```

checkForMigration

- **Simple:** ~100 lines of code (C and bash) total
- **Portable:** uses client API
- **Scalable:** can check a disk volume with 300,000 segments in ~20 minutes

Nagios

- Open source software for monitoring
- Executes standard and custom health check scripts on remote hosts
- Many alert and reporting features

Nagios

- Used to augment existing tools
- Two components:
 - Code added to existing tools to create a Nagios status file
 - Standard Nagios service check script in libexec to query the status file and report results
- Existing tools continue to run out of root or ACSLS crontabs
- Nagios checks do not require elevated privileges

Nagios – Augmentation Code

```
COUNT=`${GREP} Degraded acsss_event.log|grep -v ^Cannot \  
|wc -l|tr -d " "`
```

```
if [[ "${COUNT}" -gt 0 ]]
```

```
then
```

```
  ${GREP} Degraded acsss_event.log > ${MSG}
```

```
  diff ${MSG} ${DEGFND} 1>/dev/null 2>/dev/null
```

```
  if [[ $? -ne 0 ]]
```

```
  then
```

```
    echo "[CRITICAL] - SL8500 Degraded Components Found!" \  
    > /tmp/ck.degraded.nagios.out
```

```
  fi
```

```
else
```

```
  echo "[OK] - No SL8500 Degraded Components Found." \  
  > /tmp/ck.degraded.nagios.out
```

```
fi
```

Nagios – Service Status Check Code

```
STATUS="/tmp/ck.degraded.nagios.out"
```

```
grep "\[OK\]" ${STATUS} 1>/dev/null 2>&1
```

```
if [[ "$?" -eq "0" ]]
```

```
then
```

```
  cat ${STATUS}
```

```
  exit 0
```

```
fi
```

```
grep "\[CRITICAL\]" ${STATUS} 1>/dev/null 2>&1
```

```
if [[ "$?" -eq "0" ]]
```

```
then
```

```
  cat ${STATUS}
```

```
  exit 2
```

```
fi
```

```
echo "[UNKNOWN] - Status File Missing Or Logic Error!"
```

```
exit 3
```

Nagios

- **Simple:** Uses existing tools with minor modification & trivial Nagios service check code
- **Portable:** Any cron, any language, any tool type, any operating system
- **Scalable:** Nagios service check code leverages existing crontab entries (root, ACSLS, etc.) to minimize performance impact on the servers

Conclusion

- tapeinfo
- checkForMigration
- Nagios



Thanks!

Questions?