



소프트웨어 기반 노드 설치 StorageGRID software

NetApp
July 23, 2026

목차

소프트웨어 기반 노드 설치	1
소프트웨어 기반 노드에 StorageGRID를 설치하는 빠른 시작	1
설치 자동화	1
소프트웨어 기반 노드에 대한 설치 계획 및 준비	2
StorageGRID에 필요한 정보 및 자료	2
StorageGRID 설치 파일 다운로드 및 압축 해제	3
StorageGRID 설치 파일을 확인하십시오	9
StorageGRID의 소프트웨어 요구 사항	10
StorageGRID의 CPU 및 RAM 요구 사항	14
StorageGRID의 스토리지 및 성능 요구 사항	15
Linux용 StorageGRID 노드 컨테이너 마이그레이션 요구 사항	21
호스트(Linux) 준비	23
소프트웨어 기반 노드 설치 자동화	40
Linux에서 StorageGRID 호스트 서비스 설치 및 그리드 노드 구성 자동화	40
VMware vSphere를 사용하여 StorageGRID에서 그리드 노드 배포 자동화	43
가상 그리드 노드 배포	56
StorageGRID용 VMware 환경에 대한 정보 수집	57
Linux 배포용 StorageGRID 노드 구성 파일을 생성합니다	58
그리드 노드가 StorageGRID에서 기본 관리 노드를 검색하는 방법	75
VMware vSphere를 사용하여 StorageGRID용 그리드 노드를 가상 머신으로 배포합니다	75
Linux의 StorageGRID 노드에 대한 구성 파일 예	82
Linux에서 StorageGRID 구성 검증	84
Linux에서 StorageGRID 호스트 서비스를 시작합니다	85
StorageGRID 설치 REST API에 대해 알아보세요	86
StorageGRID 설치 API	86
StorageGRID 설치 문제 해결	87
스크립트 예시	89
RHEL 기반 StorageGRID에서 4개의 물리적 인터페이스를 하나의 본드로 통합하는 예시	89
Ubuntu 및 Debian에서 StorageGRID용 물리적 인터페이스를 통합하는 예	92

소프트웨어 기반 노드 설치

소프트웨어 기반 노드에 **StorageGRID**를 설치하는 빠른 시작

Linux 또는 VMware StorageGRID 노드를 설치하려면 다음의 주요 단계를 따르십시오.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

1

준비

- "StorageGRID 아키텍처 및 네트워크 토폴로지"에 대해 자세히 알아보십시오.
- "StorageGRID 네트워킹"의 구체적인 내용을 알아보세요.
- 재료를 모으고 준비하세요 ["필수 정보 및 자료"](#).
- (VMware 전용) ["VMware vSphere 하이퍼바이저, vCenter 및 ESX 호스트"](#)을(를) 설치 및 구성합니다.
- 필요한 ["CPU 및 RAM"](#)을(를) 준비하세요.
- ["스토리지 및 성능 요구 사항"](#)을(를) 제공하십시오.
- (Linux 전용) ["Linux 서버 준비"](#) StorageGRID 노드를 호스팅할 시스템입니다.

2

배포

그리드 노드를 배포합니다. 그리드 노드를 배포하면 해당 노드는 StorageGRID 시스템의 일부로 생성되어 하나 이상의 네트워크에 연결됩니다.

- (Linux 전용) 1단계에서 준비한 호스트에 소프트웨어 기반 그리드 노드를 배포하려면 Linux 명령줄과 ["노드 구성 파일"](#)을 사용합니다.
- (VMware 전용) VMware vSphere 웹 클라이언트, .vmdk 파일 및 .ovf 파일 템플릿 세트를 사용하여 1단계에서 준비한 서버에서 ["소프트웨어 기반 노드를 가상 머신\(VM\)으로 배포합니다."](#)하십시오.
- StorageGRID 어플라이언스 노드를 배포하려면 ["하드웨어 설치 빠른 시작"](#)을 따르십시오.

3

구성

모든 노드가 배포되면 Grid Manager를 사용하여 ["그리드를 구성하고 설치를 완료합니다"](#).

설치 자동화



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

Linux

시간을 절약하고 일관성을 유지하기 위해 StorageGRID 호스트 서비스 설치 및 그리드 노드 구성을 자동화할 수 있습니다.

- Ansible, Puppet 또는 Chef와 같은 표준 오케스트레이션 프레임워크를 사용하여 다음을 자동화하십시오.
 - Linux 설치
 - 네트워킹 및 스토리지 구성
 - 컨테이너 엔진 및 StorageGRID 호스트 서비스 설치
 - 가상 그리드 노드 배포

"[StorageGRID 호스트 서비스의 설치 및 구성 자동화](#)"을 참조하십시오.

- 그리드 노드를 배포한 후 "[StorageGRID 시스템의 구성을 자동화합니다.](#)" 설치 아카이브에 제공된 Python 구성 스크립트를 사용하십시오.
- "[어플라이언스 그리드 노드의 설치 및 구성을 자동화합니다.](#)"
- StorageGRID 배포에 대한 고급 개발자라면 "[설치 REST API](#)"를 사용하여 그리드 노드 설치를 자동화할 수 있습니다.

VMware

시간을 절약하고 일관성을 유지하기 위해 그리드 노드의 배포 및 구성과 StorageGRID 시스템의 구성을 자동화할 수 있습니다.

- "[VMware vSphere를 사용하여 그리드 노드 배포 자동화](#)".
- 그리드 노드를 배포한 후 "[StorageGRID 시스템의 구성을 자동화합니다.](#)" 설치 아카이브에 제공된 Python 구성 스크립트를 사용하십시오.
- "[어플라이언스 그리드 노드의 설치 및 구성을 자동화합니다.](#)"
- StorageGRID 배포에 대한 고급 개발자라면 "[설치 REST API](#)"를 사용하여 그리드 노드 설치를 자동화할 수 있습니다.

소프트웨어 기반 노드에 대한 설치 계획 및 준비

StorageGRID에 필요한 정보 및 자료

StorageGRID를 설치하기 전에 필요한 정보와 자료를 수집하고 준비하십시오.

필수 정보

네트워크 계획

각 StorageGRID 노드에 연결할 네트워크를 선택하십시오. StorageGRID는 트래픽 분리, 보안 및 관리 편의성을 위해 여러 네트워크를 지원합니다.

StorageGRID "[네트워킹 가이드라인](#)"을 참조하십시오.

네트워크 정보

각 그리드 노드에 할당할 IP 주소와 DNS 및 NTP 서버의 IP 주소입니다.

그리드 노드용 서버

배포하려는 StorageGRID 노드의 수와 유형을 지원하기에 충분한 리소스를 집합적으로 제공하는 서버 세트(물리적, 가상 또는 둘 다)를 식별하십시오.



StorageGRID 설치에 StorageGRID 어플라이언스(하드웨어) 스토리지 노드를 사용하지 않는 경우, 배터리 백업 쓰기 캐시(BBWC)가 있는 하드웨어 RAID 스토리지를 사용해야 합니다. StorageGRID는 가상 스토리지 영역 네트워크(vSAN), 소프트웨어 RAID 또는 RAID 보호 기능이 없는 구성을 지원하지 않습니다.

노드 마이그레이션 (Ubuntu 및 Debian에만 해당, 필요한 경우)

서비스 중단 없이 물리적 호스트에 대한 예약된 유지 보수를 수행하려면 ["노드 마이그레이션 요구 사항"](#)을(를) 이해해야 합니다.

관련 정보

["NetApp 상호 운용성 매트릭스 도구"](#)

필수 재료

NetApp StorageGRID 라이선스

유효하고 디지털 서명이 된 NetApp 라이선스가 있어야 합니다.



StorageGRID 설치 아카이브에는 테스트 및 개념 증명 그리드에 사용할 수 있는 비프로덕션 라이선스가 포함되어 있습니다.

StorageGRID 설치 아카이브

["StorageGRID 설치 아카이브를 다운로드하고 파일을 압축 해제합니다."](#)

서비스 노트북

StorageGRID 시스템은 서비스용 노트북을 통해 설치됩니다.

서비스용 노트북은 다음 조건을 충족해야 합니다.

- 네트워크 포트
- SSH 클라이언트(예: PuTTY)
- ["지원되는 웹 브라우저"](#)

StorageGRID 문서

- ["릴리스 노트"](#). NetApp 계정으로 로그인하거나 계정을 생성하여 릴리스 노트에 액세스해야 합니다.
- ["StorageGRID 관리 지침"](#)

StorageGRID 설치 파일 다운로드 및 압축 해제

StorageGRID 설치 아카이브를 다운로드하고 필요한 파일을 압축 해제해야 합니다. 필요에 따라 설치 패키지에 포함된 파일을 수동으로 확인할 수 있습니다.

단계

1. "[NetApp StorageGRID 다운로드 페이지](#)"로 이동합니다.
2. 최신 버전을 다운로드하려면 버튼을 선택하거나 드롭다운 메뉴에서 다른 버전을 선택한 후 *Go*를 선택하십시오.
3. NetApp 계정의 사용자 이름과 비밀번호로 로그인하세요.
4. 주의/MustRead 문구가 표시되면 해당 문구를 읽고 확인란을 선택하십시오.



StorageGRID 릴리스를 설치한 후에는 필요한 핫픽스를 적용해야 합니다. 자세한 내용은 "[복구 및 유지 관리 지침의 핫픽스 절차](#)"

5. 최종 사용자 라이선스 계약을 읽고 확인란을 선택한 다음 *Accept & Continue*를 선택하십시오.
6. **StorageGRID** 설치 열에서 사용 중인 소프트웨어 기반 노드 유형(RHEL, Ubuntu 또는 Debian, 또는 VMware)에 맞는 .tgz 또는 .zip 설치 아카이브를 선택하십시오.



서비스용 노트북에서 Windows를 실행 중인 경우 .zip 파일을 사용하십시오.

7. 설치 아카이브를 저장합니다.
8. 코드 서명 검증은 Linux 노드에서 수동으로 수행됩니다. 선택적으로 설치 아카이브를 검증해야 하는 경우:
 - a. StorageGRID 코드 서명 검증 패키지를 다운로드하십시오. 이 패키지의 파일 이름은 `StorageGRID_<version-number>_Code_Signature_Verification_Package.tar.gz` 형식을 사용하며, 여기서 `<version-number>` 은(는) StorageGRID 소프트웨어 버전입니다.
 - b. 다음 단계를 따르십시오 "[설치 파일을 수동으로 확인합니다](#)".
9. 설치 압축 파일에서 파일을 추출하세요.
10. 필요한 파일을 선택하십시오.

필요한 파일은 계획된 그리드 토폴로지와 StorageGRID 시스템 배포 방식에 따라 다릅니다.



표에 나열된 경로는 압축 해제된 설치 아카이브가 설치한 최상위 디렉터리를 기준으로 한 상대 경로입니다.

RHEL

경로 및 파일 이름	설명
	StorageGRID 다운로드 파일에 포함된 모든 파일을 설명하는 텍스트 파일입니다.
	제품에 대한 어떠한 지원 권한도 제공하지 않는 무료 라이선스입니다.
	RHEL 호스트에 StorageGRID 노드 이미지를 설치하기 위한 RPM 패키지입니다.
	RHEL 호스트에 StorageGRID 호스트 서비스를 설치하기 위한 RPM 패키지입니다.
배포 스크립팅 도구	설명
	StorageGRID 시스템 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	StorageGRID 어플라이언스의 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	<pre>`configure-storagegrid.py`</pre> 스크립트와 함께 사용할 수 있는 예제 구성 파일입니다.
	싱글 사인온(SSO)이 활성화된 경우 그리드 관리 API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제입니다. 이 스크립트는 Ping Federate 통합에도 사용할 수 있습니다.
	<pre>`configure-storagegrid.py`</pre> 스크립트와 함께 사용할 빈 구성 파일입니다.
	StorageGRID 컨테이너 배포를 위해 RHEL 호스트를 구성하는 Ansible 역할 및 플레이북 예시입니다. 필요에 따라 역할 또는 플레이북을 사용자 지정할 수 있습니다.

경로 및 파일 이름	설명
	Active Directory 또는 Ping Federate를 사용하여 단일 로그인(SSO)이 활성화된 경우 그리드 관리 API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제입니다.
	이 스크립트는 함께 실행되는 `storagegrid-ssoauth-azure.py` Python 스크립트에서 호출되어 Azure와의 SSO 상호 작용을 수행합니다.
	StorageGRID에 대한 API 스키마. 참고: 업그레이드를 수행하기 전에, 업그레이드 호환성 테스트를 위한 비프로덕션 StorageGRID 환경이 없는 경우, 이러한 스키마를 사용하여 StorageGRID 관리 API를 사용하도록 작성한 코드가 새 StorageGRID 릴리스와 호환되는지 확인할 수 있습니다.

Ubuntu 또는 Debian

경로 및 파일 이름	설명
	StorageGRID 다운로드 파일에 포함된 모든 파일을 설명하는 텍스트 파일입니다.
	테스트 및 개념 증명 배포에 사용할 수 있는 비프로덕션 NetApp 라이선스 파일입니다.
	Ubuntu 또는 Debian 호스트에 StorageGRID 노드 이미지를 설치하기 위한 DEB 패키지입니다.
	파일의 MD5 체크섬 `/debs/storagegrid-webscale-images-version-SHA.deb`입니다.
	Ubuntu 또는 Debian 호스트에 StorageGRID 호스트 서비스를 설치하기 위한 DEB 패키지입니다.
배포 스크립팅 도구	설명
	StorageGRID 시스템 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	StorageGRID 어플라이언스의 구성을 자동화하는 데 사용되는 Python 스크립트입니다.

경로 및 파일 이름	설명
	싱글 사인온(SSO)이 활성화된 경우 그리드 관리 API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제입니다. 이 스크립트는 Ping Federate 통합에도 사용할 수 있습니다.
	<code>`configure-storagegrid.py`</code> 스크립트와 함께 사용할 수 있는 예제 구성 파일입니다.
	<code>`configure-storagegrid.py`</code> 스크립트와 함께 사용할 빈 구성 파일입니다.
	StorageGRID 컨테이너 배포를 위해 Ubuntu 또는 Debian 호스트를 구성하는 Ansible 역할 및 플레이북 예시입니다. 필요에 따라 역할 또는 플레이북을 사용자 지정할 수 있습니다.
	Active Directory 또는 Ping Federate를 사용하여 단일 로그인(SSO)이 활성화된 경우 그리드 관리 API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제입니다.
	이 스크립트는 함께 실행되는 <code>`storagegrid-ssoauth-azure.py`</code> Python 스크립트에서 호출되어 Azure와의 SSO 상호 작용을 수행합니다.
	StorageGRID에 대한 API 스키마. 참고: 업그레이드를 수행하기 전에, 업그레이드 호환성 테스트를 위한 비프로덕션 StorageGRID 환경이 없는 경우, 이러한 스키마를 사용하여 StorageGRID 관리 API를 사용하도록 작성한 코드가 새 StorageGRID 릴리스와 호환되는지 확인할 수 있습니다.

VMware

경로 및 파일 이름	설명
	StorageGRID 다운로드 파일에 포함된 모든 파일을 설명하는 텍스트 파일입니다.

경로 및 파일 이름	설명
	제품에 대한 어떠한 지원 권한도 제공하지 않는 무료 라이선스입니다.
	그리드 노드 가상 머신을 생성하기 위한 템플릿으로 사용되는 가상 머신 디스크 파일입니다.
	기본 관리 노드를 배포하기 위한 Open Virtualization Format 템플릿 파일(.ovf)과 매니페스트 파일(.mf)입니다.
	기본이 아닌 관리 노드를 배포하기 위한 템플릿 파일(.ovf)과 매니페스트 파일(.mf)입니다.
	게이트웨이 노드를 배포하기 위한 템플릿 파일(.ovf) 및 매니페스트 파일(.mf)입니다.
	가상 머신 기반 스토리지 노드를 배포하기 위한 템플릿 파일(.ovf)과 매니페스트 파일(.mf)입니다.
배포 스크립팅 도구	설명
	가상 그리드 노드 배포를 자동화하는 데 사용되는 Bash 셸 스크립트입니다.
	<pre>`deploy-vmware-ovftool.sh`</pre> 스크립트와 함께 사용할 수 있는 예제 구성 파일입니다.
	StorageGRID 시스템 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	StorageGRID 어플라이언스의 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	싱글 사인온(SSO)이 활성화된 경우 그리드 관리 API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제입니다. 이 스크립트는 Ping Federate 통합에도 사용할 수 있습니다.

경로 및 파일 이름	설명
	<code>`configure-storagegrid.py`</code> 스크립트와 함께 사용할 수 있는 예제 구성 파일입니다.
	<code>`configure-storagegrid.py`</code> 스크립트와 함께 사용할 빈 구성 파일입니다.
	Active Directory 또는 Ping Federate를 사용하여 단일 로그인(SSO)이 활성화된 경우 그리드 관리 API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제입니다.
	이 스크립트는 함께 실행되는 <code>`storagegrid-ssoauth-azure.py`</code> Python 스크립트에서 호출되어 Azure와의 SSO 상호 작용을 수행합니다.
	StorageGRID에 대한 API 스키마. 참고: 업그레이드를 수행하기 전에, 업그레이드 호환성 테스트를 위한 비프로덕션 StorageGRID 환경이 없는 경우, 이러한 스키마를 사용하여 StorageGRID 관리 API를 사용하도록 작성한 코드가 새 StorageGRID 릴리스와 호환되는지 확인할 수 있습니다.

StorageGRID 설치 파일을 확인하십시오

필요한 경우 StorageGRID 설치 아카이브에 있는 파일을 수동으로 확인할 수 있습니다.

시작하기 전에

다음에서 ["검증 패키지를 다운로드했습니다."](#)을(를) 받았습니다. ["NetApp StorageGRID 다운로드 페이지"](#)

단계

1. 검증 패키지에서 아티팩트를 추출합니다.

```
tar -xf StorageGRID_12.0.0_Code_Signature_Verification_Package.tar.gz
```

2. 이러한 아티팩트가 추출되었는지 확인하십시오.

- 리프 인증서: Leaf-Cert.pem
- 인증서 체인: CA-Int-Cert.pem

- 타임스탬프 응답 체인: TS-Cert.pem
- 체크섬 파일: sha256sum
- 체크섬 서명: sha256sum.sig
- 타임스탬프 응답 파일: sha256sum.sig.tsr

3. 체인을 사용하여 리프 인증서가 유효한지 확인하십시오.

예: `openssl verify -CAfile CA-Int-Cert.pem Leaf-Cert.pem`

예상 출력: Leaf-Cert.pem: OK

4. 2단계에서 리프 인증서가 만료되어 실패한 경우, tsr 파일을 사용하여 확인하십시오.

예: `openssl ts -CAfile CA-Int-Cert.pem -untrusted TS-Cert.pem -verify -data sha256sum.sig -in sha256sum.sig.tsr`

예상 출력 내용: Verification: OK

5. 리프 인증서에서 공개 키 파일을 생성합니다.

예: `openssl x509 -pubkey -noout -in Leaf-Cert.pem > Leaf-Cert.pub`

예상 출력: 없음

6. 공개 키를 사용하여 sha256sum 파일을 'sha256sum.sig'에 대해 검증하십시오.

예: `openssl dgst -sha256 -verify Leaf-Cert.pub -signature sha256sum.sig sha256sum`

예상 출력: Verified OK

7. 새로 생성된 체크섬과 sha256sum 파일 내용을 대조하여 검증합니다.

예: `sha256sum -c sha256sum`

예상 출력: `<filename>: OK`
 ``<filename>``는 다운로드한 압축 파일의 이름입니다.

8. "나머지 단계를 완료합니다" 설치 아카이브에서 적절한 파일을 추출하고 선택합니다.

StorageGRID의 소프트웨어 요구 사항

가상 머신을 사용하여 모든 유형의 StorageGRID 노드를 호스팅할 수 있습니다. 그리드 노드 하나당 가상 머신이 하나씩 필요합니다.

RHEL

RHEL에 StorageGRID를 설치하려면 일부 타사 소프트웨어 패키지를 설치해야 합니다. 일부 지원되는 Linux 배포판에는 이러한 패키지가 기본적으로 포함되어 있지 않습니다. StorageGRID 설치 테스트에 사용된 소프트웨어 패키지 버전은 이 페이지에 나열되어 있습니다.

Linux 배포판 및 컨테이너 런타임 설치 옵션에서 이러한 패키지 중 하나라도 필요로 하는 경우, 해당 패키지가 Linux 배포판에서 자동으로 설치되지 않으면, 공급업체 또는 Linux 배포판 지원 업체에서 제공하는 여기에 나열된 버전 중 하나를 설치하십시오. 그렇지 않으면 공급업체에서 제공하는 기본 패키지 버전을 사용하십시오.

모든 설치 옵션에는 Podman 또는 Docker 중 하나가 필요합니다. 두 패키지를 모두 설치하지 마십시오. 설치 옵션에 필요한 패키지만 설치하십시오.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로서 Docker 지원이 더 이상 권장되지 않습니다. Docker는 향후 릴리스에서 다른 컨테이너 엔진으로 대체될 예정입니다.

테스트에 사용된 **Python** 버전

- 3.5.2-2
- 3.6.8-2
- 3.6.8-38
- 3.6.9-1
- 3.7.3-1
- 3.8.10-0
- 3.9.2-1
- 3.9.10-2
- 3.9.16-1
- 3.10.6-1
- 3.11.2-6

Podman 버전 테스트 완료

- 3.2.3-0
- 3.4.4+ds1
- 4.1.1-7
- 4.2.0-11
- 4.3.1+ds1-8+b1
- 4.4.1-8
- 4.4.1-12

테스트 완료된 **Docker** 버전



Docker 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다.

- Docker-CE 20.10.7

- Docker-CE 20.10.20-3
- Docker-CE 23.0.6-1
- Docker-CE 24.0.2-1
- Docker-CE 24.0.4-1
- Docker-CE 24.0.5-1
- Docker-CE 24.0.7-1
- 1.5-2

Ubuntu 및 Debian

Ubuntu 또는 Debian에 StorageGRID를 설치하려면 일부 타사 소프트웨어 패키지를 설치해야 합니다. 일부 지원되는 Linux 배포판에는 이러한 패키지가 기본적으로 포함되어 있지 않습니다. StorageGRID 설치 테스트에 사용된 소프트웨어 패키지 버전은 이 페이지에 나열되어 있습니다.

Linux 배포판 및 컨테이너 런타임 설치 옵션에서 이러한 패키지 중 하나라도 필요로 하는 경우, 해당 패키지가 Linux 배포판에서 자동으로 설치되지 않으면, 공급업체 또는 Linux 배포판 지원 업체에서 제공하는 여기에 나열된 버전 중 하나를 설치하십시오. 그렇지 않으면 공급업체에서 제공하는 기본 패키지 버전을 사용하십시오.

모든 설치 옵션에는 Podman 또는 Docker 중 하나가 필요합니다. 두 패키지를 모두 설치하지 마십시오. 설치 옵션에 필요한 패키지만 설치하십시오.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로서 Docker 지원이 더 이상 권장되지 않습니다. Docker는 향후 릴리스에서 다른 컨테이너 엔진으로 대체될 예정입니다.

테스트에 사용된 Python 버전

- 3.5.2-2
- 3.6.8-2
- 3.6.8-38
- 3.6.9-1
- 3.7.3-1
- 3.8.10-0
- 3.9.2-1
- 3.9.10-2
- 3.9.16-1
- 3.10.6-1
- 3.11.2-6

Podman 버전 테스트 완료

- 3.2.3-0
- 3.4.4+ds1
- 4.1.1-7
- 4.2.0-11
- 4.3.1+ds1-8+b1

- 4.4.1-8
- 4.4.1-12

테스트 완료된 **Docker** 버전



Docker 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다.

- Docker-CE 20.10.7
- Docker-CE 20.10.20-3
- Docker-CE 23.0.6-1
- Docker-CE 24.0.2-1
- Docker-CE 24.0.4-1
- Docker-CE 24.0.5-1
- Docker-CE 24.0.7-1
- 1.5-2

VMware

VMware vSphere 하이퍼바이저

VMware vSphere Hypervisor를 준비된 물리적 서버에 설치해야 합니다. VMware 소프트웨어를 설치하기 전에 하드웨어가 올바르게 구성되어 있어야 합니다(펌웨어 버전 및 BIOS 설정 포함).

- 설치하려는 StorageGRID 시스템의 네트워킹을 지원하도록 하이퍼바이저에서 필요한 네트워킹을 구성하십시오.

"네트워킹 가이드라인"

- 그리드 노드를 호스팅하는 데 필요한 가상 머신과 가상 디스크를 저장할 수 있을 만큼 데이터 저장소의 크기가 충분한지 확인하십시오.
- 데이터 저장소를 두 개 이상 생성하는 경우, 가상 머신을 생성할 때 각 그리드 노드에 사용할 데이터 저장소를 쉽게 식별할 수 있도록 각 데이터 저장소에 이름을 지정하십시오.

ESX 호스트 구성 요구 사항



각 ESX 호스트에서 NTP(네트워크 시간 프로토콜)를 올바르게 구성해야 합니다. 호스트 시간이 올바르게 맞지 않으면 데이터 손실을 포함한 부정적인 영향이 발생할 수 있습니다.

VMware 구성 요구 사항

StorageGRID 노드를 배포하기 전에 VMware vSphere 및 vCenter를 설치하고 구성해야 합니다.

지원되는 VMware vSphere 하이퍼바이저 및 VMware vCenter Server 소프트웨어 버전에 대한 자세한 내용은 "[NetApp 상호 운용성 매트릭스 도구](#)"를 참조하십시오.

이러한 VMware 제품을 설치하는 데 필요한 단계는 VMware 설명서를 참조하십시오.

StorageGRID의 CPU 및 RAM 요구 사항

StorageGRID 소프트웨어를 설치하기 전에 하드웨어가 StorageGRID 시스템을 지원할 수 있도록 준비되었는지 확인하고 구성하십시오.

각 StorageGRID 노드에는 다음과 같은 최소 리소스가 필요합니다.

- CPU 코어: 노드당 8개
- RAM: 사용 가능한 총 RAM 용량과 시스템에서 실행 중인 non-StorageGRID 소프트웨어 양에 따라 달라집니다.
 - 일반적으로 노드당 최소 24GB, 그리고 전체 시스템 RAM보다 2~16GB 적은 용량이 필요합니다.
 - 약 5,000개의 버킷을 보유할 각 테넌트에 대해 최소 64GB

소프트웨어 기반 메타데이터 전용 노드 리소스는 기존 스토리지 노드 리소스와 일치해야 합니다. 예를 들면 다음과 같습니다.

- 기존 StorageGRID 사이트에서 SG6000 또는 SG6100 어플라이언스를 사용하는 경우, 소프트웨어 기반 메타데이터 전용 노드는 다음 최소 요구 사항을 충족해야 합니다.
 - 128GB RAM
 - 8코어 CPU
 - Cassandra 데이터베이스(rangedb/0)용 8TB SSD 또는 동급 스토리지
- 기존 StorageGRID 사이트에서 24GB RAM, 8코어 CPU, 3TB 또는 4TB의 메타데이터 스토리지를 갖춘 가상 스토리지 노드를 사용하는 경우 소프트웨어 기반 메타데이터 전용 노드도 유사한 리소스(24GB RAM, 8코어 CPU, 4TB의 메타데이터 스토리지(rangedb/0))를 사용해야 합니다.

새로운 StorageGRID 사이트를 추가할 때, 새 사이트의 총 메타데이터 용량은 최소한 기존 StorageGRID 사이트와 일치해야 하며, 새 사이트의 리소스는 기존 StorageGRID 사이트의 스토리지 노드와 일치해야 합니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

Linux

각 물리적 또는 가상 호스트에서 실행할 StorageGRID 노드 수가 사용 가능한 CPU 코어 수 또는 물리적 RAM을 초과하지 않도록 하십시오. 호스트가 StorageGRID 실행 전용이 아닌 경우(권장하지 않음) 다른 애플리케이션의 리소스 요구 사항도 고려해야 합니다.

VMware

VMware는 가상 머신당 하나의 노드를 지원합니다. StorageGRID 노드가 사용 가능한 물리적 RAM을 초과하지 않도록 하십시오. 각 가상 머신은 StorageGRID 실행 전용이어야 합니다.



CPU 및 메모리 사용량을 정기적으로 모니터링하여 이러한 리소스가 워크로드를 지속적으로 수용할 수 있도록 하십시오. 예를 들어 가상 스토리지 노드의 RAM 및 CPU 할당량을 두 배로 늘리면 StorageGRID 어플라이언스 노드에 제공되는 것과 유사한 리소스를 확보할 수 있습니다. 또한 노드당 메타데이터 양이 500GB를 초과하는 경우 노드당 RAM을 48GB 이상으로 늘리는 것을 고려하십시오. 객체 메타데이터 스토리지 관리, 메타데이터 예약 공간 설정 증가, CPU 및 메모리 사용량 모니터링에 대한 자세한 내용은 ["관리"](#), ["모니터링"](#) 및 ["업그레이드"](#) StorageGRID에 대한 지침을 참조하십시오.

기본 물리적 호스트에서 하이퍼스레딩이 활성화된 경우 노드당 8개의 가상 코어(4개의 물리적 코어)를 제공할 수 있습니다. 기본 물리적 호스트에서 하이퍼스레딩이 활성화되지 않은 경우 노드당 8개의 물리적 코어를 제공해야 합니다.

가상 머신을 호스트로 사용하고 VM의 크기와 개수를 제어할 수 있는 경우, StorageGRID 노드마다 하나의 VM을 사용하고 VM의 크기를 그에 맞게 조정해야 합니다.

(RHEL, Debian 및 Ubuntu에만 해당) 운영 구축 환경에서는 동일한 물리적 스토리지 하드웨어 또는 가상 호스트에서 여러 스토리지 노드를 실행해서는 안 됩니다. 단일 StorageGRID 배포 환경의 각 스토리지 노드는 자체적으로 격리된 장애 도메인에 있어야 합니다. 단일 하드웨어 장애가 단일 스토리지 노드에만 영향을 미치도록 하면 객체 데이터의 내구성과 가용성을 극대화할 수 있습니다.

참고 항목: "[스토리지 및 성능 요구 사항](#)".

StorageGRID의 스토리지 및 성능 요구 사항

StorageGRID 노드의 스토리지 요구 사항을 이해해야 초기 구성 및 향후 스토리지 확장을 지원하기에 충분한 공간을 제공할 수 있습니다.

스토리지 및 성능 요구 사항은 소프트웨어 기반 노드 구현 방식에 따라 다릅니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 "[NetApp 상호 운용성 매트릭스 툴\(IMT\)](#)"을 참조하십시오.

스토리지 범주

StorageGRID 노드에는 세 가지 논리적 범주의 스토리지가 필요합니다.

- 컨테이너 풀 — 노드 컨테이너용 성능 등급(10K SAS 또는 SSD) 스토리지입니다. 이 스토리지는 StorageGRID 노드를 지원할 호스트에 컨테이너 엔진을 설치 및 구성할 때 컨테이너 엔진 스토리지 드라이버에 할당됩니다.
- 시스템 데이터 — 시스템 데이터 및 트랜잭션 로그의 노드별 영구 스토리지를 위한 성능 등급(10K SAS 또는 SSD) 스토리지로, StorageGRID 호스트 서비스가 이를 사용하여 개별 노드에 매핑합니다.
- 객체 데이터 — 객체 데이터 및 객체 메타데이터의 영구 스토리지를 위한 성능 계층(10K SAS 또는 SSD) 스토리지 및 용량 계층(NL-SAS/SATA) 대용량 스토리지.

모든 스토리지 범주에는 RAID 기반 블록 장치를 사용해야 합니다. 비중복 디스크, SSD 또는 JBOD는 지원되지 않습니다. 공유 또는 로컬 RAID 스토리지를 모든 스토리지 범주에 사용할 수 있지만, StorageGRID에서 노드 마이그레이션 기능을 사용하려면 시스템 데이터와 객체 데이터를 모두 공유 스토리지에 저장해야 합니다. 자세한 내용은 "[Node 컨테이너 마이그레이션 요구 사항](#)"을 참조하십시오.

성능 요구사항

컨테이너 풀, 시스템 데이터 및 객체 메타데이터에 사용되는 볼륨의 성능은 시스템의 전반적인 성능에 상당한 영향을 미칩니다. 지연 시간, 초당 입출력 작업 수(IOPS) 및 처리량 측면에서 적절한 디스크 성능을 확보하려면 이러한 볼륨에는 성능 등급(10K SAS 또는 SSD) 스토리지를 사용해야 합니다. 객체 데이터의 영구 스토리지에는 용량 등급(NL-SAS/SATA) 스토리지를 사용할 수 있습니다.

컨테이너 풀, 시스템 데이터 및 객체 데이터에 사용되는 볼륨에는 쓰기 캐시가 활성화되어 있어야 합니다. 캐시는 보호된 미디어 또는 영구 미디어에 있어야 합니다.

NetApp ONTAP 스토리지를 사용하는 호스트에 대한 요구 사항

StorageGRID 노드가 NetApp ONTAP 시스템에서 할당된 스토리지를 사용하는 경우 해당 볼륨에 FabricPool 티어링 정책이 활성화되어 있지 않은지 확인하십시오. StorageGRID 노드에서 사용하는 볼륨에 대해 FabricPool 티어링을 비활성화하면 문제 해결 및 스토리지 운영이 간소화됩니다.



FabricPool를 사용하여 StorageGRID와 관련된 데이터를 StorageGRID 자체로 계층화하지 마십시오. StorageGRID 데이터를 StorageGRID로 다시 계층화하면 문제 해결 및 운영 복잡성이 증가합니다.

필요한 호스트 수

각 StorageGRID 사이트에는 최소 3개의 스토리지 노드가 필요합니다.



운영 구축 시 단일 물리적 또는 가상 호스트에서 스토리지 노드를 두 개 이상 실행하지 마십시오. 각 스토리지 노드에 전용 호스트를 사용하면 장애 발생 시 격리된 환경을 구축할 수 있습니다.

`${post_edited_translations.segment}`



디스크 스냅샷은 그리드 노드를 복원하는 데 사용할 수 없습니다. 대신 "[그리드 노드 복구](#)" 각 노드 유형에 대한 절차를 참조하십시오.

노드당 스토리지 볼륨 수

다음 표는 각 호스트에 필요한 스토리지 볼륨(LUN)의 수와 각 LUN에 필요한 최소 크기를 보여줍니다. 이는 해당 호스트에 배포될 노드를 기준으로 합니다.

테스트된 최대 LUN 크기는 39TB입니다.



이 수치는 전체 그리드가 아닌 각 호스트에 대한 수치입니다.

<code>\${post_edited_translations.segment}</code>	<code>\${post_edited_translations.segment}</code>	LUN 개수	최소 크기/LUN
컨테이너 엔진 스토리지 풀	컨테이너 풀	1	<code>\${post_edited_translations.segment}</code>
/var/local 볼륨	시스템 데이터	이 호스트의 각 노드마다 1개씩	100 GB

<code>\${post_edited_translation.segment}</code>	<code>\${post_edited_translation.segment}</code>	LUN 개수	최소 크기/LUN
스토리지 노드	객체 데이터	<code>\${post_edited_translation.segment}</code> 참고: Linux 소프트웨어 기반 스토리지 노드는 1~48개의 스토리지 볼륨을 가질 수 있습니다. VMware 소프트웨어 기반 스토리지 노드는 1~16개의 스토리지 볼륨을 가질 수 있습니다. 최소 3개의 스토리지 볼륨을 권장합니다.	12 TB (최소 4 TB/LUN) 최대 테스트 LUN 크기: 39TB. 자세한 내용은 스토리지 노드에 필요한 스토리지 용량 을 참조하십시오.
스토리지 노드(메타데이터 전용)	객체 메타데이터	1	최소 4TB/LUN 최대 테스트 LUN 크기: 39TB. 자세한 내용은 스토리지 노드에 필요한 스토리지 용량 을 참조하십시오. 참고: 메타데이터 전용 스토리지 노드에는 rangedb가 하나만 필요합니다.
관리자 노드 감사 로그	시스템 데이터	이 호스트의 각 관리 노드마다 1개	200 GB
관리자 노드 테이블	시스템 데이터	이 호스트의 각 관리 노드마다 1개	200 GB



구성된 감사 수준, S3 객체 키 이름과 같은 사용자 입력 크기, 보존해야 하는 감사 로그 데이터 양에 따라 각 관리 노드의 감사 로그 LUN 크기를 늘려야 할 수 있습니다. 일반적으로 그리드 환경에서는 S3 작업당 약 1KB의 감사 데이터가 생성되므로 200GB LUN은 하루 7천만 건의 작업 또는 초당 800건의 작업을 2~3일 동안 지원할 수 있습니다.

호스트에 필요한 최소 스토리지 공간

다음 표는 각 노드 유형에 필요한 최소 스토리지 공간을 보여줍니다. 이 표를 사용하여 호스트에 배포할 노드 유형에 따라 각 스토리지 범주별로 호스트에 제공해야 하는 최소 스토리지 공간을 확인할 수 있습니다.



디스크 스냅샷은 그리드 노드를 복원하는 데 사용할 수 없습니다. 대신 "[그리드 노드 복구](#)" 각 노드 유형에 대한 절차를 참조하십시오.

각 노드 호스트에는 운영 체제용으로 100GB LUN이 필요합니다.

노드 유형	컨테이너 풀	시스템 데이터	객체 데이터
스토리지 노드	100 GB	100 GB	4,000 GB
관리자 노드	100 GB	500GB (LUN 3개)	해당 사항 없음
게이트웨이 노드	100 GB	100 GB	해당 사항 없음

예시: 호스트 또는 가상 머신의 스토리지 요구사항 계산

동일한 호스트 또는 가상 머신에 스토리지 노드, 관리 노드, 게이트웨이 노드 이렇게 세 개의 노드를 배포할 계획이라고 가정해 보겠습니다. 호스트에는 최소 9개의 스토리지 볼륨을 제공해야 합니다. 노드 컨테이너에는 최소 300GB의 성능 계층 스토리지가, 시스템 데이터 및 트랜잭션 로그에는 700GB의 성능 계층 스토리지가, 객체 데이터에는 12TB의 용량 계층 스토리지가 필요합니다.

Linux 호스트 예시

노드 유형	<code>\${post_edited_translations.segment}</code>	LUN 개수	LUN 크기
스토리지 노드	컨테이너 엔진 스토리지 풀	1	<code>\${post_edited_translations.segment}</code>
스토리지 노드	<code>/var/local</code> 볼륨	1	100 GB
스토리지 노드	객체 데이터	3	<code>\${post_edited_translations.segment}</code>
관리자 노드	<code>/var/local</code> 볼륨	1	100 GB
관리자 노드	관리자 노드 감사 로그	1	200 GB
관리자 노드	관리자 노드 테이블	1	200 GB
게이트웨이 노드	<code>/var/local</code> 볼륨	1	100 GB
총		9	컨테이너 풀: 300 GB 시스템 데이터: 700 GB 객체 데이터: 12,000 GB

VMware 가상 머신 예시

노드 유형	<code>\${post_edited_translations.segment}</code>	LUN 개수	LUN 크기
스토리지 노드	OS 볼륨	1	100 GB
스토리지 노드	객체 데이터	3	<code>\${post_edited_translations.segment}</code>
관리자 노드	OS 볼륨	1	100 GB
관리자 노드	관리자 노드 감사 로그	1	200 GB
관리자 노드	관리자 노드 테이블	1	200 GB
게이트웨이 노드	OS 볼륨	1	100 GB

노드 유형	<code>\${post_edited_translations.segment}</code>	LUN 개수	LUN 크기
총		8	시스템 데이터: 700 GB 객체 데이터: 12,000 GB

스토리지 노드에 대한 특정 스토리지 요구 사항

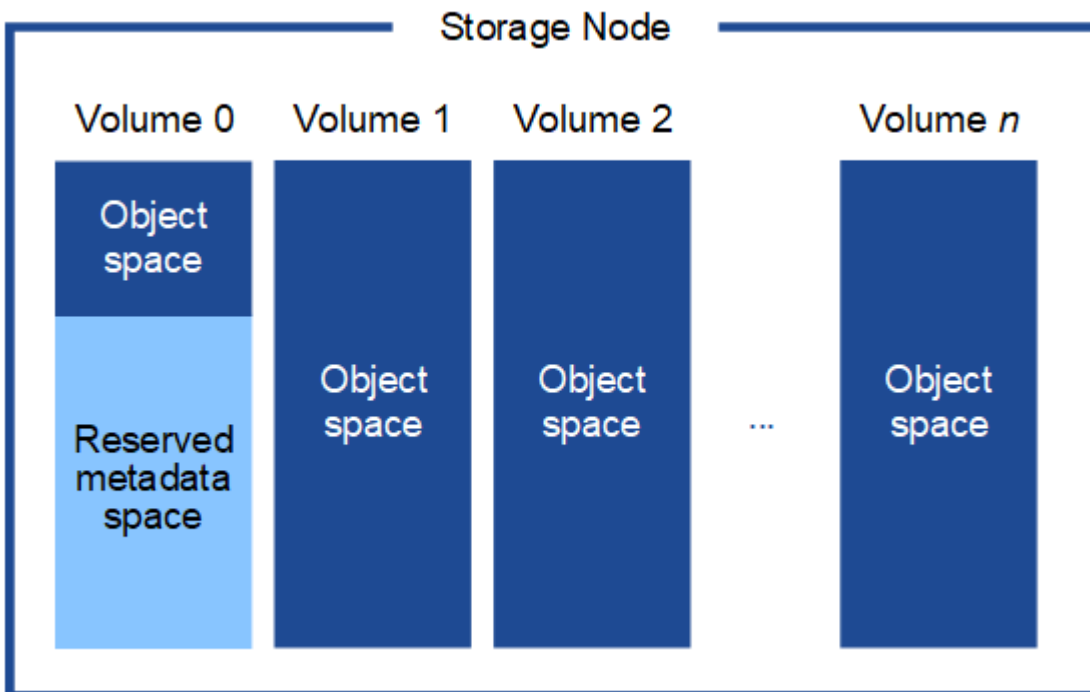
Linux와 VMware는 스토리지 노드에 대한 스토리지 요구 사항이 다릅니다.

- `${post_edited_translations.segment}`
- VMware 소프트웨어 기반 스토리지 노드는 1개에서 16개까지의 스토리지 볼륨을 가질 수 있습니다.
- 3개 이상의 스토리지 볼륨을 권장합니다.
- 각 스토리지 볼륨은 4TB 이상이어야 합니다.



어플라이언스 스토리지 노드는 최대 48개의 스토리지 볼륨을 가질 수 있습니다.

그림에서 보는 바와 같이 StorageGRID는 각 스토리지 노드의 스토리지 볼륨 0에 객체 메타데이터를 위한 공간을 예약합니다. 스토리지 볼륨 0 및 스토리지 노드의 다른 스토리지 볼륨에 남아 있는 공간은 모두 객체 데이터 저장에만 사용됩니다.



이중화를 제공하고 객체 메타데이터를 손실로부터 보호하기 위해 StorageGRID는 시스템 내 모든 객체의 메타데이터 사본 3개를 각 사이트에 저장합니다. 이 3개의 객체 메타데이터 사본은 각 사이트의 모든 스토리지 노드에 고르게 분산됩니다.

메타데이터 전용 스토리지 노드를 사용하여 그리드를 설치할 경우, 그리드에는 객체 스토리지를 위한 최소 개수의 노드도 포함되어야 합니다. 메타데이터 전용 스토리지 노드에 대한 자세한 내용은 "[스토리지 노드의 유형](#)"을

참조하십시오.

- 단일 사이트 그리드의 경우 객체 및 메타데이터에 대해 최소 두 개의 스토리지 노드가 구성됩니다.
- `#{post_edited_translations.segment}`

`#{post_edited_translations.segment}`

- `#{post_edited_translations.segment}`



스토리지 노드에 스토리지 볼륨을 하나만 사용하고 해당 볼륨에 4TB 이하의 용량을 할당하는 경우, 스토리지 노드는 시작 시 스토리지 읽기 전용 상태로 전환되어 객체 메타데이터만 저장할 수 있습니다.



볼륨 0(비프로덕션 용도 전용)에 500GB 미만의 용량을 할당하면 스토리지 볼륨 용량의 10%가 메타데이터용으로 예약됩니다.

- 소프트웨어 기반 메타데이터 전용 노드 리소스는 기존 스토리지 노드 리소스와 일치해야 합니다. 예를 들면 다음과 같습니다.
 - 기존 StorageGRID 사이트에서 SG6000 또는 SG6100 어플라이언스를 사용하는 경우, 소프트웨어 기반 메타데이터 전용 노드는 다음 최소 요구 사항을 충족해야 합니다.

- 128GB RAM
- 8코어 CPU
- Cassandra 데이터베이스(rangedb/0)용 8TB SSD 또는 동급 스토리지

- 기존 StorageGRID 사이트에서 24GB RAM, 8코어 CPU, 3TB 또는 4TB의 메타데이터 스토리지를 갖춘 가상 스토리지 노드를 사용하는 경우 소프트웨어 기반 메타데이터 전용 노드도 유사한 리소스(24GB RAM, 8코어 CPU, 4TB의 메타데이터 스토리지(rangedb/0))를 사용해야 합니다.

새로운 StorageGRID 사이트를 추가할 때, 새 사이트의 총 메타데이터 용량은 최소한 기존 StorageGRID 사이트와 일치해야 하며, 새 사이트의 리소스는 기존 StorageGRID 사이트의 스토리지 노드와 일치해야 합니다.

- 새 시스템(StorageGRID 11.6 이상)을 설치하고 각 스토리지 노드에 128GB 이상의 RAM이 있는 경우, 볼륨 0에 8TB 이상을 할당하십시오. 볼륨 0에 더 큰 값을 사용하면 각 스토리지 노드의 메타데이터에 허용되는 공간을 늘릴 수 있습니다.
- 사이트에 대해 서로 다른 스토리지 노드를 구성할 때 가능하면 볼륨 0에 대해 동일한 설정을 사용하십시오. 사이트에 크기가 다른 스토리지 노드가 포함된 경우 볼륨 0의 크기가 가장 작은 스토리지 노드가 해당 사이트의 메타데이터 용량을 결정합니다.

자세한 내용은 "[객체 메타데이터 저장소 관리](#)"를 참조하십시오.

Linux용 StorageGRID 노드 컨테이너 마이그레이션 요구 사항

노드 마이그레이션 기능을 사용하면 노드를 한 호스트에서 다른 호스트로 수동으로 이동할 수 있습니다. 일반적으로 두 호스트는 동일한 물리적 데이터 센터에 있습니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 "[NetApp 상호 운용성 매트릭스 툴\(IMT\)](#)"을 참조하십시오.

노드 마이그레이션을 통해 그리드 운영을 중단하지 않고 물리적 호스트 유지 관리를 수행할 수 있습니다. 모든 StorageGRID 노드를 한 번에 하나씩 다른 호스트로 이동한 후 물리적 호스트를 오프라인으로 전환합니다. 노드 마이그레이션에는 각 노드당 짧은 다운타임만 필요하며 그리드 서비스의 운영이나 가용성에 영향을 미치지 않습니다.

StorageGRID 노드 마이그레이션 기능을 사용하려면 배포 환경이 추가 요구 사항을 충족해야 합니다.

- 단일 물리적 데이터 센터 내 호스트 전체에서 일관된 네트워크 인터페이스 이름
- StorageGRID 메타데이터 및 객체 리포지토리 볼륨을 위한 공유 스토리지로, 단일 물리적 데이터 센터 내의 모든 호스트에서 액세스할 수 있습니다. 예를 들어 NetApp E-Series 스토리지 어레이를 사용할 수 있습니다.

가상 호스트를 사용하고 있고 기본 하이퍼바이저 계층에서 VM 마이그레이션을 지원하는 경우, StorageGRID의 노드 마이그레이션 기능 대신 이 기능을 사용하는 것이 좋습니다. 이 경우 추가 요구 사항은 무시해도 됩니다.

마이그레이션 또는 하이퍼바이저 유지 관리를 수행하기 전에 노드를 정상적으로 종료하십시오. 자세한 내용은 "[그리드 노드 종료](#)"에 대한 지침을 참조하십시오.

VMware Live Migration은 지원되지 않습니다.

VMware VM에 베어메탈 설치를 수행할 때 OpenStack Live Migration 및 VMware live vMotion은 가상 머신의 시계 시간을 변경시키며, 모든 유형의 그리드 노드에서 지원되지 않습니다. 드물지만 시계 시간이 잘못되면 데이터 손실이나 구성 업데이트 오류가 발생할 수 있습니다.

콜드 마이그레이션이 지원됩니다. 콜드 마이그레이션에서는 StorageGRID 노드를 종료한 후 호스트 간에 마이그레이션합니다. 자세한 내용은 "[그리드 노드 종료](#)" 지침을 참조하십시오.

일관된 네트워크 인터페이스 이름

노드를 한 호스트에서 다른 호스트로 이동하려면 StorageGRID 호스트 서비스는 노드가 현재 위치에서 사용하는 외부 네트워크 연결이 새 위치에서도 동일하게 유지될 수 있다는 확신을 가져야 합니다. 이러한 확신은 호스트에서 일관된 네트워크 인터페이스 이름을 사용함으로써 확보됩니다.

예를 들어 Host1에서 실행 중인 StorageGRID NodeA가 다음과 같은 인터페이스 매핑으로 구성되었다고 가정해 보겠습니다.

eth0 → bond0.1001

eth1 → bond0.1002

eth2 → bond0.1003

화살표의 왼쪽은 StorageGRID 컨테이너 내에서 본 기존 인터페이스(즉, 각각 Grid, Admin 및 Client Network 인터페이스)에 해당합니다. 화살표의 오른쪽은 이러한 네트워크를 제공하는 실제 호스트 인터페이스로, 동일한 물리적 인터페이스 본드에 종속된 세 개의 VLAN 인터페이스입니다.

이제 NodeA를 Host2로 마이그레이션한다고 가정해 보겠습니다. Host2에도 bond0.1001, bond0.1002, bond0.1003과 같은 이름의 인터페이스가 있다면, 시스템은 동일한 이름의 인터페이스가 Host1에서와 마찬가지로 Host2에서도 동일한 연결성을 제공한다고 판단하여 마이그레이션을 허용합니다. 하지만 Host2에 동일한 이름의 인터페이스가 없다면 마이그레이션은 허용되지 않습니다.

여러 호스트에서 일관된 네트워크 인터페이스 이름을 지정하는 방법은 여러 가지가 있습니다. 몇 가지 예는 "[호스트 네트워크 구성](#)"을 참조하십시오.

공유 스토리지

StorageGRID 노드 마이그레이션 기능은 빠르고 오버헤드가 적은 노드 마이그레이션을 구현하기 위해 노드 데이터를 물리적으로 이동하지 않습니다. 대신, 노드 마이그레이션은 다음과 같은 내보내기 및 가져오기 작업 쌍으로 수행됩니다.

- "노드 내보내기" 작업 중에 HostA에서 실행 중인 노드 컨테이너에서 소량의 영구 상태 데이터가 추출되어 해당 노드의 시스템 데이터 볼륨에 캐시됩니다. 그런 다음 HostA의 노드 컨테이너가 인스턴스 해제됩니다.
- "노드 가져오기" 작업 중에는 HostA에서 사용되던 것과 동일한 네트워크 인터페이스 및 블록 스토리지 매핑을 사용하는 HostB의 노드 컨테이너가 인스턴스화됩니다. 그런 다음 캐시된 영구 상태 데이터가 새 인스턴스에 삽입됩니다.

이러한 운영 모드에서 마이그레이션이 허용되고 정상적으로 작동하려면 노드의 모든 시스템 데이터 및 객체 스토리지 볼륨에 HostA와 HostB 모두에서 액세스할 수 있어야 합니다. 또한, HostA와 HostB에서 동일한 LUN을 참조하도록 보장된 이름을 사용하여 노드에 매핑되어야 합니다.

다음 예는 StorageGRID 스토리지 노드의 블록 장치 매핑에 대한 한 가지 솔루션을 보여줍니다. 이 솔루션에서는 호스트에서 DM 멀티패싱이 사용되고 있으며, 별칭 필드가 `/etc/multipath.conf`에서 사용되어 모든 호스트에서 일관되고 친숙한 블록 장치 이름을 제공합니다.

```
/var/local    → /dev/mapper/sgws-sn1-var-local
rangedb0     → /dev/mapper/sgws-sn1-rangedb0
rangedb1     → /dev/mapper/sgws-sn1-rangedb1
rangedb2     → /dev/mapper/sgws-sn1-rangedb2
rangedb3     → /dev/mapper/sgws-sn1-rangedb3
```

호스트(Linux) 준비

StorageGRID에서 호스트 전체 설정을 변경하는 방법

베어메탈 시스템에서 StorageGRID는 호스트 전체 `sysctl` 설정에 일부 변경 사항을 적용합니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 "[NetApp 상호 운용성 매트릭스 툴\(IMT\)](#)"을 참조하십시오.

다음과 같은 변경 사항이 적용됩니다.

```
# Recommended Cassandra setting: CASSANDRA-3563, CASSANDRA-13008, DataStax
```

```

documentation
vm.max_map_count = 1048575

# core file customization
# Note: for cores generated by binaries running inside containers, this
# path is interpreted relative to the container filesystem namespace.
# External cores will go nowhere, unless /var/local/core also exists on
# the host.
kernel.core_pattern = /var/local/core/%e.core.%p

# Set the kernel minimum free memory to the greater of the current value
or
# 512MiB if the host has 48GiB or less of RAM or 1.83GiB if the host has
more than 48GiB of RAM
vm.min_free_kbytes = 524288

# Enforce current default swappiness value to ensure the VM system has
some
# flexibility to garbage collect behind anonymous mappings. Bump
watermark_scale_factor
# to help avoid OOM conditions in the kernel during memory allocation
bursts. Bump
# dirty_ratio to 90 because we explicitly fsync data that needs to be
persistent, and
# so do not require the dirty_ratio safety net. A low dirty_ratio combined
with a large
# working set (nr_active_pages) can cause us to enter synchronous I/O mode
unnecessarily,
# with deleterious effects on performance.
vm.swappiness = 60
vm.watermark_scale_factor = 200
vm.dirty_ratio = 90

# Turn off slow start after idle
net.ipv4.tcp_slow_start_after_idle = 0

# Tune TCP window settings to improve throughput
net.core.rmem_max = 8388608
net.core.wmem_max = 8388608
net.ipv4.tcp_rmem = 4096 524288 8388608
net.ipv4.tcp_wmem = 4096 262144 8388608
net.core.netdev_max_backlog = 2500

# Turn on MTU probing
net.ipv4.tcp_mtu_probing = 1

```

```
# Be more liberal with firewall connection tracking
net.ipv4.netfilter.ip_conntrack_tcp_be_liberal = 1

# Reduce TCP keepalive time to reasonable levels to terminate dead
connections
net.ipv4.tcp_keepalive_time = 270
net.ipv4.tcp_keepalive_probes = 3
net.ipv4.tcp_keepalive_intvl = 30

# Increase the ARP cache size to tolerate being in a /16 subnet
net.ipv4.neigh.default.gc_thresh1 = 8192
net.ipv4.neigh.default.gc_thresh2 = 32768
net.ipv4.neigh.default.gc_thresh3 = 65536
net.ipv6.neigh.default.gc_thresh1 = 8192
net.ipv6.neigh.default.gc_thresh2 = 32768
net.ipv6.neigh.default.gc_thresh3 = 65536

# Disable IP forwarding, we are not a router
net.ipv4.ip_forward = 0

# Follow security best practices for ignoring broadcast ping requests
net.ipv4.icmp_echo_ignore_broadcasts = 1

# Increase the pending connection and accept backlog to handle larger
connection bursts.
net.core.somaxconn=4096
net.ipv4.tcp_max_syn_backlog=4096
```

Linux 그리드 호스트에 StorageGRID 설치

모든 Linux 그리드 호스트에 StorageGRID를 설치해야 합니다. 지원되는 버전 목록은 NetApp 상호 운용성 매트릭스 도구를 참조하십시오.

시작하기 전에

운영 체제가 아래에 명시된 StorageGRID의 최소 커널 버전 요구 사항을 충족하는지 확인하십시오. `uname -r` 명령을 사용하여 운영 체제의 커널 버전을 확인하거나 운영 체제 공급업체에 문의하십시오.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

RHEL

RHEL 버전	최소 커널 버전	커널 패키지 이름
8.8 (사용 중단됨)	4.18.0-477.10.1.el8_8.x86_64	kernel-4.18.0-477.10.1.el8_8.x86_64
8.10	4.18.0-553.el8_10.x86_64	kernel-4.18.0-553.el8_10.x86_64
9.0 (사용 중단됨)	5.14.0-70.22.1.el9_0.x86_64	kernel-5.14.0-70.22.1.el9_0.x86_64
9.2 (사용 중단됨)	5.14.0-284.11.1.el9_2.x86_64	kernel-5.14.0-284.11.1.el9_2.x86_64
9.4	5.14.0-427.18.1.el9_4.x86_64	kernel-5.14.0-427.18.1.el9_4.x86_64
9.6	5.14.0-570.18.1.el9_6.x86_64	kernel-5.14.0-570.18.1.el9_6.x86_64

Ubuntu

참고: Ubuntu 18.04 및 20.04 버전에 대한 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다.

Ubuntu 버전	최소 커널 버전	커널 패키지 이름
22.04.1	5.15.0-47-generic	linux-image-5.15.0-47-generic/jammy-updates,jammy-security,now 5.15.0-47.51
24.04	6.8.0-31-generic	linux-image-6.8.0-31-generic/noble,now 6.8.0-31.31

Debian

참고: Debian 버전 11 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다.

Debian 버전	최소 커널 버전	커널 패키지 이름
11(지원 중단됨)	5.10.0-18-amd64	linux-image-5.10.0-18-amd64/stable,now 5.10.150-1
12	6.1.0-9-amd64	linux-image-6.1.0-9-amd64/stable,now 6.1.27-1

단계

1. 총판의 지침 또는 표준 절차에 따라 모든 물리적 또는 가상 그리드 호스트에 Linux를 설치하십시오.



그래픽 데스크톱 환경을 설치하지 마십시오.

- RHEL을 설치할 때 표준 Linux 설치 프로그램을 사용하는 경우, 가능하다면 "컴퓨팅 노드" 소프트웨어 구성을 선택하거나 "최소 설치" 기본 환경을 선택하십시오.

- 우분투를 설치할 때 *표준 시스템 유틸리티*를 선택해야 합니다. 우분투 호스트에 SSH로 접속하려면 *OpenSSH 서버*를 선택하는 것이 좋습니다. 나머지 옵션은 모두 선택 해제된 상태로 두어도 됩니다.

2. 모든 호스트가 RHEL용 Extras 채널을 포함한 패키지 저장소에 액세스할 수 있는지 확인하십시오.

3. 스왑 기능이 활성화된 경우:

- 다음 명령을 실행합니다. `$ sudo swapoff --all`
- 설정을 유지하려면 `/etc/fstab`에서 모든 스왑 항목을 제거하십시오.



스왑 기능을 완전히 비활성화하지 않으면 성능이 심각하게 저하될 수 있습니다.

Ubuntu 및 Debian에서 StorageGRID용 AppArmor 프로필에 대해 알아보십시오

자체 배포된 Ubuntu 환경에서 AppArmor 강제 액세스 제어 시스템을 사용하는 경우, 기본 시스템에 설치한 패키지와 연결된 AppArmor 프로필이 StorageGRID와 함께 설치된 해당 패키지에 의해 차단될 수 있습니다.

기본적으로 AppArmor 프로필은 기본 운영 체제에 설치하는 패키지에 대해 설치됩니다. StorageGRID 시스템 컨테이너에서 이러한 패키지를 실행하면 AppArmor 프로필이 차단됩니다. DHCP, MySQL, NTP 및 tcdump 기본 패키지는 AppArmor와 충돌하며, 다른 기본 패키지도 충돌할 수 있습니다.

AppArmor 프로필을 처리하는 방법에는 두 가지가 있습니다.

- StorageGRID 시스템 컨테이너의 패키지와 겹치는 기본 시스템에 설치된 패키지에 대한 개별 프로필을 비활성화합니다. 개별 프로필을 비활성화하면 StorageGRID 로그 파일에 AppArmor가 활성화되었다는 항목이 나타납니다.

다음 명령을 사용합니다.

```
sudo ln -s /etc/apparmor.d/<profile.name> /etc/apparmor.d/disable/  
sudo apparmor_parser -R /etc/apparmor.d/<profile.name>
```

예:

```
sudo ln -s /etc/apparmor.d/bin.ping /etc/apparmor.d/disable/  
sudo apparmor_parser -R /etc/apparmor.d/bin.ping
```

- AppArmor를 완전히 비활성화하세요. Ubuntu 9.10 이상 버전의 경우 Ubuntu 온라인 커뮤니티의 지침을 따르세요: "[AppArmor 비활성화](#)". 최신 Ubuntu 버전에서는 AppArmor를 완전히 비활성화하는 것이 불가능할 수 있습니다.

AppArmor를 비활성화하면 AppArmor가 활성화되었음을 나타내는 항목이 StorageGRID 로그 파일에 더 이상 나타나지 않습니다.

Linux 배포를 위한 StorageGRID 호스트 네트워크 구성

호스트에 Linux 설치를 완료한 후에 나중에 배포할 StorageGRID 노드에 매핑하기에 적합한

네트워크 인터페이스 세트를 각 호스트에서 준비하기 위해 추가 구성을 수행해야 할 수 있습니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

시작하기 전에

- ["StorageGRID 네트워킹 가이드라인"](#)를 검토했습니다.
- ["노드 컨테이너 마이그레이션 요구 사항"](#)에 대한 정보를 검토하셨습니다.
- 가상 호스트를 사용하는 경우 호스트 네트워크를 구성하기 전에 [MAC 주소 복제에 대한 고려 사항 및 권장 사항](#)을 읽어보시기 바랍니다.



가상 머신을 호스트로 사용하는 경우 가상 네트워크 어댑터로 VMXNET 3을 선택해야 합니다. VMware E1000 네트워크 어댑터는 특정 Linux 배포판에 배포된 StorageGRID 컨테이너에서 연결 문제를 일으킨 것으로 알려져 있습니다.

이 작업 정보

그리드 노드는 그리드 네트워크에 액세스할 수 있어야 하며, 선택적으로 관리 네트워크 및 클라이언트 네트워크에도 액세스할 수 있어야 합니다. 이러한 액세스는 호스트의 물리적 인터페이스를 각 그리드 노드의 가상 인터페이스에 연결하는 매핑을 생성하여 제공합니다. 호스트 인터페이스를 생성할 때는 모든 호스트에 걸쳐 배포를 용이하게 하고 마이그레이션을 지원하기 위해 친숙한 이름을 사용하십시오.

호스트와 하나 이상의 노드 간에 동일한 인터페이스를 공유할 수 있습니다. 예를 들어, 호스트 및 노드 유지 관리를 용이하게 하기 위해 호스트 액세스와 노드 관리 네트워크 액세스에 동일한 인터페이스를 사용할 수 있습니다. 호스트와 개별 노드 간에 동일한 인터페이스를 공유할 수 있지만, 모든 노드는 서로 다른 IP 주소를 가져야 합니다. 노드 간 또는 호스트와 노드 간에 IP 주소를 공유할 수 없습니다.

호스트의 모든 StorageGRID 노드에 그리드 네트워크 인터페이스를 제공하기 위해 동일한 호스트 네트워크 인터페이스를 사용할 수도 있고, 각 노드마다 다른 호스트 네트워크 인터페이스를 사용할 수도 있으며, 그 중간 방식을 사용할 수도 있습니다. 하지만 일반적으로 단일 노드에 대해 그리드 네트워크 인터페이스와 관리 네트워크 인터페이스로 동일한 호스트 네트워크 인터페이스를 제공하거나, 한 노드에는 그리드 네트워크 인터페이스를, 다른 노드에는 클라이언트 네트워크 인터페이스를 제공하지는 않습니다.

이 작업은 여러 가지 방법으로 완료할 수 있습니다. 예를 들어 호스트가 가상 머신이고 각 호스트에 StorageGRID 노드를 하나 또는 두 개 배포하는 경우 하이퍼바이저에서 올바른 수의 네트워크 인터페이스를 생성하고 1:1 매핑을 사용할 수 있습니다. 프로덕션 환경에서 베어메탈 호스트에 여러 노드를 배포하는 경우 Linux 네트워킹 스택의 VLAN 및 LACP 지원 기능을 활용하여 내결함성과 대역폭 공유를 구현할 수 있습니다. 다음 섹션에서는 이 두 가지 예시에 대한 자세한 접근 방식을 제공합니다. 이러한 예시 중 어느 것도 반드시 사용해야 하는 것은 아니며, 필요에 맞는 다른 접근 방식을 사용할 수 있습니다.



본드 또는 브리지 장치를 컨테이너 네트워크 인터페이스로 직접 사용하지 마십시오. 이렇게 하면 컨테이너 네임스페이스에서 본드 및 브리지 장치와 함께 MACVLAN을 사용할 때 발생하는 커널 문제로 인해 노드 시작이 중단될 수 있습니다. 대신 VLAN 또는 가상 이더넷(veth) 쌍과 같은 비본드 장치를 사용하십시오. 이 장치를 노드 구성 파일의 네트워크 인터페이스로 지정하십시오.

MAC 주소 복제에 대한 고려 사항 및 권장 사항

MAC 주소 복제를 사용하면 컨테이너는 호스트의 MAC 주소를 사용하고, 호스트는 사용자가 지정한 주소 또는 임의로 생성된 주소의 MAC 주소를 사용하게 됩니다. 무차별 모드 네트워크 구성을 피하려면 MAC 주소 복제를 사용하는 것이 좋습니다.

MAC 복제 활성화

특정 환경에서는 MAC 주소 복제를 통해 보안을 강화할 수 있습니다. 이를 통해 관리 네트워크, 그리드 네트워크 및 클라이언트 네트워크에 각각 전용 가상 NIC를 사용할 수 있기 때문입니다. 컨테이너가 호스트의 전용 NIC의 MAC 주소를 사용하도록 설정하면 무차별 모드 네트워크 구성을 사용하지 않아도 됩니다.



MAC 주소 복제는 가상 서버 설치 환경에서 사용하도록 설계되었으며, 모든 물리적 어플라이언스 구성에서 제대로 작동하지 않을 수 있습니다.



MAC 주소 복제를 위해 지정된 인터페이스가 사용 중이어서 노드 시작에 실패하는 경우, 노드를 시작하기 전에 해당 링크를 "down" 상태로 설정해야 할 수 있습니다. 또한, 링크가 활성화된 상태에서 가상 환경이 네트워크 인터페이스의 MAC 주소 복제를 차단할 수 있습니다. 인터페이스 사용 중으로 인해 노드가 MAC 주소를 설정하고 시작하지 못하는 경우, 노드를 시작하기 전에 해당 링크를 "down" 상태로 설정하면 문제가 해결될 수 있습니다.

MAC 주소 복제는 기본적으로 비활성화되어 있으며 노드 구성 키를 통해 설정해야 합니다. StorageGRID를 설치할 때 활성화해야 합니다.

각 네트워크마다 하나의 키가 있습니다.

- ADMIN_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC
- GRID_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC
- CLIENT_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC

이 키를 "true"로 설정하면 컨테이너가 호스트 NIC의 MAC 주소를 사용하게 됩니다. 또한 호스트는 지정된 컨테이너 네트워크의 MAC 주소를 사용하게 됩니다. 기본적으로 컨테이너 주소는 임의로 생성된 주소이지만, `NETWORK_MAC` 노드 구성 키를 사용하여 주소를 설정한 경우 해당 주소가 대신 사용됩니다. 호스트와 컨테이너는 항상 서로 다른 MAC 주소를 갖게 됩니다.



하이퍼바이저에서 무차별 모드를 활성화하지 않고 가상 호스트에서 MAC 복제를 활성화하면 호스트 인터페이스를 사용하는 Linux 호스트 네트워킹이 작동을 멈출 수 있습니다.

MAC 복제 활용 사례

MAC 복제에는 두 가지 사용 사례가 있습니다.

- MAC 복제 미사용: 노드 구성 파일의 `_CLONE_MAC` 키가 설정되지 않았거나 "false"로 설정된 경우, 호스트는 호스트 NIC MAC 주소를 사용하고 컨테이너는 `NETWORK_MAC` 키에 MAC 주소가 지정되지 않은 한 StorageGRID에서 생성한 MAC 주소를 사용합니다. `NETWORK_MAC` 키에 주소가 설정된 경우 컨테이너는 `NETWORK_MAC` 키에 지정된 주소를 사용합니다. 이러한 키 구성에는 무차별 모드(promiscuous mode) 사용이 필요합니다.
- MAC 복제 활성화: 노드 구성 파일의 `_CLONE_MAC` 키가 "true"로 설정된 경우 컨테이너는 호스트 NIC의 MAC 주소를 사용하고, 호스트는 `NETWORK_MAC` 키에 MAC 주소가 지정되지 않은 경우 StorageGRID에서 생성한 MAC 주소를 사용합니다. `NETWORK_MAC` 키에 주소가 설정된 경우 호스트는 생성된 주소 대신 지정된 주소를 사용합니다. 이 키 구성에서는 무차별 모드를 사용해서는 안 됩니다.



MAC 주소 복제를 사용하지 않고 하이퍼바이저가 할당한 MAC 주소 이외의 MAC 주소에 대해 모든 인터페이스가 데이터를 송수신할 수 있도록 허용하려면 가상 스위치 및 포트 그룹 수준의 보안 속성에서 무차별 모드, MAC 주소 변경 및 위조 전송에 대해 *허용*으로 설정해야 합니다. 가상 스위치에서 설정한 값은 포트 그룹 수준에서 설정한 값으로 재정의될 수 있으므로 두 위치의 설정이 동일한지 확인하십시오.

MAC 복제를 활성화하려면 "노드 구성 파일 생성 지침"을 참조하십시오.

MAC 복제 예시

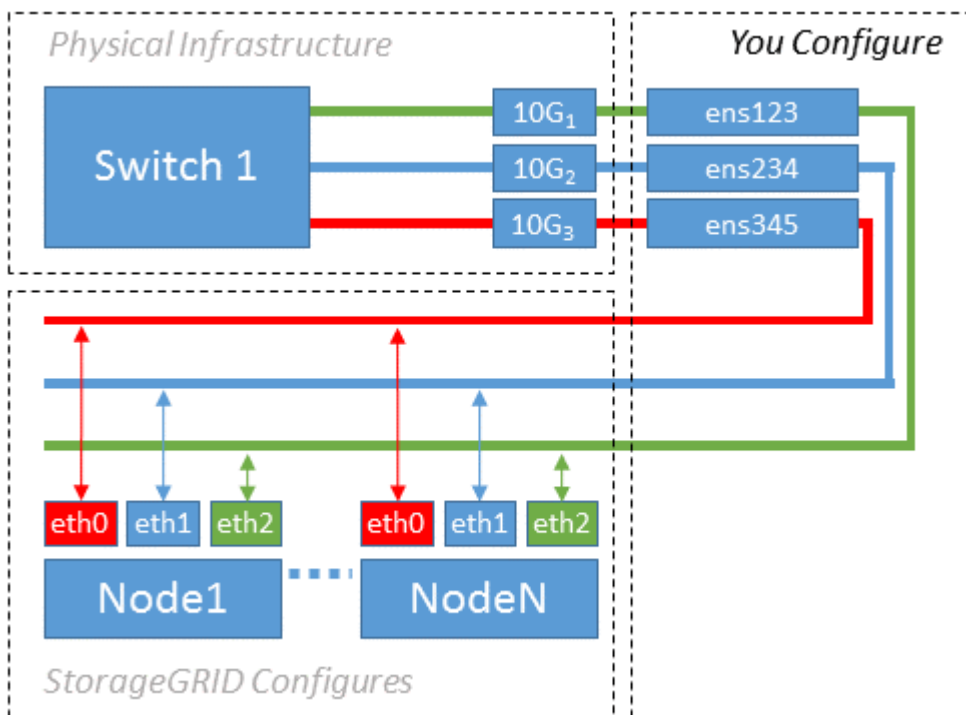
MAC 주소가 11:22:33:44:55:66인 호스트에서 인터페이스 ens256에 대해 MAC 클로닝이 활성화된 예이며, 노드 구성 파일에 다음 키가 있습니다.

- ADMIN_NETWORK_TARGET = ens256
- ADMIN_NETWORK_MAC = b2:9c:02:c2:27:10
- ADMIN_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC = true

결과: ens256의 호스트 MAC 주소는 b2:9c:02:c2:27:10이고 관리 네트워크 MAC 주소는 11:22:33:44:55:66입니다.

예시 1: 물리적 또는 가상 NIC에 대한 1:1 매핑

예제 1은 호스트 측 구성이 거의 또는 전혀 필요하지 않은 간단한 물리적 인터페이스 매핑에 대해 설명합니다.



Linux 운영 체제는 설치 또는 부팅 중에, 혹은 인터페이스를 핫 추가할 때 `ensXYZ` 인터페이스를 자동으로 생성합니다. 부팅 후 인터페이스가 자동으로 활성화되도록 설정하는 것 외에는 별도의 구성이 필요하지 않습니다. 다만, 나중에 구성 과정에서 올바른 매핑을 제공할 수 있도록 어떤 ensXYZ 인터페이스가 어떤 StorageGRID 네트워크(Grid, Admin 또는 Client)에 해당하는지 미리 파악해 두어야 합니다.

그림에는 여러 StorageGRID 노드가 나와 있지만, 일반적으로 이 구성은 단일 노드 VM에 사용됩니다.

스위치 1이 물리적 스위치인 경우, 인터페이스 10G1부터 10G3에 연결된 포트를 액세스 모드로 구성하고 적절한 VLAN에 배치해야 합니다.

예시 2: VLAN을 전송하는 LACP 본딩

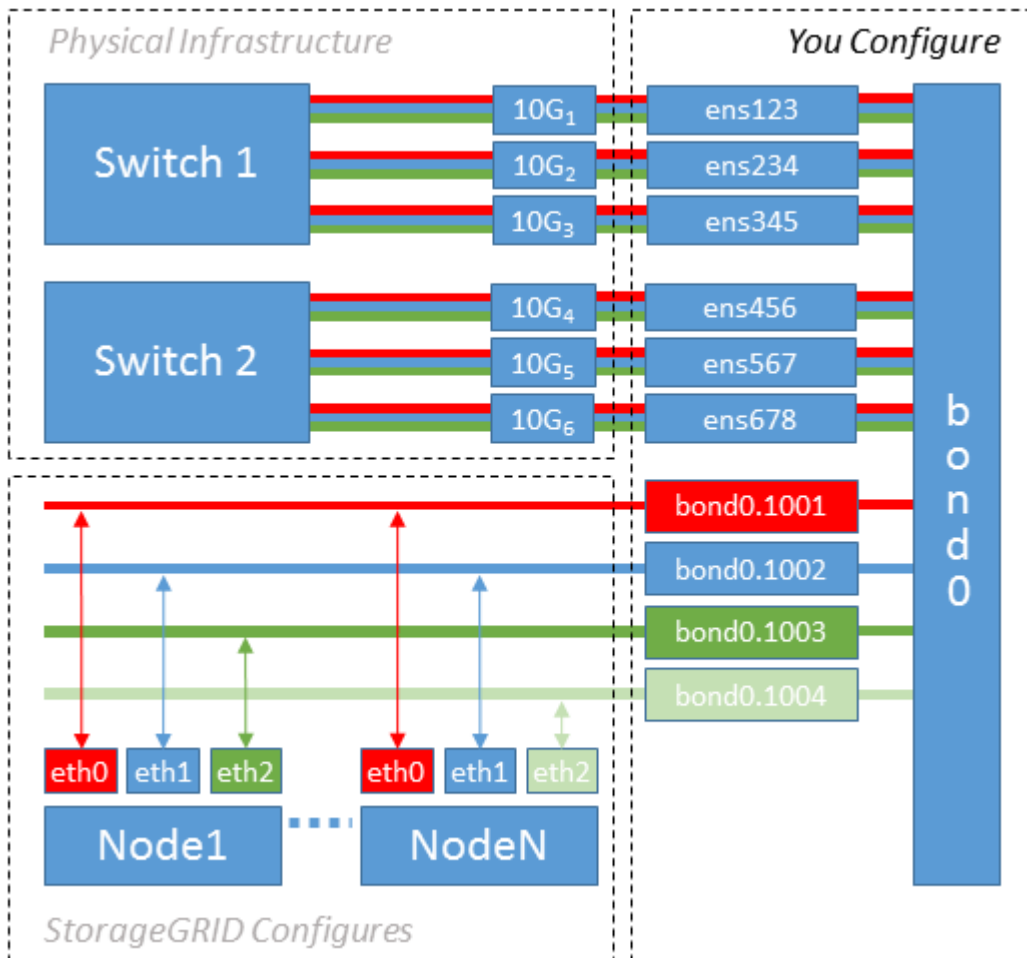
예제 2는 사용자가 네트워크 인터페이스 본딩과 사용 중인 Linux 배포판에서 VLAN 인터페이스를 생성하는 방법에 익숙하다고 가정합니다.

이 작업 정보

예제 2는 단일 호스트의 모든 노드에서 사용 가능한 네트워크 대역폭을 공유할 수 있도록 하는 일반적이고 유연한 VLAN 기반 체계를 설명합니다. 이 예제는 특히 베어메탈 호스트에 적용 가능합니다.

이 예시를 이해하기 위해 각 데이터 센터에 그리드, 관리 및 클라이언트 네트워크에 대해 각각 별도의 서브넷이 있다고 가정해 보겠습니다. 이 서브넷들은 서로 다른 VLAN(1001, 1002, 1003)에 있으며, LACP로 연결된 트렁크 포트(bond0)를 통해 호스트에 제공됩니다. 본드에 bond0.1001, bond0.1002, bond0.1003의 세 가지 VLAN 인터페이스를 구성해야 합니다.

동일한 호스트에서 노드 네트워크에 대해 별도의 VLAN 및 서브넷이 필요한 경우, 본드에 VLAN 인터페이스를 추가하고 이를 호스트에 매핑할 수 있습니다(그림에서 bond0.1004로 표시됨).



단계

1. StorageGRID 네트워크 연결에 사용될 모든 물리적 네트워크 인터페이스를 단일 LACP 본드로 집계합니다.

모든 호스트에서 동일한 본드 이름을 사용하십시오. 예를 들어, bond0.

2. 이 본드를 관련 "물리적 장치"로 사용하는 VLAN 인터페이스를 표준 VLAN 인터페이스 명명 규칙 `physdev-name.VLAN ID`을 사용하여 생성합니다.

1단계와 2단계를 수행하려면 네트워크 링크의 다른 끝단에 있는 에지 스위치에 적절한 구성이 필요합니다. 에지 스위치 포트도 LACP 포트 채널로 집계되고 트렁크로 구성되어야 하며 필요한 모든 VLAN이 통과할 수 있도록 허용되어야 합니다.

호스트별 네트워킹 구성 체계에 대한 샘플 인터페이스 구성 파일이 제공됩니다.

관련 정보

- ["Ubuntu 및 Debian의 /etc/network/interfaces 예시"](#)
- ["RHEL용 /etc/sysconfig/network-scripts 예시"](#)

Linux 배포를 위한 StorageGRID 호스트 스토리지 구성

각 Linux 호스트에 블록 스토리지 볼륨을 할당해야 합니다.

시작하기 전에

다음 항목을 검토하셨으며, 이 항목들은 이 작업을 수행하는 데 필요한 정보를 제공합니다.

- ["스토리지 및 성능 요구 사항"](#)
- ["Node 컨테이너 마이그레이션 요구 사항"](#)



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

이 작업 정보

호스트에 블록 스토리지 볼륨(LUN)을 할당할 때는 "스토리지 요구 사항"의 표를 사용하여 다음 사항을 확인하십시오.

- 호스트당 필요한 볼륨 수(해당 호스트에 배포될 노드의 수 및 유형에 따라 결정됨)
- 각 볼륨의 저장 범주(즉, 시스템 데이터 또는 객체 데이터)
- 각 볼륨의 크기

호스트에 StorageGRID 노드를 배포할 때 이 정보와 Linux에서 각 물리적 볼륨에 할당한 영구 이름을 사용하게 됩니다.



이러한 볼륨을 파티션하거나 포맷하거나 마운트할 필요는 없습니다. 호스트에서 해당 볼륨이 보이도록만 하면 됩니다.



메타데이터 전용 스토리지 노드에는 객체 데이터 LUN이 하나만 필요합니다.

볼륨 이름 목록을 작성할 때 "raw" 특수 장치 파일(예: (/dev/sdb) 사용을 피하십시오. 이러한 파일은 호스트 재부팅 시 변경될 수 있으며, 이는 시스템의 정상적인 작동에 영향을 미칠 수 있습니다. iSCSI LUN 및 Device Mapper 다중 경로를 사용하는 경우, 특히 SAN 토폴로지에 공유 스토리지에 대한 중복 네트워크 경로가 포함된 경우 /dev/mapper 디렉토리에서 다중 경로 별칭을 사용하는 것을 고려하십시오. 또는 영구 장치 이름에 /dev/disk/by-path/ 아래의 시스템에서 생성된 소프트링크를 사용할 수도 있습니다.

예를 들면 다음과 같습니다.

```
ls -l
$ ls -l /dev/disk/by-path/
total 0
lrwxrwxrwx 1 root root 9 Sep 19 18:53 pci-0000:00:07.1-ata-2 -> ../../sr0
lrwxrwxrwx 1 root root 9 Sep 19 18:53 pci-0000:03:00.0-scsi-0:0:0:0 ->
../../sda
lrwxrwxrwx 1 root root 10 Sep 19 18:53 pci-0000:03:00.0-scsi-0:0:0:0-part1
-> ../../sda1
lrwxrwxrwx 1 root root 10 Sep 19 18:53 pci-0000:03:00.0-scsi-0:0:0:0-part2
-> ../../sda2
lrwxrwxrwx 1 root root 9 Sep 19 18:53 pci-0000:03:00.0-scsi-0:0:1:0 ->
../../sdb
lrwxrwxrwx 1 root root 9 Sep 19 18:53 pci-0000:03:00.0-scsi-0:0:2:0 ->
../../sdc
lrwxrwxrwx 1 root root 9 Sep 19 18:53 pci-0000:03:00.0-scsi-0:0:3:0 ->
../../sdd
```

설치 환경에 따라 결과가 다를 수 있습니다.

StorageGRID 설치 및 향후 유지 관리 절차를 간소화하기 위해 각 블록 스토리지 볼륨에 친숙한 이름을 지정하십시오. 공유 스토리지 볼륨에 대한 이중화 액세스를 위해 장치 매퍼 다중 경로 드라이버를 사용하는 경우 `alias` 파일의 `/etc/multipath.conf` 필드를 사용할 수 있습니다.

예를 들면 다음과 같습니다.

```

multipaths {
    multipath {
        wwid 3600a09800059d6df00005df2573c2c30
        alias docker-storage-volume-hostA
    }
    multipath {
        wwid 3600a09800059d6df00005df3573c2c30
        alias sgws-adml-var-local
    }
    multipath {
        wwid 3600a09800059d6df00005df4573c2c30
        alias sgws-adml-audit-logs
    }
    multipath {
        wwid 3600a09800059d6df00005df5573c2c30
        alias sgws-adml-tables
    }
    multipath {
        wwid 3600a09800059d6df00005df6573c2c30
        alias sgws-gw1-var-local
    }
    multipath {
        wwid 3600a09800059d6df00005df7573c2c30
        alias sgws-sn1-var-local
    }
    multipath {
        wwid 3600a09800059d6df00005df7573c2c30
        alias sgws-sn1-rangedb-0
    }
    ...
}

```

이와 같이 별칭 필드를 사용하면 별칭이 호스트의 `/dev/mapper` 디렉터리에 블록 장치로 나타나므로 구성 또는 유지 관리 작업에서 블록 스토리지 볼륨을 지정해야 할 때마다 친숙하고 유효성 검사가 쉬운 이름을 지정할 수 있습니다.

StorageGRID 노드 마이그레이션을 지원하기 위해 공유 스토리지를 설정하고 Device Mapper 다중 경로를 사용하는 경우, 모든 동일 위치 호스트에 공통 `/etc/multipath.conf`를 생성하고 설치할 수 있습니다. 단, 각 호스트에서 서로 다른 컨테이너 엔진 스토리지 볼륨을 사용해야 합니다. 각 컨테이너 엔진 스토리지 볼륨 LUN에 별칭을 사용하고 별칭에 대상 호스트 이름을 포함시키면 기억하기 쉬우므로 권장됩니다.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로서 Docker 지원이 더 이상 권장되지 않습니다. Docker는 향후 릴리스에서 다른 컨테이너 엔진으로 대체될 예정입니다.

관련 정보

- ["컨테이너 엔진 스토리지 볼륨 구성"](#)
- ["스토리지 및 성능 요구 사항"](#)

- ["Node 컨테이너 마이그레이션 요구 사항"](#)

Linux에서 StorageGRID용 컨테이너 엔진 스토리지 볼륨 구성

Docker 또는 Podman 컨테이너 엔진을 설치하기 전에 스토리지 볼륨을 포맷하고 마운트해야 할 수도 있습니다.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로서 Docker 지원이 더 이상 권장되지 않습니다. Docker는 향후 릴리스에서 다른 컨테이너 엔진으로 대체될 예정입니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

이 작업 정보

Docker 또는 Podman 스토리지 볼륨에 루트 볼륨을 사용할 계획이고 다음을 포함하는 호스트 파티션에 충분한 여유 공간이 있는 경우 이 단계를 건너뛸 수 있습니다.

- Podman: /var/lib/containers
- Docker: /var/lib/docker

단계

1. 컨테이너 엔진 스토리지 볼륨에 파일 시스템을 생성합니다.

RHEL

```
sudo mkfs.ext4 container-engine-storage-volume-device
```

Ubuntu 또는 Debian

```
sudo mkfs.ext4 docker-storage-volume-device
```

2. 컨테이너 엔진 저장 볼륨을 마운트합니다.

RHEL

◦ Docker의 경우:

```
sudo mkdir -p /var/lib/docker
sudo mount container-storage-volume-device /var/lib/docker
```

◦ Podman의 경우:

```
sudo mkdir -p /var/lib/containers
sudo mount container-storage-volume-device /var/lib/containers
```

Ubuntu 또는 Debian

```
sudo mkdir -p /var/lib/docker
sudo mount docker-storage-volume-device /var/lib/docker
```

◦ Podman의 경우:

```
sudo mkdir -p /var/lib/podman
sudo mount container-storage-volume-device /var/lib/podman
```

3. 컨테이너 스토리지 볼륨 장치에 대한 항목을 /etc/fstab 파일에 추가합니다.

- RHEL: container-storage-volume-device
- Ubuntu 또는 Debian: docker-storage-volume-device

이 단계를 통해 호스트 재부팅 후 스토리지 볼륨이 자동으로 다시 마운트됩니다.

Docker 설치

StorageGRID 시스템은 컨테이너 모음으로 Linux에서 실행될 수 있습니다.

- Ubuntu 또는 Debian용 StorageGRID를 설치하기 전에 Docker를 설치해야 합니다.
- Docker 컨테이너 엔진을 사용하기로 선택했다면 다음 단계를 따라 Docker를 설치하십시오. 그렇지 않은 경우, [Podman 설치](#).



소프트웨어 전용 배포를 위한 컨테이너 엔진으로서 Docker 지원이 더 이상 권장되지 않습니다. Docker는 향후 릴리스에서 다른 컨테이너 엔진으로 대체될 예정입니다.

단계

1. 사용하는 Linux 배포판의 지침에 따라 Docker를 설치합니다.



사용하는 Linux 배포판에 Docker가 포함되어 있지 않다면 Docker 웹사이트에서 다운로드할 수 있습니다.

2. Docker가 활성화되고 시작되었는지 확인하려면 다음 두 명령을 실행하십시오.

```
sudo systemctl enable docker
```

```
sudo systemctl start docker
```

3. 다음 명령어를 입력하여 예상되는 버전의 Docker가 설치되었는지 확인하십시오.

```
sudo docker version
```

클라이언트 및 서버 버전은 1.11.0 이상이어야 합니다.

Podman 설치

StorageGRID 시스템은 컨테이너 모음으로 실행됩니다. Podman 컨테이너 엔진을 사용하기로 선택한 경우 다음 단계를 따라 Podman을 설치하십시오. 그렇지 않은 경우 [Docker 설치](#).

단계

1. 사용하는 Linux 배포판의 지침에 따라 Podman과 Podman-Docker를 설치하십시오.



Podman을 설치할 때 Podman-Docker 패키지도 함께 설치해야 합니다.

2. 다음 명령어를 입력하여 Podman 및 Podman-Docker의 예상 버전이 설치되었는지 확인하십시오.

```
sudo docker version
```



Podman-Docker 패키지를 사용하면 Docker 명령어를 사용할 수 있습니다.

클라이언트 및 서버 버전은 3.2.3 이상이어야 합니다.

```
Version: 3.2.3
API Version: 3.2.3
Go Version: go1.15.7
Built: Tue Jul 27 03:29:39 2021
OS/Arch: linux/amd64
```

관련 정보

"호스트 스토리지 구성"

Linux에 **StorageGRID** 호스트 서비스를 설치합니다

운영 체제 유형에 맞는 StorageGRID 패키지를 사용하여 StorageGRID 호스트 서비스를 설치합니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

RHEL

StorageGRID 호스트 서비스를 설치하려면 StorageGRID RPM 패키지를 사용합니다.

이 작업 정보

이 지침에서는 RPM 패키지를 사용하여 호스트 서비스를 설치하는 방법을 설명합니다. 또는 설치 아카이브에 포함된 DNF 저장소 메타데이터를 사용하여 RPM 패키지를 원격으로 설치할 수도 있습니다. 사용 중인 Linux 운영 체제에 맞는 DNF 저장소 지침을 참조하십시오.

단계

1. StorageGRID RPM 패키지를 각 호스트에 복사하거나 공유 스토리지에서 사용할 수 있도록 합니다.

예를 들어, 다음 단계에서 예제 명령을 사용할 수 있도록 해당 파일들을 /tmp 디렉토리에 넣어 두십시오.

2. 루트 사용자 또는 sudo 권한이 있는 계정으로 각 호스트에 로그인한 다음, 지정된 순서대로 다음 명령어를 실행하십시오.

```
sudo dnf --nogpgcheck localinstall /tmp/StorageGRID-Webscale-Images-  
version-SHA.rpm
```

```
sudo dnf --nogpgcheck localinstall /tmp/StorageGRID-Webscale-  
Service-version-SHA.rpm
```



이미지 패키지를 먼저 설치하고, 그 다음에 서비스 패키지를 설치해야 합니다.



패키지를 /tmp 이외의 다른 디렉토리에 배치한 경우, 사용한 경로를 반영하도록 명령을 수정하십시오.

Ubuntu 또는 Debian

StorageGRID DEB 패키지를 사용하여 Ubuntu 또는 Debian용 StorageGRID 호스트 서비스를 설치합니다.

이 작업 정보

이 지침에서는 DEB 패키지에서 호스트 서비스를 설치하는 방법을 설명합니다. 또는 설치 아카이브에 포함된 APT 저장소 메타데이터를 사용하여 DEB 패키지를 원격으로 설치할 수도 있습니다. 사용 중인 Linux 운영 체제에 맞는 APT 저장소 지침을 참조하십시오.

단계

1. StorageGRID DEB 패키지를 각 호스트에 복사하거나 공유 스토리지에서 사용할 수 있도록 합니다.

예를 들어, 다음 단계에서 예제 명령을 사용할 수 있도록 해당 파일들을 /tmp 디렉토리에 넣어 두십시오.

2. 루트 사용자 또는 sudo 권한이 있는 계정으로 각 호스트에 로그인한 후 다음 명령을 실행하십시오.

먼저 images 패키지를 설치하고, 그 다음에 service 패키지를 설치해야 합니다. 패키지를 /tmp 이외의 다른 디렉토리에 배치한 경우, 사용한 경로를 반영하도록 명령을 수정하십시오.

```
sudo dpkg --install /tmp/storagegrid-webscale-images-version-SHA.deb
```

```
sudo dpkg --install /tmp/storagegrid-webscale-service-version-SHA.deb
```



StorageGRID 패키지를 설치하려면 Python 3이 이미 설치되어 있어야 합니다. `sudo dpkg --install /tmp/storagegrid-webscale-images-version-SHA.deb` 명령은 Python 3을 설치하기 전까지 실패합니다.

소프트웨어 기반 노드 설치 자동화

Linux에서 StorageGRID 호스트 서비스 설치 및 그리드 노드 구성 자동화

StorageGRID 호스트 서비스 설치 및 그리드 노드 구성을 자동화할 수 있습니다.



이 작업 정보

"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

배포 자동화는 다음과 같은 경우에 유용할 수 있습니다.

- 이미 Ansible, Puppet 또는 Chef와 같은 표준 오케스트레이션 프레임워크를 사용하여 물리적 또는 가상 호스트를 배포하고 구성하고 있습니다.
- 여러 개의 StorageGRID 인스턴스를 배포할 계획입니다.
- 규모가 크고 복잡한 StorageGRID 인스턴스를 배포하고 있습니다.

StorageGRID 호스트 서비스는 패키지 형태로 설치되며 구성 파일에 의해 관리됩니다. 다음 방법 중 하나를 사용하여 구성 파일을 생성할 수 있습니다.

- ["구성 파일을 생성합니다"](#) 수동 설치 중에 대화형으로 작동합니다.
- 이 문서에 설명된 대로 표준 오케스트레이션 프레임워크를 사용하여 자동 설치를 가능하게 하려면 구성 파일을 미리(또는 프로그래밍 방식으로) 준비하십시오.

StorageGRID는 StorageGRID 어플라이언스와 전체 StorageGRID 시스템("그리드")의 구성을 자동화하기 위한 선택적 Python 스크립트를 제공합니다. 이러한 스크립트를 직접 사용하거나, 스크립트를 검토하여 직접 개발한 그리드 배포 및 구성 도구에서 ["StorageGRID 설치 REST API"](#)을 사용하는 방법을 학습할 수 있습니다.

StorageGRID 호스트 서비스의 설치 및 구성 자동화

Ansible, Puppet, Chef, Fabric 또는 SaltStack과 같은 표준 오케스트레이션 프레임워크를 사용하여 StorageGRID 호스트 서비스 설치를 자동화할 수 있습니다.

StorageGRID 호스트 서비스는 DEB(Ubuntu 또는 Debian) 또는 RPM(RHEL) 패키지로 제공되며, 자동 설치를 위해

사전에 준비하거나 프로그래밍 방식으로 생성할 수 있는 구성 파일을 기반으로 작동합니다. 이미 표준 오케스트레이션 프레임워크를 사용하여 Linux 배포판을 설치 및 구성하는 경우, 플레이북이나 레시피에 StorageGRID를 추가하는 것은 간단합니다.

호스트 준비 및 가상 그리드 노드 배포에 필요한 모든 단계를 자동화할 수 있습니다.

Ansible 역할 및 플레이북 예시

설치 아카이브의 `/extras` 폴더에 예제 Ansible 역할과 플레이북이 포함되어 있습니다. Ansible 플레이북은 `storagegrid` 역할이 호스트를 준비하고 대상 서버에 StorageGRID를 설치하는 방법을 보여줍니다. 필요에 따라 역할 또는 플레이북을 사용자 지정할 수 있습니다.



예제 플레이북에는 StorageGRID 호스트 서비스를 시작하기 전에 네트워크 장치를 생성하는 데 필요한 단계가 포함되어 있지 않습니다. 플레이북을 최종 확정하고 사용하기 전에 이러한 단계를 추가하십시오.

RHEL

RHEL의 경우, 제공된 `storagegrid` 역할 예제의 설치 작업은 `ansible.builtin.dnf` 모듈을 사용하여 로컬 RPM 파일 또는 원격 Yum 저장소에서 설치를 수행합니다. 모듈을 사용할 수 없거나 지원되지 않는 경우, 다음 파일에서 해당 Ansible 작업을 수정하여 `yum` 또는 `ansible.builtin.yum` 모듈을 사용하도록 해야 할 수 있습니다.

- `roles/storagegrid/tasks/rhel_install_from_repo.yml`
- `roles/storagegrid/tasks/rhel_install_from_local.yml`

Ubuntu 또는 Debian

Ubuntu 또는 Debian의 경우, 제공된 `storagegrid` 역할 예제의 설치 작업은 `ansible.builtin.apt` 모듈을 사용하여 로컬 DEB 파일 또는 원격 apt 저장소에서 설치를 수행합니다. 모듈을 사용할 수 없거나 지원되지 않는 경우, 다음 파일에서 해당 Ansible 작업을 수정하여 `ansible.builtin.apt` 모듈을 사용해야 할 수 있습니다.

- `roles/storagegrid/tasks/deb_install_from_repo.yml`
- `roles/storagegrid/tasks/deb_install_from_local.yml`

StorageGRID 구성 자동화

그리드 노드를 배포한 후에는 StorageGRID 시스템의 구성을 자동화할 수 있습니다.

시작하기 전에

- 설치 아카이브에서 다음 파일의 위치를 알고 있습니다.

파일 이름	설명
<code>configure-storagegrid.py</code>	구성을 자동화하는 데 사용되는 Python 스크립트
<code>configure-storagegrid.sample.json</code>	스크립트와 함께 사용할 예시 구성 파일
<code>configure-storagegrid.blank.json</code>	스크립트와 함께 사용할 빈 구성 파일

- 사용자가 `configure-storagegrid.json` 구성 파일을 생성했습니다. 이 파일을 생성하려면 예제 구성 파일 (`configure-storagegrid.sample.json`) 또는 빈 구성 파일 (`configure-storagegrid.blank.json`)을 수정하면 됩니다.



수정된 `configure-storagegrid.json` 구성 파일의 암호 섹션에 있는 관리 암호와 프로비저닝 암호를 안전한 장소에 보관하십시오. 이러한 암호는 설치, 확장 및 유지 관리 절차에 필요합니다. 또한 수정된 `configure-storagegrid.json` 구성 파일을 백업하여 안전한 장소에 보관해야 합니다.

이 작업 정보

``configure-storagegrid.py`` Python 스크립트와 ``configure-storagegrid.json`` 구성 파일을 사용하여 StorageGRID 시스템의 구성을 자동화할 수 있습니다.



그리드 관리자 또는 설치 API를 사용하여 시스템을 구성할 수도 있습니다.

단계

1. Python 스크립트를 실행하는 데 사용하는 Linux 시스템에 로그인합니다.
2. 설치 압축 파일을 압축 해제한 디렉토리로 이동합니다.

예를 들면 다음과 같습니다.

```
cd StorageGRID-Webscale-version/platform
```

여기서 `platform``는 ``debs`, `rpms`, 또는 ``vsphere``입니다.

3. Python 스크립트를 실행하고 생성한 구성 파일을 사용하십시오.

예를 들면 다음과 같습니다.

```
./configure-storagegrid.py ./configure-storagegrid.json --start-install
```

결과

복구 패키지 `.zip` 파일은 구성 과정 중에 생성되어 설치 및 구성 프로세스를 실행하는 디렉토리로 다운로드됩니다. 하나 이상의 그리드 노드에 장애가 발생할 경우 StorageGRID 시스템을 복구할 수 있도록 복구 패키지 파일을 백업해야 합니다. 예를 들어, 안전한 네트워크 백업 위치와 안전한 클라우드 스토리지 위치에 복사해 두십시오.



복구 패키지 파일에는 StorageGRID 시스템에서 데이터를 가져오는 데 사용할 수 있는 암호화 키와 암호가 포함되어 있으므로 보안을 유지해야 합니다.

무작위 암호 생성을 지정한 경우, `Passwords.txt` 파일을 열고 StorageGRID 시스템에 액세스하는 데 필요한 암호를 찾으십시오.

```
#####
##### The StorageGRID "Recovery Package" has been downloaded as: #####
#####      ./sgws-recovery-package-994078-rev1.zip      #####
#####   Safeguard this file as it will be needed in case of a   #####
#####           StorageGRID node recovery.           #####
#####
```

StorageGRID 시스템 설치 및 구성이 완료되면 확인 메시지가 표시됩니다.

```
StorageGRID has been configured and installed.
```

VMware vSphere를 사용하여 StorageGRID에서 그리드 노드 배포 자동화

VMware OVF 도구를 사용하여 그리드 노드 배포를 자동화할 수 있습니다. 또한 StorageGRID의 구성을 자동화할 수 있습니다.

그리드 노드 배포 자동화

VMware OVF Tool을 사용하여 그리드 노드 배포를 자동화하십시오.

시작하기 전에

- Bash 3.2 이상 버전이 설치된 Linux/Unix 시스템에 접근할 수 있어야 합니다.
- VMware vSphere와 vCenter가 있습니다
- VMware OVF Tool이 설치되어 있고 올바르게 구성되어 있습니다.
- OVF Tool을 사용하여 VMware vSphere에 액세스하는 데 필요한 사용자 이름과 암호를 알고 있습니다.
- OVF 파일에서 가상 머신을 배포하고 전원을 켤 수 있는 충분한 권한이 있으며, 가상 머신에 연결할 추가 볼륨을 생성할 수 있는 권한도 있습니다. 자세한 내용은 `ovftool` 설명서를 참조하십시오.
- StorageGRID 가상 머신을 배포하려는 vSphere 위치의 가상 인프라(VI) URL을 알고 있습니다. 이 URL은 일반적으로 vApp 또는 리소스 풀입니다. 예를 들면 다음과 같습니다.
`vi://vcenter.example.com/vi/sgws`



VMware `ovftool` 유틸리티를 사용하여 이 값을 확인할 수 있습니다(자세한 내용은 `ovftool` 설명서를 참조하십시오).



vApp에 배포하는 경우, 가상 머신은 처음에 자동으로 시작되지 않으므로 수동으로 전원을 켜야 합니다.

- 배포 구성 파일에 필요한 모든 정보를 수집했습니다. 자세한 내용은 ["배포 환경에 대한 정보를 수집합니다."](#)을 참조하십시오.
- StorageGRID용 VMware 설치 아카이브에서 다음 파일에 액세스할 수 있습니다.

파일 이름	설명
NetApp-SG-version-SHA.vmdk	그리드 노드 가상 머신을 생성하기 위한 템플릿으로 사용되는 가상 머신 디스크 파일입니다. 참고: 이 파일은 .ovf 및 .mf 파일과 동일한 폴더에 있어야 합니다.
vsphere-primary-admin.ovf vsphere-primary-admin.mf	기본 관리 노드를 배포하기 위한 Open Virtualization Format 템플릿 파일(.ovf)과 매니페스트 파일(.mf)입니다.
vsphere-non-primary-admin.ovf vsphere-non-primary-admin.mf	기본이 아닌 관리 노드를 배포하기 위한 템플릿 파일(.ovf)과 매니페스트 파일(.mf)입니다.
vsphere-gateway.ovf vsphere-gateway.mf	게이트웨이 노드를 배포하기 위한 템플릿 파일 (.ovf) 및 매니페스트 파일 (.mf)입니다.
vsphere-storage.ovf vsphere-storage.mf	가상 머신 기반 스토리지 노드를 배포하기 위한 템플릿 파일(.ovf)과 매니페스트 파일(.mf)입니다.
deploy-vsphere-ovftool.sh	가상 그리드 노드 배포 자동화에 사용되는 Bash 셸 스크립트입니다.
deploy-vsphere-ovftool-sample.ini	<pre>`deploy-vsphere-ovftool.sh` 스크립트와 함께 사용할 예제 구성 파일입니다.</pre>

배포를 위한 구성 파일을 정의합니다.

StorageGRID의 가상 그리드 노드를 배포하는 데 필요한 정보를 구성 파일에 지정하며, 이 파일은 `deploy-vsphere-ovftool.sh` Bash 스크립트에서 사용됩니다. 예제 구성 파일을 수정하면 처음부터 파일을 만들 필요가 없습니다.

단계

1. 예제 구성 파일 (`deploy-vsphere-ovftool.sample.ini`)을 복사합니다. 새 파일을 ``deploy-vsphere-ovftool.ini``로 저장하되, ``deploy-vsphere-ovftool.sh``와 동일한 디렉토리에 저장합니다.
2. 열기 `deploy-vsphere-ovftool.ini`.
3. VMware 가상 그리드 노드를 배포하는 데 필요한 모든 정보를 입력하십시오.

자세한 내용은 [구성 파일 설정](#)을 참조하십시오.

4. 필요한 정보를 모두 입력하고 확인했으면 파일을 저장하고 닫으십시오.

구성 파일 설정

The `deploy-vsphere-ovftool.ini` 구성 파일에는 가상 그리드 노드를 배포하는 데 필요한 설정이 포함되어 있습니다.

구성 파일은 먼저 전역 매개변수를 나열하고, 그 다음 노드 이름으로 정의된 섹션에 노드별 매개변수를 나열합니다. 이 파일이 사용될 때:

- 전역 매개변수는 모든 그리드 노드에 적용됩니다.
- 노드별 매개변수는 전역 매개변수를 재정의합니다.

전역 매개변수

전역 매개변수는 개별 섹션의 설정으로 재정의되지 않는 한 모든 그리드 노드에 적용됩니다. 여러 노드에 적용되는 매개변수는 전역 매개변수 섹션에 배치한 다음, 개별 노드 섹션에서 필요에 따라 이러한 설정을 재정의하십시오.

- **OVFTOOL_ARGUMENTS:** OVFTOOL_ARGUMENTS는 전역 설정으로 지정하거나 특정 노드에 개별적으로 적용할 수 있습니다. 예를 들면 다음과 같습니다.

```
OVFTOOL_ARGUMENTS = --powerOn --noSSLVerify --diskMode=eagerZeroedThick
--datastore='datastore_name'
```

`--powerOffTarget` 및 `--overwrite` 옵션을 사용하여 기존 가상 머신을 종료하고 교체할 수 있습니다.



노드를 서로 다른 데이터 저장소에 배포하고 OVFTOOL_ARGUMENTS를 전역적으로 지정하는 대신 각 노드에 대해 지정해야 합니다.

- 출처: StorageGRID 가상 머신 템플릿 (.vmdk 파일과 개별 그리드 노드용 .ovf 및 .mf 파일의 경로입니다. 기본값은 현재 디렉토리입니다.

```
SOURCE = /downloads/StorageGRID-Webscale-version/vsphere
```

- 대상: StorageGRID가 배포될 위치의 VMware vSphere 가상 인프라(vi) URL입니다. 예:

```
TARGET = vi://vcenter.example.com/vm/sgws
```

- **GRID_NETWORK_CONFIG:** IP 주소를 가져오는 데 사용되는 방법(STATIC 또는 DHCP). 기본값은 STATIC입니다. 모든 노드 또는 대부분의 노드가 동일한 IP 주소 가져오기 방법을 사용하는 경우 여기에 해당 방법을 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
GRID_NETWORK_CONFIG = STATIC
```

- **GRID_NETWORK_TARGET:** 그리드 네트워크에 사용할 기존 VMware 네트워크의 이름입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 이름을 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
GRID_NETWORK_TARGET = SG Admin Network
```

- **GRID_NETWORK_MASK:** 그리드 네트워크의 네트워크 마스크입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 마스크를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
GRID_NETWORK_MASK = 255.255.255.0
```

- **GRID_NETWORK_GATEWAY:** 그리드 네트워크의 네트워크 게이트웨이입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 게이트웨이를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
GRID_NETWORK_GATEWAY = 10.1.0.1
```

- **GRID_NETWORK_MTU:** 선택 사항입니다. 그리드 네트워크의 최대 전송 단위(MTU)입니다. 지정하는 경우 값은 1280에서 9216 사이여야 합니다. 예:

```
GRID_NETWORK_MTU = 9000
```

생략할 경우 1400이 사용됩니다.

점보 프레임을 사용하려면 MTU 값을 9000과 같이 점보 프레임에 적합한 값으로 설정하십시오. 그렇지 않으면 기본값을 유지하십시오.



네트워크의 MTU 값은 노드가 연결된 vSphere의 가상 스위치 포트에 구성된 값과 일치해야 합니다. 그렇지 않으면 네트워크 성능 문제 또는 패킷 손실이 발생할 수 있습니다.



최적의 네트워크 성능을 위해서는 모든 노드의 그리드 네트워크 인터페이스에 유사한 MTU 값을 구성해야 합니다. 개별 노드의 그리드 네트워크 MTU 설정에 상당한 차이가 있는 경우 그리드 네트워크 **MTU** 불일치 경고가 발생합니다. 모든 네트워크 유형에 대해 MTU 값이 동일할 필요는 없습니다.

- **ADMIN_NETWORK_CONFIG:** IP 주소를 가져오는 데 사용되는 방법으로, DISABLED, STATIC 또는 DHCP 중 하나를 선택할 수 있습니다. 기본값은 DISABLED입니다. 모든 노드 또는 대부분의 노드가 동일한 IP 주소 가져오기 방법을 사용하는 경우, 여기에 해당 방법을 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
ADMIN_NETWORK_CONFIG = STATIC
```

- **ADMIN_NETWORK_TARGET:** 관리자 네트워크에 사용할 기존 VMware 네트워크의 이름입니다. 관리자 네트워크가 비활성화된 경우를 제외하고 이 설정은 필수입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 이름을 사용하는 경우 여기에 지정할 수 있습니다. 그리드 네트워크와 달리 모든 노드가 동일한 관리자 네트워크에 연결될 필요는 없습니다. 따라서 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
ADMIN_NETWORK_TARGET = SG Admin Network
```

- **ADMIN_NETWORK_MASK:** 관리자 네트워크의 네트워크 마스크입니다. 고정 IP 주소를 사용하는 경우 이 설정이 필수입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 마스크를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
ADMIN_NETWORK_MASK = 255.255.255.0
```

- **ADMIN_NETWORK_GATEWAY:** 관리 네트워크의 네트워크 게이트웨이입니다. 고정 IP 주소를 사용하고 ADMIN_NETWORK_ESL 설정에서 외부 서브넷을 지정하는 경우 이 설정이 필요합니다. (즉, ADMIN_NETWORK_ESL이 비어 있는 경우에는 필요하지 않습니다.) 모든 노드 또는 대부분의 노드가 동일한 네트워크 게이트웨이를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
ADMIN_NETWORK_GATEWAY = 10.3.0.1
```

- **ADMIN_NETWORK_ESL:** 관리자 네트워크의 외부 서브넷 목록(경로)입니다. CIDR 경로 대상을 쉼표로 구분하여 지정합니다. 모든 노드 또는 대부분의 노드가 동일한 외부 서브넷 목록을 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
ADMIN_NETWORK_ESL = 172.16.0.0/21,172.17.0.0/21
```

- **ADMIN_NETWORK_MTU:** 선택 사항입니다. 관리자 네트워크의 최대 전송 단위(MTU)입니다. ADMIN_NETWORK_CONFIG = DHCP인 경우에는 지정하지 마십시오. 지정하는 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1400이 사용됩니다. 점보 프레임을 사용하려면 MTU를 9000과 같이 점보 프레임에 적합한 값으로 설정하십시오. 그렇지 않으면 기본값을 유지하십시오. 모든 노드 또는 대부분의 노드가 관리자 네트워크에 대해 동일한 MTU를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
ADMIN_NETWORK_MTU = 8192
```

- **CLIENT_NETWORK_CONFIG:** IP 주소를 가져오는 데 사용되는 방법으로, DISABLED, STATIC 또는 DHCP 중 하나를 지정합니다. 기본값은 DISABLED입니다. 모든 노드 또는 대부분의 노드가 동일한 IP 주소 가져오기 방법을 사용하는 경우 여기에 해당 방법을 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
CLIENT_NETWORK_CONFIG = STATIC
```

- **CLIENT_NETWORK_TARGET:** 클라이언트 네트워크에 사용할 기존 VMware 네트워크의 이름입니다. 클라이언트 네트워크가 비활성화된 경우를 제외하고 이 설정은 필수입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 이름을 사용하는 경우 여기에 지정할 수 있습니다. 그리드 네트워크와 달리 모든 노드가 동일한 클라이언트 네트워크에 연결될 필요는 없습니다. 따라서 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
CLIENT_NETWORK_TARGET = SG Client Network
```

- **CLIENT_NETWORK_MASK:** 클라이언트 네트워크의 네트워크 마스크입니다. 고정 IP 주소를 사용하는 경우 이 설정이 필수입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 마스크를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
CLIENT_NETWORK_MASK = 255.255.255.0
```

- **CLIENT_NETWORK_GATEWAY:** 클라이언트 네트워크의 네트워크 게이트웨이입니다. 고정 IP 주소를 사용하는 경우 이 설정이 필수입니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 게이트웨이를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
CLIENT_NETWORK_GATEWAY = 10.4.0.1
```

- **CLIENT_NETWORK_MTU:** 선택 사항입니다. 클라이언트 네트워크의 최대 전송 단위(MTU)입니다. CLIENT_NETWORK_CONFIG = DHCP인 경우에는 지정하지 마십시오. 지정하는 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1400이 사용됩니다. 점보 프레임을 사용하려면 MTU를 9000과 같이 점보 프레임에 적합한 값으로 설정하십시오. 그렇지 않으면 기본값을 유지하십시오. 모든 노드 또는 대부분의 노드가 클라이언트 네트워크에 대해 동일한 MTU를 사용하는 경우 여기에 지정할 수 있습니다. 그런 다음 하나 이상의 개별 노드에 대해 다른 설정을 지정하여 전역 설정을 재정의할 수 있습니다. 예를 들면 다음과 같습니다.

```
CLIENT_NETWORK_MTU = 8192
```

- **PORT_REMAP:** 노드가 내부 그리드 노드 통신 또는 외부 통신에 사용하는 포트를 다시 매핑합니다. 엔터프라이즈 네트워킹 정책으로 인해 StorageGRID에서 사용하는 하나 이상의 포트가 제한되는 경우 포트 재매핑이 필요합니다. StorageGRID에서 사용하는 포트 목록은 "[네트워킹 가이드라인](#)"의 내부 그리드 노드 통신 및 외부 통신을 참조하십시오.



로드 밸런서 엔드포인트를 구성하는 데 사용할 포트를 다시 매핑하지 마십시오.



PORT_REMAP만 설정된 경우, 지정한 매핑이 수신 및 송신 통신 모두에 사용됩니다. PORT_REMAP_INBOUND도 지정된 경우, PORT_REMAP은 송신 통신에만 적용됩니다.

사용되는 형식은 다음과 같습니다. *network type/protocol/default port used by grid*

node/new port 여기서 네트워크 유형은 `grid`, `admin` 또는 `client`이고 프로토콜은 `tcp` 또는 `udp`입니다.

예를 들면 다음과 같습니다.

```
PORT_REMAP = client/tcp/18082/443
```

이 예시 설정을 단독으로 사용하면 그리드 노드의 18082번 포트에서 443번 포트로의 수신 및 송신 통신을 대칭적으로 매핑합니다. `PORT_REMAP_INBOUND`와 함께 사용하면 이 예시 설정은 18082번 포트에서 443번 포트로의 송신 통신을 매핑합니다.

심표로 구분된 목록을 사용하여 여러 포트를 재매핑할 수도 있습니다.

예를 들면 다음과 같습니다.

```
PORT_REMAP = client/tcp/18082/443, client/tcp/18083/80
```

- **PORT_REMAP_INBOUND:** 지정된 포트에 대한 수신 통신을 재매핑합니다. `PORT_REMAP_INBOUND`를 지정했지만 `PORT_REMAP` 값을 지정하지 않으면 해당 포트의 송신 통신은 변경되지 않습니다.



로드 밸런서 엔드포인트를 구성하는 데 사용할 포트를 다시 매핑하지 마십시오.

사용되는 형식은 다음과 같습니다. *network type/protocol/_default port used by grid node / new port* 여기서 네트워크 유형은 `grid`, `admin` 또는 `client`이고 프로토콜은 `tcp` 또는 `udp`입니다.

예를 들면 다음과 같습니다.

```
PORT_REMAP_INBOUND = client/tcp/443/18082
```

이 예시는 포트 443으로 전송되는 트래픽을 내부 방화벽을 통과시켜 그리드 노드가 S3 요청을 수신 대기하는 포트 18082로 전달하는 방법을 보여줍니다.

심표로 구분된 목록을 사용하여 여러 개의 수신 포트를 다시 매핑할 수도 있습니다.

예를 들면 다음과 같습니다.

```
PORT_REMAP_INBOUND = grid/tcp/3022/22, admin/tcp/3022/22
```

- **TEMPORARY_PASSWORD_TYPE:** 노드가 그리드에 참여하기 전에 VM 콘솔이나 StorageGRID 설치 API에 액세스하거나 SSH를 사용할 때 사용할 임시 설치 암호 유형입니다.



모든 노드 또는 대부분의 노드에서 동일한 유형의 임시 설치 암호를 사용하는 경우, 전역 매개변수 섹션에서 해당 유형을 지정하십시오. 그런 다음, 선택적으로 개별 노드에 대해 다른 설정을 사용할 수 있습니다. 예를 들어, 전역적으로 `*Use Custom Password*`를 선택한 경우 `*CUSTOM_TEMPORARY_PASSWORD=<password>*`를 사용하여 각 노드의 암호를 설정할 수 있습니다.

TEMPORARY_PASSWORD_TYPE 은 다음 중 하나일 수 있습니다.

- 노드 이름 사용: 노드 이름은 임시 설치 암호로 사용되며 VM 콘솔, StorageGRID 설치 API 및 SSH에 대한 액세스를 제공합니다.
- 암호 비활성화: 임시 설치 암호는 사용되지 않습니다. 설치 문제를 디버깅하기 위해 VM에 액세스해야 하는 경우 ["설치 문제 해결"](#)을 참조하십시오.
- 사용자 지정 암호 사용: *CUSTOM_TEMPORARY_PASSWORD=<password>*에 제공된 값은 임시 설치 암호로 사용되며 VM 콘솔, StorageGRID 설치 API 및 SSH에 액세스할 수 있도록 합니다.



선택적으로 **TEMPORARY_PASSWORD_TYPE** 매개변수를 생략하고 *CUSTOM_TEMPORARY_PASSWORD=<password>*만 지정할 수 있습니다.

- **CUSTOM_TEMPORARY_PASSWORD=<password>** (선택 사항). VM 콘솔, StorageGRID 설치 API 및 SSH에 액세스할 때 설치 중에 사용할 임시 암호입니다. **TEMPORARY_PASSWORD_TYPE***이 *노드 이름 사용 또는 *암호 비활성화*로 설정된 경우 무시됩니다.

노드별 매개변수

각 노드는 구성 파일의 자체 섹션에 있습니다. 각 노드에는 다음 설정이 필요합니다.

- 섹션 제목은 Grid Manager에 표시될 노드 이름을 정의합니다. 노드에 대해 선택적 **NODE_NAME** 매개변수를 지정하여 해당 값을 재정의할 수 있습니다.
- **NODE_TYPE**: VM_Admin_Node, VM_Storage_Node 또는 VM_API_Gateway_Node
- **STORAGE_TYPE**: combined, data 또는 metadata. 스토리지 노드에 대한 이 선택적 매개변수는 지정되지 않은 경우 기본적으로 combined(데이터 및 메타데이터)로 설정됩니다. 자세한 내용은 ["스토리지 노드의 유형"](#)을 참조하십시오.
- **GRID_NETWORK_IP**: 그리드 네트워크에서 노드의 IP 주소입니다.
- **ADMIN_NETWORK_IP**: 관리 네트워크에서 노드의 IP 주소입니다. 노드가 관리 네트워크에 연결되어 있고 ADMIN_NETWORK_CONFIG가 STATIC으로 설정된 경우에만 필요합니다.
- **CLIENT_NETWORK_IP**: 클라이언트 네트워크의 노드에 대한 IP 주소입니다. 노드가 클라이언트 네트워크에 연결되어 있고 해당 노드의 CLIENT_NETWORK_CONFIG가 STATIC으로 설정된 경우에만 필요합니다.
- **ADMIN_IP**: 그리드 네트워크의 기본 관리 노드 IP 주소입니다. 기본 관리 노드의 경우 GRID_NETWORK_IP로 지정한 값을 사용하십시오. 이 매개변수를 생략하면 노드는 mDNS를 사용하여 기본 관리 노드 IP를 검색하려고 시도합니다. 자세한 내용은 ["그리드 노드가 기본 관리 노드를 찾는 방법"](#)을 참조하십시오.



ADMIN_IP 매개변수는 기본 관리 노드의 경우 무시됩니다.

- 전역적으로 설정되지 않은 매개변수입니다. 예를 들어 노드가 관리 네트워크에 연결되어 있고 ADMIN_NETWORK 매개변수를 전역적으로 지정하지 않은 경우 해당 노드에 대해 해당 매개변수를 지정해야 합니다.

기본 관리자 노드

기본 관리자 노드에는 다음과 같은 추가 설정이 필요합니다.

- **NODE_TYPE**: VM_Admin_Node
- **ADMIN_ROLE**: 기본

이 예시 항목은 세 개의 네트워크 모두에 연결된 기본 관리 노드에 대한 것입니다.

```
[DC1-ADM1]
ADMIN_ROLE = Primary
NODE_TYPE = VM_Admin_Node
TEMPORARY_PASSWORD_TYPE = Use custom password
CUSTOM_TEMPORARY_PASSWORD = Passw0rd

GRID_NETWORK_IP = 10.1.0.2
ADMIN_NETWORK_IP = 10.3.0.2
CLIENT_NETWORK_IP = 10.4.0.2
```

다음 추가 설정은 기본 관리 노드에 대해 선택 사항입니다.

- **DISK:** 기본적으로 관리 노드에는 감사 및 데이터베이스 용도로 200GB 하드 디스크 두 개가 추가로 할당됩니다. DISK 매개변수를 사용하여 이 설정을 늘릴 수 있습니다. 예를 들면 다음과 같습니다.

```
DISK = INSTANCES=2, CAPACITY=300
```



관리자 노드의 경우 INSTANCES는 항상 2여야 합니다.

스토리지 노드

스토리지 노드에는 다음과 같은 추가 설정이 필요합니다.

- **NODE_TYPE:** VM_Storage_Node

이 예시 항목은 그리드 및 관리 네트워크에는 연결되어 있지만 클라이언트 네트워크에는 연결되어 있지 않은 스토리지 노드에 대한 것입니다. 이 노드는 ADMIN_IP 설정을 사용하여 그리드 네트워크에서 기본 관리 노드의 IP 주소를 지정합니다.

```
[DC1-S1]
NODE_TYPE = VM_Storage_Node

GRID_NETWORK_IP = 10.1.0.3
ADMIN_NETWORK_IP = 10.3.0.3

ADMIN_IP = 10.1.0.2
```

두 번째 예시 항목은 고객의 엔터프라이즈 네트워킹 정책에 따라 S3 클라이언트 애플리케이션이 포트 80 또는 443을 통해서만 스토리지 노드에 액세스할 수 있도록 설정된 클라이언트 네트워크의 스토리지 노드에 대한 것입니다. 이 예시 구성 파일은 PORT_REMAP을 사용하여 스토리지 노드가 포트 443에서 S3 메시지를 송수신할 수 있도록 합니다.

```
[DC2-S1]
NODE_TYPE = VM_Storage_Node

GRID_NETWORK_IP = 10.1.1.3
CLIENT_NETWORK_IP = 10.4.1.3
PORT_REMAP = client/tcp/18082/443

ADMIN_IP = 10.1.0.2
```

마지막 예제는 포트 22에서 포트 3022로의 SSH 트래픽에 대한 대칭 재매핑을 생성하지만, 수신 및 송신 트래픽 모두에 대한 값을 명시적으로 설정합니다.

```
[DC1-S3]
NODE_TYPE = VM_Storage_Node

GRID_NETWORK_IP = 10.1.1.3

PORT_REMAP = grid/tcp/22/3022
PORT_REMAP_INBOUND = grid/tcp/3022/22

ADMIN_IP = 10.1.0.2
```

스토리지 노드에 대해 다음 추가 설정은 선택 사항입니다.

- **DISK:** 기본적으로 스토리지 노드에는 RangeDB 사용을 위해 4TB 디스크 3개가 할당됩니다. DISK 매개변수를 사용하여 이러한 설정을 늘릴 수 있습니다. 예를 들면 다음과 같습니다.

```
DISK = INSTANCES=16, CAPACITY=4096
```

- **STORAGE_TYPE:** 기본적으로 모든 새 스토리지 노드는 객체 데이터와 메타데이터를 모두 저장하도록 구성되며, 이를 결합형 스토리지 노드라고 합니다. STORAGE_TYPE 매개변수를 사용하여 스토리지 노드 유형을 데이터만 저장하거나 메타데이터만 저장하도록 변경할 수 있습니다. 예를 들면 다음과 같습니다.

```
STORAGE_TYPE = data
```

게이트웨이 노드

게이트웨이 노드에는 다음과 같은 추가 설정이 필요합니다.

- **NODE_TYPE:** VM_API_Gateway

이 예시 항목은 세 가지 네트워크 모두에서 게이트웨이 노드를 사용하는 예시입니다. 이 예시에서는 구성 파일의 전역 섹션에 클라이언트 네트워크 매개변수가 지정되지 않았으므로 노드에 대해 해당 매개변수를 지정해야 합니다.

```
[DC1-G1]
NODE_TYPE = VM_API_Gateway

GRID_NETWORK_IP = 10.1.0.5
ADMIN_NETWORK_IP = 10.3.0.5

CLIENT_NETWORK_CONFIG = STATIC
CLIENT_NETWORK_TARGET = SG Client Network
CLIENT_NETWORK_MASK = 255.255.255.0
CLIENT_NETWORK_GATEWAY = 10.4.0.1
CLIENT_NETWORK_IP = 10.4.0.5

ADMIN_IP = 10.1.0.2
```

비기본 관리 노드

기본 관리 노드가 아닌 경우 다음과 같은 추가 설정이 필요합니다.

- **NODE_TYPE:** VM_Admin_Node
- **ADMIN_ROLE:** 비기본

이 예시 항목은 클라이언트 네트워크에 연결되어 있지 않은 비기본 관리 노드에 대한 것입니다.

```
[DC2-ADM1]
ADMIN_ROLE = Non-Primary
NODE_TYPE = VM_Admin_Node

GRID_NETWORK_TARGET = SG Grid Network
GRID_NETWORK_IP = 10.1.0.6
ADMIN_NETWORK_IP = 10.3.0.6

ADMIN_IP = 10.1.0.2
```

다음 추가 설정은 기본 관리 노드가 아닌 경우 선택 사항입니다.

- **DISK:** 기본적으로 관리 노드에는 감사 및 데이터베이스 용도로 200GB 하드 디스크 두 개가 추가로 할당됩니다. DISK 매개변수를 사용하여 이 설정을 늘릴 수 있습니다. 예를 들면 다음과 같습니다.

```
DISK = INSTANCES=2, CAPACITY=300
```



관리자 노드의 경우 INSTANCES는 항상 2여야 합니다.

Bash 스크립트를 실행합니다

수정된 Bash 스크립트와 deploy-vmware-ovftool.ini 구성 파일을 사용하여 deploy-vmware-ovftool.sh VMware vSphere에서 StorageGRID 노드 배포를 자동화할 수 있습니다.

시작하기 전에

사용자 환경에 맞는 deploy-vmware-ovftool.ini 구성 파일을 생성했습니다.

Bash 스크립트에서 제공하는 도움말을 사용하려면 help 명령어를 입력하십시오((-h/--help). 예를 들면 다음과 같습니다.

```
./deploy-vmware-ovftool.sh -h
```

또는

```
./deploy-vmware-ovftool.sh --help
```

단계

1. Bash 스크립트를 실행하는 데 사용하는 Linux 시스템에 로그인하십시오.
2. 설치 압축 파일을 압축 해제한 디렉토리로 이동합니다.

예를 들면 다음과 같습니다.

```
cd StorageGRID-WebScale-version/vsphere
```

3. 모든 그리드 노드를 배포하려면 환경에 맞는 옵션을 사용하여 Bash 스크립트를 실행하십시오.

예를 들면 다음과 같습니다.

```
./deploy-vmware-ovftool.sh --username=user --password=pwd ./deploy-vmware-ovftool.ini
```

4. 그리드 노드 배포에 오류가 발생한 경우, 오류를 해결하고 해당 노드에 대해서만 Bash 스크립트를 다시 실행하십시오.

예를 들면 다음과 같습니다.

```
./deploy-vmware-ovftool.sh --username=user --password=pwd --single -node="DC1-S3" ./deploy-vmware-ovftool.ini
```

각 노드의 상태가 "Passed"로 표시되면 배포가 완료된 것입니다.

Deployment Summary

node	attempts	status
DC1-ADM1	1	Passed
DC1-G1	1	Passed
DC1-S1	1	Passed
DC1-S2	1	Passed
DC1-S3	1	Passed

StorageGRID 구성 자동화

그리드 노드를 배포한 후에는 StorageGRID 시스템의 구성을 자동화할 수 있습니다.

시작하기 전에

- 설치 아카이브에서 다음 파일의 위치를 알고 있습니다.

파일 이름	설명
configure-storagegrid.py	구성을 자동화하는 데 사용되는 Python 스크립트
configure-storagegrid.sample.json	스크립트와 함께 사용할 예시 구성 파일
configure-storagegrid.blank.json	스크립트와 함께 사용할 빈 구성 파일

- 사용자가 configure-storagegrid.json 구성 파일을 생성했습니다. 이 파일을 생성하려면 예제 구성 파일 (configure-storagegrid.sample.json) 또는 빈 구성 파일 (configure-storagegrid.blank.json)을 수정하면 됩니다.



수정된 configure-storagegrid.json 구성 파일의 암호 섹션에 있는 관리 암호와 프로비저닝 암호를 안전한 장소에 보관하십시오. 이러한 암호는 설치, 확장 및 유지 관리 절차에 필요합니다. 또한 수정된 configure-storagegrid.json 구성 파일을 백업하여 안전한 장소에 보관해야 합니다.

이 작업 정보

`configure-storagegrid.py` Python 스크립트와 `configure-storagegrid.json` 그리드 구성 파일을 사용하여 StorageGRID 시스템의 구성을 자동화할 수 있습니다.



그리드 관리자 또는 설치 API를 사용하여 시스템을 구성할 수도 있습니다.

단계

1. Python 스크립트를 실행하는 데 사용하는 Linux 시스템에 로그인합니다.
2. 설치 압축 파일을 압축 해제한 디렉토리로 이동합니다.

예를 들면 다음과 같습니다.

```
cd StorageGRID-Webscale-version/platform
```

여기서 platform 는 debs, rpms, 또는 vsphere입니다.

3. Python 스크립트를 실행하고 생성한 구성 파일을 사용하십시오.

예를 들면 다음과 같습니다.

```
./configure-storagegrid.py ./configure-storagegrid.json --start-install
```

결과

복구 패키지 .zip 파일은 구성 과정 중에 생성되어 설치 및 구성 프로세스를 실행하는 디렉터리로 다운로드됩니다. 하나 이상의 그리드 노드에 장애가 발생할 경우 StorageGRID 시스템을 복구할 수 있도록 복구 패키지 파일을 백업해야 합니다. 예를 들어, 안전한 네트워크 백업 위치와 안전한 클라우드 스토리지 위치에 복사해 두십시오.



복구 패키지 파일에는 StorageGRID 시스템에서 데이터를 가져오는 데 사용할 수 있는 암호화 키와 암호가 포함되어 있으므로 보안을 유지해야 합니다.

무작위 암호 생성을 지정한 경우, Passwords.txt 파일을 열고 StorageGRID 시스템에 액세스하는 데 필요한 암호를 찾으십시오.

```
#####
##### The StorageGRID "Recovery Package" has been downloaded as: #####
#####      ./sgws-recovery-package-994078-rev1.zip      #####
#####   Safeguard this file as it will be needed in case of a   #####
#####                        StorageGRID node recovery.                        #####
#####
```

StorageGRID 시스템 설치 및 구성이 완료되면 확인 메시지가 표시됩니다.

```
StorageGRID has been configured and installed.
```

관련 정보

- ["Grid Manager로 이동합니다."](#)
- ["설치 REST API"](#)

가상 그리드 노드 배포

StorageGRID용 VMware 환경에 대한 정보 수집

그리드 노드를 배포하기 전에 네트워크 구성 및 VMware 환경에 대한 정보를 수집해야 합니다.



일부 노드를 지금 설치하고 나머지 노드를 나중에 설치하는 것보다 모든 노드를 한 번에 단일 설치하는 것이 더 효율적입니다.

VMware 정보

배포 환경에 액세스하여 VMware 환경에 대한 정보, 그리드, 관리 및 클라이언트 네트워크를 위해 생성된 네트워크, 그리고 스토리지 노드에 사용할 스토리지 볼륨 유형에 대한 정보를 수집해야 합니다.

VMware 환경에 대한 다음 정보를 수집해야 합니다.

- 배포를 완료하는 데 필요한 권한이 있는 VMware vSphere 계정의 사용자 이름과 암호입니다.
- 각 StorageGRID 노드 가상 머신의 호스트, 데이터 저장소 및 네트워크 구성 정보입니다.



VMware live vMotion은 가상 머신의 시계 시간을 변경시키며, 모든 유형의 그리드 노드에서는 지원되지 않습니다. 드물지만, 시계 시간이 잘못되면 데이터 손실이나 구성 업데이트 오류가 발생할 수 있습니다.

그리드 네트워크 정보

StorageGRID Grid Network용으로 생성된 VMware 네트워크에 대한 정보를 수집해야 합니다(필수). 여기에는 다음 정보가 포함됩니다.

- 네트워크 이름입니다.
- IP 주소를 할당하는 데 사용되는 방법으로, 정적 또는 DHCP입니다.
 - 고정 IP 주소를 사용하는 경우 각 그리드 노드에 필요한 네트워킹 세부 정보(IP 주소, 게이트웨이, 네트워크 마스크).
 - DHCP를 사용하는 경우, 그리드 네트워크의 기본 관리 노드의 IP 주소입니다. 자세한 내용은 ["그리드 노드가 기본 관리 노드를 찾는 방법"](#)을 참조하십시오.

관리자 네트워크 정보

선택적 StorageGRID 관리 네트워크에 연결될 노드의 경우, 이 네트워크를 위해 생성된 VMware 네트워크에 대한 정보를 수집해야 합니다. 여기에는 다음 정보가 포함됩니다.

- 네트워크 이름입니다.
- IP 주소를 할당하는 데 사용되는 방법으로, 정적 또는 DHCP입니다.
 - 고정 IP 주소를 사용하는 경우 각 그리드 노드에 필요한 네트워킹 세부 정보(IP 주소, 게이트웨이, 네트워크 마스크).
 - DHCP를 사용하는 경우, 그리드 네트워크의 기본 관리 노드의 IP 주소입니다. 자세한 내용은 ["그리드 노드가 기본 관리 노드를 찾는 방법"](#)을 참조하십시오.
- 관리자 네트워크용 외부 서브넷 목록(ESL)입니다.

클라이언트 네트워크 정보

선택적 StorageGRID 클라이언트 네트워크에 연결될 노드의 경우, 이 네트워크를 위해 생성된 VMware 네트워크에 대한 정보를 수집해야 합니다. 여기에는 다음 정보가 포함됩니다.

- 네트워크 이름입니다.
- IP 주소를 할당하는 데 사용되는 방법으로, 정적 또는 DHCP입니다.
- 고정 IP 주소를 사용하는 경우 각 그리드 노드에 필요한 네트워킹 세부 정보(IP 주소, 게이트웨이, 네트워크 마스크).

추가 인터페이스에 대한 정보

노드를 설치한 후 선택적으로 vCenter의 VM에 트렁크 또는 액세스 인터페이스를 추가할 수 있습니다. 예를 들어, 관리 노드 또는 게이트웨이 노드에 트렁크 인터페이스를 추가하여 VLAN 인터페이스를 통해 서로 다른 애플리케이션이나 테넌트의 트래픽을 분리할 수 있습니다. 또는고가용성(HA) 그룹에서 사용할 액세스 인터페이스를 추가할 수도 있습니다.

추가한 인터페이스는 Grid Manager의 VLAN 인터페이스 페이지와 HA 그룹 페이지에 표시됩니다.

- 트렁크 인터페이스를 추가하는 경우, 새 상위 인터페이스마다 하나 이상의 VLAN 인터페이스를 구성해야 합니다. "[VLAN 인터페이스 구성](#)"을 참조하십시오.
- 액세스 인터페이스를 추가하는 경우 HA 그룹에 직접 추가해야 합니다. "[고가용성 그룹 구성](#)"을 참조하십시오.

가상 스토리지 노드용 스토리지 볼륨

가상 머신 기반 스토리지 노드의 경우 다음 정보를 수집해야 합니다.

- 추가할 스토리지 볼륨(스토리지 LUN)의 개수 및 크기입니다. "[스토리지 및 성능 요구 사항](#)"을 참조하십시오.

그리드 구성 정보

그리드를 구성하려면 정보를 수집해야 합니다.

- 그리드 라이선스
- 네트워크 시간 프로토콜(NTP) 서버 IP 주소
- DNS 서버 IP 주소

Linux 배포용 StorageGRID 노드 구성 파일을 생성합니다

노드 구성 파일은 StorageGRID 호스트 서비스가 노드를 시작하고 적절한 네트워크 및 블록 스토리지 리소스에 연결하는 데 필요한 정보를 제공하는 작은 텍스트 파일입니다. 노드 구성 파일은 가상 노드에 사용되며 어플라이언스 노드에는 사용되지 않습니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 "[NetApp 상호 운용성 매트릭스 툴\(IMT\)](#)"을 참조하십시오.

노드 구성 파일의 위치

각 StorageGRID 노드의 구성 파일을 해당 노드가 실행될 호스트의 `/etc/storagegrid/nodes` 디렉터리에

배치하십시오. 예를 들어, HostA에서 관리 노드 1개, 게이트웨이 노드 1개, 스토리지 노드 1개를 실행하려면 HostA의 `/etc/storagegrid/nodes`에 노드 구성 파일 3개를 배치해야 합니다.

vim 또는 nano 등의 텍스트 편집기를 사용하여 각 호스트에 직접 구성 파일을 만들거나, 다른 곳에서 만든 후 각 호스트로 옮길 수 있습니다.

노드 구성 파일 이름 지정

구성 파일의 이름은 중요합니다. 형식은 ``node-name.conf``이며, 여기서 ``node-name``은 노드에 지정하는 이름입니다. 이 이름은 StorageGRID 설치 프로그램에 표시되며 노드 마이그레이션과 같은 노드 유지 관리 작업에 사용됩니다.

노드 이름은 다음 규칙을 따라야 합니다.

- 고유해야 합니다.
- 반드시 문자로 시작해야 합니다.
- A부터 Z까지의 문자와 a부터 z까지의 문자를 포함할 수 있습니다.
- 0부터 9까지의 숫자를 포함할 수 있습니다.
- 하나 이상의 하이픈(-)을 포함할 수 있습니다.
- 확장자를 제외하고 32자 이내여야 합니다. `.conf`

``/etc/storagegrid/nodes``에서 이러한 명명 규칙을 따르지 않는 파일은 호스트 서비스에서 구문 분석되지 않습니다.

그리드에 다중 사이트 토폴로지를 계획하고 있다면 일반적인 노드 명명 체계는 다음과 같습니다.

`site-nodetype-nodenum.conf`

예를 들어, 데이터 센터 1의 첫 번째 관리 노드에는 ``dc1-adm1.conf``를, 데이터 센터 2의 세 번째 스토리지 노드에는 ``dc2-sn3.conf``를 사용할 수 있습니다. 하지만 모든 노드 이름이 명명 규칙을 준수하는 한 원하는 어떤 체계든 사용할 수 있습니다.

노드 구성 파일의 내용

구성 파일에는 키/값 쌍이 포함되어 있으며, 각 줄에는 키 하나와 값 하나가 있습니다. 각 키/값 쌍에 대해 다음 규칙을 따르십시오.

- 키와 값은 등호(=)와 선택적 공백으로 구분해야 합니다.
- 키에는 공백이 포함될 수 없습니다.
- 값에는 공백이 포함될 수 있습니다.
- 앞뒤 공백은 무시됩니다.

다음 표는 지원되는 모든 키에 대한 값을 정의합니다. 각 키는 다음 중 하나의 지정자를 가집니다.

- 필수: 모든 노드 또는 지정된 노드 유형에 대해 필수입니다.
- 모범 사례: 선택 사항이지만 권장합니다.

- 선택 사항: 모든 노드에 대해 선택 사항입니다.

관리자 네트워크 키

ADMIN_IP

가치	지정
Linux 기반 노드를 설치하는 데 사용할 관리 노드의 그리드 네트워크 IPv4 주소입니다. 복구를 위해서는 가능한 경우 기본 관리 노드의 IP 주소를 사용하고, 그렇지 않으면 기본 관리 노드가 아닌 다른 관리 노드의 IP 주소를 사용하십시오. 이 매개변수를 생략하면 노드는 mDNS를 사용하여 기본 관리 노드를 검색하려고 시도합니다. "그리드 노드가 기본 관리 노드를 찾는 방법" 참고: 이 값은 기본 관리 노드에서 무시되며 사용이 금지될 수 있습니다.	모범 사례

ADMIN_NETWORK_CONFIG

가치	지정
DHCP, STATIC 또는 DISABLED	선택 사항

ADMIN_NETWORK_ESL

가치	지정
이 노드가 관리 네트워크 게이트웨이를 사용하여 통신해야 하는 서브넷의 쉼표로 구분된 CIDR 표기법 목록입니다. 예: 172.16.0.0/21, 172.17.0.0/21	선택 사항

ADMIN_NETWORK_GATEWAY

가치	지정
이 노드의 로컬 관리 네트워크 게이트웨이의 IPv4 주소입니다. ADMIN_NETWORK_IP 및 ADMIN_NETWORK_MASK로 정의된 서브넷에 있어야 합니다. DHCP로 구성된 네트워크의 경우 이 값은 무시됩니다. 예: 1.1.1.1 10.224.4.81	`ADMIN_NETWORK_ESL`이 (가) 지정된 경우 필수 항목입니다. 그렇지 않은 경우 선택 사항입니다.

ADMIN_NETWORK_IP

가치	지정
관리 네트워크에서 이 노드의 IPv4 주소입니다. 이 키는 ADMIN_NETWORK_CONFIG = STATIC인 경우에만 필요하며, 다른 값의 경우에는 지정하지 마십시오. 예: 1.1.1.1 10.224.4.81	ADMIN_NETWORK_CONFIG = STATIC인 경우 필수입니다. 그렇지 않으면 선택 사항입니다.

ADMIN_NETWORK_MAC

가치	지정
컨테이너의 관리 네트워크 인터페이스에 대한 MAC 주소입니다. 이 필드는 선택 사항입니다. 생략할 경우 MAC 주소가 자동으로 생성됩니다. 6쌍의 16진수 숫자로 구성되어야 하며, 각 숫자는 콜론으로 구분되어야 합니다. 예: b2:9c:02:c2:27:10	선택 사항

ADMIN_NETWORK_MASK

가치	지정
관리자 네트워크에서 이 노드에 사용할 IPv4 넷마스크입니다. ADMIN_NETWORK_CONFIG = STATIC인 경우 이 키를 지정하고, 다른 값의 경우에는 지정하지 마십시오. 예: 255.255.255.0 255.255.248.0	ADMIN_NETWORK_IP가 지정되고 ADMIN_NETWORK_CONFIG = STATIC인 경우 필수입니다. 그렇지 않으면 선택 사항입니다.

ADMIN_NETWORK_MTU

가치	지정
관리자 네트워크에서 이 노드의 최대 전송 단위(MTU)입니다. ADMIN_NETWORK_CONFIG가 DHCP인 경우에는 지정하지 마십시오. 지정하는 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1500이 사용됩니다. 점보 프레임을 사용하려면 MTU 값을 9000과 같이 점보 프레임에 적합한 값으로 설정하십시오. 그렇지 않으면 기본값을 유지하십시오. 중요: 네트워크의 MTU 값은 노드가 연결된 스위치 포트에 구성된 값과 일치해야 합니다. 그렇지 않으면 네트워크 성능 문제 또는 패킷 손실이 발생할 수 있습니다. 예: 1500 8192	선택 사항

ADMIN_NETWORK_TARGET

가치	지정
StorageGRID 노드에서 관리 네트워크에 액세스하는 데 사용할 호스트 장치의 이름입니다. 네트워크 인터페이스 이름만 지원됩니다. 일반적으로 GRID_NETWORK_TARGET 또는 CLIENT_NETWORK_TARGET에 지정된 인터페이스 이름과 다른 인터페이스 이름을 사용합니다. 참고: 네트워크 대상으로 본드 또는 브리지 장치를 사용하지 마십시오. 본드 장치 위에 VLAN(또는 다른 가상 인터페이스)을 구성하거나 브리지와 가상 이더넷(veth) 쌍을 사용하십시오. 모범 사례: 이 노드에 처음에 관리 네트워크 IP 주소가 없더라도 값을 지정하십시오. 그러면 나중에 호스트에서 노드를 재구성할 필요 없이 관리 네트워크 IP 주소를 추가할 수 있습니다. 예: bond0.1002 ens256	모범 사례

ADMIN_NETWORK_TARGET_TYPE

가치	지정
인터페이스 (이 값이 유일하게 지원되는 값입니다.)	선택 사항

ADMIN_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC

가치	지정
참 또는 거짓 StorageGRID 컨테이너가 관리 네트워크에서 호스트 대상 인터페이스의 MAC 주소를 사용하도록 하려면 해당 키를 "true"로 설정하십시오. 모범 사례: 무차별 모드가 필요한 네트워크에서는 ADMIN_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC 키를 대신 사용하십시오. Linux용 MAC 복제에 대한 자세한 내용은 "MAC 주소 복제에 대한 고려 사항 및 권장 사항"	모범 사례

ADMIN_ROLE

가치	지정
기본 또는 비기본 이 키는 NODE_TYPE이 VM_Admin_Node인 경우에만 필요하며, 다른 노드 유형에는 지정하지 마십시오.	NODE_TYPE이 VM_Admin_Node인 경우 필수 항목입니다. 그렇지 않으면 선택 사항입니다.

블록 장치 키

BLOCK_DEVICE_AUDIT_LOGS

가치	지정
이 노드가 감사 로그의 영구 스토리지에 사용할 블록 장치 특수 파일의 경로와 이름입니다. 예: /dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:0:0 /dev/disk/by-id/wwn-0x600a09800059d6df000060d757b475fd /dev/mapper/sgws-adm1-audit-logs	NODE_TYPE이 VM_Admin_Node인 노드에 필수입니다. 다른 노드 유형에는 지정하지 마십시오.

BLOCK_DEVICE_RANGEDB_nnn

가치	지정
이 노드가 영구 객체 저장소로 사용할 블록 장치 특수 파일의 경로 및 이름입니다. 이 키는 <code>NODE_TYPE = VM_Storage_Node</code>인 노드에만 필요하며, 다른 노드 유형에는 지정하지 마십시오. <code>BLOCK_DEVICE_RANGEDB_000</code>만 필수 항목이며, 나머지는 선택 사항입니다. <code>BLOCK_DEVICE_RANGEDB_000</code>에 지정된 블록 장치는 최소 4TB 이상이어야 하며, 나머지는 그보다 작을 수 있습니다. 공백을 남기지 마십시오. <code>BLOCK_DEVICE_RANGEDB_005</code>를 지정하는 경우 <code>BLOCK_DEVICE_RANGEDB_004</code>도 지정해야 합니다. 참고: 기존 배포 환경과의 호환성을 위해 업그레이드된 노드에서는 두 자리 키가 지원됩니다. 예: <pre>/dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:0:0</pre> <pre>/dev/disk/by-id/wwn-0x600a09800059d6df000060d757b475fd</pre> <pre>/dev/mapper/sgws-sn1-rangedb-000</pre>	필수: <code>BLOCK_DEVICE_RANGEDB_000</code> 선택 사항: <code>BLOCK_DEVICE_RANGEDB_001</code> <code>BLOCK_DEVICE_RANGEDB_002</code> <code>BLOCK_DEVICE_RANGEDB_003</code> <code>BLOCK_DEVICE_RANGEDB_004</code> <code>BLOCK_DEVICE_RANGEDB_005</code> <code>BLOCK_DEVICE_RANGEDB_006</code> <code>BLOCK_DEVICE_RANGEDB_007</code> <code>BLOCK_DEVICE_RANGEDB_008</code> <code>BLOCK_DEVICE_RANGEDB_009</code> <code>BLOCK_DEVICE_RANGEDB_010</code> <code>BLOCK_DEVICE_RANGEDB_011</code> <code>BLOCK_DEVICE_RANGEDB_012</code> <code>BLOCK_DEVICE_RANGEDB_013</code> <code>BLOCK_DEVICE_RANGEDB_014</code> <code>BLOCK_DEVICE_RANGEDB_015</code>

BLOCK_DEVICE_TABLES

가치	지정
이 노드가 데이터베이스 테이블의 영구 스토리지에 사용할 블록 장치 특수 파일의 경로 및 이름입니다. 이 키는 <code>NODE_TYPE</code>이 <code>VM_Admin_Node</code>인 노드에만 필요하며, 다른 노드 유형에는 지정하지 마십시오. 예: <pre>/dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:0:0</pre> <pre>/dev/disk/by-id/wwn-0x600a09800059d6df000060d757b475fd</pre> <pre>/dev/mapper/sgws-adm1-tables</pre>	필수

BLOCK_DEVICE_VAR_LOCAL

가치	지정
이 노드가 <code>/var/local</code> 영구 스토리지에 사용할 블록 디바이스 특수 파일의 경로 및 이름입니다. 예: <pre>/dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:0:0</pre> <pre>/dev/disk/by-id/wwn-0x600a09800059d6df000060d757b475fd</pre> <pre>/dev/mapper/sgws-sn1-var-local</pre>	필수

클라이언트 네트워크 키

CLIENT_NETWORK_CONFIG

가치	지정
DHCP, STATIC 또는 DISABLED	선택 사항

CLIENT_NETWORK_GATEWAY

가치	지정
----	----

이 노드의 로컬 클라이언트 네트워크 게이트웨이의 IPv4 주소입니다. 이 주소는 <code>CLIENT_NETWORK_IP</code> 및 <code>CLIENT_NETWORK_MASK</code>로 정의된 서브넷에 있어야 합니다. DHCP로 구성된 네트워크의 경우 이 값은 무시됩니다. 예: <code>1.1.1.1</code> <code>10.224.4.81</code>	선택 사항
---	-------

CLIENT_NETWORK_IP

가치	지정
클라이언트 네트워크에서 이 노드의 IPv4 주소입니다. 이 키는 <code>CLIENT_NETWORK_CONFIG = STATIC</code>인 경우에만 필요하며, 다른 값에 대해서는 지정하지 마십시오. 예: <code>1.1.1.1</code> <code>10.224.4.81</code>	<code>CLIENT_NETWORK_CONFIG = STATIC</code>인 경우 필수 항목입니다. 그렇지 않으면 선택 사항입니다.

CLIENT_NETWORK_MAC

가치	지정
컨테이너의 클라이언트 네트워크 인터페이스에 대한 MAC 주소입니다. 이 필드는 선택 사항입니다. 생략할 경우 MAC 주소가 자동으로 생성됩니다. 6쌍의 16진수 숫자로 구성되어야 하며, 각 숫자는 콜론으로 구분되어야 합니다. 예: <code>b2:9c:02:c2:27:20</code>	선택 사항

CLIENT_NETWORK_MASK

가치	지정
클라이언트 네트워크에서 이 노드의 IPv4 넷마스크입니다. CLIENT_NETWORK_CONFIG = STATIC인 경우 이 키를 지정하고, 다른 값의 경우에는 지정하지 마십시오. 예: 255.255.255.0 255.255.248.0	CLIENT_NETWORK_IP가 지정되고 CLIENT_NETWORK_CONFIG = STATIC인 경우 필수입니다. 그렇지 않으면 선택 사항입니다.

CLIENT_NETWORK_MTU

가치	지정
클라이언트 네트워크에서 이 노드의 최대 전송 단위(MTU)입니다. CLIENT_NETWORK_CONFIG = DHCP인 경우에는 지정하지 마십시오. 지정하는 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1500이 사용됩니다. 정보 프레임을 사용하려면 MTU 값을 9000과 같이 정보 프레임에 적합한 값으로 설정하십시오. 그렇지 않으면 기본값을 유지하십시오. 중요: 네트워크의 MTU 값은 노드가 연결된 스위치 포트에 구성된 값과 일치해야 합니다. 그렇지 않으면 네트워크 성능 문제 또는 패킷 손실이 발생할 수 있습니다. 예: 1500 8192	선택 사항

CLIENT_NETWORK_TARGET

가치	지정
StorageGRID 노드에서 클라이언트 네트워크에 액세스하는 데 사용할 호스트 장치의 이름입니다. 네트워크 인터페이스 이름만 지원됩니다. 일반적으로 GRID_NETWORK_TARGET 또는 ADMIN_NETWORK_TARGET에 지정된 인터페이스 이름과 다른 인터페이스 이름을 사용합니다. 참고: 네트워크 대상으로 본드 또는 브리지 장치를 사용하지 마십시오. 본드 장치 위에 VLAN(또는 다른 가상 인터페이스)을 구성하거나 브리지와 가상 이더넷(veth) 쌍을 사용하십시오. 권장 사항: 이 노드에 처음에 클라이언트 네트워크 IP 주소가 없더라도 값을 지정하십시오. 그러면 나중에 호스트에서 노드를 재구성할 필요 없이 클라이언트 네트워크 IP 주소를 추가할 수 있습니다. 예: bond0.1003 ens423	모범 사례

CLIENT_NETWORK_TARGET_TYPE

가치	지정
인터페이스 (이 값만 지원됩니다.)	선택 사항

CLIENT_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC

가치	지정
참 또는 거짓 이 키를 "true"로 설정하면 StorageGRID 컨테이너가 클라이언트 네트워크에서 호스트 대상 인터페이스의 MAC 주소를 사용하게 됩니다. 모범 사례: 무차별 모드가 필요한 네트워크에서는 CLIENT_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC 키를 대신 사용하십시오. Linux용 MAC 복제에 대한 자세한 내용은 "MAC 주소 복제에 대한 고려 사항 및 권장 사항"	모범 사례

그리드 네트워크 키

GRID_NETWORK_CONFIG

가치	지정
STATIC 또는 DHCP 별도로 지정하지 않으면 기본값은 STATIC입니다.	모범 사례

GRID_NETWORK_GATEWAY

가치	지정
이 노드의 로컬 그리드 네트워크 게이트웨이의 IPv4 주소입니다. 이 주소는 GRID_NETWORK_IP 및 GRID_NETWORK_MASK로 정의된 서브넷에 있어야 합니다. DHCP로 구성된 네트워크의 경우 이 값은 무시됩니다. 그리드 네트워크가 게이트웨이가 없는 단일 서브넷인 경우 서브넷의 표준 게이트웨이 주소(X.Y.Z.1) 또는 이 노드의 GRID_NETWORK_IP 값을 사용하십시오. 두 값 중 하나를 사용하면 향후 그리드 네트워크 확장이 간소화됩니다.	필수

GRID_NETWORK_IP

가치	지정
그리드 네트워크에서 이 노드의 IPv4 주소입니다. 이 키는 GRID_NETWORK_CONFIG = STATIC인 경우에만 필요하며, 다른 값의 경우에는 지정하지 마십시오. 예: 1.1.1.1 10.224.4.81	GRID_NETWORK_CONFIG = STATIC인 경우 필수입니다. 그렇지 않으면 선택 사항입니다.

GRID_NETWORK_MAC

가치	지정
컨테이너의 그리드 네트워크 인터페이스에 대한 MAC 주소입니다. 6쌍의 16진수 숫자로 구성되어야 하며, 각 숫자는 콜론으로 구분되어야 합니다. 예: b2:9c:02:c2:27:30	선택 사항 생략할 경우 MAC 주소가 자동으로 생성됩니다.

GRID_NETWORK_MASK

가치	지정
그리드 네트워크에서 이 노드에 사용할 IPv4 넷마스크입니다. GRID_NETWORK_CONFIG = STATIC인 경우 이 키를 지정하고, 다른 값의 경우에는 지정하지 마십시오. 예: 255.255.255.0 255.255.248.0	GRID_NETWORK_IP가 지정되고 GRID_NETWORK_CONFIG = STATIC인 경우 필수입니다. 그렇지 않으면 선택 사항입니다.

GRID_NETWORK_MTU

가치	지정
그리드 네트워크에서 이 노드의 최대 전송 단위(MTU)입니다. GRID_NETWORK_CONFIG가 DHCP인 경우에는 지정하지 마십시오. 지정하는 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1500이 사용됩니다. 점보 프레임을 사용하려면 MTU 값을 9000과 같이 점보 프레임에 적합한 값으로 설정하십시오. 그렇지 않으면 기본값을 유지하십시오. 중요: 네트워크의 MTU 값은 노드가 연결된 스위치 포트에 구성된 값과 일치해야 합니다. 그렇지 않으면 네트워크 성능 문제 또는 패킷 손실이 발생할 수 있습니다. 중요: 최상의 네트워크 성능을 위해서는 모든 노드의 그리드 네트워크 인터페이스에 유사한 MTU 값을 구성해야 합니다. 개별 노드의 그리드 네트워크 MTU 설정에 상당한 차이가 있는 경우 그리드 네트워크 MTU 불일치 경고가 발생합니다. 모든 네트워크 유형에 대해 MTU 값이 동일할 필요는 없습니다. 예: 1500 8192	선택 사항

GRID_NETWORK_TARGET

가치	지정
StorageGRID 노드가 그리드 네트워크에 액세스하는 데 사용할 호스트 장치의 이름입니다. 네트워크 인터페이스 이름만 지원됩니다. 일반적으로 ADMIN_NETWORK_TARGET 또는 CLIENT_NETWORK_TARGET에 지정된 인터페이스 이름과 다른 인터페이스 이름을 사용합니다. 참고: 네트워크 대상으로 본드 또는 브리지 장치를 사용하지 마십시오. 본드 장치 위에 VLAN(또는 다른 가상 인터페이스)을 구성하거나 브리지와 가상 이더넷(veth) 쌍을 사용하십시오. 예: bond0.1001 ens192	필수

GRID_NETWORK_TARGET_TYPE

가치	지정
인터페이스 (이 값이 유일하게 지원되는 값입니다.)	선택 사항

GRID_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC

가치	지정
참 또는 거짓 키 값을 "true"로 설정하면 StorageGRID 컨테이너가 그리드 네트워크에서 호스트 대상 인터페이스의 MAC 주소를 사용하게 됩니다. 모범 사례: 무차별 모드가 필요한 네트워크에서는 GRID_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC 키를 대신 사용하십시오. Linux용 MAC 복제에 대한 자세한 내용은 "MAC 주소 복제에 대한 고려 사항 및 권장 사항"	모범 사례

설치 암호 키(임시)

CUSTOM_TEMPORARY_PASSWORD_HASH

가치	지정
기본 관리 노드의 경우, StorageGRID 설치 중에 StorageGRID 설치 API에 사용할 기본 임시 암호를 설정하십시오. 참고: 설치 암호는 기본 관리자 노드에만 설정하십시오. 다른 노드 유형에 암호를 설정하려고 하면 노드 구성 파일 유효성 검사가 실패합니다. 설치가 완료된 후에는 이 값을 설정해도 아무런 효과가 없습니다. 이 키를 생략하면 기본적으로 임시 암호가 설정되지 않습니다. 또는 StorageGRID 설치 API를 사용하여 임시 암호를 설정할 수 있습니다. 최소 8자, 최대 32자 길이의 비밀번호에 대한 <code>crypt()</code> SHA-512 비밀번호 해시 형식이어야 합니다 <code>\$6\$<salt>\$<password hash></code>. 이 해시는 SHA-512 모드의 <code>openssl passwd</code> 명령어와 같은 CLI 도구를 사용하여 생성할 수 있습니다.	모범 사례

인터페이스 키

INTERFACE_TARGET_nnnn

가치	지정
이 노드에 추가할 추가 인터페이스의 이름과 선택적 설명입니다. 각 노드에 여러 개의 추가 인터페이스를 추가할 수 있습니다. _nnnn_의 경우, 추가하는 각 INTERFACE_TARGET 항목에 대해 고유한 번호를 지정하십시오. 값에는 베어메탈 호스트의 물리적 인터페이스 이름을 지정하십시오. 그런 다음 선택적으로 쉼표를 추가하고 VLAN 인터페이스 페이지와 HA 그룹 페이지에 표시되는 인터페이스 설명을 제공할 수 있습니다. 예: <code>INTERFACE_TARGET_0001=ens256, Trunk</code> 트렁크 인터페이스를 추가하는 경우 StorageGRID에서 VLAN 인터페이스를 구성해야 합니다. 액세스 인터페이스를 추가하는 경우에는 인터페이스를 HA 그룹에 직접 추가할 수 있으며, VLAN 인터페이스를 구성할 필요가 없습니다.	선택 사항

최대 RAM 키

MAXIMUM_RAM

가치	지정
이 노드가 사용할 수 있는 최대 RAM 용량입니다. 이 키를 생략하면 노드에 메모리 제한이 없습니다. 운영 환경의 노드에 이 필드를 설정할 때는 최소 24GB, 전체 시스템 RAM보다 16~32GB 적은 값을 지정하십시오. 참고: RAM 값은 노드의 실제 메타데이터 예약 공간에 영향을 줍니다. 자세한 내용은 "메타데이터 예약 공간에 대한 설명"을 참조하십시오. 이 필드의 형식은 <code>numberunit`</code>이며, 여기서 <code>`unit`</code>은 <code>`b</code>, <code>k</code>, <code>m</code>, 또는 <code>`g`</code>가 될 수 있습니다. 예: 24g 38654705664b 참고: 이 옵션을 사용하려면 메모리 cgroups에 대한 커널 지원을 활성화해야 합니다.	선택 사항

노드 유형 키

NODE_TYPE

가치	지정
노드 유형: • VM_Admin_Node • VM_Storage_Node • VM_Archive_Node • VM_API_Gateway	필수

STORAGE_TYPE

가치	지정
스토리지 노드에 포함되는 객체 유형을 정의합니다. 자세한 내용은 "스토리지 노드의 유형"을 참조하십시오. 이 키는 <code>NODE_TYPE = VM_Storage_Node</code>인 노드에만 필요하며, 다른 노드 유형에는 지정하지 마십시오. 스토리지 유형: • 결합된 • 데이터 • 메타데이터 참고: <code>STORAGE_TYPE</code>이 지정되지 않은 경우 스토리지 노드 유형은 기본적으로 결합형(데이터 및 메타데이터)으로 설정됩니다.	선택 사항

포트 재매핑 키



포트 재매핑 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다. 재매핑된 포트를 제거하려면 "[베어메탈 호스트에서 포트 재매핑 제거](#)"을 참조하십시오.

PORT_REMAP

가치	지정
노드가 내부 그리드 노드 통신 또는 외부 통신에 사용하는 모든 포트를 다시 매핑합니다. 기업 네트워킹 정책으로 인해 StorageGRID에서 사용하는 하나 이상의 포트가 제한되는 경우 포트 재매핑이 필요합니다. 자세한 내용은 "내부 그리드 노드 통신" 또는 "외부 통신"을 참조하십시오. 중요: 로드 밸런서 엔드포인트를 구성하는 데 사용할 포트를 다시 매핑하지 마십시오. 참고: <code>PORT_REMAP</code>만 설정된 경우, 지정한 매핑이 수신 및 송신 통신 모두에 사용됩니다. <code>PORT_REMAP_INBOUND</code>도 지정된 경우, <code>PORT_REMAP</code>은 송신 통신에만 적용됩니다. 사용되는 형식은 다음과 같습니다. <code>network type/protocol/default port used by grid node/new port</code>, 여기서 <code>network type</code>는 <code>grid</code>, <code>admin</code> 또는 <code>client</code>이고 <code>protocol</code>는 <code>tcp</code> 또는 <code>udp</code>입니다. 예: <code>PORT_REMAP = client/tcp/18082/443</code> 쉼표로 구분된 목록을 사용하여 여러 포트를 재매핑할 수도 있습니다. 예: <code>PORT_REMAP = client/tcp/18082/443, client/tcp/18083/80</code>	선택 사항

PORT_REMAP_INBOUND

가치	지정
수신 통신을 지정된 포트로 다시 매핑합니다. <code>PORT_REMAP_INBOUND</code>를 지정했지만 <code>PORT_REMAP</code> 값을 지정하지 않으면 해당 포트의 송신 통신은 변경되지 않습니다. 중요: 로드 밸런서 엔드포인트를 구성하는 데 사용할 포트를 다시 매핑하지 마십시오. 사용되는 형식은 다음과 같습니다. <code>network type/protocol /remapped port/default port used by grid node</code>, 여기서 <code>'network type'</code>는 <code>grid</code>, <code>admin</code> 또는 <code>client</code>이고 <code>'protocol'</code>는 <code>tcp</code> 또는 <code>udp</code>입니다. 예: <code>PORT_REMAP_INBOUND = grid/tcp/3022/22</code> 쉼표로 구분된 목록을 사용하여 여러 개의 수신 포트를 다시 매핑할 수도 있습니다. 예: <code>PORT_REMAP_INBOUND = grid/tcp/3022/22, admin/tcp/3022/22</code>	선택 사항

그리드 노드가 **StorageGRID**에서 기본 관리 노드를 검색하는 방법

그리드 노드는 구성 및 관리를 위해 기본 관리 노드와 통신합니다. 각 그리드 노드는 그리드 네트워크에 있는 기본 관리 노드의 IP 주소를 알고 있어야 합니다.

그리드 노드가 기본 관리 노드에 액세스할 수 있도록 하려면 노드를 배포할 때 다음 중 하나를 수행할 수 있습니다.

- `ADMIN_IP` 매개변수를 사용하여 기본 관리 노드의 IP 주소를 수동으로 입력할 수 있습니다.
- `ADMIN_IP` 매개변수를 생략하면 그리드 노드가 자동으로 값을 검색합니다. 자동 검색은 그리드 네트워크에서 DHCP를 사용하여 기본 관리 노드에 IP 주소를 할당하는 경우 특히 유용합니다.

기본 관리 노드의 자동 검색은 멀티캐스트 도메인 이름 시스템(mDNS)을 사용하여 수행됩니다. 기본 관리 노드가 처음 시작될 때 mDNS를 사용하여 자신의 IP 주소를 게시합니다. 그러면 동일한 서브넷에 있는 다른 노드들이 해당 IP 주소를 쿼리하여 자동으로 가져올 수 있습니다. 그러나 멀티캐스트 IP 트래픽은 일반적으로 서브넷 간에 라우팅되지 않으므로 다른 서브넷에 있는 노드는 기본 관리 노드의 IP 주소를 직접 가져올 수 없습니다.

자동 검색을 사용하는 경우:



- 기본 관리 노드가 직접 연결되지 않은 서브넷의 경우, 최소 하나 이상의 그리드 노드에 대해 `ADMIN_IP` 설정을 포함해야 합니다. 그러면 해당 그리드 노드가 기본 관리 노드의 IP 주소를 게시하여 서브넷의 다른 노드들이 mDNS를 통해 검색할 수 있도록 합니다.
- 네트워크 인프라가 서브넷 내에서 멀티캐스트 IP 트래픽 전송을 지원하는지 확인하십시오.

VMware vSphere를 사용하여 **StorageGRID**용 그리드 노드를 가상 머신으로 배포합니다

VMware vSphere 웹 클라이언트를 사용하여 각 그리드 노드를 가상 머신으로 배포합니다. 배포 과정에서 각 그리드 노드가 생성되고 하나 이상의 StorageGRID 네트워크에 연결됩니다.

StorageGRID 어플라이언스 스토리지 노드를 배포해야 하는 경우 다음을 참조하십시오 ["어플라이언스 스토리지 노드 배포"](#).

필요에 따라 노드 전원을 켜기 전에 노드 포트를 다시 매핑하거나 노드의 CPU 또는 메모리 설정을 높일 수 있습니다.

시작하기 전에

- 이미 ["설치 계획 및 준비"](#) 방법을 검토하셨고, 소프트웨어, CPU 및 RAM, 스토리지 및 성능에 대한 요구 사항을 이해하고 계십니다.
- 귀하는 VMware vSphere Hypervisor에 익숙하고 해당 환경에 가상 머신을 배포한 경험이 있습니다.



The open-vm-tools 패키지는 VMware Tools와 유사한 오픈 소스 구현체로, StorageGRID 가상 머신에 포함되어 있습니다. VMware Tools를 수동으로 설치할 필요가 없습니다.

- VMware용 StorageGRID 설치 아카이브의 올바른 버전을 다운로드하고 압축을 해제했습니다.



확장 또는 복구 작업의 일환으로 새 노드를 배포하는 경우, 현재 그리드에서 실행 중인 StorageGRID 버전을 사용해야 합니다.

- StorageGRID 가상 머신 디스크 (.vmdk) 파일이 있습니다.

```
NetApp-SG-version-SHA.vmdk
```

- 배포하려는 각 그리드 노드 유형에 대한 .ovf 및 .mf 파일이 있습니다.

파일 이름	설명
vsphere-primary-admin.ovf vsphere-primary-admin.mf	기본 관리자 노드에 대한 템플릿 파일 및 매니페스트 파일입니다.
vsphere-non-primary-admin.ovf vsphere-non-primary-admin.mf	기본 관리자 노드가 아닌 관리자 노드에 대한 템플릿 파일 및 매니페스트 파일입니다.
vsphere-storage.ovf vsphere-storage.mf	스토리지 노드용 템플릿 파일 및 매니페스트 파일입니다.
vsphere-gateway.ovf vsphere-gateway.mf	게이트웨이 노드용 템플릿 파일 및 매니페스트 파일입니다.

- .vmdk, .ovf 및 .mf 파일은 모두 같은 디렉토리에 있습니다.
- 장애 발생 영역을 최소화하기 위한 계획이 있습니다. 예를 들어, 모든 게이트웨이 노드를 단일 vSphere ESXi 호스트에 배포해서는 안 됩니다.



운영 구축 시 단일 가상 머신에서 스토리지 노드를 두 개 이상 실행하지 마십시오. 또한, 장애 도메인에 허용할 수 없는 문제가 발생할 수 있는 경우 동일한 ESXi 호스트에서 여러 가상 머신을 실행하지 마십시오.

- 확장 또는 복구 작업의 일부로 노드를 배포하는 경우 ["StorageGRID 시스템 확장 지침"](#) 또는 ["복구 및 유지 관리"](#)

지침"이(가) 있습니다.

- StorageGRID 노드를 NetApp ONTAP 시스템에서 스토리지를 할당받아 가상 머신으로 배포하는 경우, 해당 볼륨에 FabricPool 티어링 정책이 활성화되어 있지 않은지 확인해야 합니다. 예를 들어, StorageGRID 노드가 VMware 호스트에서 가상 머신으로 실행되는 경우, 해당 노드의 데이터스토어를 지원하는 볼륨에 FabricPool 티어링 정책이 활성화되어 있지 않은지 확인하십시오. StorageGRID 노드와 함께 사용되는 볼륨에 대해 FabricPool 티어링을 비활성화하면 문제 해결 및 스토리지 운영이 간소화됩니다.



FabricPool를 사용하여 StorageGRID와 관련된 데이터를 StorageGRID 자체로 계층화하지 마십시오. StorageGRID 데이터를 StorageGRID로 다시 계층화하면 문제 해결 및 운영 복잡성이 증가합니다.

이 작업 정보

VMware 노드를 처음 배포하거나, 확장 시 새 VMware 노드를 추가하거나, 복구 작업의 일환으로 VMware 노드를 교체하려면 다음 지침을 따르십시오. 단계에 명시된 경우를 제외하고, 관리 노드, 스토리지 노드, 게이트웨이 노드를 포함한 모든 노드 유형에 대한 노드 배포 절차는 동일합니다.

새로운 StorageGRID 시스템을 설치하는 경우:

- 노드는 어떤 순서로든 배포할 수 있습니다.
- 각 가상 머신이 그리드 네트워크를 통해 기본 관리 노드에 연결할 수 있는지 확인해야 합니다.
- 그리드를 구성하기 전에 모든 그리드 노드를 배포해야 합니다.

확장 또는 복구 작업을 수행하는 경우:

- 새 가상 머신이 그리드 네트워크의 다른 모든 노드에 연결될 수 있는지 확인해야 합니다.

노드의 포트를 다시 매핑해야 하는 경우, 포트 재매핑 구성이 완료될 때까지 새 노드의 전원을 켜지 마십시오.



포트 재매핑 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다. 재매핑된 포트를 제거하려면 "[베어메탈 호스트에서 포트 재매핑 제거](#)"를 참조하십시오.

단계

1. VCenter를 사용하여 OVF 템플릿을 배포합니다.

URL을 지정하는 경우 다음 파일들이 포함된 폴더를 가리키도록 하십시오. 그렇지 않으면 로컬 디렉터리에서 각 파일을 선택하십시오.

```
NetApp-SG-version-SHA.vmdk  
vsphere-node.ovf  
vsphere-node.mf
```

예를 들어, 처음으로 노드를 배포하는 경우 다음 파일을 사용하여 StorageGRID 시스템의 기본 관리 노드를 배포하십시오.

```
NetApp-SG-version-SHA.vmdk
vsphere-primary-admin.ovf
vsphere-primary-admin.mf
```

2. 가상 머신의 이름을 지정하십시오.

일반적으로 가상 머신과 그리드 노드에 동일한 이름을 사용합니다.

3. 가상 머신을 적절한 vApp 또는 리소스 풀에 배치합니다.

4. 기본 관리 노드를 배포하는 경우 최종 사용자 라이선스 계약을 읽고 동의하십시오.

vCenter 버전에 따라 최종 사용자 라이선스 계약 동의, 가상 머신 이름 지정 및 데이터 저장소 선택 단계의 순서가 다를 수 있습니다.

5. 가상 머신에 사용할 스토리지를 선택합니다.

복구 작업의 일환으로 노드를 배포하는 경우, [스토리지 복구 단계](#)의 지침에 따라 새 가상 디스크를 추가하거나, 장애가 발생한 그리드 노드에서 가상 하드 디스크를 다시 연결하거나, 또는 둘 다를 수행하십시오.

스토리지 노드를 배포할 때는 3개 이상의 스토리지 볼륨을 사용하고, 각 볼륨의 용량은 4TB 이상이어야 합니다. 특히 볼륨 0에는 최소 4TB를 할당해야 합니다.



스토리지 노드의 .ovf 파일에는 스토리지용 VMDK 파일이 여러 개 정의되어 있습니다. 이러한 VMDK 파일이 스토리지 요구 사항을 충족하지 못하는 경우, 노드 전원을 켜기 전에 해당 VMDK 파일을 제거하고 적절한 VMDK 파일 또는 RDM 파일을 스토리지에 할당해야 합니다. VMDK 파일은 VMware 환경에서 더 일반적으로 사용되며 관리가 더 쉽습니다. 반면 RDM 파일은 객체 크기가 큰 워크로드(예: 100MB 이상)에 더 나은 성능을 제공할 수 있습니다.



일부 StorageGRID 설치 환경에서는 일반적인 가상화 워크로드보다 더 크고 활성도가 높은 스토리지 볼륨을 사용할 수 있습니다. 최적의 성능을 얻으려면 하이퍼바이저 매개변수(예: MaxAddressableSpaceTB)를 조정해야 할 수 있습니다. 성능 저하가 발생하는 경우 가상화 지원 담당자에게 문의하여 워크로드별 구성 조정이 환경에 도움이 될 수 있는지 확인하십시오.

6. 네트워크를 선택합니다.

각 소스 네트워크에 대한 대상 네트워크를 선택하여 노드가 사용할 StorageGRID 네트워크를 결정합니다.

- 그리드 네트워크가 필요합니다. vSphere 환경에서 대상 네트워크를 선택해야 합니다. + 그리드 네트워크는 모든 내부 StorageGRID 트래픽에 사용됩니다. 그리드의 모든 노드 간, 모든 사이트 및 서브넷 간 연결을 제공합니다. 그리드 네트워크의 모든 노드는 다른 모든 노드와 통신할 수 있어야 합니다.
- 관리자 네트워크를 사용하는 경우 vSphere 환경에서 다른 대상 네트워크를 선택하십시오. 관리자 네트워크를 사용하지 않는 경우 그리드 네트워크에 대해 선택한 것과 동일한 대상 네트워크를 선택하십시오.
- 클라이언트 네트워크를 사용하는 경우 vSphere 환경에서 다른 대상 네트워크를 선택하십시오. 클라이언트 네트워크를 사용하지 않는 경우 그리드 네트워크에 대해 선택한 것과 동일한 대상 네트워크를 선택하십시오.
- 관리자 네트워크 또는 클라이언트 네트워크를 사용하는 경우 노드는 동일한 관리자 네트워크 또는 클라이언트 네트워크에 있을 필요가 없습니다.

7. *템플릿 사용자 지정*의 경우, 필요한 StorageGRID 노드 속성을 구성하십시오.

a. *노드 이름*을 입력합니다.



그리드 노드를 복구하는 경우 복구할 노드의 이름을 입력해야 합니다.

b. 새 노드가 그리드에 참여하기 전에 VM 콘솔이나 StorageGRID 설치 API에 액세스하거나 SSH를 사용할 수 있도록 임시 설치 암호 드롭다운을 사용하여 임시 설치 암호를 지정하십시오.



임시 설치 비밀번호는 노드 설치 중에만 사용됩니다. 노드가 그리드에 추가된 후에는 Recovery Package의 Passwords.txt 파일에 나열된 "노드 콘솔 비밀번호"을(를) 사용하여 해당 노드에 액세스할 수 있습니다.

- 노드 이름 사용: 노드 이름 필드에 입력한 값이 임시 설치 암호로 사용됩니다.
- 사용자 지정 암호 사용: 사용자 지정 암호가 임시 설치 암호로 사용됩니다.
- 암호 비활성화: 임시 설치 암호는 사용되지 않습니다. 설치 문제를 디버깅하기 위해 VM에 액세스해야 하는 경우 "설치 문제 해결"을 참조하십시오.

c. 사용자 지정 암호 사용*을 선택한 경우, *사용자 지정 암호 필드에 사용할 임시 설치 암호를 지정하십시오.

d. 그리드 네트워크(eth0) 섹션에서 *그리드 네트워크 IP 구성*에 대해 STATIC 또는 DHCP를 선택하십시오.

- STATIC을 선택하는 경우 **Grid network IP**, **Grid network mask**, **Grid network gateway** 및 *Grid network MTU*를 입력하십시오.
- DHCP를 선택하면 그리드 네트워크 IP, 그리드 네트워크 마스크, *그리드 네트워크 게이트웨이*가 자동으로 할당됩니다.

e. 기본 관리자 IP 필드에 그리드 네트워크의 기본 관리 노드의 IP 주소를 입력하십시오.



배포하려는 노드가 기본 관리자 노드인 경우에는 이 단계가 적용되지 않습니다.

기본 관리 노드 IP 주소를 생략하면, 기본 관리 노드 또는 ADMIN_IP가 구성된 다른 그리드 노드가 동일한 서브넷에 있는 경우 IP 주소가 자동으로 검색됩니다. 하지만 여기에서 기본 관리 노드 IP 주소를 설정하는 것이 좋습니다.

a. 관리자 네트워크(eth1) 섹션에서 *관리자 네트워크 IP 구성*에 대해 STATIC, DHCP 또는 DISABLED를 선택합니다.

- 관리자 네트워크를 사용하지 않으려면 DISABLED를 선택하고 관리자 네트워크 IP 주소에 *0.0.0.0*을 입력하십시오. 나머지 필드는 비워 두셔도 됩니다.
- 정적(STATIC)을 선택하는 경우 관리자 네트워크 IP, 관리자 네트워크 마스크, 관리자 네트워크 게이트웨이, *관리자 네트워크 MTU*를 입력하십시오.
- 정적(STATIC)을 선택하는 경우 *관리 네트워크 외부 서브넷 목록*을 입력하십시오. 게이트웨이가 구성해야 합니다.
- DHCP를 선택하면 관리자 네트워크 IP, 관리자 네트워크 마스크, *관리자 네트워크 게이트웨이*가 자동으로 할당됩니다.

b. 클라이언트 네트워크(eth2) 섹션에서 *클라이언트 네트워크 IP 구성*에 대해 STATIC, DHCP 또는 DISABLED를 선택합니다.

- 클라이언트 네트워크를 사용하지 않으려면 DISABLED를 선택하고 클라이언트 네트워크 IP 주소에 *0.0.0.0*을 입력하십시오. 나머지 필드는 비워 두셔도 됩니다.
- 정적(STATIC)을 선택하는 경우 클라이언트 네트워크 IP, 클라이언트 네트워크 마스크, 클라이언트 네트워크 게이트웨이, *클라이언트 네트워크 MTU*를 입력하십시오.

- DHCP를 선택하면 클라이언트 네트워크 IP, 클라이언트 네트워크 마스크, *클라이언트 네트워크 게이트웨이*가 자동으로 할당됩니다.

- 가상 머신 구성을 검토하고 필요한 변경 사항을 적용하십시오.
- 준비가 되면 *완료*를 선택하여 가상 머신 업로드를 시작하십시오.
- 복구 작업의 일부로 이 노드를 배포했고 전체 노드 복구가 아닌 경우, 배포가 완료된 후 다음 단계를 수행하십시오.
 - 가상 머신을 마우스 오른쪽 버튼으로 클릭하고 *설정 편집*을 선택합니다.
 - 스토리지로 지정된 기본 가상 하드 디스크를 각각 선택하고 *제거*를 선택합니다.
 - 데이터 복구 상황에 따라 스토리지 요구 사항에 맞춰 새 가상 디스크를 추가하거나, 이전에 제거된 장애 그리드 노드에서 보존된 가상 하드 디스크를 다시 연결하거나, 또는 둘 다를 수행할 수 있습니다.

다음의 중요한 지침을 참고하십시오:

- 새 디스크를 추가하는 경우 노드 복구 이전에 사용했던 것과 동일한 유형의 스토리지 장치를 사용해야 합니다.
- 스토리지 노드의 .ovf 파일에는 스토리지용 VMDK 파일이 여러 개 정의되어 있습니다. 이러한 VMDK 파일이 스토리지 요구 사항을 충족하지 못하는 경우, 노드 전원을 켜기 전에 해당 VMDK 파일을 제거하고 적절한 VMDK 파일 또는 RDM 파일을 스토리지에 할당해야 합니다. VMDK 파일은 VMware 환경에서 더 일반적으로 사용되며 관리가 더 쉽습니다. 반면 RDM 파일은 객체 크기가 큰 워크로드(예: 100MB 이상)에 더 나은 성능을 제공할 수 있습니다.

- 이 노드에서 사용하는 포트를 다시 매핑해야 하는 경우 다음 단계를 따르십시오.

엔터프라이즈 네트워킹 정책이 StorageGRID에서 사용하는 하나 이상의 포트에 대한 액세스를 제한하는 경우 포트를 다시 매핑해야 할 수 있습니다. StorageGRID에서 사용하는 포트는 "[네트워킹 가이드라인](#)"을 참조하십시오.



로드 밸런싱 엔드포인트에서 사용하는 포트를 재매핑하지 마십시오.

- 새 VM을 선택합니다.
- 구성 탭에서 설정 > *vApp Options*를 선택합니다. *vApp Options*의 위치는 vCenter 버전에 따라 다릅니다.
- Properties** 테이블에서 PORT_REMAP_INBOUND와 PORT_REMAP을 찾으십시오.
- 포트에 대한 수신 및 송신 통신을 대칭적으로 매핑하려면 *PORT_REMAP*을 선택하십시오.



포트 재매핑 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다. 재매핑된 포트를 제거하려면 "[베어메탈 호스트에서 포트 재매핑 제거](#)"을 참조하십시오.



PORT_REMAP만 설정된 경우, 지정한 매핑이 수신 및 송신 통신 모두에 적용됩니다. PORT_REMAP_INBOUND도 지정된 경우, PORT_REMAP은 송신 통신에만 적용됩니다.

- *값 설정*을 선택합니다.
- 포트 매핑을 입력하십시오.

```
<network type>/<protocol>/<default port used by grid node>/<new port>
```

`<network type>`는 grid, admin 또는 client이고, `<protocol>`는 tcp 또는 udp입니다.

예를 들어, SSH 트래픽을 포트 22에서 포트 3022로 재매핑하려면 다음을 입력합니다.

client/tcp/22/3022

심표로 구분된 목록을 사용하여 여러 포트를 다시 매핑할 수 있습니다.

예를 들면 다음과 같습니다.

client/tcp/18082/443, client/tcp/18083/80

i. *확인*을 선택합니다.

e. 노드로 들어오는 통신에 사용할 포트를 지정하려면 *PORT_REMAP_INBOUND*를 선택하십시오.



PORT_REMAP_INBOUND를 지정하고 PORT_REMAP 값을 지정하지 않으면 해당 포트의 아웃바운드 통신은 변경되지 않습니다.

i. *값 설정*을 선택합니다.

ii. 포트 매핑을 입력하십시오.

<network type>/<protocol>/<remapped inbound port>/<default inbound port used by grid node>

`<network type>`는 grid, admin 또는 client이고, `<protocol>`는 tcp 또는 udp입니다.

예를 들어, 3022번 포트에 전송되는 수신 SSH 트래픽을 그리드 노드의 22번 포트에 수신되도록 재매핑하려면 다음을 입력합니다.

client/tcp/3022/22

심표로 구분된 목록을 사용하여 여러 개의 수신 포트를 다시 매핑할 수 있습니다.

예를 들면 다음과 같습니다.

grid/tcp/3022/22, admin/tcp/3022/22

i. *확인*을 선택합니다

12. 노드의 CPU 또는 메모리를 기본 설정보다 높이려면 다음을 수행합니다.

a. 가상 머신을 마우스 오른쪽 버튼으로 클릭하고 *설정 편집*을 선택합니다.

b. 필요에 따라 CPU 개수 또는 메모리 용량을 변경하십시오.

메모리 예약*을 가상 머신에 할당된 *메모리 크기*와 동일하게 설정하십시오.

c. *확인*을 선택합니다.

13. 가상 머신의 전원을 켭니다.

완료한 후

확장 또는 복구 절차의 일환으로 이 노드를 배포한 경우 해당 지침으로 돌아가 절차를 완료하십시오.

Linux의 StorageGRID 노드에 대한 구성 파일 예

예제 노드 구성 파일을 사용하여 StorageGRID 시스템의 노드 구성 파일을 설정할 수 있습니다. 예제에는 모든 유형의 그리드 노드에 대한 노드 구성 파일이 나와 있습니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

대부분의 노드에서는 그리드 관리자 또는 설치 API를 사용하여 그리드를 구성할 때 관리 및 클라이언트 네트워크 주소 정보(IP, 서브넷 마스크, 게이트웨이 등)를 추가할 수 있습니다. 단, 기본 관리 노드는 예외입니다. 그리드 구성을 완료하기 위해 기본 관리 노드의 관리 네트워크 IP에 접속해야 하는 경우(예: 그리드 네트워크가 라우팅되지 않는 경우), 해당 노드의 노드 구성 파일에서 기본 관리 노드의 관리 네트워크 연결을 구성해야 합니다. 이는 예시에서 확인할 수 있습니다.



예시에서는 클라이언트 네트워크가 기본적으로 비활성화되어 있음에도 불구하고 모범 사례로 클라이언트 네트워크 대상을 구성했습니다.

기본 관리자 노드의 예시

파일 이름 예시: /etc/storagegrid/nodes/dc1-adm1.conf

파일 내용 예시:

```
NODE_TYPE = VM_Admin_Node
ADMIN_ROLE = Primary
TEMPORARY_PASSWORD_TYPE = Use custom password
CUSTOM_TEMPORARY_PASSWORD = Passw0rd
BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/dc1-adm1-var-local
BLOCK_DEVICE_AUDIT_LOGS = /dev/mapper/dc1-adm1-audit-logs
BLOCK_DEVICE_TABLES = /dev/mapper/dc1-adm1-tables
GRID_NETWORK_TARGET = bond0.1001
ADMIN_NETWORK_TARGET = bond0.1002
CLIENT_NETWORK_TARGET = bond0.1003

GRID_NETWORK_IP = 10.1.0.2
GRID_NETWORK_MASK = 255.255.255.0
GRID_NETWORK_GATEWAY = 10.1.0.1

ADMIN_NETWORK_CONFIG = STATIC
ADMIN_NETWORK_IP = 192.168.100.2
ADMIN_NETWORK_MASK = 255.255.248.0
ADMIN_NETWORK_GATEWAY = 192.168.100.1
ADMIN_NETWORK_ESL = 192.168.100.0/21,172.16.0.0/21,172.17.0.0/21
```

스토리지 노드 예시

파일 이름 예시: /etc/storagegrid/nodes/dc1-sn1.conf

파일 내용 예시:

```
NODE_TYPE = VM_Storage_Node
ADMIN_IP = 10.1.0.2
BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/dc1-sn1-var-local
BLOCK_DEVICE_RANGEDB_00 = /dev/mapper/dc1-sn1-rangedb-0
BLOCK_DEVICE_RANGEDB_01 = /dev/mapper/dc1-sn1-rangedb-1
BLOCK_DEVICE_RANGEDB_02 = /dev/mapper/dc1-sn1-rangedb-2
BLOCK_DEVICE_RANGEDB_03 = /dev/mapper/dc1-sn1-rangedb-3
GRID_NETWORK_TARGET = bond0.1001
ADMIN_NETWORK_TARGET = bond0.1002
CLIENT_NETWORK_TARGET = bond0.1003

GRID_NETWORK_IP = 10.1.0.3
GRID_NETWORK_MASK = 255.255.255.0
GRID_NETWORK_GATEWAY = 10.1.0.1
```

게이트웨이 노드 예시

파일 이름 예시: /etc/storagegrid/nodes/dc1-gw1.conf

파일 내용 예시:

```
NODE_TYPE = VM_API_Gateway
ADMIN_IP = 10.1.0.2
BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/dc1-gw1-var-local
GRID_NETWORK_TARGET = bond0.1001
ADMIN_NETWORK_TARGET = bond0.1002
CLIENT_NETWORK_TARGET = bond0.1003
GRID_NETWORK_IP = 10.1.0.5
GRID_NETWORK_MASK = 255.255.255.0
GRID_NETWORK_GATEWAY = 10.1.0.1
```

기본 관리 노드가 아닌 노드의 예시

파일 이름 예시: /etc/storagegrid/nodes/dc1-adm2.conf

파일 내용 예시:

```

NODE_TYPE = VM_Admin_Node
ADMIN_ROLE = Non-Primary
ADMIN_IP = 10.1.0.2
BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/dc1-adm2-var-local
BLOCK_DEVICE_AUDIT_LOGS = /dev/mapper/dc1-adm2-audit-logs
BLOCK_DEVICE_TABLES = /dev/mapper/dc1-adm2-tables
GRID_NETWORK_TARGET = bond0.1001
ADMIN_NETWORK_TARGET = bond0.1002
CLIENT_NETWORK_TARGET = bond0.1003

GRID_NETWORK_IP = 10.1.0.6
GRID_NETWORK_MASK = 255.255.255.0
GRID_NETWORK_GATEWAY = 10.1.0.1

```

Linux에서 StorageGRID 구성 검증

각 StorageGRID 노드에 대한 구성 파일을 /etc/storagegrid/nodes 생성한 후에는 해당 파일의 내용을 검증해야 합니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

구성 파일의 내용을 검증하려면 각 호스트에서 다음 명령을 실행하십시오.

```
sudo storagegrid node validate all
```

파일이 올바르면 예시와 같이 각 구성 파일에 대해 *PASSED*가 출력됩니다.



메타데이터 전용 노드에서 LUN을 하나만 사용하는 경우 무시해도 되는 경고 메시지가 표시될 수 있습니다.

```

Checking for misnamed node configuration files... PASSED
Checking configuration file for node dc1-adm1... PASSED
Checking configuration file for node dc1-gw1... PASSED
Checking configuration file for node dc1-sn1... PASSED
Checking configuration file for node dc1-sn2... PASSED
Checking configuration file for node dc1-sn3... PASSED
Checking for duplication of unique values between nodes... PASSED

```



자동 설치의 경우, storagegrid 명령에서 -q 또는 --quiet 옵션을 사용하여 이 출력을 표시하지 않을 수 있습니다(예: storagegrid --quiet...). 출력을 표시하지 않는 경우, 구성 경고 또는 오류가 감지되면 명령은 0이 아닌 종료 값을 갖게 됩니다.

구성 파일이 올바르지 않으면 예시와 같이 **WARNING** 및 ***ERROR***로 표시됩니다. 구성 오류가 발견되면 설치를 계속하기 전에 수정해야 합니다.

```
Checking for misnamed node configuration files...
WARNING: ignoring /etc/storagegrid/nodes/dcl-adml
WARNING: ignoring /etc/storagegrid/nodes/dcl-sn2.conf.keep
WARNING: ignoring /etc/storagegrid/nodes/my-file.txt
Checking configuration file for node dcl-adml...
ERROR: NODE_TYPE = VM_Foo_Node
      VM_Foo_Node is not a valid node type.  See *.conf.sample
ERROR: ADMIN_ROLE = Foo
      Foo is not a valid admin role.  See *.conf.sample
ERROR: BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/sgws-gw1-var-local
      /dev/mapper/sgws-gw1-var-local is not a valid block device
Checking configuration file for node dcl-gw1...
ERROR: GRID_NETWORK_TARGET = bond0.1001
      bond0.1001 is not a valid interface.  See `ip link show`
ERROR: GRID_NETWORK_IP = 10.1.3
      10.1.3 is not a valid IPv4 address
ERROR: GRID_NETWORK_MASK = 255.248.255.0
      255.248.255.0 is not a valid IPv4 subnet mask
Checking configuration file for node dcl-sn1...
ERROR: GRID_NETWORK_GATEWAY = 10.2.0.1
      10.2.0.1 is not on the local subnet
ERROR: ADMIN_NETWORK_ESL = 192.168.100.0/21,172.16.0foo
      Could not parse subnet list
Checking configuration file for node dcl-sn2... PASSED
Checking configuration file for node dcl-sn3... PASSED
Checking for duplication of unique values between nodes...
ERROR: GRID_NETWORK_IP = 10.1.0.4
      dcl-sn2 and dcl-sn3 have the same GRID_NETWORK_IP
ERROR: BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/sgws-sn2-var-local
      dcl-sn2 and dcl-sn3 have the same BLOCK_DEVICE_VAR_LOCAL
ERROR: BLOCK_DEVICE_RANGEDB_00 = /dev/mapper/sgws-sn2-rangedb-0
      dcl-sn2 and dcl-sn3 have the same BLOCK_DEVICE_RANGEDB_00
```

Linux에서 StorageGRID 호스트 서비스를 시작합니다

StorageGRID 노드를 시작하고 호스트 재부팅 후에도 노드가 다시 시작되도록 하려면 StorageGRID 호스트 서비스를 활성화하고 시작해야 합니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

1. 각 호스트에서 다음 명령을 실행하십시오.

```
sudo systemctl enable storagegrid
sudo systemctl start storagegrid
```

2. 배포가 제대로 진행되고 있는지 확인하려면 다음 명령을 실행하십시오.

```
sudo storagegrid node status node-name
```

3. 어떤 노드라도 상태가 "Not Running" 또는 "Stopped"로 표시되면 다음 명령을 실행하십시오.

```
sudo storagegrid node start node-name
```

4. 이전에 StorageGRID 호스트 서비스를 활성화하고 시작한 경우(또는 서비스가 활성화되고 시작되었는지 확실하지 않은 경우) 다음 명령도 실행하십시오.

```
sudo systemctl reload-or-restart storagegrid
```

StorageGRID 설치 REST API에 대해 알아보세요

StorageGRID는 설치 작업을 수행하기 위한 StorageGRID 설치 API를 제공합니다.

이 API는 오픈 소스 API 플랫폼인 Swagger를 사용하여 API 문서를 제공합니다. Swagger를 통해 개발자와 비개발자 모두 API가 매개변수와 옵션에 어떻게 응답하는지 보여주는 사용자 인터페이스를 통해 API와 상호 작용할 수 있습니다. 이 문서는 사용자가 표준 웹 기술과 JSON 데이터 형식에 익숙하다고 가정합니다.



API 문서 웹페이지를 사용하여 수행하는 모든 API 작업은 실시간 작업입니다. 실수로 구성 데이터 또는 기타 데이터를 생성, 업데이트 또는 삭제하지 않도록 주의하십시오.

각 REST API 명령에는 API URL, HTTP 작업, 필수 또는 선택적 URL 매개변수, 그리고 예상되는 API 응답이 포함됩니다.

StorageGRID 설치 API

StorageGRID 설치 API는 StorageGRID 시스템을 처음 구성할 때와 기본 관리 노드 복구가 필요한 경우에만 사용할 수 있습니다. 설치 API는 Grid Manager에서 HTTPS를 통해 액세스할 수 있습니다.

API 문서를 보려면 기본 관리 노드의 설치 웹 페이지로 이동하여 메뉴 모음에서 도움말 > *API 문서*를 선택하십시오.

StorageGRID 설치 API는 다음 섹션으로 구성됩니다.

- **config** — 제품 릴리스 및 API 버전과 관련된 작업입니다. 제품 릴리스 버전과 해당 릴리스에서 지원하는 API의 주요 버전을 나열할 수 있습니다.

- **그리드** — 그리드 수준 구성 작업입니다. 그리드 세부 정보, 그리드 네트워크 서브넷, 그리드 암호, NTP 및 DNS 서버 IP 주소를 포함한 그리드 설정을 가져오고 업데이트할 수 있습니다.
- **노드** — 노드 수준 구성 작업입니다. 그리드 노드 목록을 가져오거나, 그리드 노드를 삭제하거나, 그리드 노드를 구성하거나, 그리드 노드를 보거나, 그리드 노드의 구성을 재설정할 수 있습니다.
- **provision** — 프로비저닝 작업. 프로비저닝 작업을 시작하고 프로비저닝 작업의 상태를 확인할 수 있습니다.
- **복구** — 기본 관리 노드 복구 작업입니다. 정보를 재설정하고, 복구 패키지를 업로드하고, 복구를 시작하고, 복구 작업 상태를 확인할 수 있습니다.
- **recovery-package** — 복구 패키지를 다운로드하는 작업입니다.
- **sites** — 사이트 수준 구성 작업입니다. 사이트를 생성, 조회, 삭제 및 수정할 수 있습니다.
- **temporary-password** — 설치 중 mgmt-api를 보호하기 위한 임시 암호에 대한 작업입니다.

StorageGRID 설치 문제 해결

StorageGRID 시스템 설치 중 문제가 발생하면 설치 로그 파일을 통해 확인할 수 있습니다. 기술 지원팀에서도 문제 해결을 위해 설치 로그 파일을 사용할 수 있습니다.



"Linux"는 RHEL, Ubuntu 또는 Debian 배포판을 의미합니다. 지원되는 버전 목록은 ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#)을 참조하십시오.

Linux

각 노드를 실행 중인 컨테이너에서 다음 설치 로그 파일을 확인할 수 있습니다.

- /var/local/log/install.log (모든 그리드 노드에서 발견됨)
- /var/local/log/gdu-server.log (기본 Admin Node에서 찾을 수 있음)

`${post_edited_translations.segment}`

- /var/log/storagegrid/daemon.log
- /var/log/storagegrid/nodes/node-name.log

로그 파일에 액세스하는 방법을 알아보려면 ["로그 파일 및 시스템 데이터 수집"](#)을(를) 참조하십시오.

VMware

다음은 기술 지원에서 문제를 해결하는 데 필요할 수 있는 주요 설치 로그 파일입니다.

- /var/local/log/install.log (모든 그리드 노드에서 발견됨)
- /var/local/log/gdu-server.log (기본 Admin Node에서 찾을 수 있음)

`${post_edited_translations.segment}`

OVF 파일에는 각 그리드 노드가 효율적으로 작동하는 데 필요한 충분한 RAM과 CPU를 확보하기 위한 리소스 예약이 포함되어 있습니다. 이러한 OVF 파일을 VMware에 배포하여 가상 머신을 생성할 때, 미리 정의된 리소스 수만큼 사용 가능한 리소스가 없으면 가상 머신이 시작되지 않습니다.

이 작업 정보

`${post_edited_translations.segment}`

단계

1. VMware vSphere Hypervisor 클라이언트 트리에서 시작되지 않은 가상 머신을 선택합니다.
2. `${post_edited_translations.segment}`
3. `${post_edited_translations.segment}`
4. 가상 머신에 할당된 리소스를 조정합니다.
 - a. *CPU*를 선택한 다음 예약 슬라이더를 사용하여 이 가상 머신에 예약된 MHz를 조정하십시오.
 - b. *메모리*를 선택한 다음 예약 슬라이더를 사용하여 이 가상 머신에 예약된 MB를 조정하십시오.
5. *확인*을 클릭합니다.
6. 동일한 VM 호스트에서 호스팅되는 다른 가상 머신에 대해서도 필요에 따라 반복하십시오.

임시 설치 암호가 비활성화되었습니다

VMware 노드를 배포할 때 선택적으로 임시 설치 암호를 지정할 수 있습니다. 새 노드가 그리드에 참여하기 전에 VM 콘솔에 액세스하거나 SSH를 사용하려면 이 암호가 필요합니다.

임시 설치 암호를 사용하지 않도록 선택한 경우 설치 문제를 해결하기 위해 추가 단계를 수행해야 합니다.

다음 두 가지 방법 중 하나를 선택할 수 있습니다.

- 가상 머신을 다시 배포하되, 콘솔에 액세스하거나 SSH를 사용하여 설치 문제를 디버깅할 수 있도록 임시 설치 암호를 지정하십시오.
- vCenter를 사용하여 암호를 설정하십시오.
 - a. VM의 전원을 끄십시오.
 - b. VM으로 이동하여 구성 탭을 선택한 다음 *vApp 옵션*을 선택합니다.
 - c. 설정할 임시 설치 암호 유형을 지정하십시오.
 - 사용자 지정 임시 암호를 설정하려면 *CUSTOM_TEMPORARY_PASSWORD*를 선택하십시오.
 - 노드 이름을 임시 암호로 사용하려면 *TEMPORARY_PASSWORD_TYPE*을 선택하십시오.
 - d. *값 설정*을 선택합니다.
 - e. 임시 비밀번호를 설정합니다.
 - *CUSTOM_TEMPORARY_PASSWORD*를 사용자 지정 암호 값으로 변경하십시오.
 - **TEMPORARY_PASSWORD_TYPE** 을(를) 노드 이름 값으로 업데이트합니다.
 - f. 새 암호를 적용하려면 가상 머신을 다시 시작하십시오.

관련 정보

- 로그 파일에 액세스하는 방법을 알아보려면 ["로그 파일 참조"](#)을 참조하십시오.
- ["StorageGRID 시스템 문제 해결"](#)
- 추가적인 도움이 필요하시면 ["NetApp 지원"](#)에 문의하십시오.

스크립트 예시

RHEL 기반 StorageGRID에서 4개의 물리적 인터페이스를 하나의 본드로 통합하는 예시

제공된 예제 파일을 사용하여 4개의 Linux 물리적 인터페이스를 단일 LACP 본드로 집계한 다음, 해당 본드를 하위로 하는 3개의 VLAN 인터페이스를 StorageGRID Grid, 관리 및 클라이언트 네트워크 인터페이스로 설정할 수 있습니다.

물리적 인터페이스

참고로 링크의 다른 끝에 있는 스위치도 네 개의 포트를 단일 LACP 트렁크 또는 포트 채널로 처리해야 하며, 참조된 VLAN 세 개 이상을 태그와 함께 전달해야 합니다.

```
/etc/sysconfig/network-scripts/ifcfg-ens160
```

```
TYPE=Ethernet
NAME=ens160
UUID=011b17dd-642a-4bb9-acae-d71f7e6c8720
DEVICE=ens160
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

/etc/sysconfig/network-scripts/ifcfg-ens192

```
TYPE=Ethernet
NAME=ens192
UUID=e28eb15f-76de-4e5f-9a01-c9200b58d19c
DEVICE=ens192
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

/etc/sysconfig/network-scripts/ifcfg-ens224

```
TYPE=Ethernet
NAME=ens224
UUID=b0e3d3ef-7472-4cde-902c-ef4f3248044b
DEVICE=ens224
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

/etc/sysconfig/network-scripts/ifcfg-ens256

```
TYPE=Ethernet
NAME=ens256
UUID=7cf7aabc-3e4b-43d0-809a-1e2378faa4cd
DEVICE=ens256
ONBOOT=yes
MASTER=bond0
SLAVE=yes
```

결합 인터페이스

/etc/sysconfig/network-scripts/ifcfg-bond0

```
DEVICE=bond0
TYPE=Bond
BONDING_MASTER=yes
NAME=bond0
ONBOOT=yes
BONDING_OPTS=mode=802.3ad
```

VLAN 인터페이스

/etc/sysconfig/network-scripts/ifcfg-bond0.1001

```
VLAN=yes
TYPE=Vlan
DEVICE=bond0.1001
PHYSDEV=bond0
VLAN_ID=1001
REORDER_HDR=0
BOOTPROTO=none
UUID=296435de-8282-413b-8d33-c4dd40fca24a
ONBOOT=yes
```

/etc/sysconfig/network-scripts/ifcfg-bond0.1002

```
VLAN=yes
TYPE=Vlan
DEVICE=bond0.1002
PHYSDEV=bond0
VLAN_ID=1002
REORDER_HDR=0
BOOTPROTO=none
UUID=dbaaec72-0690-491c-973a-57b7dd00c581
ONBOOT=yes
```

/etc/sysconfig/network-scripts/ifcfg-bond0.1003

```
VLAN=yes
TYPE=Vlan
DEVICE=bond0.1003
PHYSDEV=bond0
VLAN_ID=1003
REORDER_HDR=0
BOOTPROTO=none
UUID=d1af4b30-32f5-40b4-8bb9-71a2fbf809a1
ONBOOT=yes
```

Ubuntu 및 Debian에서 StorageGRID용 물리적 인터페이스를 통합하는 예

이 `/etc/network/interfaces` 파일은 물리적 인터페이스, 본드 인터페이스 및 VLAN 인터페이스를 정의하는 세 부분으로 구성되어 있습니다. 세 가지 예제 섹션을 하나의 파일로 결합하면 4개의 Linux 물리적 인터페이스를 단일 LACP 본드로 통합한 다음, 해당 본드를 하위로 하는 3개의 VLAN 인터페이스를 설정하여 StorageGRID Grid, 관리 및 클라이언트 네트워크 인터페이스로 사용할 수 있습니다.

물리적 인터페이스

참고로 링크의 다른 끝에 있는 스위치도 네 개의 포트를 단일 LACP 트렁크 또는 포트 채널로 처리해야 하며, 참조된 VLAN 세 개 이상을 태그와 함께 전달해야 합니다.

```
# loopback interface
auto lo
iface lo inet loopback

# ens160 interface
auto ens160
iface ens160 inet manual
    bond-master bond0
    bond-primary en160

# ens192 interface
auto ens192
iface ens192 inet manual
    bond-master bond0

# ens224 interface
auto ens224
iface ens224 inet manual
    bond-master bond0

# ens256 interface
auto ens256
iface ens256 inet manual
    bond-master bond0
```

결합 인터페이스

```
# bond0 interface
auto bond0
iface bond0 inet manual
    bond-mode 4
    bond-miimon 100
    bond-slaves ens160 ens192 ens224 ens256
```

VLAN 인터페이스

```
# 1001 vlan
auto bond0.1001
iface bond0.1001 inet manual
vlan-raw-device bond0

# 1002 vlan
auto bond0.1002
iface bond0.1002 inet manual
vlan-raw-device bond0

# 1003 vlan
auto bond0.1003
iface bond0.1003 inet manual
vlan-raw-device bond0
```

저작권 정보

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

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이선스와 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 라이선스도 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 없으며 재사용이 불가능하며 취소 불가능한 라이선스를 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 라이선스는 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.