



AFF/FAS 시스템에 대한 스토리지 구성

Enterprise applications

NetApp
February 10, 2026

목차

AFF/FAS 시스템에 대한 스토리지 구성	1
FC SAN	1
LUN 정렬	1
LUN 사이징 및 LUN 수	1
LUN 배치	2
LUN 리사이징과 LVM 리사이징	4
LVM 스트라이핑	4
NFS 를 참조하십시오	5
개요	5
Oracle Direct NFS(dNFS)	9
NFS 리스 및 잠금	13
NFS 캐싱	17
NFS 전송 크기	17
NVFAIL(* NVFAIL	18
ASM 재생 유틸리티(ASMRU)	18

AFF/FAS 시스템에 대한 스토리지 구성

FC SAN

LUN 정렬

LUN 정렬은 기본 파일 시스템 레이아웃과 관련하여 I/O를 최적화하는 것을 칭합니다.

ONTAP 시스템에서 스토리지는 4KB 장치로 구성됩니다. 데이터베이스 또는 파일 시스템 8KB 블록은 정확히 2개의 4KB 블록에 매핑되어야 합니다. LUN 구성 오류로 인해 정렬이 어느 방향으로든 1KB 이동되면 각 8KB 블록은 2개가 아닌 3개의 서로 다른 4KB 스토리지 블록에 존재하게 됩니다. 이 방식은 스토리지 시스템 내에서 지연 시간이 늘어나고 추가적인 I/O가 수행되도록 합니다.

정렬은 LVM 아키텍처에도 영향을 미칩니다. 논리적 볼륨 그룹 내의 물리적 볼륨은 전체 드라이브 장치에서 정의되며 (생성되는 파티션 없음), LUN의 첫 번째 4KB 블록이 스토리지 시스템의 첫 번째 4KB 블록에 맞춰 정렬됩니다. 이것이 올바른 정렬입니다. 문제는 파티션에서 발생하는데, 운영 체제가 LUN을 사용하는 시작 위치를 파티션이 바꾸기 때문입니다. 오프셋이 4KB 단위로 이동하는 한 LUN은 정렬됩니다.

Linux 환경에서는 전체 드라이브 장치에 논리적 볼륨 그룹이 구축됩니다. 파티션이 필요하면 를 실행하고 각 파티션의 시작이 8의 배수인지 확인하여 정렬을 `fdisk -u` 점검합니다. 이는 512바이트 섹터를 곱한 값인 4KB에서 파티션이 시작된다는 뜻입니다.

또한 압축 블록 정렬에 대한 내용은 섹션을 ["효율성"](#) 참조하십시오. 8KB 압축 블록 경계에 맞춰 정렬된 레이아웃은 4KB 경계에도 맞춰집니다.

정렬 불량 경고

데이터베이스 재실행/트랜잭션 로깅은 일반적으로 정렬되지 않은 I/O를 생성하여 ONTAP에서 정렬 불량 LUN과 관련된 경고를 야기할 수 있으며 이는 사용자를 호도할 수 있습니다.

로깅은 다양한 크기의 쓰기로 로그 파일의 순차적 쓰기를 수행합니다. 4KB 경계에 맞춰 정렬되지 않는 로그 쓰기 작업은 일반적으로 성능 문제를 야기하지 않는데 다음 로그 쓰기 작업에서 블록이 완료되기 때문입니다. 결과적으로 4KB 블록이 완료되면 ONTAP은 거의 모든 쓰기를 처리할 수 있으며, 이는 일부 4KB 블록의 데이터가 2개의 개별 작업으로 작성되었더라도 마찬가지입니다.

과 같은 유ти리티를 사용하여 정렬을 확인합니다 `sio` 또는 `dd` 이를 통해 정의된 블록 크기로 I/O를 생성할 수 있습니다. 스토리지 시스템의 I/O 정렬 통계는 를 에서 확인할 수 있습니다 `stats` 명령. 을 참조하십시오 ["WAFL 정렬 확인"](#) 를 참조하십시오.

Solaris 환경에서는 정렬이 더 복잡합니다. 을 참조하십시오 ["ONTAP SAN 호스트 구성"](#) 를 참조하십시오.

주의

Solaris x86 환경에서는 대부분의 구성에 여러 개의 파티션 계층이 있기 때문에 정렬이 제대로 되는지 더 주의를 기울여야 합니다. Solaris x86 파티션 조각은 일반적으로 표준 마스터 부트 레코드 파티션 테이블의 상단에 존재합니다.

LUN 사이징 및 LUN 수

최적의 LUN 크기와 사용할 LUN 수를 선택하는 것은 Oracle 데이터베이스의 성능 및 관리

효율성을 최적화하는 데 매우 중요합니다.

LUN은 ONTAP의 가상화 오브젝트로서 호스팅 애그리게이트의 모든 드라이브에 존재합니다. 그 결과, LUN은 어떤 크기를 선택하든 애그리게이트의 잠재력을 최대한으로 활용하기 때문에 크기가 LUN의 성능에 영향을 미치지 않습니다.

고객이 편의상 특정 크기의 LUN을 사용하려 할 수 있습니다. 예를 들어, 데이터베이스가 각각 1TB인 2개의 LUN으로 구성된 LVM 또는 Oracle ASM 디스크 그룹에 구축되어 있는 경우 이 디스크 그룹은 1TB의 증분으로 확장되어야 합니다. 디스크 그룹이 더 적은 증분으로 증가할 수 있도록 각각 500GB인 8개의 LUN으로 디스크 그룹을 구축하는 것이 더 낫습니다.

범용 표준 LUN 크기를 설정하는 방법은 관리가 복잡할 수 있으므로 사용하지 않는 것이 좋습니다. 예를 들어, 100GB의 표준 LUN 크기는 1TB~2TB 범위의 데이터베이스에서는 효과적일 수 있지만 20TB 크기의 데이터베이스나 데이터 저장소에는 200개의 LUN이 필요할 것입니다. 이에 따라 서버 재부팅 시간이 길어지고, 다양한 UI에서 더 많은 오브젝트를 관리해야 하며, SnapCenter 같은 제품을 통해 여러 오브젝트를 탐색해야 합니다. LUN의 개수는 줄이고 크기는 늘리면 이 문제를 피할 수 있습니다.

- LUN 수가 LUN 크기보다 더 중요합니다.
- LUN 크기는 주로 LUN 개수 요구사항에 따라 결정됩니다.
- LUN을 필요한 것보다 많이 생성하지 마십시오.

LUN 수입니다

LUN 크기와 달리 LUN 개수는 성능에 영향을 미칩니다. 애플리케이션 성능은 SCSI 계층을 통해 병렬 I/O를 수행하는 능력에 따라 달라집니다. 그 결과 2개의 LUN이 1개의 LUN보다 나은 성능을 제공합니다. Veritas VxVM, Linux LVM2 또는 Oracle ASM 같은 LVM은 병렬 처리를 늘리는 가장 단순한 방법입니다.

부하가 매우 큰 랜덤 I/O의 100% SSD 환경 테스트에서 최대 64개 LUN이라는 추가 개선이 입증되었음에도 NetApp 고객은 일반적으로 LUN의 개수가 16을 초과하여 증가할 때 최소한의 이점을 경험합니다.

- NetApp는 * 다음을 권장합니다.



일반적으로 특정 데이터베이스 워크로드의 입출력 요구를 지원하는 데는 4~16개의 LUN으로 충분합니다. 4개 미만의 LUN은 호스트 SCSI 구현의 제한으로 인해 성능 제한이 발생할 수 있습니다.

LUN 배치

ONTAP 볼륨 내에 데이터베이스 LUN을 최적으로 배치하는 것은 주로 다양한 ONTAP 기능을 사용하는 방법에 따라 달라집니다.

볼륨

ONTAP를 처음 접하는 고객에게 혼동을 주는 한 가지 일반적인 점은 일반적으로 "볼륨"이라고 하는 FlexVol을 사용하는 것입니다.

볼륨이 LUN이 아닙니다. 이러한 용어는 클라우드 공급자를 포함한 다른 여러 공급업체 제품과 동일하게 사용됩니다. ONTAP 볼륨은 단순한 관리 컨테이너입니다. 이러한 애플리케이션은 자체적으로 데이터를 제공하지 않으며 공간을 점유하지 않습니다. 파일 또는 LUN용 컨테이너이며, 관리 효율성을 개선하고 단순화하기 위해 존재하며, 특히 규모에 맞게 사용됩니다.

볼륨 및 LUN

관련 LUN은 일반적으로 단일 볼륨에 함께 배치됩니다. 예를 들어, 10개의 LUN이 필요한 데이터베이스에서는 일반적으로 10개의 LUN이 모두 동일한 볼륨에 배치되어 있습니다.

- 볼륨에 대한 LUN의 1:1 비율을 사용하는 것은 볼륨당 하나의 LUN을 사용하는 것은 공식적인 모범 사례가 아닙니다.
- 대신, 볼륨을 워크로드 또는 데이터 세트의 컨테이너로 보아야 합니다. 볼륨당 하나의 LUN이 있거나 여러 LUN이 있을 수 있습니다. 정답은 관리 효율 요구사항에 따라 다릅니다.
- 불필요한 볼륨 수에 걸쳐 LUN을 분산시키면 스냅샷 작업, UI에 표시되는 개체 수가 너무 많아지고 LUN 제한에 도달하기 전에 플랫폼 볼륨 제한에 도달할 수 있는 등의 작업에 추가적인 오버헤드 및 스케줄링 문제가 발생할 수 있습니다.

볼륨, LUN 및 스냅샷

스냅샷 정책과 일정은 LUN이 아닌 볼륨에 배치됩니다. 10개의 LUN으로 구성된 데이터 세트에는 해당 LUN이 동일한 볼륨에 공존할 경우 단일 스냅샷 정책만 필요합니다.

또한 특정 데이터 세트에 대한 모든 관련 LUN을 단일 볼륨에 함께 배치하면 원자 단위 스냅샷 작업이 가능합니다. 예를 들어, 10개의 LUN에 상주하는 데이터베이스 또는 10개의 서로 다른 OS로 구성된 VMware 기반 애플리케이션 환경은 기본 LUN이 모두 단일 볼륨에 배치되어 있는 경우 일관된 단일 오브젝트로 보호할 수 있습니다. 서로 다른 볼륨에 배치되는 경우 동시에 예약된 경우에도 스냅샷이 100% 동기화될 수도 있고 그렇지 않을 수도 있습니다.

경우에 따라 복구 요구 사항으로 인해 관련 LUN 세트를 두 개의 다른 볼륨으로 분할해야 할 수도 있습니다. 예를 들어, 데이터베이스에 데이터 파일에는 LUN 4개와 로그에는 LUN 2개가 포함될 수 있습니다. 이 경우 4개의 LUN이 있는 데이터 파일 볼륨과 2개의 LUN이 포함된 로그 볼륨이 최상의 옵션이 될 수 있습니다. 그 이유는 독립적인 복구 기능입니다. 예를 들어, 데이터 파일 볼륨을 선택적으로 이전 상태로 복원할 수 있습니다. 즉, 4개의 LUN이 모두 스냅샷 상태로 되돌려지고 중요 데이터가 있는 로그 볼륨은 영향을 받지 않습니다.

볼륨, LUN 및 SnapMirror를 지원합니다

SnapMirror 정책 및 작업은 스냅샷 작업과 마찬가지로 LUN이 아닌 볼륨에 대해 수행됩니다.

단일 볼륨에서 관련 LUN을 함께 배치하면 단일 SnapMirror 관계를 생성하고 포함된 모든 데이터를 단일 업데이트로 업데이트할 수 있습니다. 스냅샷과 마찬가지로 업데이트는 원자 단위 작업이기도 합니다. SnapMirror 대상은 소스 LUN의 단일 시점 복제본을 보유하도록 보장됩니다. LUN이 여러 볼륨에 분산되어 있는 경우 복제본이 서로 정합성이 보장되거나 일치하지 않을 수 있습니다.

볼륨, LUN 및 QoS를 지원합니다

QoS를 개별 LUN에 선택적으로 적용할 수 있지만 일반적으로 볼륨 수준에서 설정하기가 더 쉽습니다. 예를 들어, 특정 ESX 서버에서 게스트가 사용하는 모든 LUN을 단일 볼륨에 배치한 다음ONTAP 적응형 QoS 정책을 적용할 수 있습니다. 그 결과 모든 LUN에 적용되는 TB당 자체 확장 IOPS 제한이 적용됩니다.

마찬가지로, 데이터베이스에 100K IOPS가 필요하고 10개의 LUN을 사용하는 경우 각 LUN에 하나씩 10개의 개별 10K IOPS 제한을 설정하는 것보다 단일 볼륨에 단일 100K IOPS 제한을 설정하는 것이 더 쉽습니다.

다중 볼륨 레이아웃

여러 볼륨에 LUN을 분산하는 것이 도움이 되는 경우도 있습니다. 주된 이유는 컨트롤러 스트라이핑입니다. 예를 들어, HA 스토리지 시스템은 각 컨트롤러의 완전한 처리와 캐싱 잠재성이 필요한 단일 데이터베이스를 호스팅하고 있을 수

있습니다. 이 경우 일반적인 설계에서는 LUN의 절반을 컨트롤러 1의 단일 볼륨에 배치하고 나머지 절반은 컨트롤러 2의 단일 볼륨에 배치하는 것입니다.

마찬가지로, 로드 밸런싱에 컨트롤러 스트라이핑을 사용할 수도 있습니다. 10개의 LUN으로 구성된 100개의 데이터베이스를 호스팅하는 HA 시스템은 각 데이터베이스가 2개의 컨트롤러 각각에서 5개의 LUN 볼륨을 받을 수 있도록 설계될 수 있습니다. 그 결과, 추가 데이터베이스를 프로비저닝할 때 각 컨트롤러를 대칭으로 로드할 수 있습니다.

단, 이 예에서는 볼륨 대 LUN 비율에 1:1이 관련되지 않습니다. 목표는 볼륨에 관련된 LUN을 함께 배치함으로써 관리 효율성을 최적화하는 것입니다.

1:1 LUN 대 볼륨 비율이 적합한 한 가지 예는 컨테이너화입니다. 각 LUN이 실제로 단일 워크로드를 나타내므로 각 워크로드를 개별적으로 관리해야 합니다. 이 경우 1:1 비율이 최적일 수 있습니다.

LUN 리사이징과 LVM 리사이징

SAN 기반 파일 시스템이 용량 제한에 도달하면 사용 가능한 공간을 늘리기 위한 두 가지 옵션이 있습니다.

- LUN의 크기를 늘립니다
- 기존 볼륨 그룹에 LUN을 추가하고 포함된 논리적 볼륨을 늘립니다

LUN 리사이징이 용량을 증대하는 옵션이기는 하지만 Oracle ASM을 포함한 LVM을 사용하는 것이 대체로 더 낫습니다. LVM이 존재하는 주된 이유 중 하나는 LUN 리사이징이 필요하지 않도록 하는 것입니다. LVM은 여러 개의 LUN을 스토리지 가상 풀에 함께 바인딩합니다. 이 풀에서 생성된 논리적 볼륨은 LVM이 관리하며 쉽게 리사이징할 수 있습니다. 이 외에도 특정 논리적 볼륨을 사용 LUN 전반에 분산하여 특정 드라이브의 핫스팟을 피할 수 있다는 이점이 있습니다. 일반적으로 볼륨 관리자를 사용하여 투명한 마이그레이션을 수행함으로써 논리적 볼륨의 기본 익스텐트를 새 LUN에 재배치할 수 있습니다.

LVM 스트라이핑

LVM 스트라이핑은 여러 LUN에 데이터를 배포하는 것을 의미합니다. 그 결과 많은 데이터베이스의 성능이 크게 향상되었습니다.

플래시 드라이브의 시대 이전에는 스트라이핑이 회전식 드라이브의 성능 제한을 극복하는 데 사용되었습니다. 예를 들어, OS에서 1MB 읽기 작업을 수행해야 하는 경우 1MB가 느리게 전송되기 때문에 단일 드라이브에서 1MB의 데이터를 읽으려면 많은 드라이브 헤드가 필요합니다. 이 1MB의 데이터가 8개의 LUN에 스트라이핑된 경우 운영 체제에서는 8개의 128K의 읽기 작업을 병렬로 실행하고 1MB 전송을 완료하는 데 필요한 시간을 줄일 수 있습니다.

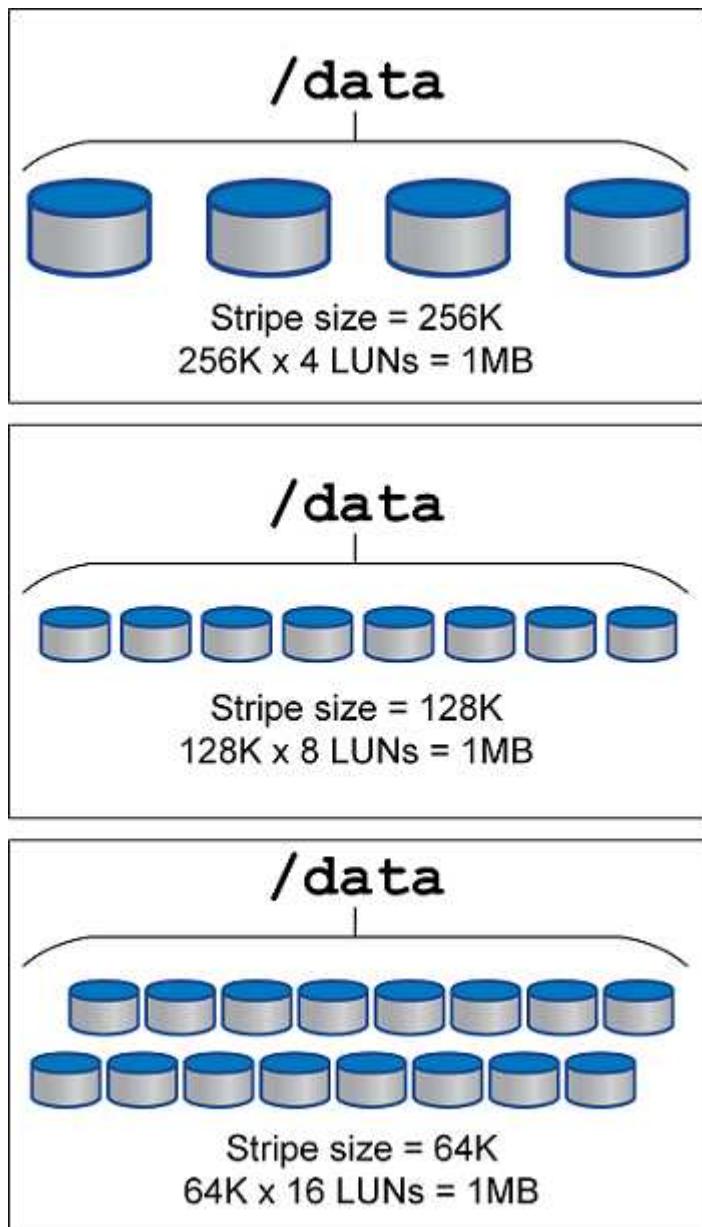
회전식 드라이브를 사용한 스트라이핑은 I/O 패턴을 사전에 알고 있어야 하므로 더 어려웠습니다. 스트라이핑이 실제 I/O 패턴에 맞게 올바르게 조정되지 않은 경우 스트라이핑된 구성이 성능을 저하시킬 수 있습니다. Oracle 데이터베이스, 특히 All-Flash 구성을 사용하면 스트라이핑이 훨씬 쉽게 구성되고 성능이 크게 향상된다는 사실이 입증되었습니다.

기본적으로 Oracle ASM 스트라이프와 같은 논리적 볼륨 관리자는 있지만 기본 OS LVM은 그렇지 않습니다. 이 중 일부는 여러 LUN을 연결된 장치로 연결하므로 하나의 LUN 디바이스와 하나의 LUN 디바이스에 데이터 파일이 존재합니다. 이로 인해 핫스팟이 발생합니다. 다른 LVM 구현은 기본적으로 분산 익스텐트로 설정됩니다. 이는 스트라이핑과 비슷하지만 더 거칠입니다. 볼륨 그룹의 LUN은 익스텐트라고 하는 큰 조각으로 분할되며 일반적으로 메가바이트 단위로 측정되며 논리적 볼륨은 이러한 익스텐트에 분산됩니다. 그 결과 파일에 대한 랜덤 I/O가 LUN 전체에 분산되어야 하지만 순차적 I/O 작업의 효율성이 최대한 높지는 않습니다.

높은 성능을 필요로 하는 애플리케이션 I/O는 거의 항상 (a) 기본 블록 크기 단위 또는 (b) 1MB입니다.

스트라이핑 구성의 기본적인 목표는 단일 파일 I/O를 단일 유닛으로 수행하고, 1MB 크기여야 하는 다중 블록 I/O를 스트라이핑 볼륨의 모든 LUN에 걸쳐 균등하게 병렬 처리할 수 있도록 지원하는 것입니다. 즉, 스트라이프 크기가 데이터베이스 블록 크기보다 작아서는 안 되며 스트라이프 크기를 LUN 수를 곱한 크기가 1MB여야 합니다.

다음 그림에서는 스트라이프 크기 및 폭 조정에 사용할 수 있는 세 가지 옵션을 보여 줍니다. 위에서 설명한 대로 성능 요구 사항을 충족하기 위해 LUN 수를 선택하지만 모든 경우에 단일 스트라이프의 총 데이터는 1MB입니다.



NFS 를 참조하십시오

개요

NetApp은 30년 이상 엔터프라이즈급 NFS 스토리지를 제공하고 있으며 단순성 덕분에 클라우드 기반 인프라로의 전환과 함께 사용 사례가 증가하고 있습니다.

NFS 프로토콜에는 다양한 요구사항이 있는 여러 버전이 포함되어 있습니다. ONTAP를 사용한 NFS 구성에 대한 자세한 내용은을 참조하십시오 "[ONTAP 모범 사례 기반 TR-4067 NFS](#)". 다음 섹션에서는 보다 중요한 요구 사항 및 일반적인 사용자 오류에 대해 설명합니다.

NFS 버전

운영 체제 NFS 클라이언트는 NetApp에서 지원되어야 합니다.

- NFSv3은 NFSv3 표준 OS에서 지원됩니다.
- NFSv3은 Oracle dNFS 클라이언트에서 지원됩니다.
- NFSv4는 NFSv4 표준을 따르는 모든 OS에서 지원됩니다.
- NFSv4.1 및 NFSv4.2에는 특정 OS 지원이 필요합니다. 을 참조하십시오 "[NetApp IMT](#)" 지원되는 운영 체제.
- NFSv4.1에 대해 Oracle dNFS를 지원하려면 Oracle 12.2.0.2 이상이 필요합니다.

를 클릭합니다 "[NetApp 지원 매트릭스](#)" NFSv3 및 NFSv4의 경우 특정 운영 체제가 포함되지 않습니다.



RFC를 준수하는 모든 OS는 일반적으로 지원됩니다. IMT에서 NFSv3 또는 NFSv4 지원을 찾을 때 일치하는 항목이 표시되지 않으므로 특정 OS를 선택하지 마십시오. 모든 OS는 일반 정책에 의해 암시적으로 지원됩니다.

Linux NFSv3 TCP 슬롯 테이블

TCP 슬롯 테이블은 호스트 버스 어댑터(HBA) 큐 길이(queue depth)와 동등한 NFSv3의 기능입니다. 이들 테이블은 한 번에 수행될 수 있는 최대 NFS 작업의 수를 제어합니다. 기본값은 보통 16이며 최적의 성능을 발휘하기에 너무 낮습니다. 최신 Linux 커널에서는 반대의 문제가 발생하는데, 요청을 통해 NFS 서버를 포화시키는 수준까지 TCP 슬롯 테이블의 한계를 자동으로 높일 수 있습니다.

최적의 성능을 얻으면서 성능 문제를 방지하려면 TCP 슬롯 테이블을 제어하는 커널 매개 변수를 조정하십시오.

를 실행합니다 `sysctl -a | grep tcp.*.slot_table` 명령을 실행하고 다음 매개 변수를 관찰합니다.

```
# sysctl -a | grep tcp.*.slot_table
sunrpc.tcp_max_slot_table_entries = 128
sunrpc.tcp_slot_table_entries = 128
```

모든 Linux 시스템에는 가 포함되어 있습니다 `sunrpc.tcp_slot_table_entries` 그러나 일부에만 가 포함됩니다 `sunrpc.tcp_max_slot_table_entries`. 둘 다 128로 설정해야 합니다.



이러한 매개 변수를 설정하지 않으면 성능에 상당한 영향을 미칠 수 있습니다. 경우에 따라 Linux 운영 체제가 충분한 I/O를 실행하지 않기 때문에 성능이 제한됩니다 또는 Linux 운영 체제가 처리할 수 있는 것보다 더 많은 I/O를 발급하려고 하면 I/O 지연 시간이 늘어납니다.

ADR과 NFS

에서 데이터와 관련하여 과도한 양의 I/O가 초래하는 성능 문제를 보고하는 고객이 있습니다 ADR 위치. 일반적으로, 성능 데이터가 대량으로 누적되기 전에는 이 문제가 발생하지 않습니다. 과도한 I/O의 이유는 알려지지 않았지만 이 문제는 Oracle 프로세스가 변경을 위해 타겟 디렉토리를 반복하여 스캐닝했기 때문인 것으로 추측됩니다.

의 제거 noac 및/또는 actimeo=0 마운트 옵션을 사용하면 호스트 운영 체제 캐싱이 수행되고 스토리지 I/O 레벨이 낮아집니다.

* NetApp는 배치하지 않을 것을 권장합니다 ADR 을 사용하는 파일 시스템 데이터 noac 또는 actimeo=0 성능 문제를 일으킬 수 있기 때문입니다. 분리 ADR 필요한 경우 데이터를 다른 마운트 지점으로 이동합니다.

NFS - rootonly 및 mount-rootonly

ONTAP에는 이라는 NFS 옵션이 포함되어 있습니다 nfs-rootonly 서버가 높은 포트의 NFS 트래픽 연결을 허용할지 여부를 제어합니다. 보안 조치로서 루트 사용자만 1024 미만의 소스 포트를 사용하여 TCP/IP 연결을 열 수 있습니다. 이러한 포트는 일반적으로 사용자 프로세스가 아닌 OS 사용을 위해 예약되기 때문입니다. 이러한 제한을 통해 NFS 트래픽이 NFS 클라이언트를 에뮬레이트하는 악의적인 프로세스가 아닌 실제 운영 체제 NFS 클라이언트에서 발생하도록 할 수 있습니다. Oracle dNFS 클라이언트는 사용자 공간 드라이버이지만 프로세스는 루트로 실행되므로 일반적으로 의 값을 변경할 필요가 없습니다 nfs-rootonly. 이 연결은 로우 포트로 이루어집니다.

를 클릭합니다 mount-rootonly 옵션은 NFSv3에만 적용됩니다. 1024 이상의 포트에서 RPC 마운트 호출을 수락할지 여부를 제어합니다. dNFS를 사용하면 클라이언트가 루트로 다시 실행되므로 1024 미만의 포트를 열 수 있습니다. 이 매개 변수는 효과가 없습니다.

프로세스 NFS 버전 4.0 이상을 통해 dNFS 연결을 열면 루트로 실행되지 않으므로 1024를 초과하는 포트가 필요합니다. 를 클릭합니다 nfs-rootonly dNFS가 연결을 완료하려면 매개 변수를 disabled로 설정해야 합니다.

If(경우 nfs-rootonly 이 활성화되면 dNFS 접속을 여는 마운트 단계 동안 작동이 중단됩니다. sqlplus 출력은 다음과 유사합니다.

```
SQL>startup
ORACLE instance started.
Total System Global Area 4294963272 bytes
Fixed Size          8904776 bytes
Variable Size       822083584 bytes
Database Buffers   3456106496 bytes
Redo Buffers        7868416 bytes
```

파라미터는 다음과 같이 변경할 수 있다.

```
Cluster01::> nfs server modify -nfs-rootonly disabled
```

드문 경우지만 nfs-rootonly와 mount-rootonly를 모두 disabled로 변경해야 할 수도 있습니다. 서버가 매우 많은 수의 TCP 연결을 관리하는 경우 1024 미만의 포트를 사용할 수 없으며 OS가 더 높은 포트를 사용하도록 강제됩니다. 연결을 완료하려면 이 두 ONTAP 매개 변수를 변경해야 합니다.

NFS 내보내기 정책: superuser 및 setuid

Oracle 바이너리가 NFS 공유에 있는 경우 내보내기 정책에 수퍼유저 및 setuid 권한이 포함되어야 합니다.

사용자 홈 디렉토리와 같은 일반 파일 서비스에 사용되는 공유 NFS 내보내기는 일반적으로 루트 사용자를 스夸시

처리합니다. 즉, 파일 시스템을 마운트한 호스트에서 루트 사용자의 요청이 더 낮은 권한을 가진 다른 사용자로 다시 매핑됩니다. 이렇게 하면 특정 서버의 루트 사용자가 공유 서버의 데이터에 액세스하지 못하도록 하여 데이터를 보호할 수 있습니다. setuid 비트는 공유 환경에서 보안 위험이 될 수도 있습니다. setuid 비트를 사용하면 명령을 호출하는 사용자와 다른 사용자로 프로세스를 실행할 수 있습니다. 예를 들어, setuid 비트가 있는 루트가 소유한 쉘 스크립트는 루트로 실행됩니다. 다른 사용자가 쉘 스크립트를 변경할 수 있는 경우, 루트가 아닌 사용자는 스크립트를 업데이트하여 명령을 루트로 실행할 수 있습니다.

Oracle 바이너리는 루트가 소유한 파일을 포함하며 setuid 비트를 사용합니다. Oracle 바이너리가 NFS 공유에 설치된 경우 내보내기 정책에 적절한 수퍼 사용자 및 setuid 권한이 포함되어야 합니다. 아래 예제에서는 규칙에 두 가지가 모두 포함되어 있습니다 allow-suid 있습니다 superuser 시스템 인증을 사용하여 NFS 클라이언트에 대한 (루트) 액세스.

```
Cluster01::> export-policy rule show -vserver vserver1 -policyname orabin  
-fields allow-suid,superuser  
vserver   policyname ruleindex superuser allow-suid  
----- ----- ----- -----  
vserver1  orabin      1          sys      true
```

NFSv4/4.1 구성

대부분의 애플리케이션에서 NFSv3과 NFSv4 사이에는 아주 작은 차이가 있습니다. 애플리케이션 입출력은 대개 매우 간단한 입출력이며 NFSv4에서 제공되는 일부 고급 기능의 이점은 크게 활용되지 않습니다. 더 높은 버전의 NFS는 데이터베이스 스토리지의 관점에서 "업그레이드"로 간주해서는 안 되며, 추가 기능이 포함된 NFS 버전으로 간주해야 합니다. 예를 들어 Kerberos 개인 정보 보호 모드(krb5p)의 엔드 투 엔드 보안이 필요한 경우 NFSv4가 필요합니다.



* NetApp는 NFSv4 기능이 필요한 경우 NFSv4.1을 사용할 것을 권장합니다. NFSv4.1에서는 NFSv4 프로토콜의 기능이 개선되어 특정 엣지 경우 복원력을 향상할 수 있습니다.

NFSv4로 전환하는 것은 마운트 옵션을 단순히 vers=3에서 vers=4.1로 변경하는 것보다 더 복잡합니다. 운영 체제 구성에 대한 지침을 포함하여 ONTAP를 사용한 NFSv4 구성에 대한 자세한 설명은을 참조하십시오 ["ONTAP 모범 사례 기반 TR-4067 NFS"](#). 이 TR의 다음 섹션에서는 NFSv4 사용을 위한 몇 가지 기본 요구 사항에 대해 설명합니다.

NFSv4 도메인입니다

NFSv4/4.1 구성에 대한 자세한 설명은 이 문서의 범위를 벗어나지만 도메인 매핑이 일치하지 않는 문제가 흔히 발생합니다. sysadmin 관점에서 NFS 파일 시스템은 정상적으로 작동하는 것처럼 보이지만 애플리케이션이 특정 파일에 대한 권한 및/또는 setuid에 대한 오류를 보고합니다. 경우에 따라 관리자는 응용 프로그램 바이너리의 사용 권한이 손상되었다는 잘못된 결론을 내리고 실제 문제가 도메인 이름일 때 chown 또는 chmod 명령을 실행했습니다.

NFSv4 도메인 이름은 ONTAP SVM에 설정됩니다.

```
Cluster01::> nfs server show -fields v4-id-domain  
vserver   v4-id-domain  
-----  
vserver1  my.lab
```

호스트의 NFSv4 도메인 이름은 에 설정되어 있습니다 /etc/idmap.cfg

```
[root@host1 etc]# head /etc/idmapd.conf
[General]
#Verbosity = 0
# The following should be set to the local NFSv4 domain name
# The default is the host's DNS domain name.
Domain = my.lab
```

도메인 이름이 일치해야 합니다. 그렇지 않은 경우에 다음과 유사한 매피ング 오류가 나타납니다. /var/log/messages:

```
Apr 12 11:43:08 host1 nfsidmap[16298]: nss_getpwnam: name 'root@my.lab'
does not map into domain 'default.com'
```

Oracle 데이터베이스 바이너리와 같은 애플리케이션 바이너리에는 setuid 비트가 있는 루트가 소유한 파일이 포함됩니다. 즉, NFSv4 도메인 이름이 일치하지 않으면 Oracle 시작 시 오류가 발생하고 라는 파일의 소유권이나 사용 권한에 대한 경고가 발생합니다 oradism`에 있습니다 `\$ORACLE_HOME/bin 디렉토리. 다음과 같이 나타납니다.

```
[root@host1 etc]# ls -l /orabin/product/19.3.0.0/dbhome_1/bin/oradism
-rwsr-x--- 1 root oinstall 147848 Apr 17 2019
/orabin/product/19.3.0.0/dbhome_1/bin/oradism
```

이 파일의 소유권이 아무도 없는 경우 NFSv4 도메인 매피ング 문제가 있을 수 있습니다.

```
[root@host1 bin]# ls -l oradism
-rwsr-x--- 1 nobody oinstall 147848 Apr 17 2019 oradism
```

이 문제를 해결하려면 을 선택합니다 /etc/idmap.cfg ONTAP에서 v4-id-domain 설정을 기준으로 한 파일로, 일관성이 있는지 확인합니다. 그렇지 않은 경우 필요한 변경 작업을 수행하고 를 실행합니다 nfsidmap -c`를 클릭하고 변경 사항이 전파될 때까지 잠시 기다립니다. 그런 다음 파일 소유권이 루트로 올바르게 인식되어야 합니다. 사용자가 실행을 시도한 경우 `chown root NFS 도메인 구성이 수정되기 전에 이 파일에서 를 실행해야 할 수도 있습니다 chown root 다시?

Oracle Direct NFS(dNFS)

Oracle 데이터베이스는 두 가지 방법으로 NFS를 사용할 수 있습니다.

먼저 운영 체제의 일부인 네이티브 NFS 클라이언트를 사용하여 마운트된 파일 시스템을 사용할 수 있습니다. 이를 커널 NFS 또는 kNFS라고도 합니다. NFS 파일 시스템은 다른 애플리케이션이 NFS 파일 시스템을 사용하는 것과 정확히 동일한 방식으로 Oracle 데이터베이스에 의해 마운트되고 사용됩니다.

두 번째 방법은 Oracle Direct NFS(dNFS)입니다. Oracle 데이터베이스 소프트웨어 내에서 NFS 표준을 구현한 것입니다. Oracle 데이터베이스가 DBA에 의해 구성되거나 관리되는 방식은 변경되지 않습니다. 스토리지 시스템 자체에 올바른 설정이 있는 한 dNFS의 사용은 DBA 팀과 최종 사용자에게 투명해야 합니다.

dNFS 기능이 활성화된 데이터베이스에는 여전히 일반적인 NFS 파일 시스템이 마운트되어 있습니다. 데이터베이스가

열리면 Oracle 데이터베이스가 일련의 TCP/IP 세션을 열고 NFS 작업을 직접 수행합니다.

직접 NFS

Oracle Direct NFS의 기본 가치는 호스트 NFS 클라이언트를 우회하고 NFS 파일 작업을 NFS 서버에서 직접 수행하는 것입니다. 이를 활성화하려면 ODM(Oracle Disk Manager) 라이브러리를 변경하기만 하면 됩니다. 이 프로세스 관련 지침은 Oracle 설명서에 제공되어 있습니다.

dNFS를 사용하면 입출력이 가능한 가장 효율적인 방식으로 수행되기 때문에 입출력 성능이 크게 향상되고 호스트와 스토리지 시스템의 부하가 감소합니다.

또한 Oracle dNFS에는 네트워크 인터페이스 다중 경로 및 내결함성을 위한 * 옵션 * 이 포함되어 있습니다. 예를 들어, 2개의 10Gb 인터페이스를 서로 바인딩하여 20Gb 대역폭을 제공할 수 있습니다. 한 인터페이스에 장애가 발생하면 다른 인터페이스에서 I/O가 재시도됩니다. 전반적인 작동 방식은 FC 다중 경로와 매우 비슷합니다. 다중 경로는 1Gb 이더넷이 가장 일반적인 표준이었던 수년 전에 일반적이었습니다. 대부분의 Oracle 워크로드에는 10Gb NIC로 충분하지만 더 많은 작업이 필요한 경우 10Gb NIC를 연결할 수 있습니다.

dNFS를 사용할 때는 Oracle Doc 1495104.1에 설명된 모든 패치를 설치하는 것이 중요합니다. 패치를 설치할 수 없는 경우 환경을 평가하여 문서에 설명된 버그가 문제를 유발하지 않는지 확인해야 합니다. 필요한 패치를 설치할 수 없어 dNFS를 사용할 수 없는 경우도 있습니다.

DNS, DDNS, NIS 또는 기타 방법을 포함하여 어떤 유형의 라운드 로빈 이름 확인과도 dNFS를 함께 사용하지 마십시오. ONTAP에서 사용할 수 있는 DNS 로드 밸런싱 기능도 여기에 포함됩니다. dNFS를 사용하는 Oracle 데이터베이스가 호스트 이름을 IP 주소로 확인하면 이후 조회에서 호스트 이름을 변경해서는 안 됩니다. 변경할 경우 Oracle 데이터베이스 충돌과 잠재적 데이터 손상을 야기할 수 있습니다.

dNFS를 설정하는 중입니다

Oracle dNFS는 dNFS 라이브러리를 활성화하는 것 외에 별도의 구성 없이도 NFSv3과 함께 사용할 수 있습니다 (필요한 특정 명령에 대한 Oracle 설명서 참조). 그러나 dNFS가 연결을 설정할 수 없는 경우 커널 NFS 클라이언트로 자동 되돌아갈 수 있습니다. 이 경우 성능에 심각한 영향을 미칠 수 있습니다.

NFSv4.X를 사용하여 여러 인터페이스에서 dNFS 멀티플렉싱을 사용하거나 암호화를 사용하려면 orafstab 파일을 구성해야 합니다. 구문은 매우 엄격합니다. 파일의 작은 오류로 인해 시작 시 중단되거나 orafstab 파일을 우회할 수 있습니다.

쓰기 당시에는 dNFS 다중 경로가 최신 버전의 Oracle Database와 NFSv4.1에서 작동하지 않습니다. NFSv4.1을 프로토콜로 지정하는 orafstab 파일은 지정된 내보내기에 대해 단일 PATH 명령만 사용할 수 있습니다. 그 이유는 ONTAP가 ClientID 트렁킹을 지원하지 않기 때문입니다. 이 제한을 해결하기 위한 Oracle Database 패치는 향후 제공될 수 있습니다.

특정 dNFS가 예상대로 작동하는 유일한 방법은 v\$dnfs 테이블을 쿼리하는 것입니다.

아래는 /etc에 있는 샘플 orafstab 파일입니다 이것은 orafstab 파일을 배치할 수 있는 여러 위치 중 하나입니다.

```
[root@jfs11 trace]# cat /etc/oranfstab
server: NFSv3test
path: jfs_svmdr-nfs1
path: jfs_svmdr-nfs2
export: /dbf mount: /oradata
export: /logs mount: /logs
nfs_version: NFSv3
```

첫 번째 단계는 지정된 파일 시스템에 대해 dNFS가 작동하는지 확인하는 것입니다.

```
SQL> select dirname,nfsversion from v$dnfs_servers;

DIRNAME
-----
NFSVERSION
-----
/logs
NFSv3.0

/dbf
NFSv3.0
```

이 출력은 dNFS가 이 두 파일 시스템에서 사용 중임을 나타내지만 * 이는 oranfstab이 작동 중임을 의미하지는 않습니다. 오류가 있는 경우 dNFS는 호스트의 NFS 파일 시스템을 자동으로 검색하며 이 명령의 출력도 동일하게 표시될 수 있습니다.

경로 다중화는 다음과 같이 확인할 수 있습니다.

```
SQL> select svrname,path,ch_id from v$dnfs_channels;

SVRNAME
-----
PATH
-----
CH_ID
-----
NFSv3test
jfs_svmdr-nfs1
          0

NFSv3test
jfs_svmdr-nfs2
          1
```

```
SVRNAME
-----
PATH
-----
CH_ID
-----

NFSv3test
jfs_svmdr-nfs1
    0

NFSv3test
jfs_svmdr-nfs2

[output truncated]

SVRNAME
-----
PATH
-----
CH_ID
-----

NFSv3test
jfs_svmdr-nfs2
    1

NFSv3test
jfs_svmdr-nfs1
    0

SVRNAME
-----
PATH
-----
CH_ID
-----

NFSv3test
jfs_svmdr-nfs2
    1

66 rows selected.
```

dNFS가 사용하는 연결입니다. 각 SVRNAME 항목에 대해 두 개의 경로와 채널이 표시됩니다. 이는 다중 경로가 작동 중임을 의미합니다. 즉, oranfstab 파일이 인식되고 처리되었음을 의미합니다.

Direct NFS 및 호스트 파일 시스템 액세스

dNFS를 사용하면 호스트에 마운트된 가시적인 파일 시스템을 사용하는 애플리케이션이나 사용자 작업에 문제가 발생할 수 있습니다. dNFS 클라이언트가 호스트 OS에서 대역 외 파일 시스템에 액세스하기 때문입니다. dNFS 클라이언트는 운영 체제에 대한 지식 없이도 파일을 생성, 삭제 및 수정할 수 있습니다.

단일 인스턴스 데이터베이스의 마운트 옵션을 사용할 때 이들 옵션은 파일과 디렉토리 특성의 캐싱을 활성화할 수 있으며 이에 따라 디렉토리의 콘텐츠가 캐싱됩니다. 따라서 dNFS는 파일을 생성할 수 있으며 OS가 디렉토리 콘텐츠를 다시 읽고 파일이 사용자에게 표시되기 전에 짧은 지연이 발생합니다. 일반적으로는 이것이 문제가 되지 않지만 드물게 SAP BR * Tools 같은 유ти리티에 문제가 발생할 수 있습니다. 이런 일이 발생하면 Oracle RAC 관련 권장사항을 사용하기 위한 마운트 옵션을 변경하여 문제를 해결하십시오. 이렇게 변경하면 모든 호스트 캐싱이 비활성화됩니다.

(a) dNFS가 사용되고 (b) 파일 가시성의 지연으로 인해 문제가 발생하는 경우에만 마운트 옵션을 변경하십시오. dNFS를 사용하지 않는 경우 단일 인스턴스 데이터베이스에서 Oracle RAC 마운트 옵션을 사용하면 성능이 저하됩니다.



비정상적인 결과를 초래할 수 있는 Linux 관련 dNFS 문제에 대해서는 의 "[Linux NFS 마운트 옵션](#)" 참고 사항을 `nosharecache` 참조하십시오.

NFS 리스 및 잠금

NFSv3은 상태를 저장하지 않습니다. 이는 사실상 NFS 서버(ONTAP)가 어떤 파일 시스템이 마운트되었는지, 누가 어떤 잠금이 설정되어 있는지 추적하지 않는다는 것을 의미합니다.

ONTAP에는 마운트 시도를 기록하는 몇 가지 기능이 있으므로 어떤 클라이언트가 데이터에 액세스할 수 있는지 알 수 있고, 권고 잠금이 있을 수 있지만 이 정보가 100% 완전하지는 않을 수 있습니다. NFS 클라이언트 상태 추적은 NFSv3 표준의 일부가 아니므로 완료할 수 없습니다.

NFSv4 상태

이와 반대로 NFSv4는 상태 저장 방식입니다. NFSv4 서버는 어떤 클라이언트가 어떤 파일 시스템을 사용하고 있는지, 어떤 파일이 있는지, 어떤 파일 및/또는 파일 영역이 잠겨 있는지 등을 추적합니다 즉, 상태 데이터를 최신 상태로 유지하려면 NFSv4 서버 간에 정기적인 통신이 필요합니다.

NFS 서버에서 관리하는 가장 중요한 상태는 NFSv4 잠금 및 NFSv4 리스 상태이며, 서로 매우 밀접하게 연관되어 있습니다. 당신은 각각의 작동 방식 및 서로 어떻게 관련되는지 이해해야 합니다.

NFSv4 잠금

NFSv3에서는 잠금은 권장됩니다. NFS 클라이언트는 "잠긴" 파일을 계속 수정하거나 삭제할 수 있습니다. NFSv3 잠금은 단독으로 만료되지 않고 제거되어야 합니다. 이로 인해 문제가 발생합니다. 예를 들어, NFSv3 잠금을 생성하는 클러스터 애플리케이션이 있는데 노드 중 하나에 장애가 발생하면 어떻게 해야 합니까? 나머지 노드에서 애플리케이션을 코딩하여 잠금을 제거할 수 있지만 안전한지 어떻게 알 수 있습니까? "장애가 발생한" 노드가 작동 중이지만 나머지 클러스터와 통신하지 않는 것 같습니까?

NFSv4에서는 잠금 기간이 제한됩니다. 잠금을 보유한 클라이언트가 NFSv4 서버에 계속 체크인하는 한 다른 클라이언트는 이러한 잠금을 가져올 수 없습니다. 클라이언트가 NFSv4를 통해 체크인하지 못하면 서버에서 잠금이 취소되고 다른 클라이언트가 잠금을 요청하고 받을 수 있습니다.

NFSv4 임대

NFSv4 잠금은 NFSv4 임대와 연결됩니다. NFSv4 클라이언트가 NFSv4 서버와 연결을 설정하면 임대됩니다. 클라이언트가 잠금을 얻는 경우(여러 유형의 잠금이 있음) 잠금이 임대와 연결됩니다.

이 임대에 정의된 시간 초과가 있습니다. 기본적으로 ONTAP는 시간 초과 값을 30초로 설정합니다.

```
Cluster01::>*> nfs server show -vserver vserver1 -fields v4-lease-seconds  
  
vserver    v4-lease-seconds  
-----  
vserver1   30
```

즉, NFSv4 클라이언트는 30초마다 NFSv4 서버에 체크인해야 리스를 갱신할 수 있습니다.

리스는 모든 활동에 의해 자동으로 갱신되므로 클라이언트가 작업을 수행하는 경우 추가 작업을 수행할 필요가 없습니다. 응용 프로그램이 조용해지고 실제 작업을 수행하지 않는 경우 대신 연결 유지 작업(시퀀스라고 함)을 수행해야 합니다. "아직 여기 있어요. 리스를 새로 고치세요."라고 말하는 것입니다.

Question: What happens if you lose network connectivity for 31 seconds?
NFSv3은 상태를 저장하지 않습니다. 클라이언트로부터 통신을 기대하지 않습니다. NFSv4는 상태 저장이며 임대 기간이 경과하면 임대가 만료되고 잠금이 해제되고 잠긴 파일을 다른 클라이언트에서 사용할 수 있게 됩니다.

NFSv3을 사용하면 네트워크 케이블을 이동할 수 있고, 네트워크 스위치를 재부팅하고, 구성을 변경하며, 나쁜 부분이 전혀 발생하지 않도록 매우 확신할 수 있습니다. 일반적으로 응용 프로그램은 네트워크 연결이 다시 작동할 때까지 인내심을 가지고 기다립니다.

NFSv4를 사용하면 30초 안에 작업을 완료할 수 있습니다(ONTAP 내에서 이 매개 변수의 값을 늘리지 않는 한). 이 값을 초과하면 리스가 시간 초과됩니다. 일반적으로 이 경우 응용 프로그램이 충돌합니다.

예를 들어, Oracle 데이터베이스가 있는데 네트워크 연결 손실("네트워크 파티션"이라고도 함)이 임대 시간 초과를 초과하면 데이터베이스가 충돌합니다.

이 경우 Oracle 경고 로그에서 발생하는 작업의 예는 다음과 같습니다.

```
2022-10-11T15:52:55.206231-04:00
Errors in file /orabin/diag/rdbms/ntap/NTAP/trace/NTAP_ckpt_25444.trc:
ORA-00202: control file: '/redo0/NTAP/ctrl/control01.ctl'
ORA-27072: File I/O error
Linux-x86_64 Error: 5: Input/output error
Additional information: 4
Additional information: 1
Additional information: 4294967295
2022-10-11T15:52:59.842508-04:00
Errors in file /orabin/diag/rdbms/ntap/NTAP/trace/NTAP_ckpt_25444.trc:
ORA-00206: error in writing (block 3, # blocks 1) of control file
ORA-00202: control file: '/redo1/NTAP/ctrl/control02.ctl'
ORA-27061: waiting for async I/Os failed
```

syslogs를 보면 다음과 같은 몇 가지 오류가 표시됩니다.

```
Oct 11 15:52:55 host1 kernel: NFS: nfs4_reclaim_open_state: Lock reclaim
failed!
Oct 11 15:52:55 host1 kernel: NFS: nfs4_reclaim_open_state: Lock reclaim
failed!
Oct 11 15:52:55 host1 kernel: NFS: nfs4_reclaim_open_state: Lock reclaim
failed!
```

로그 메시지는 일반적으로 응용 프로그램 정지 이외의 문제의 첫 번째 징후입니다. 일반적으로 프로세스와 운영 체제 자체가 NFS 파일 시스템에 대한 액세스 시도를 차단하기 때문에 네트워크가 중단되는 동안에는 아무 것도 표시되지 않습니다.

네트워크가 다시 작동하면 오류가 나타납니다. 위의 예에서 연결이 다시 설정되면 OS가 잠금을 다시 가져오려고 시도했지만 너무 늦었습니다. 임대가 만료되었고 잠금이 제거되었습니다. 이로 인해 오류가 Oracle 계층까지 전파되고 경고 로그에 메시지가 표시됩니다. 데이터베이스의 버전 및 구성에 따라 이러한 패턴의 변형이 표시될 수 있습니다.

요약하면 NFSv3은 네트워크 중단은 허용하지만 NFSv4는 더 민감하며 정의된 임대 기간을 사용합니다.

30초 시간 초과가 허용되지 않는 경우에는 어떻게 해야 합니까? 스위치가 재부팅되거나 케이블이 재배치되어 간헐적인 네트워크 중단이 발생하는 동적으로 변화하는 네트워크를 관리하는 경우 어떻게 해야 합니까? 임대 기간을 연장하도록 선택할 수 있지만, 이 작업을 수행하려면 NFSv4 유예 기간에 대한 설명이 필요합니다.

NFSv4 유예 기간

NFSv3 서버를 재부팅하면 거의 즉시 IO를 처리할 수 있습니다. 그것은 고객에 대한 어떤 종류의 상태를 유지하지 않았다. 그 결과 ONTAP 테이크오버 작업이 거의 즉각적으로 발생하는 것으로 보이는 경우가 많습니다. 컨트롤러가 데이터 제공을 시작할 준비가 되면 토플로지 변경을 알리는 ARP를 네트워크에 보냅니다. 일반적으로 클라이언트는 이를 거의 즉각적으로 감지하여 데이터가 다시 흐릅니다.

하지만 NFSv4는 잠시 일시 중지됩니다. NFSv4의 작동 방식 중 일부에 불과합니다.



다음 섹션은 ONTAP 9.15.1 현재 버전이지만 임대 및 잠금 동작과 조정 옵션이 버전 간에 변경될 수 있습니다. NFSv4 리스/잠금 시간 제한을 조정해야 하는 경우 NetApp 지원에 최신 정보를 문의하십시오.

NFSv4 서버는 임대, 잠금 및 누가 어떤 데이터를 사용하는지 추적해야 합니다. NFS 서버가 패닉 및 재부팅되거나 잠시 전원이 깨지거나 유지 보수 작업 중에 다시 시작되면 임대/잠금이 발생하고 다른 클라이언트 정보가 손실됩니다. 서버에서 작업을 재개하기 전에 어떤 클라이언트가 어떤 데이터를 사용 중인지 파악해야 합니다. 유예 기간이 시작되는 곳입니다.

NFSv4 서버의 전원을 갑자기 끈다 켜는 경우 다시 백업되면 입출력을 재개하려는 클라이언트는 기본적으로 "리스/잠금 정보가 손실되었습니다. 잠금 장치를 다시 등록하시겠습니까?" 이것이 유예 기간의 시작입니다. ONTAP에서는 기본적으로 45초로 설정됩니다.

```
Cluster01::> nfs server show -vserver vserver1 -fields v4-grace-seconds  
  
vserver      v4-grace-seconds  
-----  
vserver1    45
```

그 결과, 재시작 후 모든 클라이언트가 임대를 재확보하고 잠금을 재설정하는 동안 컨트롤러는 IO를 일시 중지합니다. 유예 기간이 종료되면 서버가 입출력 작업을 재개합니다.

이 유예 기간은 네트워크 인터페이스 변경 중에 임대 재확보를 제어하지만 스토리지 페일오버 중에 재확보를 제어하는 두 번째 유예 기간이 `locking.grace_lease_seconds` 있습니다. 노드 레벨 옵션입니다.

```
cluster01::> node run [node names or *] options  
locking.grace_lease_seconds
```

예를 들어, LIF 페일오버를 자주 수행해야 하며 유예 기간을 줄여야 하는 경우 변경을 `v4-grace-seconds` 수행합니다. 컨트롤러 페일오버 중 IO 재시작 시간을 단축하려면 을 변경해야 합니다 `locking.grace_lease_seconds`.

위험 및 결과를 완전히 이해한 후에만 이러한 값을 변경하십시오. NFSv4.X에서의 페일오버 및 마이그레이션 작업과 관련된 입출력 일시 중지는 완전히 피할 수 없습니다. 잠금, 리스 및 유예 기간은 NFS RFC의 일부입니다. 페일오버 시간이 더 빨라서 많은 고객에게 NFSv3을 사용하는 것이 더 좋습니다.

리스 시간 초과 대 유예 기간

유예 기간 및 임대 기간이 연결되었습니다. 위에서 언급한 것처럼 기본 임대 시간 초과는 30초입니다. 즉, NFSv4 클라이언트는 30초마다 서버에 체크인해야 합니다. 그렇지 않으면 리스와 잠금이 손실됩니다. NFS 서버가 임대/잠금 데이터를 재구축할 수 있는 유예 기간이 있으며 기본값은 45초입니다. 유예 기간은 임대 기간보다 길어야 합니다. 이를 통해 최소 30초마다 리스를 갱신하도록 설계된 NFS 클라이언트 환경에서는 재시작 후 서버를 통해 체크인 할 수 있습니다. 45초의 유예 기간은 최소 30초마다 리스를 갱신할 모든 고객이 확실히 그렇게 할 기회를 갖도록 합니다.

30초의 시간 초과가 허용되지 않는 경우 임대 기간을 연장할 수 있습니다.

60초의 네트워크 중단을 견디기 위해 리스 시간 제한을 60초로 늘리려면 유예 기간도 늘려야 합니다. 이는 컨트롤러 페일오버 중 IO 일시 중단이 더 길다는 것을 의미합니다.

이것은 일반적으로 문제가 되지 않습니다. 일반 사용자는 연간 1~2회ONTAP 컨트롤러를 업데이트하며, 하드웨어 장애로 인한 계획되지 않은 페일오버는 매우 드물게 발생합니다. 또한 60초 네트워크 중단이 발생할 가능성이 있는 네트워크가 있고 임대 시간 초과가 60초로 필요한 경우 드물게 발생하는 스토리지 시스템 장애 조치를 거부하여 61초 동안 일시 중지되지 않을 수 있습니다. 이미 60초 이상 일시 중지된 네트워크가 있음을 확인했습니다.

NFS 캐싱

다음과 같은 마운트 옵션이 있으면 호스트 캐싱이 비활성화됩니다.

```
cio, actimeo=0, noac, forcedirectio
```

이러한 설정은 소프트웨어 설치, 패치 적용 및 백업/복원 작업 속도에 심각한 악영향을 미칠 수 있습니다. 경우에 따라, 특히 클러스터형 애플리케이션의 경우 클러스터의 모든 노드에 캐시 정합성을 보장해야 하기 때문에 이러한 옵션이 불가피하게 필요하게 되었습니다. 고객이 이러한 매개 변수를 잘못 사용하면 불필요한 성능 저하가 발생하게 됩니다.

많은 고객들이 애플리케이션 바이너리의 설치 또는 패칭 중에 이러한 마운트 옵션을 임시로 제거합니다. 설치 또는 패칭 프로세스 중에 타겟 디렉토리를 적극적으로 사용하는 여타 프로세스가 없음이 확인되면 제거를 안전하게 수행할 수 있습니다.

NFS 전송 크기

기본적으로 ONTAP에서는 NFS I/O 크기를 64K로 제한합니다.

대부분의 애플리케이션 및 데이터베이스에서 랜덤 I/O는 최대 64K 미만으로 훨씬 작은 블록 크기를 사용합니다. 대규모 블록 I/O는 일반적으로 병렬화되므로 최대 64K 역시 최대 대역폭을 확보하는 데 제한이 없습니다.

일부 워크로드는 최대 64K로 인해 제한이 발생합니다. 특히, 데이터베이스가 적은 수의 입출력을 수행할 수 있는 경우 백업 또는 복구 작업 또는 데이터베이스 전체 테이블 스캔과 같은 단일 스레드 작업이 보다 빠르고 효율적으로 실행됩니다. ONTAP의 최적의 I/O 처리 크기는 256K입니다.

특정 ONTAP SVM의 최대 전송 크기를 다음과 같이 변경할 수 있습니다.

```
Cluster01::> set advanced
Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by NetApp personnel.
Do you want to continue? {y|n}: y
Cluster01::*> nfs server modify -vserver vserver1 -tcp-max-xfer-size
262144
Cluster01::*>
```

ONTAP에서 허용되는 최대 전송 크기를 현재 마운트된 NFS 파일 시스템의 rsize/wsize 값보다 작게 줄이지 마십시오. 이로 인해 일부 운영 체제에서 중단되거나 심지어 데이터 손상이 발생할 수 있습니다. 예를 들어, NFS 클라이언트가 현재 rsize/wsize 65536으로 설정되어 있는 경우 클라이언트 자체가 제한되기 때문에 ONTAP 최대 전송 크기를 65536에서 1048576 사이에서 아무 영향 없이 조정할 수 있습니다. 최대 전송 크기를 65536 미만으로 줄이면 가용성 또는 데이터가 손상될 수 있습니다.



NVFAIL(* NVFAIL

NVFAIL은 대규모 페일오버 시나리오 중에 무결성을 보장하는 ONTAP의 기능입니다.

데이터베이스는 대규모 내부 캐시를 유지하기 때문에 스토리지 페일오버 중에 손상에 취약합니다. 재앙적인 이벤트에서 ONTAP 페일오버를 강제 적용하거나 MetroCluster 스위치오버를 강제 적용해야 하는 경우, 전체 구성 상태와 관계없이 이전에 승인된 변경 사항을 효과적으로 삭제할 수 있기 때문입니다. 스토리지 어레이의 콘텐츠가 시간 뒤로 이동하며, 데이터베이스 캐시의 상태는 디스크에 있는 데이터의 상태를 더 이상 반영하지 않습니다. 이러한 불일치로 인해 데이터가 손상됩니다.

캐싱은 애플리케이션이나 서버 계층에서 발생할 수 있습니다. 예를 들어, 운영 사이트와 원격 사이트 모두에서 서버가 활성 상태인 Oracle RAC(Real Application Cluster) 구성은 Oracle SGA 내의 데이터를 캐시합니다. 강제 전환 작업으로 인해 데이터가 손실되면 SGA에 저장된 블록이 디스크의 블록과 일치하지 않을 수도 있기 때문에 데이터베이스가 손상될 위험에 처하게 됩니다.

운영 체제 파일 시스템 계층에서의 캐싱 사용은 이보다 덜 명확합니다. 마운트된 NFS 파일 시스템의 블록이 운영 체제에 캐싱될 수 있습니다. 또는 운영 사이트에 있는 LUN 기반의 클러스터된 파일 시스템을 원격 사이트의 서버에 마운트하여 데이터를 다시 캐시할 수 있습니다. 이러한 상황에서 NVRAM 장애 또는 강제 적용 테이크오버 또는 강제 적용 스위치오버로 인해 파일 시스템이 손상될 수 있습니다.

ONTAP는 NVFAIL 및 관련 설정을 통해 이 시나리오로부터 데이터베이스와 운영 체제를 보호합니다.

ASM 재생 유ти리티(ASMRU)

ONTAP은 인라인 압축이 활성화되었을 때 파일 또는 LUN에 기록되는 제로 블록을 효율적으로 제거합니다. Oracle ASRU(ASM Reclamation Utility)와 같은 유ти리티는 사용되지 않는 ASM 익스텐트에 0을 쓰는 방식으로 작동합니다.

따라서 DBA는 데이터가 삭제된 후 스토리지 시스템의 공간을 재확보할 수 있습니다. ONTAP는 0을 가로채고 LUN의 공간을 재할당합니다. 스토리지 시스템 내에 데이터가 기록되지 않으므로 재확보 프로세스가 매우 빠릅니다.

데이터베이스 관점에서 보면 ASM 디스크 그룹은 0을 포함하고 LUN의 해당 영역을 읽으면 스트림이 0이 되지만 ONTAP는 드라이브에 0을 저장하지 않습니다. 그 대신, 내부적으로 LUN의 제로화된 영역을 모든 데이터가 비어 있는 상태로 표시하는 간단한 메타데이터 변경이 이루어집니다.

마찬가지로 0의 블록이 실제로 스토리지 시스템 내에서 쓰기로 처리되지 않기 때문에 제로화된 데이터를 포함하는 성능 테스트는 유효하지 않습니다.



ASRU를 사용하는 경우 모든 Oracle 권장 패치가 설치되어 있는지 확인합니다.

저작권 정보

Copyright © 2026 NetApp, Inc. All Rights Reserved. 미국에서 인쇄됨 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그레픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이센스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이센스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 있으며 재사용이 불가능하며 취소 불가능한 라이센스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

NETAPP, NETAPP 로고 및 <http://www.netapp.com/TM>에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.