IBM Storage & SDI

Securing your data with Spectrum Scale

Christopher Maestas

Spectrum Scale Senior Architect

Firewalls and SELinux

Spectrum Scale - firewall

gpfs 1191/tcp General Parallel File System
gpfs 1191/udp General Parallel File System
# Dave Craft gpfs@ibm.com November 2004

Ports: https://www.ibm.com/support/knowledgecenter/STXKQY_5.0.0/com.ibm.spectrum.scale.v5r00.doc/bl1adv_firewall.htm

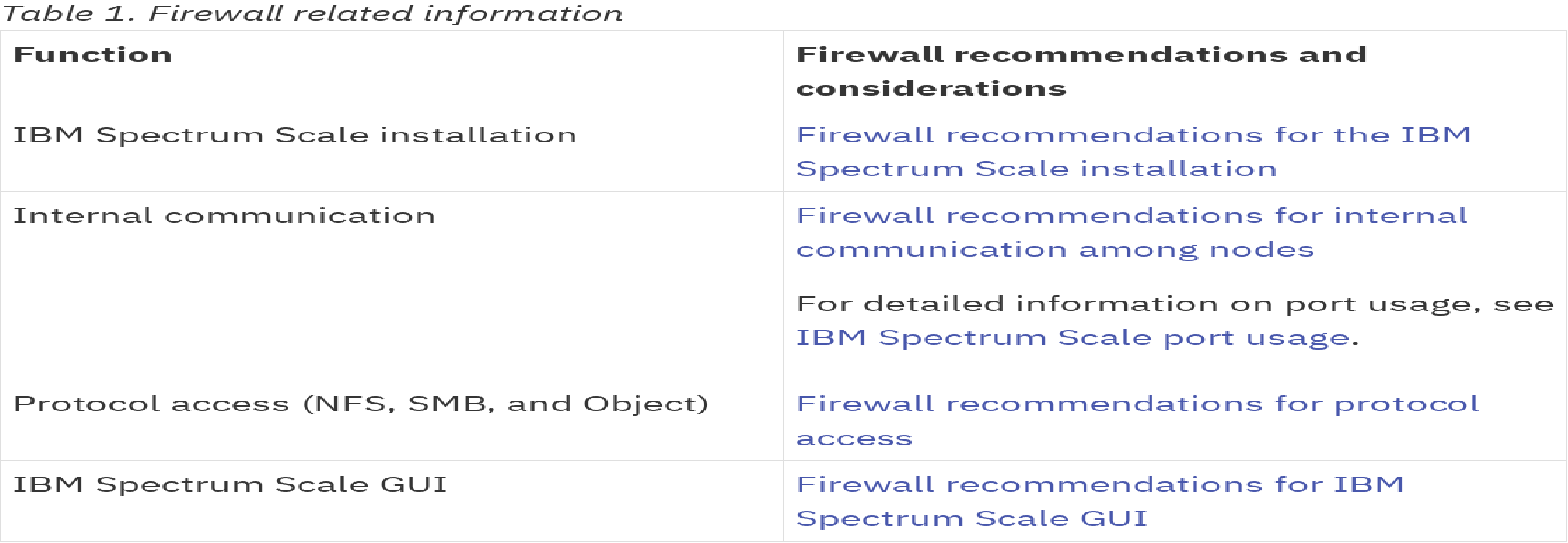


Table 1. Firewall related information

Function	Firewall recommendations and considerations
IBM Spectrum Scale installation	Firewall recommendations for the IBM Spectrum Scale installation
Internal communication	Firewall recommendations for internal communication among nodes For detailed information on port usage, see IBM Spectrum Scale port usage.
Protocol access (NFS, SMB, and Object)	Firewall recommendations for protocol access
IBM Spectrum Scale GUI	Firewall recommendations for IBM Spectrum Scale GUI

Spectrum Scale - SELinux

GPFS V3.5 and later run in

- 'permissive' mode, and
- 'enforcing' mode with 'SELINUXTYPE=targeted'

GPFS commands have to run unconfined

No SELinux profiles supplied for GPFS daemons and utilities

- Running GPFS command in a confined security context may fail
- Result in a large volume of logged security exception events.

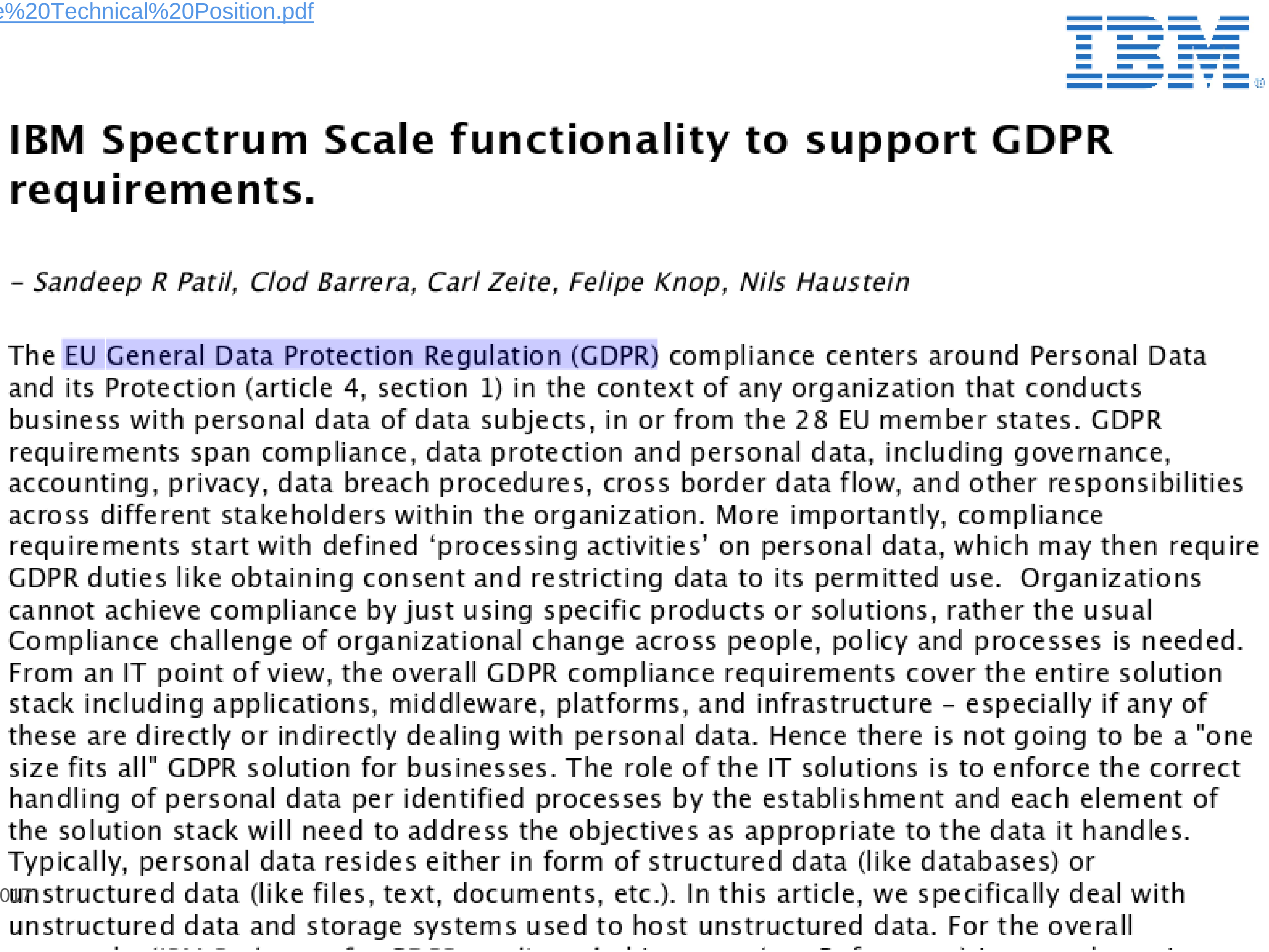
GPFS can hold files with per-inode security labels with limitations

https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/General%20Parallel%20File%20System%20(GPFS)/page/SElinux

EU GDPR

EU General Data Protection Regulation (GDPR)

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/5cb5ed706d254a8186256c71006d2e0a/1d33b61a55b2787185258251004c0566/$FILE/GDPR%20Compliance-%20Spectrum%20Scale%20Technical%20Position.pdf



IBM Spectrum Scale functionality to support GDPR requirements.

– Sandeep R Patil, Clod Barrera, Carl Zeite, Felipe Knop, Nils Haustein

The EU General Data Protection Regulation (GDPR) compliance centers around Personal Data and its Protection (article 4, section 1) in the context of any organization that conducts business with personal data of data subjects, in or from the 28 EU member states. GDPR requirements span compliance, data protection and personal data, including governance, accounting, privacy, data breach procedures, cross border data flow, and other responsibilities across different stakeholders within the organization. More importantly, compliance requirements start with defined ‘processing activities’ on personal data, which may then require GDPR duties like obtaining consent and restricting data to its permitted use. Organizations cannot achieve compliance by just using specific products or solutions, rather the usual Compliance challenge of organizational change across people, policy and processes is needed. From an IT point of view, the overall GDPR compliance requirements cover the entire solution stack including applications, middleware, platforms, and infrastructure – especially if any of these are directly or indirectly dealing with personal data. Hence there is not going to be a "one size fits all" GDPR solution for businesses. The role of the IT solutions is to enforce the correct handling of personal data per identified processes by the establishment and each element of the solution stack will need to address the objectives as appropriate to the data it handles. Typically, personal data resides either in form of structured data (like databases) or unstructured data (like files, text, documents, etc.). In this article, we specifically deal with unstructured data and storage systems used to host unstructured data. For the overall

Immutability – WORM

Spectrum Scale immutability - certified for compliance

The immutability function in IBM Spectrum Scale Version 4.2 has been assessed for compliance in accordance to **US SEC17a-4f** rules, **German and Swiss laws and regulations** by a recognized auditor.

KPMG

Review of the software IBM Spectrum Scale version 4.2

REPORT

International Business Machines Corporation
Armonk, NY

August 2016

Assessment report: http://www.kpmg.de/bescheinigungen/RequestReport.aspx?41742

Certificate: https://www.kpmg.de/bescheinigungen/RequestReport.aspx?41743

Immutability Overview

Immutability means preventing changes and deletion of files during retention time

Spectrum Scale Immutability provides WORM storage in GPFS fileset

- Immutable files cannot be changed or deleted during retention period
  - Deletion is possible when retention time is expired

Managing immutability works similar to other products

- Retention time can be set with last access date
- WORM protection can be set by removing write permission

Spectrum Scale also supports append-only mode

- An empty file can be set to append-only by removing and adding write permission
- Append-only file allows appends at the end
- Append-only file can be made immutable by removing write permission once again

Fileset Immutability Archive Manager Mode

none: Default setting for a normal fileset

advisory (ad): Allows setting retention times and WORM protection

- But files can be deleted with the proper permission

noncompliant (nc): Advisory mode plus

- Files cannot be deleted if retention time is not expired.
- But retention times can be reset and files can be deleted but not changed

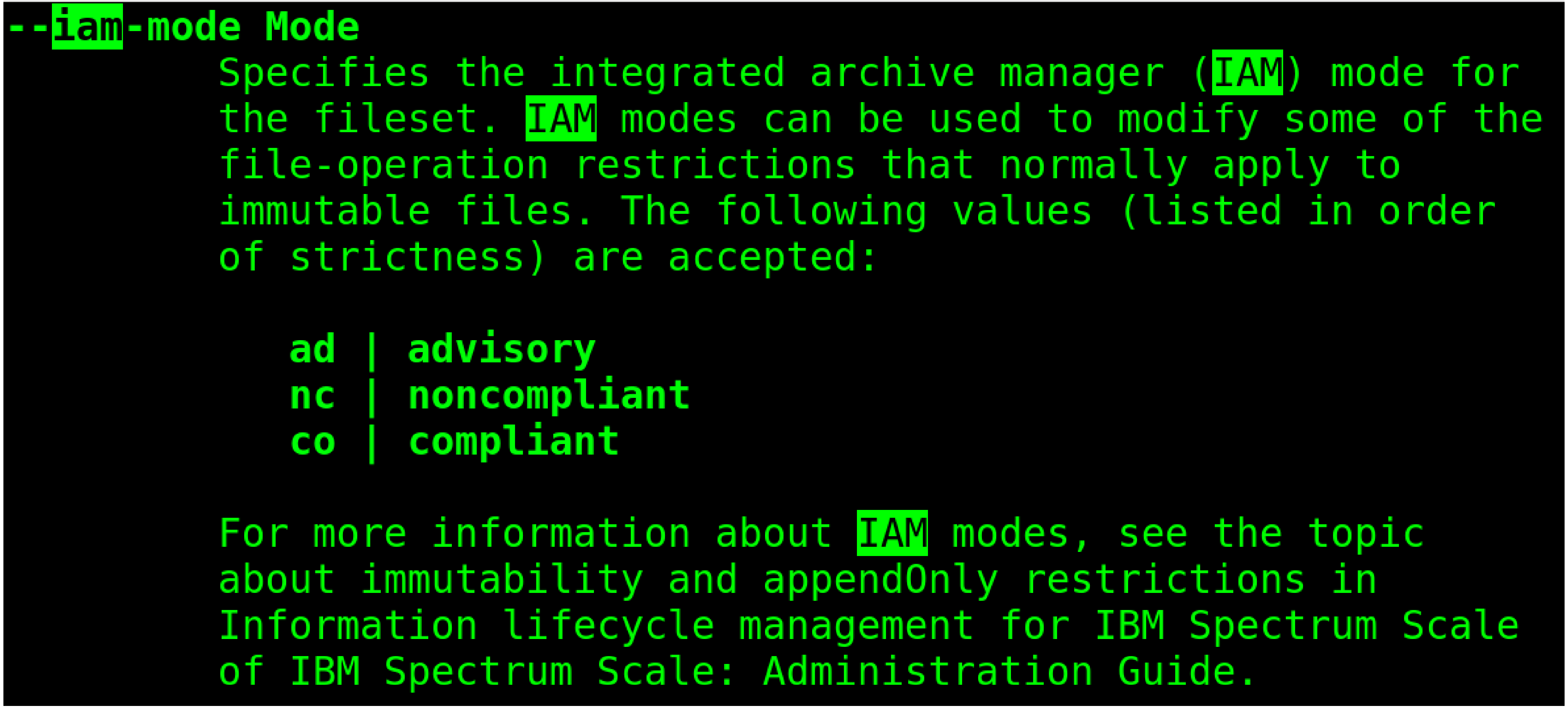
compliant (co): noncompliant mode plus

- Retention time cannot be reset.
- When retention time has expired files can be deleted but not changed

Modes can be upgraded, but not downgraded

To set IAM use command: mmchfileset–iam-mode

Look a man page! mmchfileset



```
--iam-mode Mode
        Specifies the integrated archive manager (IAM) mode for
        the fileset. IAM modes can be used to modify some of the
        file-operation restrictions that normally apply to
        immutable files. The following values (listed in order
        of strictness) are accepted:

           ad | advisory
           nc | noncompliant
           co | compliant

        For more information about IAM modes, see the topic
        about immutability and appendOnly restrictions in
        Information lifecycle management for IBM Spectrum Scale
        of IBM Spectrum Scale: Administration Guide.
```

Set commands

Setting retention time for file

- **touch –at MMddhhmmss filename**
- **mmchattr –E yyyy-mm-dd[@hh:mm:ss] filename**

Setting file immutable

- **chmod –w filename**
- **mmchattr –i yes filename**

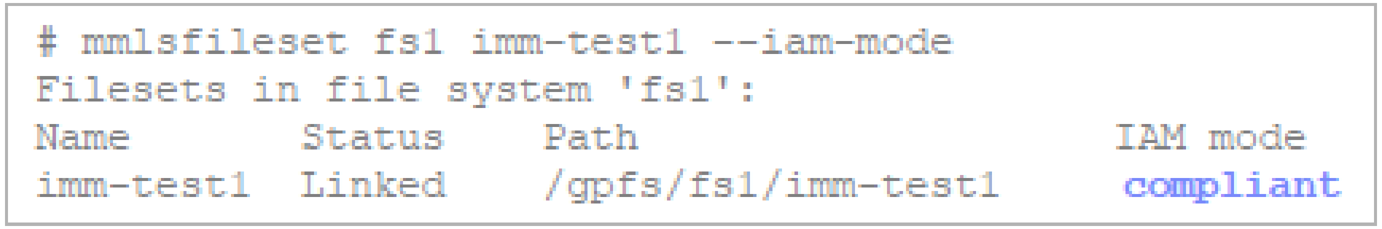
Setting file to append-only

- **Create Empty file**
  - **chmod –w filename; chmod +w filename**
  - **mmchattr –a yes**

Showing commands

View fileset immutability mode

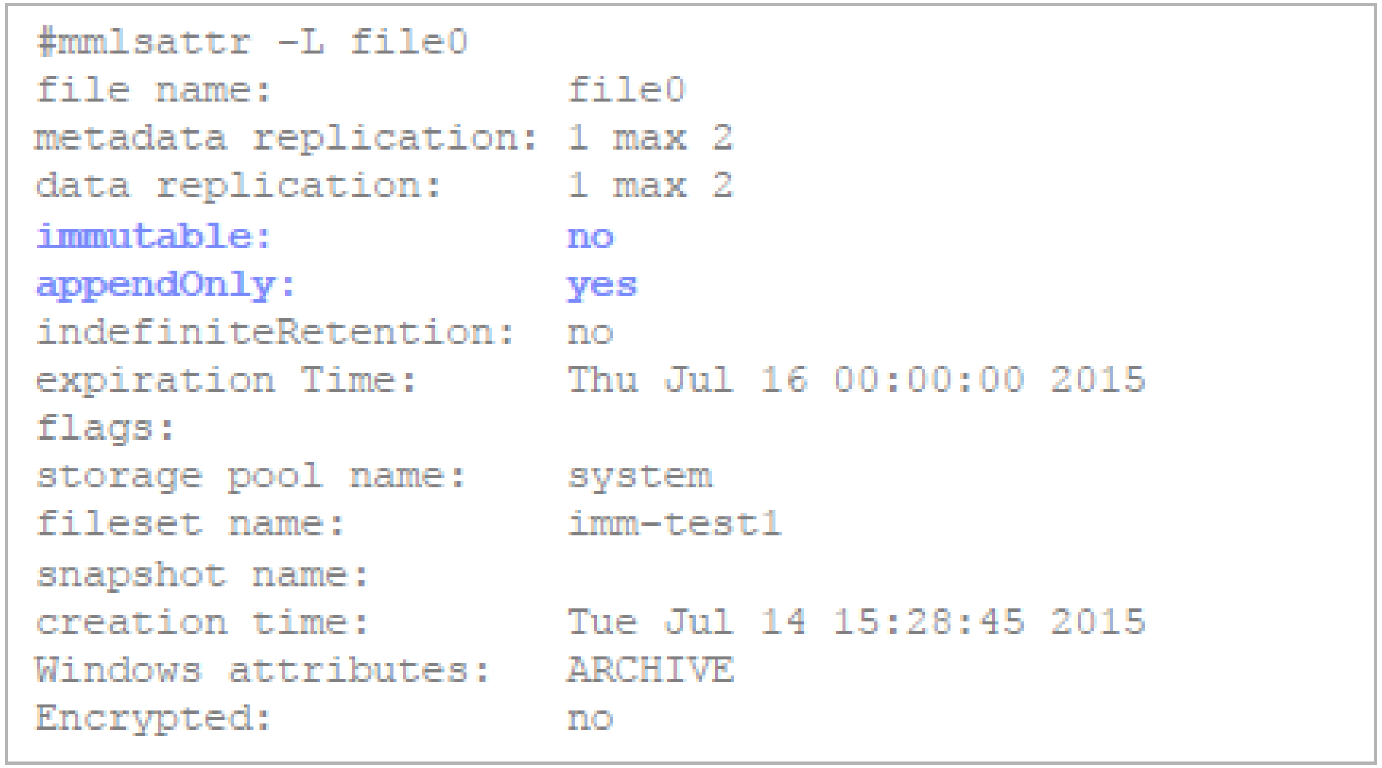
mmlsfilesetfsfset –iam-mode



```
# mmlsfileset fs1 imm-test1 --iam-mode
Filesets in file system 'fs1':
Name         Status     Path                       IAM mode
imm-test1    Linked     /gpfs/fs1/imm-test1         compliant
```

Show file immutability setting

mmlsattr –L filename



```
#mmlsattr -L file0
file name:              file0
metadata replication:   1 max 2
data replication:       1 max 2
immutable:              no
appendOnly:             yes
indefiniteRetention:    no
expiration Time:        Thu Jul 16 00:00:00 2015
flags:
storage pool name:      system
fileset name:           imm-test1
snapshot name:
creation time:          Tue Jul 14 15:28:45 2015
Windows attributes:     ARCHIVE
Encrypted:              no
```

Additional functions and options

Deletion of file systems with compliant filesets (mmdelfs)

- Cluster-wide configuration parameter “indefiniteRetentionProtection” prevents this
  - Once set to yes deletion of file system is no longer possible
  - Cannot be set back to no once set to yes

Deletion of compliant filesets (mmdelfileset)

- Not possible at GPFS 4.2 and higher

Backup and restore using mmbackup

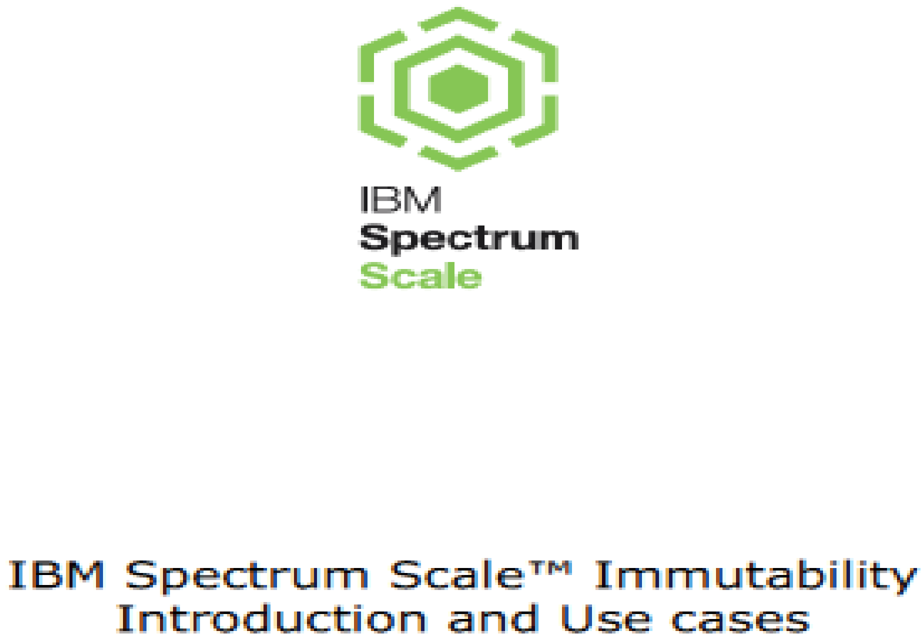
- Works with Spectrum Protect B/A client 7.1.3 and above
- In-place restore cannot overwrite and existing immutable file
- Out-of-place restore does not set the immutability attribute and retention time
  - Last access data will reflect retention time

Spectrum Protect for Space Management 7.1.4 and above supports this

Recommended reading

Spectrum Scale Immutability Whitepaper:

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102620



IBM Spectrum Scale

IBM Spectrum Scale™ Immutability
Introduction and Use cases

Encryption

Native encryption and secure erase

Native: encryption is built into the **"Advanced/Data Management Edition"** product

Protects data from security breaches, unauthorized access, and being lost, stolen or improperly discarded

Supports ISKLM and Vormetric key managers

Cryptographic erase for fast, simple and secure file deletion

Complies with **NIST SP 800-131A** and is **FIPS 140-2** certified

Supports HIPAA, Sarbanes-Oxley, EU and national data privacy law compliance

Native encryption and secure erase

Encryption of data at rest

Files are encrypted before they are stored on disk

Keys are never written to disk

No data leakage in case disks are stolen or improperly decommissioned

Secure deletion

Ability to destroy arbitrarily large subsets of a file system

No “digital shredding”, no overwriting: secure deletion is a cryptographic operation

Key-based encryption

Master Encryption Key (MEK)

Used to encrypt file encryption keys

Stored in Remote Key Management (RKM) Servers

MEK’s have a unique key name that combines the name of the key and the RKM server where it resides

File Encryption Key (FEK)

Used to encrypt sectors of an individual file

Unique key randomly generated

Encrypted (or “wrapped”) with one or more MEK’s and stored in the gpfs.

FEK must have access to MEK to be decoded

FEK can be re-wrapped to new MEK(s) in the case of a compromised key

Native encryption

Files encrypted before I/O submission

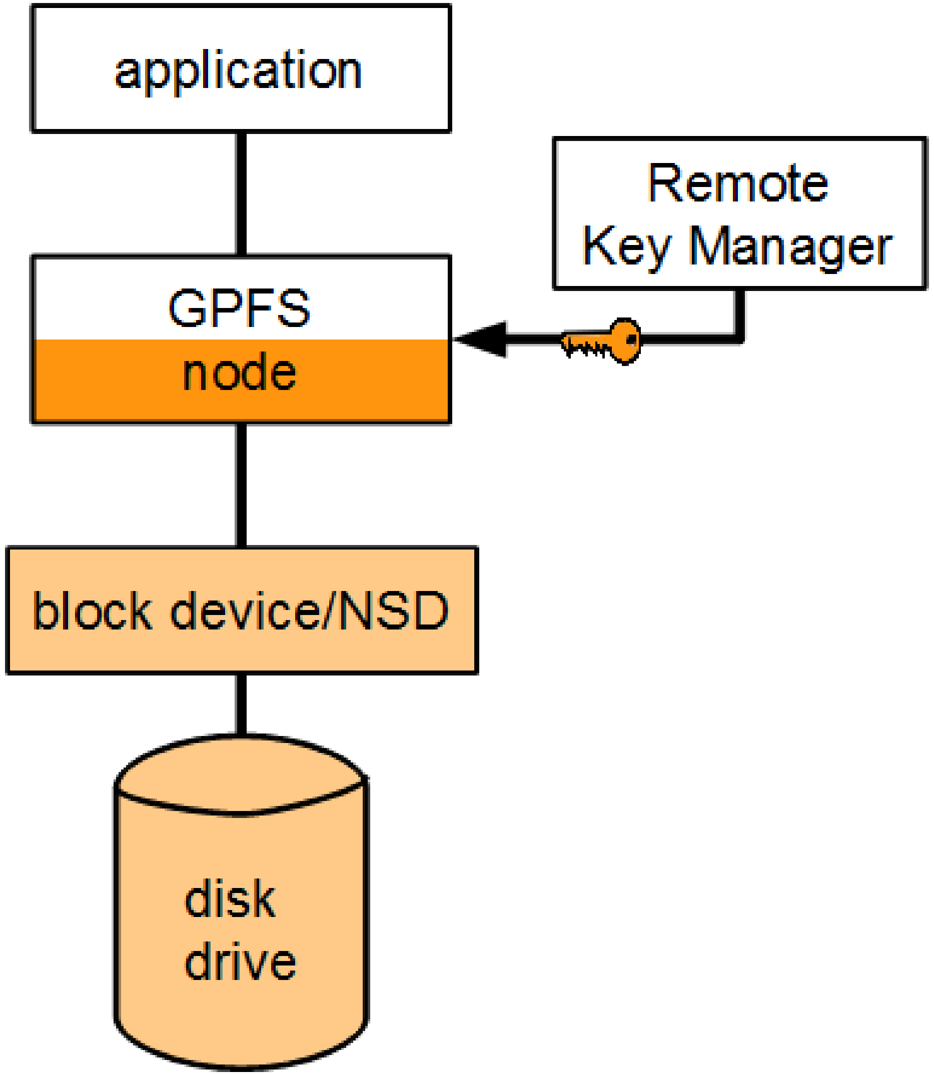
Encryption takes place on the node(s) from which the user drives the I/O

File content travels encrypted to the NSD server

Keys can be accessed by nodes that have appropriate RKM credentials

Nodes that cannot access keys cannot access files, irrespective of file permissions

Granularity is per file or per file set, as determined by encryption policies



Encryption Policies

Manage how files are encrypted and includes the following:

Which files are to be encrypted

Which algorithm is to be used for encryption

Which MEK (or MEK’s) are to be used to wrap the FEK of a file

The **mmchpolicy** command is used to configure encryption and is applied at file creation time

When a file is created, encryption rules are executed in order until the following occurs:

- The last rule is reached
- The maximum number of SET ENCRYPTION rules that can be matched (eight) is reached
- An ENCRYPTION EXCLUDE rule is matched

Encryption policies

If the file matches at least one SET ENCRYPTION rule, an FEK is generated and used to encrypt the contents of the file.

The FEK is then wrapped once for each policy it matches

Things to keep in mind:

- When an encryption policy is changed, the changes apply only to files created after the policy has been changed
- Encryption policies are defined on a per-file system basis by a system administrator.
  - Filesets can have different encryption keys and policies

Encryption policy rule syntax

The ENCRYPTION rule is used to specify how a file is to be encrypted and how the FEK is to be wrapped.

The syntax of the **ENCRYPTION IS** rule is:

```
RULE ’RuleName’ ENCRYPTION
’EncryptionSpecificationName’ IS
ALGO ’EncParamString’
COMBINE ’CombineParamString’
WRAP ’WrapParamString’
KEYS (’Keyname’[, ’Keyname’, ... ])
```

Encryption policy rule syntax

The **SET ENCRYPTION** rule is similar to the **SET POOL** rule

- If more than one **SET ENCRYPTION** rule is present, all will be considered and the FEK wrapped for each of the rules up to eight rules
- If an FEK is wrapped multiple times, only one of the wrapped FEK instances need to be unwrapped for file access
- If **no SET ENCRYPTION** rule is applicable at file creation time, the file is not encrypted.

The **SET ENCRYPTION** rule syntax is:

RULE ’RuleName’ **SET ENCRYPTION** ’EncryptionSpecificationName’[, ’EncryptionSpecificationName’,...]
[**FOR FILESET** (’FilesetName’[,’FilesetName’]...)]
[**WHERE** SqlExpression]

Encryption policy rule example

```
RULE ’myEncRule1’ ENCRYPTION ’E1’ IS
ALGO ’DEFAULTNISTSP800131A’
KEYS(’1:RKM_1’, ’2:RKM_2’)

RULE ’myEncRule2’ ENCRYPTION ’E2’ IS
ALGO ’AES:256:XTS:FEK:HMACSHA512’
COMBINE ’XOR’
WRAP ’AES:KWRAP’
KEYS(’3:RKM_1’)

RULE ’myEncRule3’ ENCRYPTION ’E3’ IS
ALGO ’AES:128:CBC:FEK:HMACSHA512’
COMBINE ’XORHMACSHA512’
WRAP ’AES:CBCIV’
KEYS(’4:RKM_2’)
```

```
RULE ’Do not encrypt files with extension enc4’
SET ENCRYPTION EXCLUDE
FOR FILESET(’fs1’)
WHERE NAME LIKE ’%.enc4’

RULE ’Encrypt files with extension enc1 with rule E1’
SET ENCRYPTION ’E1’
FOR FILESET(’fs1’)
WHERE NAME LIKE ’%.enc1’

RULE ’Encrypt files with extension enc2 with rule E2’
SET ENCRYPTION ’E2’
FOR FILESET(’fs1’)
WHERE NAME LIKE ’%.enc2’

RULE ’Encrypt files with extension enc* with rule E3’
SET ENCRYPTION ’E3’
FOR FILESET(’fs1’)
WHERE NAME LIKE ’%.enc%’
```

Secure erase

Cannot be achieved with standard methods:

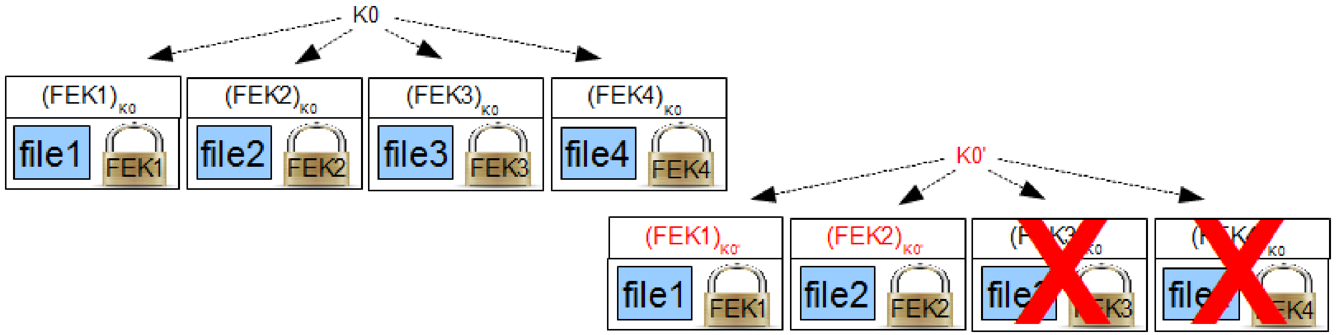
- unlink() leaves data on disk,
- overwriting is cumbersome and may not work (e.g. SSD)

Secure **Cryptographic Erase**

- When MEK is deleted, encrypted FEK is no longer retrievable
- Hence, file cannot be decrypted
- Regardless of cached copies, snapshots, backups, ...

Two-step operation

- Files are deleted with standard file system operations (e.g. rm, unlink...)
- Secure deletion committed with key management operation
  - Registration of new MEK
  - Re-encryption of FEKs that “need to stay”
  - Deletion of old MEK



Secure erase and mmdelfs

The **mmdelfs** command

not perform any secure deletion of files on its own

only removes all the structures for the file system

To securely delete files, the following steps must be performed:

Identify all MEK’s currently used to wrap FEK’s of files in the file system. This information can be obtained by:

- Invoke mmlsattr –n gpfs or through a policy
- Parse the resulting output to extract all the distinct MEK key names that are used

Determine whether MEK’s were used in other file systems

- **NOTE: If the same MEK’s were used to wrap FEK’s in other file systems, deleting those MEK’s will result in irreparable data loss in the other file systems. Before deleting such MEK’s, new MEK’s must be created and the FEK’s of the other file system rewrapped with the new MEK’s before the old MEK’s can be deleted.**

Delete the identified MEK’s from the RKM servers

File Audit Logging

Improved security and compliance

New File Audit Logging capability
(Data Management Edition only)

Track user accesses to filesystem and events

Supported across all nodes and all protocols

Parseable data stored in secure retention-protected fileset

Events that can be captured are:

Open, Close, Destroy (Delete), Rename, Unlink, Remove Directory, Extended Attributed Change, Access Control List (ACL) change

FAL - history

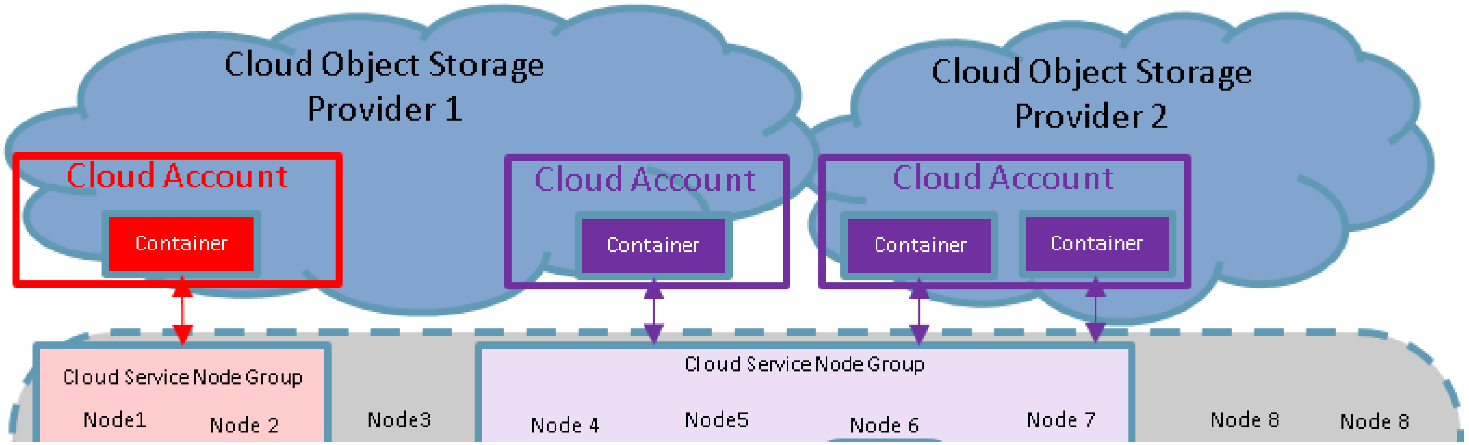
Integration with audit tools like Varonis and IBM Guardium

http://www.redbooks.ibm.com/redpapers/pdfs/redp5426.pdf

https://www.ibm.com/support/knowledgecenter/en/STXKQY_4.2.2/com.ibm.spectrum.scale.v4r22.doc/bl1adv_dpauditlogging.htm

Uses Light Weight Events (LWE) – What uses this today?

Transparent Cloud Tiering - TCT



Sample Audit POC Tasks

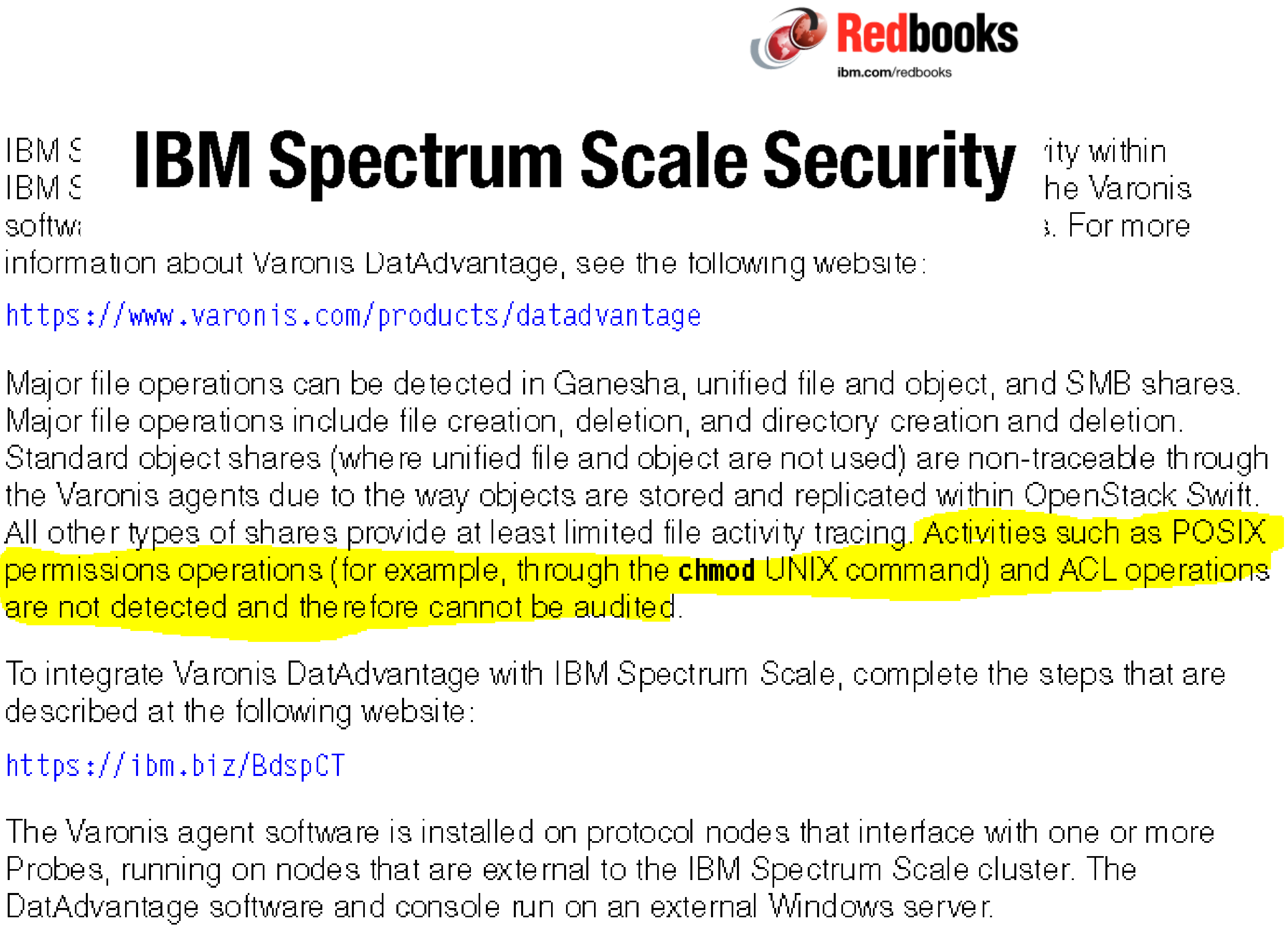
Demonstrate monitoring of file activity including user name, timestamp, and file location regardless of client type

Demonstrate monitoring of file activity without **endpoint** agent on clients

Create CSV-formatted reports of file activity and directory activity

Create report containing variable days of activity and deliver via file system, email, and api

Audit logging with Varonis DatAdvantage



Redbooks
ibm.com/redbooks

IBM Spectrum Scale Security

IBM S ity within
IBM S he Varonis
softwa . For more
information about Varonis DatAdvantage, see the following website:

https://www.varonis.com/products/datadvantage

Major file operations can be detected in Ganesha, unified file and object, and SMB shares. Major file operations include file creation, deletion, and directory creation and deletion. Standard object shares (where unified file and object are not used) are non-traceable through the Varonis agents due to the way objects are stored and replicated within OpenStack Swift. All other types of shares provide at least limited file activity tracing. Activities such as POSIX permissions operations (for example, through the **chmod** UNIX command) and ACL operations are not detected and therefore cannot be audited.

To integrate Varonis DatAdvantage with IBM Spectrum Scale, complete the steps that are described at the following website:

https://ibm.biz/BdspCT

The Varonis agent software is installed on protocol nodes that interface with one or more Probes, running on nodes that are external to the IBM Spectrum Scale cluster. The DatAdvantage software and console run on an external Windows server.

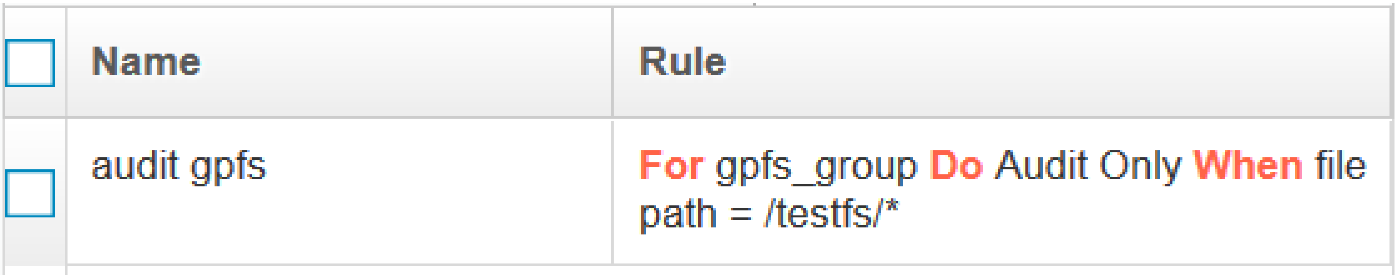
Spectrum Scale Testing with IBM Guardium

9 node cluster

Traffic

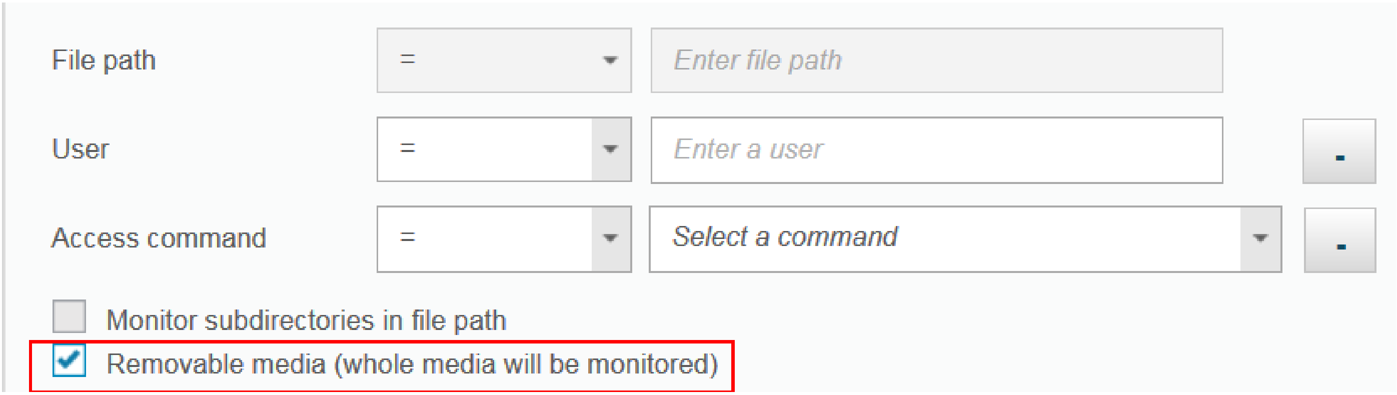
- FVT I/O Stress tests (autotest, mkfiles)
- Command Regression (as root)

STAPs installed on each node



□	Name	Rule
□	audit gpfs	**For** gpfs_group **Do** Audit Only **When** file path = /testfs/*

Audit only policy right now



File path = Enter file path

User = Enter a user

Access command = Select a command

□ Monitor subdirectories in file path

☑ Removable media (whole media will be monitored)

Audit removable media for NFS

Spectrum Scale Testing with IBM Guardium

What do we catch

Commands

DELETE

READ

WRITE

- Create file thru vi shows as a write
- We catch data in inode
- CREATE system call shows up as a WRITE

EXEC (Execution)

FILEOP (MKDIR, CHMOD, CHOWN)

Source Program

Db_user

OS-User

Object

What do we not catch

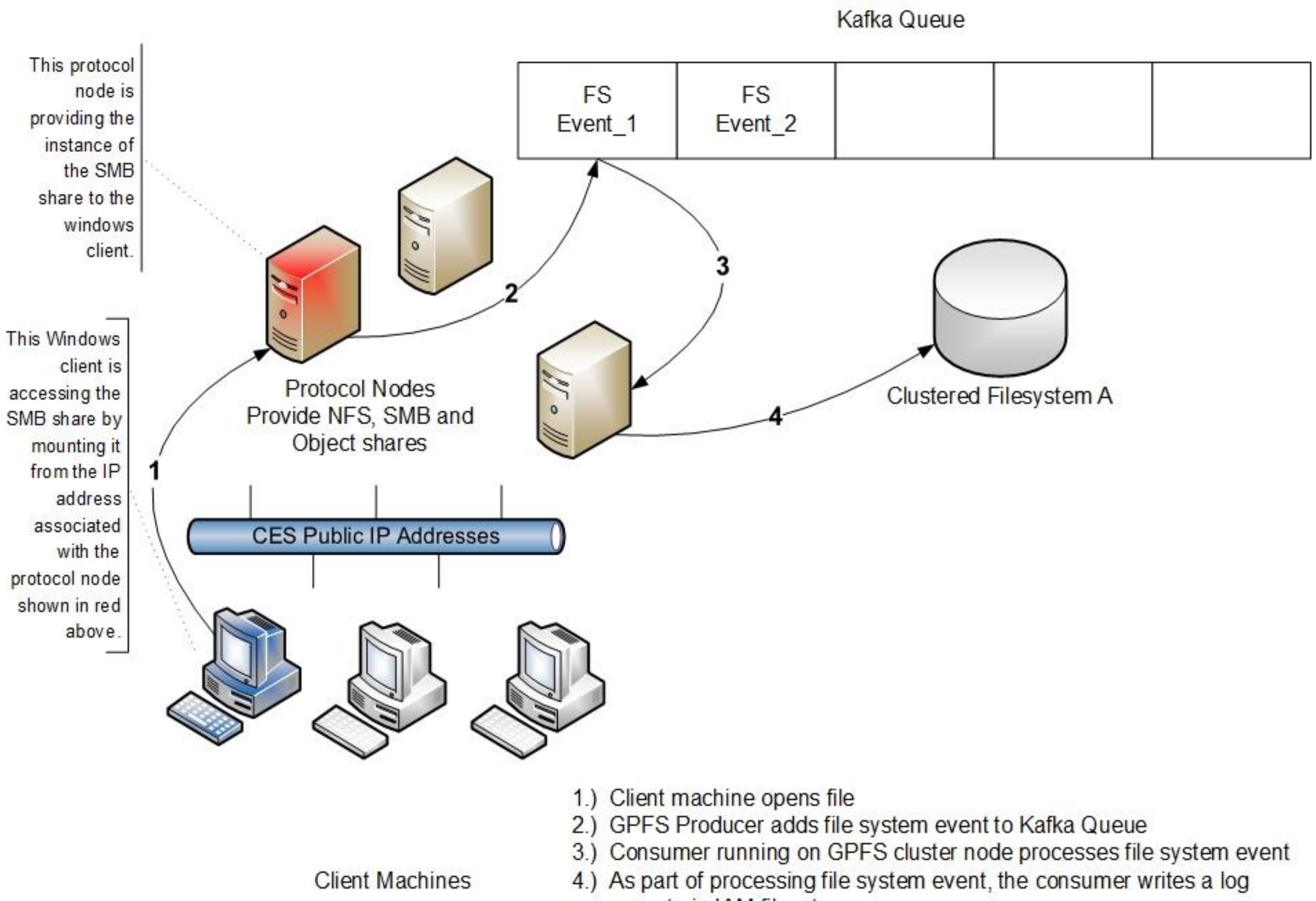
GPFS administration commands like:

```
mmchattr –P sp1 /testfs/subdir/*
#This changes the extended attributes of a file (root only)

mmapplypolicy /testfs/subdir -P mig.pol

# migrates data between storage pools (root only)
```

To monitor root

- In guard_tap.ini file add : fam_protect_privileged=1

High Level Flow


Kafka Queue
FS Event_1
FS Event_2
This protocol node is providing the instance of the SMB share to the windows client.
This Windows client is accessing the SMB share by mounting it from the IP address associated with the protocol node shown in red above.
2
3
1
4
Protocol Nodes Provide NFS, SMB and Object shares
Clustered Filesystem A
CES Public IP Addresses
Client Machines

1.) Client machine opens file
2.) GPFS Producer adds file system event to Kafka Queue
3.) Consumer running on GPFS cluster node processes file system event
4.) As part of processing file system event, the consumer writes a log message to in IAM fileset

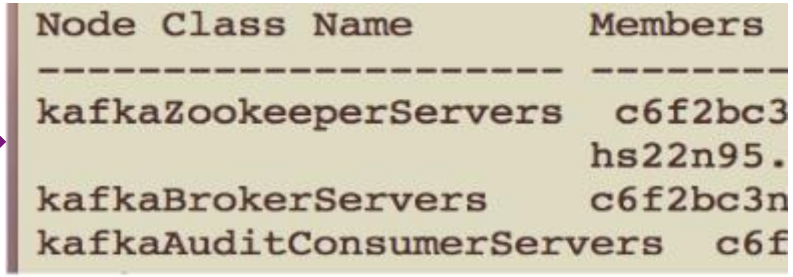
File Audit Logging (FAL)

Now an API for 3rd party software IBM Guardium and Varonis

Light Weight Events (LWE) with Apache Kafka

- **Producer** to publish stream of records: *1 million msg/s*
  - Live inside mmfsd (gpfs) daemon
- **Consumer** subscribe to one or more topics and process stream:
  - *3 million msg/s*

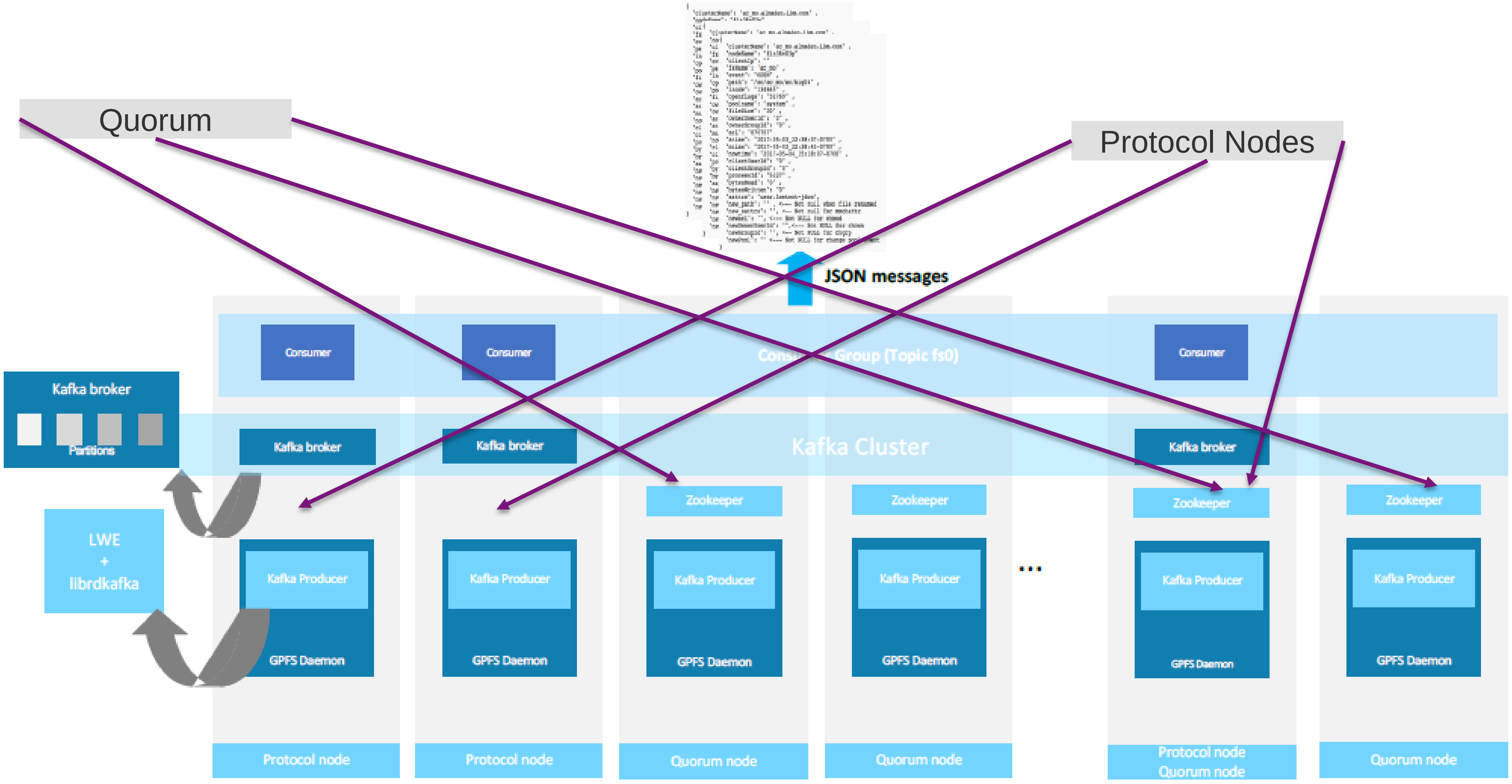
node classes – minimum of 3



```
Node Class Name          Members
----------------------- --------
kafkaZookeeperServers    c6f2bc3
                         hs22n95.
kafkaBrokerServers       c6f2bc3n
kafkaAuditConsumerServers  c6f
```

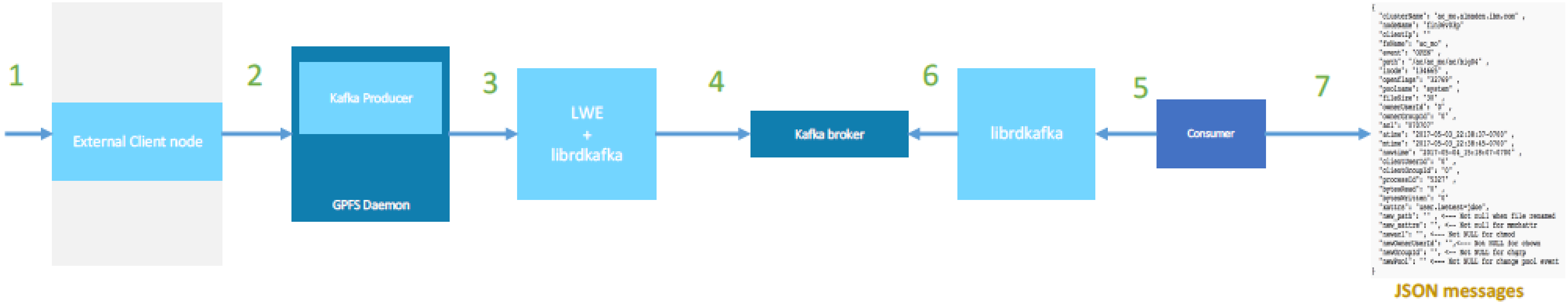
Monitor via CLI, mmhealth ,logfile, msgqueue or GUI (Events panel)!

FAL - Architecture



*Zookeeper resides on the quorum nodes
**Kafka Brokers can reside on any node (not confined to protocol nodes as depicted in this figure)
***Using the standardized JSON format, client facing API can be derived.

FAL – event flow



```
{
 "clusterName": "[illegible]" ,
 "nodeName": "[illegible]"
 "clientIp": ""
 "fsName": "[illegible]" ,
 "event": "OPEN" ,
 "path": "[illegible]" ,
 "inode": "[illegible]" ,
 "openFlags": "[illegible]" ,
 "poolName": "system" ,
 "fileSize": "[illegible]" ,
 "ownerUserId": "0" ,
 "ownerGroupId": "0" ,
 "acl": "[illegible]"
 "atime": "2017-05-03_22:[illegible]-0700" ,
 "ctime": "2017-05-03_22:[illegible]-0700" ,
 "eventTime": "2017-05-04_[illegible]-0700" ,
 "clientUserId": "0" ,
 "clientGroupId": "0" ,
 "processId": "[illegible]" ,
 "bytesRead": "0" ,
 "bytesWritten": "0"
 "xattrs": "user.[illegible]",
 "new_path": "" , <---- Not null when file renamed
 "new_xattrs": "", <-- Not null for [illegible]
 "newacl": "", <--- Not NULL for chmod
 "newOwnerUserId": "",<---- Not NULL for chown
 "newGroupId": "", <-- Not NULL for chgrp
 "newPool": "" <--- Not NULL for change pool event
}
```

SeqNbr	Description
1	Client performs a file operation (read/ write/ remove, ..) on a file in an audited filesystem
2	External client node sends the client request to the relevant gpfs-node
3	Gpfs daemon using internal LWE (lightweight events) machinery sends the events to the Kafka MsgQueue using librdkafka
4	Event messages are reliably delivered to the Kafka Broker listening on this topic.

SeqNbr	Description
5, 6	Consumers belonging to a consumerGroup listening on this event topic, will periodically pull events from the Kafka Broker queue via librdkafka
7	Consumers will write the consumed events from the MsgQueue into the audited filesystem's ".audit_log" fileset.

Install and configuration

Only Linux nodes (RHEL and Ubuntu)

Linux Kernel version above > 3.10

Minimum of 3 Linux quorum nodes

Minimum of 3 nodes must be designated as Broker nodes

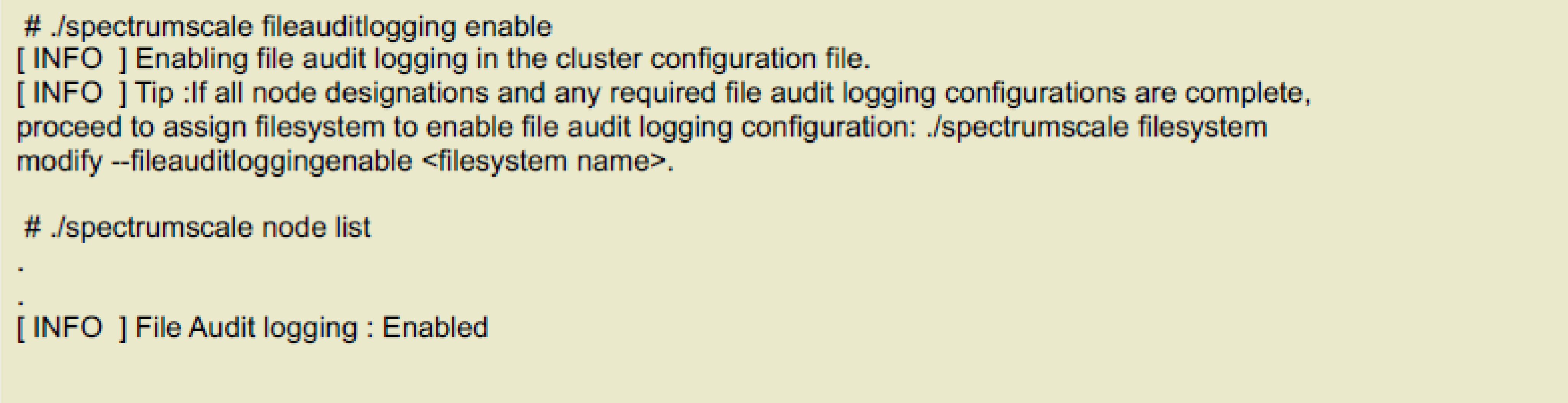
Supported hardware platforms
(x86 and PPCLE)
RHEL supported on x86 and PPC LE
Ubuntu is only supported on x86

Advanced License edition or the Data Management edition

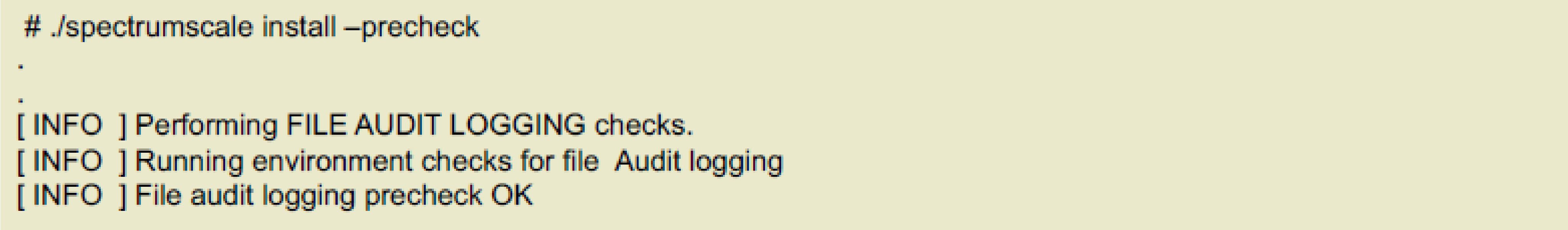
During Installation, most configuration is automatically done and stored in /opt/kafka folder

Free space requirements
>1 GB local disk space per file system being audited
> 2 GB local disk space per file system being audited on all broker nodes

Installation

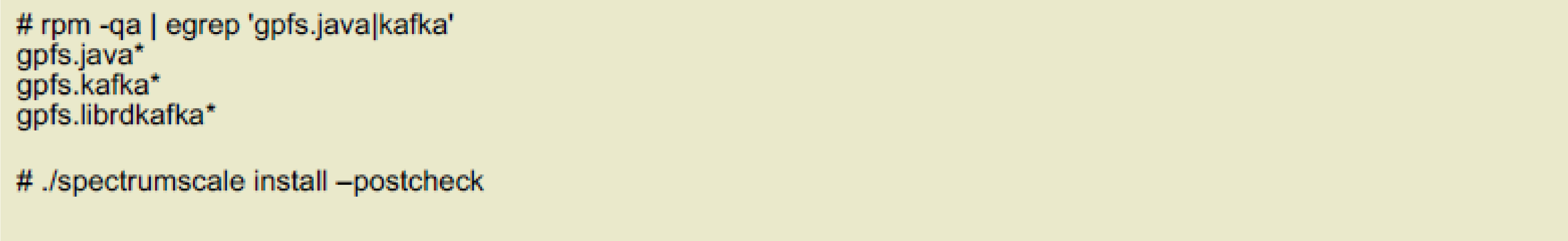


```
 # ./spectrumscale fileauditlogging enable
[ INFO  ] Enabling file audit logging in the cluster configuration file.
[ INFO  ] Tip :If all node designations and any required file audit logging configurations are complete,
proceed to assign filesystem to enable file audit logging configuration: ./spectrumscale filesystem
modify --fileauditloggingenable <filesystem name>.

 # ./spectrumscale node list
.
.
[ INFO  ] File Audit logging : Enabled
```



```
 # ./spectrumscale install –precheck
.
.
[ INFO  ] Performing FILE AUDIT LOGGING checks.
[ INFO  ] Running environment checks for file  Audit logging
[ INFO  ] File audit logging precheck OK
```

After install completes, verify that install installed the necessary GPFS rpms



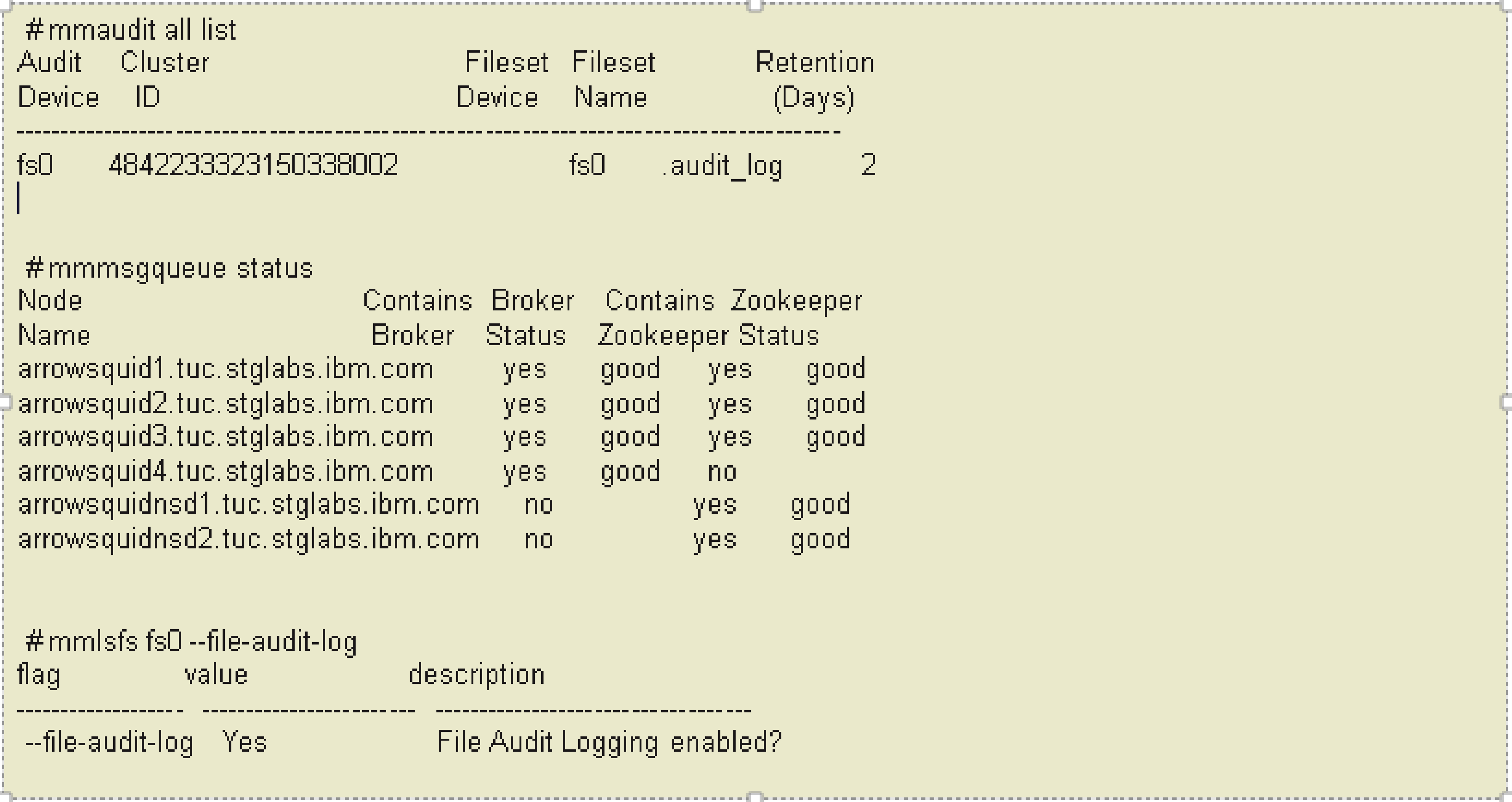
```
# rpm -qa | egrep 'gpfs.java|kafka'
gpfs.java*
gpfs.kafka*
gpfs.librdkafka*

# ./spectrumscale install –postcheck
```

Installation and verification

Validate using mm-CLI commands to ensure file audit logging is enabled

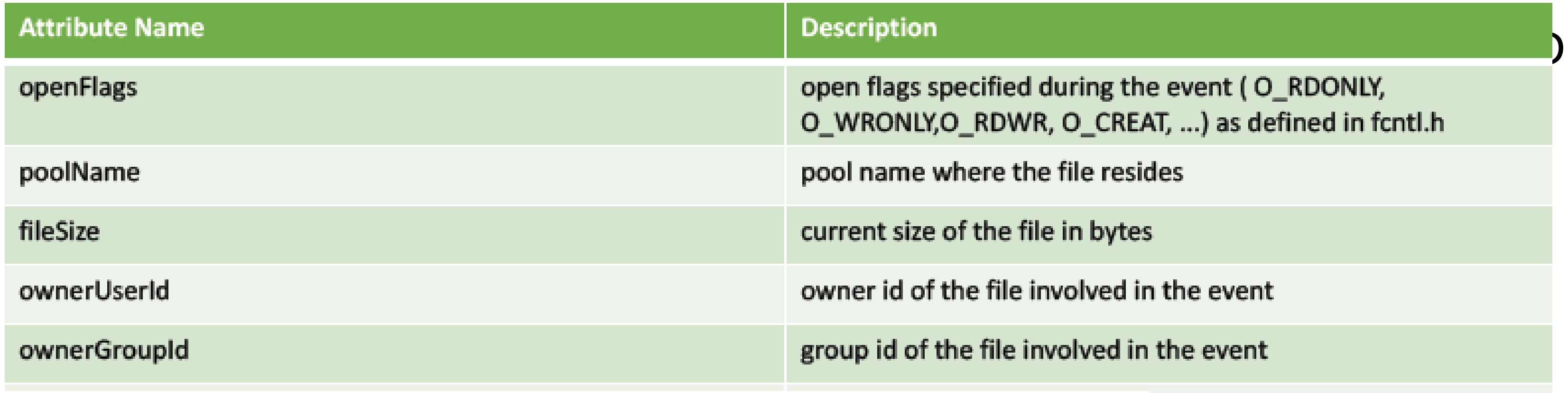
Durir

```
# ./spect
[ INFO  ]
[ INFO  ]
 './spectr
[ INFO  ]
[ INFO  ]
```

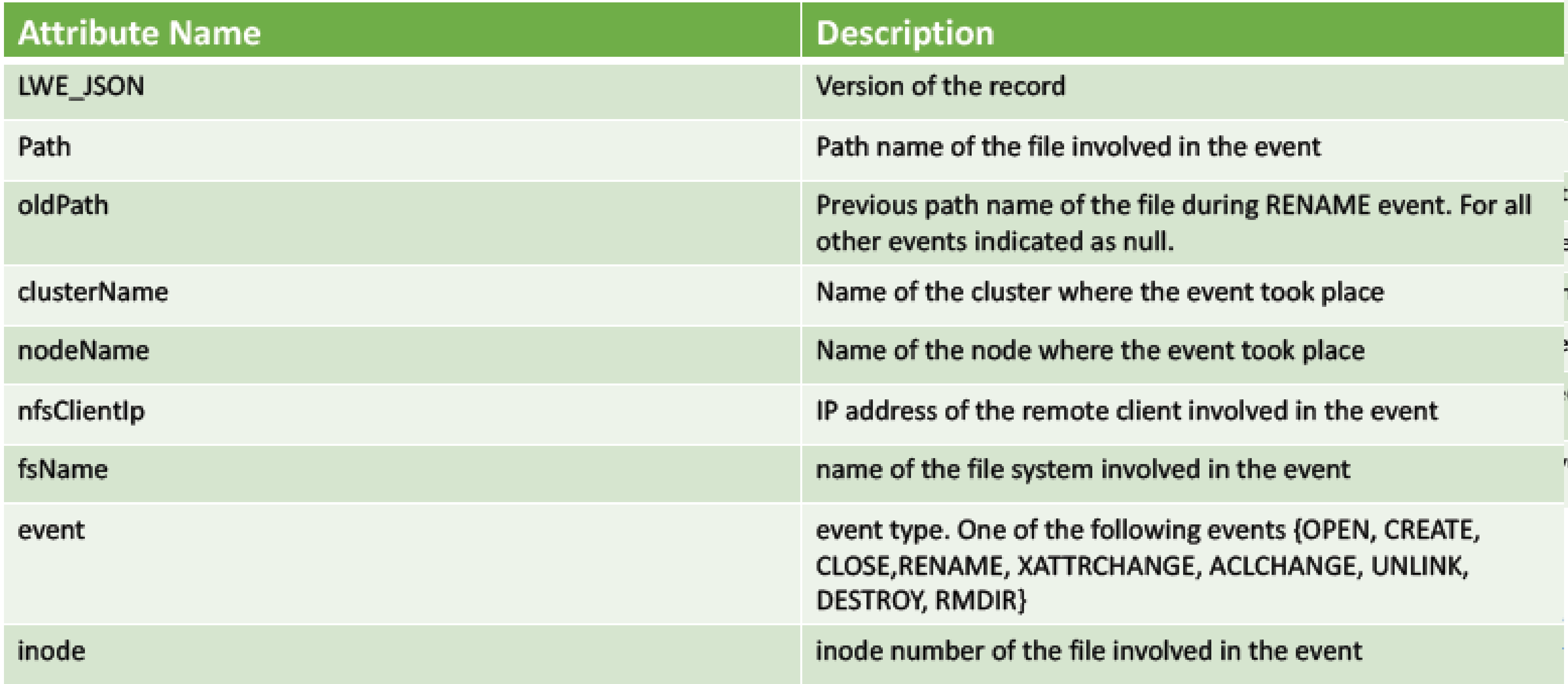


```
 #mmaudit all list
Audit     Cluster                          Fileset   Fileset           Retention
Device     ID                             Device    Name              (Days)
-------------------------------------------------------------------------------------
fs0      4842233323150338002              fs0      .audit_log         2
|

 #mmmsgqueue status
Node                              Contains  Broker    Contains  Zookeeper
Name                               Broker    Status   Zookeeper Status
arrowsquid1.tuc.stglabs.ibm.com       yes      good      yes      good
arrowsquid2.tuc.stglabs.ibm.com       yes      good      yes      good
arrowsquid3.tuc.stglabs.ibm.com       yes      good      yes      good
arrowsquid4.tuc.stglabs.ibm.com       yes      good      no
arrowsquidnsd1.tuc.stglabs.ibm.com     no                yes      good
arrowsquidnsd2.tuc.stglabs.ibm.com     no                yes      good


 #mmlsfs fs0 --file-audit-log
flag               value                    description
------------------- ------------------------ ---------------------------------
 --file-audit-log    Yes                    File Audit Logging enabled?
```

What is logged



Attribute Name	Description
openFlags	open flags specified during the event (O_RDONLY, O_WRONLY,O_RDWR, O_CREAT, ...) as defined in fcntl.h
poolName	pool name where the file resides
fileSize	current size of the file in bytes
ownerUserId	owner id of the file involved in the event
ownerGroupId	group id of the file involved in the event
	last access of the file involved in
	last status change of the file
	event
	the event
	ed in the event
	nt
	ed in the event
	d in the event (Only in case of acl
	ed in the event (Only in case of an



Attribute Name	Description
LWE_JSON	Version of the record
Path	Path name of the file involved in the event
oldPath	Previous path name of the file during RENAME event. For all other events indicated as null.
clusterName	Name of the cluster where the event took place
nodeName	Name of the node where the event took place
nfsClientIp	IP address of the remote client involved in the event
fsName	name of the file system involved in the event
event	event type. One of the following events {OPEN, CREATE, CLOSE,RENAME, XATTRCHANGE, ACLCHANGE, UNLINK, DESTROY, RMDIR}
inode	inode number of the file involved in the event

What gets Monitored

Acquire most common types of file activity:

- **open, close, delete, rename, POSIX permission changes, ACL changes, etc.**
- **Don’t capture internal operations (e.g., restripe)**

Events captured within GPFS daemon – represent attributes of filesystem action at that point

Example audit log entry:

```
{"LWE_JSON": "0.0.1", "path": "/newfs/1Kfile2.restore", "oldPath": null,
"clusterName": "pardie.cluster", "nodeName": "c6f2bc3n10", "nfsClientIp": "",
"fsName": "newfs", "event": "OPEN", "inode": "26626", "openFlags": "32962",
"poolName": "sp1", "fileSize": "0", "ownerUserId": "0", "ownerGroupId": "0",
"atime": "2017-10-25_12:36:22-0400", "ctime": "2017-10-25_12:36:22-0400",
"eventTime": "2017-10-25_12:36:22-0400", "clientUserId": "0", "clientGroupId":
"0", "processId": "10437", "permissions": "200100644", "acls": "u::rwc, g::r,
o::r, ", "xattrs": null }
```

Log Files for Auditing

Each file system enabled has a dedicated fileset where the audit logs will go.

- Default option is .audit_log at the root of the file system.

.audit_log fileset is created as **IAM mode noncompliant.**

- Advisory mode plus
  - Files cannot be deleted if retention time is not expired.
  - But retention times can be reset and files can be deleted but not changed

AuditLog files are nested within /FSNAME/.audit_log/topic/year/month/date/*

Log file is written in append only mode

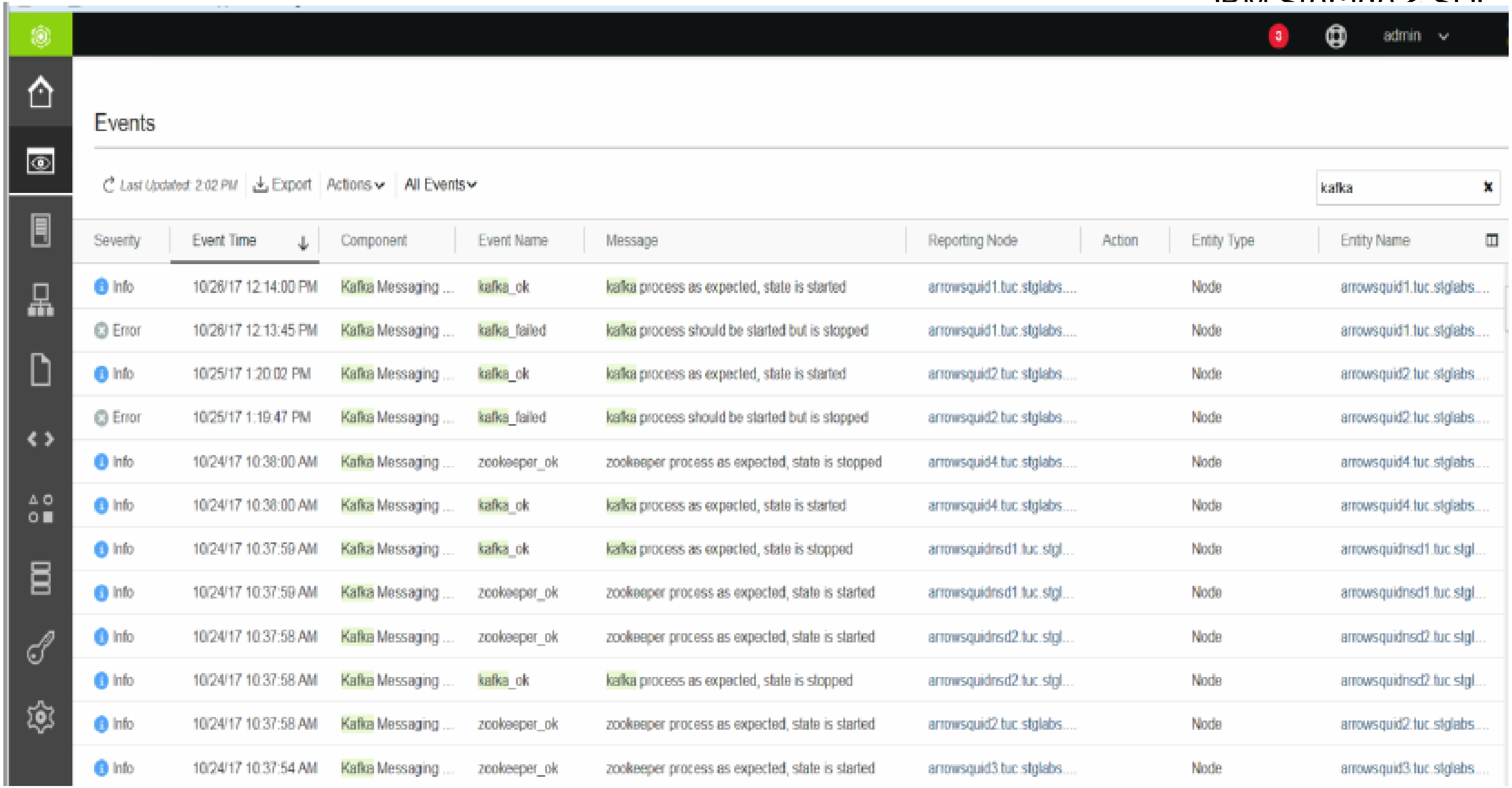
Rotation to a new log file upon reaching a threshold (500,000 events), then compressed and marked immutable for the retention period.

Default retention period is 365 days

Live events can be monitored by tailing the current auditLogFile<…>

Easy to search and consume

FAL in the GUI



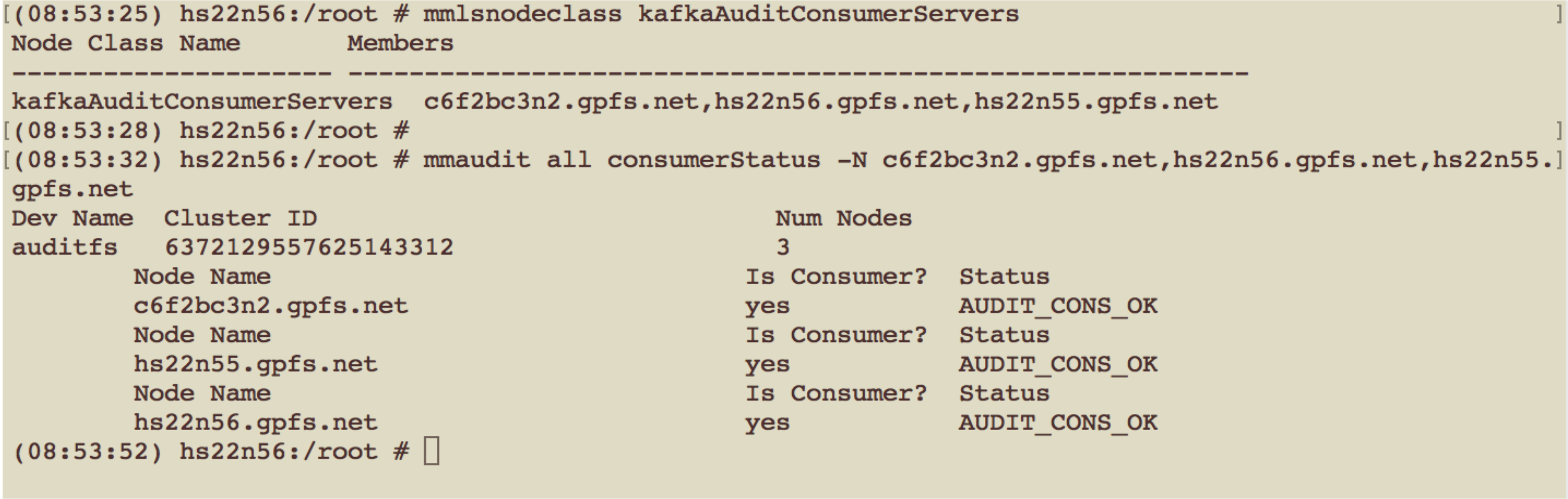
Events

kafka

Severity	Event Time	Component	Event Name	Message	Reporting Node	Action	Entity Type	Entity Name
Info	10/26/17 12:14:00 PM	Kafka Messaging ...	kafka_ok	kafka process as expected, state is started	arrowsquid1.tuc.stglabs....		Node	arrowsquid1.tuc.stglabs....
Error	10/26/17 12:13:45 PM	Kafka Messaging ...	kafka_failed	kafka process should be started but is stopped	arrowsquid1.tuc.stglabs....		Node	arrowsquid1.tuc.stglabs....
Info	10/25/17 1:20:02 PM	Kafka Messaging ...	kafka_ok	kafka process as expected, state is started	arrowsquid2.tuc.stglabs....		Node	arrowsquid2.tuc.stglabs....
Error	10/25/17 1:19:47 PM	Kafka Messaging ...	kafka_failed	kafka process should be started but is stopped	arrowsquid2.tuc.stglabs....		Node	arrowsquid2.tuc.stglabs....
Info	10/24/17 10:38:00 AM	Kafka Messaging ...	zookeeper_ok	zookeeper process as expected, state is stopped	arrowsquid4.tuc.stglabs....		Node	arrowsquid4.tuc.stglabs....
Info	10/24/17 10:38:00 AM	Kafka Messaging ...	kafka_ok	kafka process as expected, state is started	arrowsquid4.tuc.stglabs....		Node	arrowsquid4.tuc.stglabs....
Info	10/24/17 10:37:59 AM	Kafka Messaging ...	kafka_ok	kafka process as expected, state is stopped	arrowsquidnsd1.tuc.stgl...		Node	arrowsquidnsd1.tuc.stgl...
Info	10/24/17 10:37:59 AM	Kafka Messaging ...	zookeeper_ok	zookeeper process as expected, state is started	arrowsquidnsd1.tuc.stgl...		Node	arrowsquidnsd1.tuc.stgl...
Info	10/24/17 10:37:58 AM	Kafka Messaging ...	zookeeper_ok	zookeeper process as expected, state is started	arrowsquidnsd2.tuc.stgl...		Node	arrowsquidnsd2.tuc.stgl...
Info	10/24/17 10:37:58 AM	Kafka Messaging ...	kafka_ok	kafka process as expected, state is stopped	arrowsquidnsd2.tuc.stgl...		Node	arrowsquidnsd2.tuc.stgl...
Info	10/24/17 10:37:58 AM	Kafka Messaging ...	zookeeper_ok	zookeeper process as expected, state is started	arrowsquid2.tuc.stglabs....		Node	arrowsquid2.tuc.stglabs....
Info	10/24/17 10:37:54 AM	Kafka Messaging ...	zookeeper_ok	zookeeper process as expected, state is started	arrowsquid3.tuc.stglabs....		Node	arrowsquid3.tuc.stglabs....

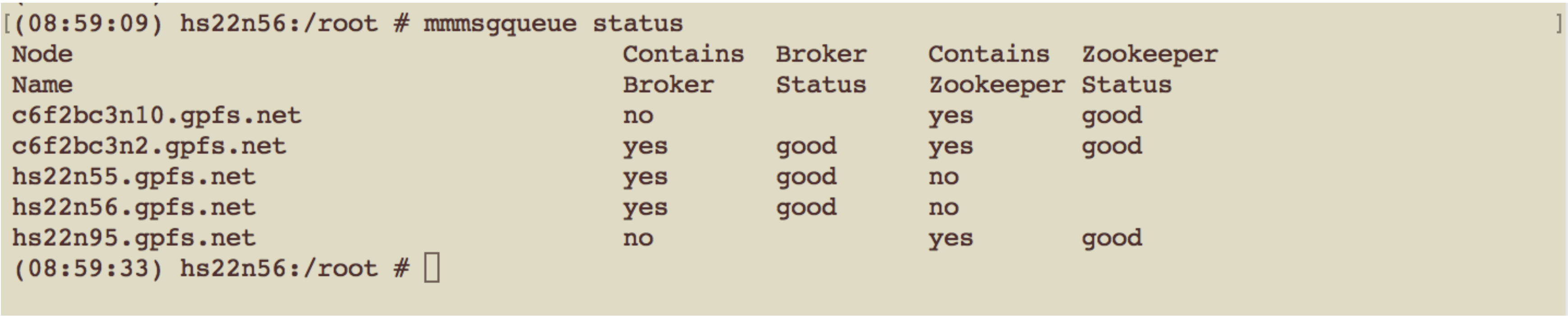
CLI Monitoring

mmaudit all consumerStatus –N …



```
[(08:53:25) hs22n56:/root # mmlsnodeclass kafkaAuditConsumerServers                                 ]
 Node Class Name       Members
 --------------------- -----------------------------------------------------------
 kafkaAuditConsumerServers  c6f2bc3n2.gpfs.net,hs22n56.gpfs.net,hs22n55.gpfs.net
[(08:53:28) hs22n56:/root #                                                                         ]
[(08:53:32) hs22n56:/root # mmaudit all consumerStatus -N c6f2bc3n2.gpfs.net,hs22n56.gpfs.net,hs22n55.]
 gpfs.net
 Dev Name  Cluster ID                               Num Nodes
 auditfs   6372129557625143312                      3
         Node Name                                Is Consumer?  Status
         c6f2bc3n2.gpfs.net                       yes           AUDIT_CONS_OK
         Node Name                                Is Consumer?  Status
         hs22n55.gpfs.net                         yes           AUDIT_CONS_OK
         Node Name                                Is Consumer?  Status
         hs22n56.gpfs.net                         yes           AUDIT_CONS_OK
 (08:53:52) hs22n56:/root # ▯
```

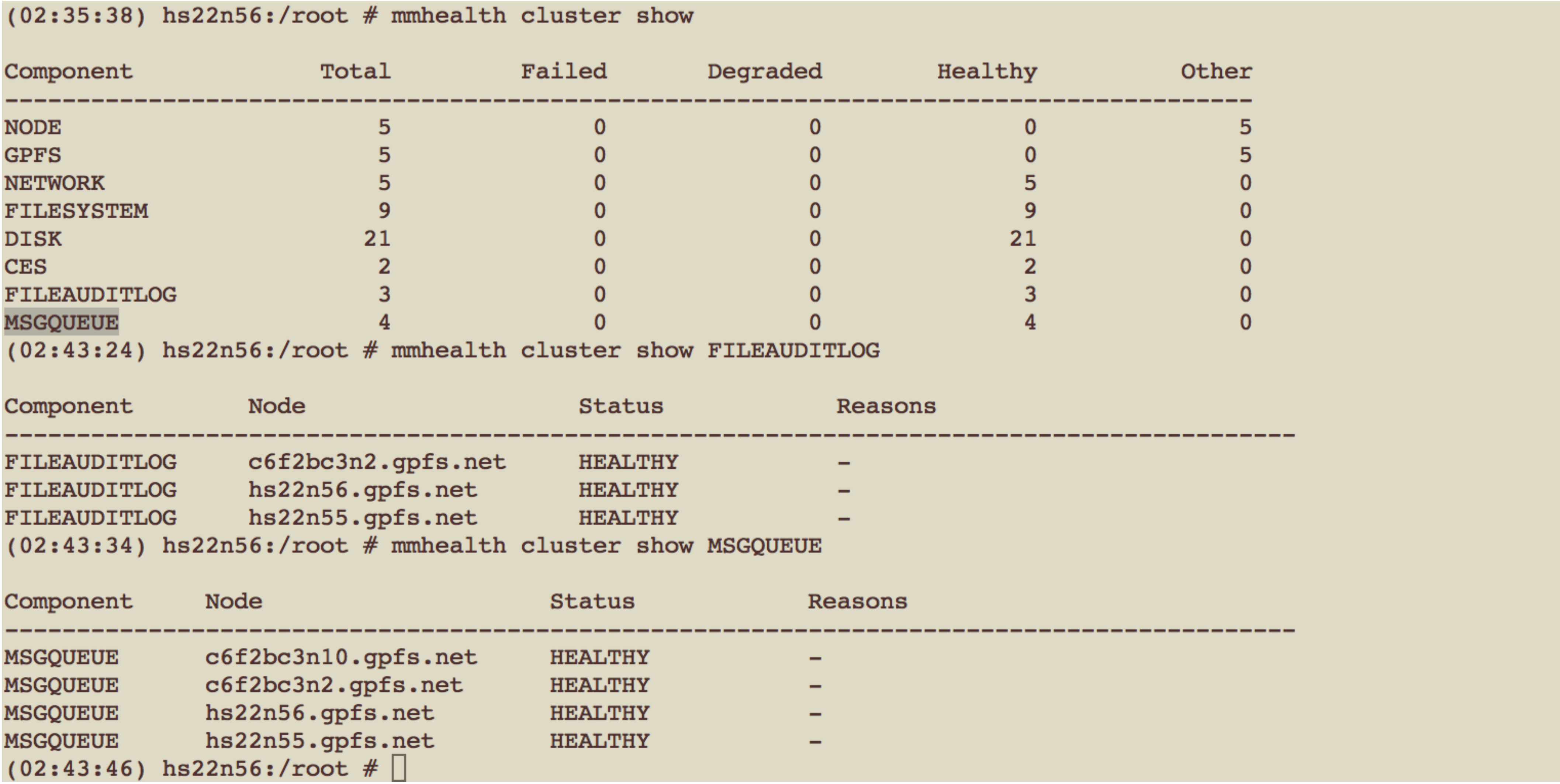
mmmsgqueue status



```
[(08:59:09) hs22n56:/root # mmmsgqueue status                                                   ]
 Node                                    Contains  Broker    Contains  Zookeeper
 Name                                    Broker    Status    Zookeeper Status
 c6f2bc3n10.gpfs.net                     no                  yes       good
 c6f2bc3n2.gpfs.net                      yes       good      yes       good
 hs22n55.gpfs.net                        yes       good      no
 hs22n56.gpfs.net                        yes       good      no
 hs22n95.gpfs.net                        no                  yes       good
 (08:59:33) hs22n56:/root # ▯
```

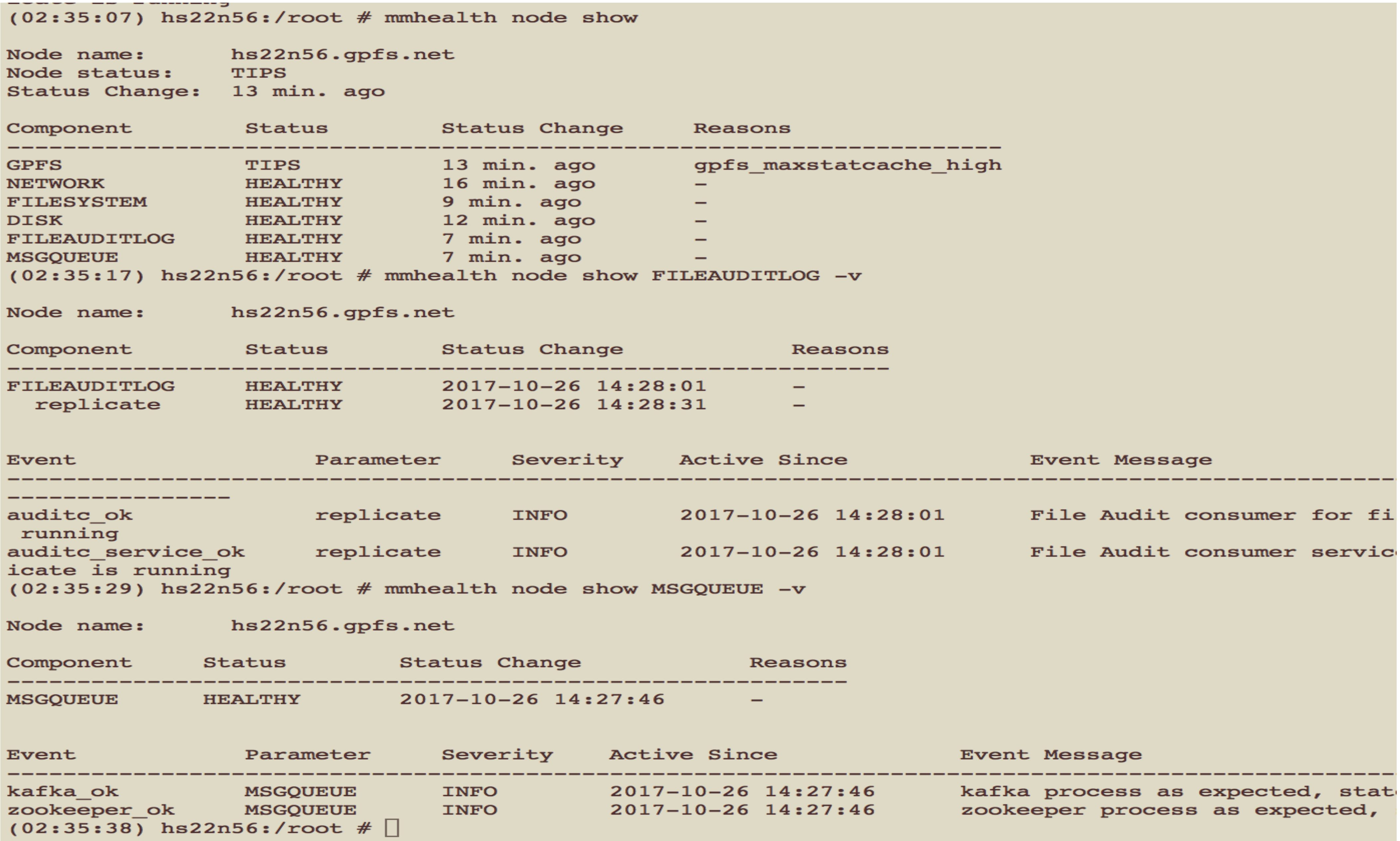
mmhealth cluster monitoring

Periodic polling and event callback registration mechanism is used.
Possible lag in determining the health due to polling constraints.



```
(02:35:38) hs22n56:/root # mmhealth cluster show

Component             Total         Failed       Degraded        Healthy          Other
---------------------------------------------------------------------------------------
NODE                      5              0              0              0              5
GPFS                      5              0              0              0              5
NETWORK                   5              0              0              5              0
FILESYSTEM                9              0              0              9              0
DISK                     21              0              0             21              0
CES                       2              0              0              2              0
FILEAUDITLOG              3              0              0              3              0
MSGQUEUE                  4              0              0              4              0
(02:43:24) hs22n56:/root # mmhealth cluster show FILEAUDITLOG

Component        Node                   Status            Reasons
------------------------------------------------------------------------------------------
FILEAUDITLOG     c6f2bc3n2.gpfs.net     HEALTHY           -
FILEAUDITLOG     hs22n56.gpfs.net       HEALTHY           -
FILEAUDITLOG     hs22n55.gpfs.net       HEALTHY           -
(02:43:34) hs22n56:/root # mmhealth cluster show MSGQUEUE

Component     Node                    Status            Reasons
------------------------------------------------------------------------------------------
MSGQUEUE      c6f2bc3n10.gpfs.net     HEALTHY           -
MSGQUEUE      c6f2bc3n2.gpfs.net      HEALTHY           -
MSGQUEUE      hs22n56.gpfs.net        HEALTHY           -
MSGQUEUE      hs22n55.gpfs.net        HEALTHY           -
(02:43:46) hs22n56:/root #
```

mmhealth node monitoring



```
nodes is running
(02:35:07) hs22n56:/root # mmhealth node show

Node name:      hs22n56.gpfs.net
Node status:    TIPS
Status Change:  13 min. ago

Component        Status        Status Change     Reasons
-----------------------------------------------------------------------
GPFS             TIPS          13 min. ago       gpfs_maxstatcache_high
NETWORK          HEALTHY       16 min. ago       -
FILESYSTEM       HEALTHY       9 min. ago        -
DISK             HEALTHY       12 min. ago       -
FILEAUDITLOG     HEALTHY       7 min. ago        -
MSGQUEUE         HEALTHY       7 min. ago        -
(02:35:17) hs22n56:/root # mmhealth node show FILEAUDITLOG -v

Node name:      hs22n56.gpfs.net

Component        Status        Status Change            Reasons
---------------------------------------------------------------
FILEAUDITLOG     HEALTHY       2017-10-26 14:28:01      -
  replicate      HEALTHY       2017-10-26 14:28:31      -


Event                 Parameter     Severity    Active Since             Event Message
---------------------------------------------------------------------------------------------------
----------------
auditc_ok             replicate     INFO        2017-10-26 14:28:01      File Audit consumer for fi
 running
auditc_service_ok     replicate     INFO        2017-10-26 14:28:01      File Audit consumer servic
icate is running
(02:35:29) hs22n56:/root # mmhealth node show MSGQUEUE -v

Node name:      hs22n56.gpfs.net

Component     Status        Status Change            Reasons
------------------------------------------------------------
MSGQUEUE      HEALTHY       2017-10-26 14:27:46      -


Event            Parameter     Severity    Active Since             Event Message
---------------------------------------------------------------------------------------------------
kafka_ok         MSGQUEUE      INFO        2017-10-26 14:27:46      kafka process as expected, stat
zookeeper_ok     MSGQUEUE      INFO        2017-10-26 14:27:46      zookeeper process as expected,
(02:35:38) hs22n56:/root #
```

Troubleshooting

/var/adm/ras/mmmsgqueue.log

Contains information regarding the set up and configuration operations that take place that affect the message queue

Valid on any node containing a broker and/or zookeeper

/var/adm/ras/mmaudit.log

Contains information regarding the set up and configuration operations that take place that affect the File Audit Logging

Valid on any node running the File Audit Logging command or location where the subcommand may be run (such as a consumer)

/var/adm/ras/mmfs.log.latest

Daemon log, and contains entries when major message queue or File Audit Logging activity occurs.

/var/log/messages (Redhat) or /var/log/syslog (Ubuntu)

Contains messages from Kafka components as well as the producer and consumers that are running on a node.

Logs collected via gpfs.snap

Where could this go in the future?

Antivirus

Take an action if something happens in a directory

TCT enhancements?!

Thank You.

IBM Storage & SDI