



소프트웨어 기반 노드 설치

StorageGRID software

NetApp
February 26, 2026

목차

소프트웨어 기반 노드 설치	1
소프트웨어 기반 노드에 StorageGRID 설치하기 위한 빠른 시작	1
설치를 자동화합니다	1
소프트웨어 기반 노드에 대한 설치를 계획하고 준비합니다.....	2
필요한 정보 및 자료	2
StorageGRID 설치 파일을 다운로드하고 압축을 풉니다	3
설치 파일 수동 확인(선택 사항)	9
소프트웨어 요구 사항	10
CPU 및 RAM 요구 사항	14
요구사항을 충족해야 합니다	15
노드 컨테이너 마이그레이션 요구 사항(Linux).....	21
호스트 준비(Linux).....	23
소프트웨어 기반 노드 설치 자동화	40
설치 자동화(Linux).....	40
설치 자동화(VMware)	43
가상 그리드 노드 배포	56
배포 환경(VMware)에 대한 정보 수집	57
Linux 배포를 위한 노드 구성 파일 만들기	58
그리드 노드가 기본 관리자 노드를 검색하는 방법	75
StorageGRID 노드를 가상 머신(VMware)으로 배포	75
예제 노드 구성 파일(Linux).....	81
StorageGRID 구성 검증(Linux)	84
StorageGRID 호스트 서비스 시작(Linux).....	85
REST API 설치	86
StorageGRID 설치 API	86
설치 문제를 해결합니다	87
스크립트 예제	89
예제 /etc/sysconfig/network-scripts(RHEL)	89
예시 /etc/network/interfaces(Ubuntu 및 Debian).....	92

소프트웨어 기반 노드 설치

소프트웨어 기반 노드에 **StorageGRID** 설치하기 위한 빠른 시작

Linux 또는 VMware StorageGRID 노드를 설치하려면 다음의 고급 단계를 따르세요.



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

1

준비

- 에 대해 자세히 "[StorageGRID 아키텍처 및 네트워크 토폴로지](#)"알아보십시오.
- 에 대해 자세히 "[StorageGRID 네트워킹](#)"알아보십시오.
- 를 수집하고 "[필요한 정보 및 자료](#)"준비합니다.
- (VMware만 해당) 설치 및 구성"[VMware vSphere Hypervisor, vCenter 및 ESX 호스트](#)".
- 필요한 를 "[CPU 및 RAM](#)"준비합니다.
- 에 대해 를 "[스토리지 및 성능 요구사항](#)"제공합니다.
- (리눅스만 해당)"[Linux 서버를 준비합니다](#)" StorageGRID 노드를 호스팅합니다.

2

구축

그리드 노드 구축 그리드 노드를 구축하면 StorageGRID 시스템의 일부로 생성되고 하나 이상의 네트워크에 연결됩니다.

- (Linux 전용) 1단계에서 준비한 호스트에 소프트웨어 기반 그리드 노드를 배포하려면 Linux 명령줄을 사용하십시오. "[노드 구성 파일](#)".
- (VMware 전용) VMware vSphere Web Client, .vmdk 파일 및 .ovf 파일 템플릿 세트를 사용하여"[소프트웨어 기반 노드를 가상 머신\(VM\)으로 구축](#)" 1단계에서 준비한 서버에서.
- StorageGRID 어플라이언스 노드를 배포하려면 를 "[하드웨어 설치를 빠르게 시작합니다](#)"따르십시오.

3

구성

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

설치를 자동화합니다



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

리눅스

StorageGRID 호스트 서비스 설치 및 그리드 노드 구성을 자동화하여 시간을 절약하고 일관성을 제공할 수 있습니다.

- Ansible, Puppet, Chef와 같은 표준 오케스트레이션 프레임워크를 사용하여 다음을 자동화합니다.
 - 리눅스 설치
 - 네트워킹 및 스토리지 구성
 - 컨테이너 엔진 및 StorageGRID 호스트 서비스 설치
 - 가상 그리드 노드 구축

보다 "[StorageGRID 호스트 서비스의 설치 및 구성을 자동화합니다](#)".

- 그리드 노드를 배포한 후 "[StorageGRID 시스템의 구성을 자동화합니다](#)" 설치 아카이브에 제공된 Python 구성 스크립트를 사용합니다.
- "[어플라이언스 그리드 노드의 설치 및 구성을 자동화합니다](#)"
- StorageGRID 구축의 고급 개발자인 경우 를 사용하여 그리드 노드 설치를 "[REST API 설치](#)" 자동화합니다.

VMware

시간을 절약하고 일관성을 제공하기 위해 그리드 노드의 구축과 구성 및 StorageGRID 시스템 구성을 자동화할 수 있습니다.

- "[VMware vSphere를 사용하여 그리드 노드 구축을 자동화합니다](#)".
- 그리드 노드를 배포한 후 "[StorageGRID 시스템의 구성을 자동화합니다](#)" 설치 아카이브에 제공된 Python 구성 스크립트를 사용합니다.
- "[어플라이언스 그리드 노드의 설치 및 구성을 자동화합니다](#)"
- StorageGRID 구축의 고급 개발자인 경우 를 사용하여 그리드 노드 설치를 "[REST API 설치](#)" 자동화합니다.

소프트웨어 기반 노드에 대한 설치를 계획하고 준비합니다.

필요한 정보 및 자료

StorageGRID를 설치하기 전에 필요한 정보와 자료를 수집하고 준비합니다.

필수 정보입니다

네트워크 계획

각 StorageGRID 노드에 연결할 네트워크 StorageGRID는 트래픽 분리, 보안 및 관리의 편의를 위해 여러 네트워크를 지원합니다.

StorageGRID 를 "[네트워킹 지침](#)" 참조하십시오.

네트워크 정보

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

그리드 노드용 서버

구축할 StorageGRID 노드의 수와 유형을 지원하기에 충분한 리소스를 제공하는 물리적 서버 세트, 가상 서버 또는 둘 다 식별합니다.



StorageGRID 설치에서 StorageGRID 어플라이언스(하드웨어) 스토리지 노드를 사용하지 않는 경우 BBWC(배터리 지원 쓰기 캐시)와 함께 하드웨어 RAID 스토리지를 사용해야 합니다. StorageGRID는 VSAN(Virtual Storage Area Network), 소프트웨어 RAID 또는 RAID 보호 사용을 지원하지 않습니다.

노드 마이그레이션(필요한 경우 **Ubuntu** 및 **Debian**만 해당)

"[노드 마이그레이션에 대한 요구사항](#)" 서비스 중단 없이 물리적 호스트에 대해 예약된 유지 관리를 수행하려는 경우를 이해합니다.

관련 정보

["NetApp 상호 운용성 매트릭스 툴"](#)

필수 자료

NetApp StorageGRID 라이선스

디지털 서명된 유효한 NetApp 라이선스가 있어야 합니다.



테스트 및 개념 증명 그리드에 사용할 수 있는 비운영 라이선스가 StorageGRID 설치 아카이브에 포함되어 있습니다.

StorageGRID 설치 아카이브

["StorageGRID 설치 아카이브를 다운로드하고 파일 압축을 풉니다"](#)..

서비스 노트북

StorageGRID 시스템은 서비스 랩톱을 통해 설치됩니다.

서비스 랩톱의 구성 요소:

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

StorageGRID 설명서

- "[릴리스 정보](#)"
- "[StorageGRID 관리 지침](#)"

StorageGRID 설치 파일을 다운로드하고 압축을 풉니다

StorageGRID 설치 아카이브를 다운로드하고 필요한 파일을 추출해야 합니다. 선택적으로 설치 패키지의 파일을 수동으로 확인할 수 있습니다.

단계

1. 로 이동합니다 "[StorageGRID용 NetApp 다운로드 페이지](#)".
2. 최신 릴리스를 다운로드하려면 버튼을 선택하거나 드롭다운 메뉴에서 다른 버전을 선택하고 * GO * 를 선택합니다.
3. NetApp 계정의 사용자 이름과 암호를 사용하여 로그인합니다.
4. Caution/MustRead 문이 나타나면 해당 문을 읽고 확인란을 선택합니다.



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

5. 최종 사용자 사용권 계약을 읽고 확인란을 선택한 다음 * 동의 및 계속 * 을 선택합니다.
6. * StorageGRID 설치* 열에서 소프트웨어 기반 노드 유형(RHEL, Ubuntu 또는 Debian, VMware)에 맞는 .tgz 또는 .zip 설치 아카이브를 선택합니다.



서비스 랩톱에서 Windows를 실행하는 경우 이 .zip 파일을 사용합니다.

7. 설치 아카이브를 저장합니다.
8. Linux 노드에서는 코드 서명 검증이 수동으로 수행됩니다. 선택적으로 설치 아카이브를 확인해야 하는 경우:
 - a. StorageGRID 코드 서명 확인 패키지를 다운로드합니다. 이 패키지의 파일 이름은 StorageGRID 소프트웨어 버전의 형식을 StorageGRID_<version-number>_Code_Signature_Verification_Package.tar.gz 사용합니다. <version-number>
 - b. 다음 단계를 따르세요 "[설치 파일을 수동으로 확인합니다](#)".
9. 설치 아카이브에서 파일 압축을 풉니다.
10. 필요한 파일을 선택합니다.

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



표에 나열된 경로는 추출된 설치 아카이브에서 설치한 최상위 디렉토리에 상대적입니다.

RHEL

경로 및 파일 이름입니다	설명
	StorageGRID 다운로드 파일에 포함된 모든 파일을 설명하는 텍스트 파일입니다.
	제품에 대한 지원 권한을 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 어플라이언스 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
/rpms/configure -StorageGrid.sample.json	스크립트와 함께 사용할 예제 구성 파일 configure-storagegrid.py
	SSO(Single Sign-On)가 활성화된 경우 Grid Management API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제 이 스크립트를 Ping 연합 통합에 사용할 수도 있습니다.
/rpms/configure -StorageGrid.blank.json을 지정합니다	스크립트와 함께 사용할 빈 구성 configure-storagegrid.py 파일입니다.
	StorageGRID 컨테이너 배포를 위해 RHEL 호스트를 구성하기 위한 Ansible 역할 및 플레이북 예 필요에 따라 역할 또는 플레이북을 사용자 지정할 수 있습니다.
	Active Directory 또는 Ping 연방을 사용하여 SSO(Single Sign-On)를 사용하도록 설정한 경우 Grid Management API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제

경로 및 파일 이름입니다	설명
/rpms/StorageGrid-ssoauth-Azure.js	Azure와의 SSO 상호 작용을 수행하기 위해 Python 스크립트에 의해 호출되는 도우미 스크립트입니다. storagegrid-ssoauth-azure.py
/rpms/Extras/API-schemas	StorageGRID에 대한 API 스키마입니다. <ul style="list-style-type: none"> 참고 *: 업그레이드를 수행하기 전에 이러한 스키마를 사용하여 StorageGRID 관리 API를 사용하도록 작성한 코드가 업그레이드 호환성 테스트를 위한 비프로덕션 StorageGRID 환경이 없는 경우 새 StorageGRID 릴리스와 호환되는지 확인할 수 있습니다.

Ubuntu 또는 Debian

경로 및 파일 이름입니다	설명
	StorageGRID 다운로드 파일에 포함된 모든 파일을 설명하는 텍스트 파일입니다.
/debs/NLF000000.txt 를 참조하십시오	테스트 및 개념 증명 배포에 사용할 수 있는 비프로덕션 NetApp 라이선스 파일.
/debs/storagegrid-webscale-images-version-SHA.deb 를 참조하십시오	StorageGRID 노드 이미지를 Ubuntu 또는 Debian 호스트에 설치하기 위한 DEB 패키지.
/debs/storagegrid-webscale-images-version-SHA.deb.md5 를 참조하십시오	파일의 MD5 체크섬 /debs/storagegrid-webscale-images-version-SHA.deb.
/debs/storagegrid-webscale-service-version-SHA.deb 를 참조하십시오	Ubuntu 또는 Debian 호스트에 StorageGRID 호스트 서비스를 설치하기 위한 DEB 패키지.
배포 스크립팅 도구	설명
/debs/configure-storagegrid.py 를 참조하십시오	StorageGRID 시스템 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
/debs/configure-sga.py 를 참조하십시오	StorageGRID 어플라이언스 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
/debs/storagegrid-ssoauth.py 를 참조하십시오	SSO(Single Sign-On)가 활성화된 경우 Grid Management API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제 이 스크립트를 Ping 연합 통합에 사용할 수도 있습니다.

경로 및 파일 이름입니다	설명
/debs/configure -StorageGrid.sample.json 을 참조하십시오	스크립트와 함께 사용할 예제 구성 파일 configure-storagegrid.py
/debs/configure -StorageGrid.blank.json 을 참조하십시오	스크립트와 함께 사용할 빈 구성 configure-storagegrid.py 파일입니다.
	StorageGRID 컨테이너 배포를 위한 Ubuntu 또는 Debian 호스트 구성을 위한 Ansible 역할 및 플레이북에 필요에 따라 역할 또는 플레이북을 사용자 지정할 수 있습니다.
	Active Directory 또는 Ping 연방을 사용하여 SSO(Single Sign-On)를 사용하도록 설정한 경우 Grid Management API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제
/debs/StorageGrid-ssoauth-Azure.js를 입력합니다	Azure와의 SSO 상호 작용을 수행하기 위해 Python 스크립트에 의해 호출되는 도우미 스크립트입니다. storagegrid-ssoauth-azure.py
/debs/Extras/API-schemas	StorageGRID에 대한 API 스키마입니다. <ul style="list-style-type: none"> 참고 *: 업그레이드를 수행하기 전에 이러한 스키마를 사용하여 StorageGRID 관리 API를 사용하도록 작성한 코드가 업그레이드 호환성 테스트를 위한 비프로덕션 StorageGRID 환경이 없는 경우 새 StorageGRID 릴리스와 호환되는지 확인할 수 있습니다.

VMware

경로 및 파일 이름입니다	설명
	StorageGRID 다운로드 파일에 포함된 모든 파일을 설명하는 텍스트 파일입니다.
	제품에 대한 지원 권한을 제공하지 않는 무료 라이선스입니다.
	그리드 노드 가상 머신을 생성하기 위한 템플릿으로 사용되는 가상 머신 디스크 파일입니다.
	Open Virtualization Format 템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 기본 관리자 노드를 배포할 수 있습니다.

경로 및 파일 이름입니다	설명
	템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 비기본 관리 노드를 배포합니다.
	템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 게이트웨이 노드를 배포할 수 있습니다.
	템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 가상 머신 기반 스토리지 노드를 구축합니다.
배포 스크립팅 도구	설명
	가상 그리드 노드의 배포를 자동화하는 데 사용되는 Bash 셸 스크립트입니다.
	스크립트와 함께 사용할 예제 구성 파일 <code>deploy-vsphere-ovftool.sh</code>
	StorageGRID 시스템 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	StorageGRID 어플라이언스 구성을 자동화하는 데 사용되는 Python 스크립트입니다.
	SSO(Single Sign-On)가 활성화된 경우 Grid Management API에 로그인하는 데 사용할 수 있는 Python 스크립트의 예 이 스크립트를 Ping 연합 통합에 사용할 수도 있습니다.
<code>/vSphere/configure -StorageGrid.sample.json</code> 을 참조하십시오	스크립트와 함께 사용할 예제 구성 파일 <code>configure-storagegrid.py</code>
<code>/vSphere/configure -StorageGrid.blank.json</code> 을 참조하십시오	스크립트와 함께 사용할 빈 구성 <code>configure-storagegrid.py</code> 파일입니다.
	Active Directory 또는 Ping 연방을 사용하여 SSO(Single Sign-On)를 사용하도록 설정한 경우 Grid Management API에 로그인하는 데 사용할 수 있는 Python 스크립트 예제
<code>/vSphere/StorageGrid-ssoauth-Azure.js</code>	Azure와의 SSO 상호 작용을 수행하기 위해 Python 스크립트에 의해 호출되는 도우미 스크립트입니다. <code>storagegrid-ssoauth-azure.py</code>

경로 및 파일 이름입니다	설명
/vSphere/Extras/API-schemas	StorageGRID에 대한 API 스키마입니다. <ul style="list-style-type: none"> 참고 *: 업그레이드를 수행하기 전에 이러한 스키마를 사용하여 StorageGRID 관리 API를 사용하도록 작성한 코드가 업그레이드 호환성 테스트를 위한 비프로덕션 StorageGRID 환경이 없는 경우 새 StorageGRID 릴리스와 호환되는지 확인할 수 있습니다.

설치 파일 수동 확인(선택 사항)

필요한 경우 StorageGRID 설치 아카이브의 파일을 수동으로 확인할 수 있습니다.

시작하기 전에

당신은 가지고있다"[검증 패키지를 다운로드했습니다](#)" 에서 "[StorageGRID용 NetApp 다운로드 페이지](#)".

단계

1. 검증 패키지에서 아티팩트를 추출합니다.

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

2. 이러한 아티팩트가 추출되었는지 확인합니다.

- Leaf 인증서: Leaf-Cert.pem
- 인증서 체인: CA-Int-Cert.pem
- 타임 스탬프 응답 체인: TS-Cert.pem
- 체크섬 파일: sha256sum
- 체크섬 서명: sha256sum.sig
- 타임 스탬프 응답 파일: sha256sum.sig.tsr

3. 체인을 사용하여 리프 인증서가 유효한지 확인합니다.

- 예 *: `openssl verify -CAfile CA-Int-Cert.pem Leaf-Cert.pem`
- 예상 출력 *: `Leaf-Cert.pem: OK`

4. leaf 인증서가 만료되어 step_2_에 실패한 경우 파일을 사용하여 tsr 확인합니다.

- 예 *: `openssl ts -CAfile CA-Int-Cert.pem -untrusted TS-Cert.pem -verify -data sha256sum.sig -in sha256sum.sig.tsr`
- 예상 출력 포함 *: `Verification: OK`

5. 리프 인증서에서 공용 키 파일을 만듭니다.

- 예 *: `openssl x509 -pubkey -noout -in Leaf-Cert.pem > Leaf-Cert.pub`
- 예상 출력 *: `_none_`

6. 공개 키를 사용하여 sha256sum 에 대해 파일을 sha256sum.sig 확인합니다.

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

- 예상 출력 *: Verified OK

7. `sha256sum` 새로 생성된 체크섬을 기준으로 파일 내용을 확인합니다.

- 예 *: sha256sum -c sha256sum

*예상 출력 * <filename>: OK:+는 <filename> 다운로드한 아카이브 파일의 이름입니다.

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

소프트웨어 요구 사항

가상 머신을 사용하여 모든 유형의 StorageGRID 노드를 호스팅할 수 있습니다. 각 그리드 노드에 대해 하나의 가상 머신이 필요합니다.

RHEL

RHEL에 StorageGRID 설치하려면 일부 타사 소프트웨어 패키지를 설치해야 합니다. 일부 지원되는 Linux 배포판에는 기본적으로 이러한 패키지가 포함되어 있지 않습니다. StorageGRID 설치가 테스트된 소프트웨어 패키지 버전은 이 페이지에 나열된 버전을 포함합니다.

이러한 패키지를 필요로 하는 Linux 배포 및 컨테이너 런타임 설치 옵션을 선택했는데 Linux 배포판에 의해 자동으로 설치되지 않은 경우, 해당 공급자 또는 Linux 배포판의 지원 공급업체에서 제공하는 경우 여기에 나열된 버전 중 하나를 설치하십시오. 그렇지 않으면 공급업체에서 제공하는 기본 패키지 버전을 사용하십시오.

모든 설치 옵션에는 Podman 또는 Docker가 필요합니다. 두 패키지를 모두 설치하지 마십시오. 설치 옵션에 필요한 패키지만 설치합니다.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로 Docker에 대한 지원은 더 이상 사용되지 않습니다. Docker는 향후 릴리즈에서 다른 컨테이너 엔진으로 대체될 예정입니다.

테스트된 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

테스트된 Podman 버전

- 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

테스트된 Docker 버전



Docker 지원은 더 이상 사용되지 않으며 향후 릴리즈에서 제거될 예정입니다.

- Docker-CE 20.10.7

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

Ubuntu 및 Debian

Ubuntu 또는 Debian에 StorageGRID를 설치하려면 타사 소프트웨어 패키지를 설치해야 합니다. 지원되는 일부 Linux 배포판에는 기본적으로 이러한 패키지가 포함되어 있지 않습니다. StorageGRID 설치를 테스트하는 소프트웨어 패키지 버전에는 이 페이지에 나열된 버전이 포함됩니다.

이러한 패키지를 필요로 하는 Linux 배포 및 컨테이너 런타임 설치 옵션을 선택했는데 Linux 배포판에 의해 자동으로 설치되지 않은 경우, 해당 공급자 또는 Linux 배포판의 지원 공급업체에서 제공하는 경우 여기에 나열된 버전 중 하나를 설치하십시오. 그렇지 않으면 공급업체에서 제공하는 기본 패키지 버전을 사용하십시오.

모든 설치 옵션에는 Podman 또는 Docker가 필요합니다. 두 패키지를 모두 설치하지 마십시오. 설치 옵션에 필요한 패키지만 설치합니다.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로 Docker에 대한 지원은 더 이상 사용되지 않습니다. Docker는 향후 릴리즈에서 다른 컨테이너 엔진으로 대체될 예정입니다.

테스트된 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

테스트된 Podman 버전

- 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

테스트된 **Docker** 버전



Docker 지원은 더 이상 사용되지 않으며 향후 릴리즈에서 제거될 예정입니다.

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

VMware

VMware vSphere 하이퍼바이저

준비된 물리적 서버에 VMware vSphere 하이퍼바이저를 설치해야 합니다. VMware 소프트웨어를 설치하기 전에 하드웨어를 올바르게 구성해야 합니다(펌웨어 버전 및 BIOS 설정 포함).

- 설치할 StorageGRID 시스템에 대한 네트워킹을 지원하기 위해 필요에 따라 하이퍼바이저에서 네트워킹을 구성합니다.

"네트워킹 지침"

- 데이터 저장소가 그리드 노드를 호스팅하는 데 필요한 가상 머신 및 가상 디스크에 충분히 크지 확인합니다.
- 둘 이상의 데이터 저장소를 생성하는 경우 가상 머신을 생성할 때 각 그리드 노드에 사용할 데이터 저장소를 쉽게 식별할 수 있도록 각 데이터 저장소의 이름을 지정합니다.

ESX 호스트 구성 요구 사항



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

VMware 구성 요구 사항

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

지원되는 VMware vSphere Hypervisor 및 VMware vCenter Server 소프트웨어 버전은 를 참조하십시오 "[NetApp 상호 운용성 매트릭스 툴](#)".

이러한 VMware 제품을 설치하는 데 필요한 단계는 VMware 설명서를 참조하십시오.

CPU 및 RAM 요구 사항

StorageGRID 소프트웨어를 설치하기 전에 StorageGRID 시스템을 지원할 준비가 되도록 하드웨어를 확인 및 구성하십시오.

각 StorageGRID 노드에는 다음과 같은 최소 리소스가 필요합니다.

- CPU 코어: 노드당 8개
- RAM: 사용 가능한 총 RAM과 시스템에서 실행되는 비 StorageGRID 소프트웨어의 양에 따라 다릅니다
 - 일반적으로 노드당 최소 24GB, 총 시스템 RAM보다 2 ~ 16GB 작습니다
 - 각 테넌트당 최소 64GB의 버킷이 약 5,000개 있습니다

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

- 기존 StorageGRID 사이트에서 SG6000 또는 SG6100 어플라이언스를 사용 중인 경우 소프트웨어 기반 메타데이터 전용 노드가 다음과 같은 최소 요구사항을 충족해야 합니다.
 - 128GB RAM
 - 8코어 CPU
 - Cassandra 데이터베이스용 8TB SSD 또는 동급 스토리지(rangedb/0)
- 기존 StorageGRID 사이트가 24GB RAM, 8코어 CPU, 3TB 또는 4TB의 메타데이터 스토리지를 갖춘 가상 스토리지 노드를 사용하는 경우, 소프트웨어 기반 메타데이터 전용 노드는 비슷한 리소스(24GB RAM, 8코어 CPU, 4TB의 메타데이터 스토리지(rangedb/0))를 사용해야 합니다.

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



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

리눅스

각 물리적 또는 가상 호스트에서 실행하려는 StorageGRID 노드 수가 사용 가능한 CPU 코어 수 또는 물리적 RAM을 초과하지 않는지 확인합니다. 호스트가 StorageGRID 실행 전용이 아닌 경우(권장되지 않음) 다른 애플리케이션의 리소스 요구 사항을 고려해야 합니다.

VMware

VMware는 가상 머신당 하나의 노드를 지원합니다. StorageGRID 노드가 사용 가능한 물리적 RAM을 초과하지 않는지 확인합니다. 각 가상 머신은 StorageGRID를 실행하기 위한 전용이어야 합니다.



CPU 및 메모리 사용량을 정기적으로 모니터링하여 이러한 리소스가 작업 부하를 지속적으로 수용할 수 있도록 합니다. 예를 들어, 가상 스토리지 노드에 대한 RAM 및 CPU 할당을 두 배로 하면 StorageGRID 어플라이언스 노드에 제공되는 것과 유사한 리소스를 제공할 수 있습니다. 또한 노드당 메타데이터 양이 500GB를 초과하는 경우 노드당 RAM을 48GB 이상으로 늘리는 것이 좋습니다. 개체 메타데이터 스토리지 관리, 메타데이터 예약 공간 설정 증가, CPU 및 메모리 사용량 모니터링에 대한 자세한 내용은 "[관리](#)" "[모니터링](#)", 및 "[업그레이드 중](#)" StorageGRID에 대한 지침을 참조하십시오.

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

가상 시스템을 호스트로 사용하고 VM의 크기와 수를 제어하는 경우 각 StorageGRID 노드에 대해 단일 VM을 사용하고 그에 따라 VM 크기를 조정해야 합니다.

(RHEL, Debian, Ubuntu에만 해당) 프로덕션 배포의 경우 동일한 물리적 스토리지 하드웨어 또는 가상 호스트에서 여러 스토리지 노드를 실행하면 안 됩니다. 단일 StorageGRID 배포에서 각 스토리지 노드는 자체적으로 격리된 장애 도메인에 있어야 합니다. 단일 하드웨어 장애가 단일 스토리지 노드에만 영향을 미치도록 하면 개체 데이터의 내구성과 가용성을 극대화할 수 있습니다.

도 ["요구사항을 충족해야 합니다"](#)참조하십시오.

요구사항을 충족해야 합니다

초기 구성과 향후 스토리지 확장을 지원할 충분한 공간을 제공할 수 있도록 StorageGRID 노드의 스토리지 요구사항을 이해해야 합니다.

저장 및 성능 요구 사항은 소프트웨어 기반 노드 구현에 따라 다릅니다.



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

저장 카테고리

StorageGRID 노드에는 다음과 같은 세 가지 논리적 스토리지 범주가 필요합니다.

- * 컨테이너 풀 * — StorageGRID 노드를 지원할 호스트에 컨테이너 엔진을 설치 및 구성할 때 컨테이너 엔진 스토리지 드라이버에 할당되는 노드 컨테이너용 성능 계층(10K SAS 또는 SSD) 스토리지입니다.
- * 시스템 데이터 * — StorageGRID 호스트 서비스가 사용하고 개별 노드에 매핑하는 시스템 데이터 및 트랜잭션 로그의 노드당 영구 스토리지를 위한 성능 계층(10K SAS 또는 SSD) 스토리지입니다.
- * 오브젝트 데이터 * — 객체 데이터 및 객체 메타데이터의 영구 스토리지를 위한 Performance-Tier(10K SAS 또는 SSD) 스토리지 및 Capacity-Tier(NL-SAS/SATA) 대용량 스토리지

모든 스토리지 범주에 RAID 지원 블록 장치를 사용해야 합니다. 비중복 디스크, SSD 또는 JBOD는 지원되지 않습니다. 모든 스토리지 범주에서 공유 또는 로컬 RAID 스토리지를 사용할 수 있지만 StorageGRID의 노드 마이그레이션 기능을 사용하려면 시스템 데이터와 오브젝트 데이터를 모두 공유 스토리지에 저장해야 합니다. 자세한 내용은 ["노드 컨테이너 마이그레이션 요구사항"](#)참조하십시오.

성능 요구사항

컨테이너 풀, 시스템 데이터 및 오브젝트 메타데이터에 사용되는 볼륨의 성능은 시스템의 전반적인 성능에 큰 영향을 미칩니다. 이러한 볼륨에 성능 계층(10K SAS 또는 SSD) 스토리지를 사용하면 지연 시간, IOPS(초당 입출력 작업) 및 처리량 측면에서 디스크 성능이 적절하게 보장됩니다. 객체 데이터의 영구 스토리지를 위해 용량 계층(NL-SAS/SATA) 스토리지를 사용할 수 있습니다.

컨테이너 풀, 시스템 데이터 및 오브젝트 데이터에 사용되는 볼륨에는 다시 쓰기 캐시가 설정되어 있어야 합니다. 캐시는 보호되거나 영구 미디어에 있어야 합니다.

NetApp ONTAP 스토리지를 사용하는 호스트의 요구 사항입니다

StorageGRID 노드가 NetApp ONTAP 시스템에서 할당된 스토리지를 사용하는 경우 볼륨에 FabricPool 계층화 정책이 활성화되어 있지 않은지 확인합니다. StorageGRID 노드와 함께 사용되는 볼륨에 대해 FabricPool 계층화를 사용하지 않도록 설정하면 문제 해결과 스토리지 작업이 간소화됩니다.



FabricPool를 사용하여 StorageGRID 관련 데이터를 StorageGRID 자체로 계층화하지 마십시오. StorageGRID 데이터를 StorageGRID로 다시 계층화하면 문제 해결과 운영 복잡성이 늘어납니다.

필요한 호스트 수입니다

각 StorageGRID 사이트에는 최소 3개의 스토리지 노드가 필요합니다.



운영 구축 시 단일 물리적 호스트 또는 가상 호스트에서 스토리지 노드를 두 개 이상 실행하지 마십시오. 각 스토리지 노드에 대해 전용 호스트를 사용하면 격리된 장애 도메인이 제공됩니다.

관리 노드 또는 게이트웨이 노드와 같은 다른 유형의 노드는 동일한 호스트에 구축하거나 필요에 따라 전용 호스트에 구축할 수 있습니다.



디스크 스냅샷을 사용하여 그리드 노드를 복원할 수 없습니다. 대신 "[그리드 노드 복구](#)" 각 노드 유형에 대한 절차를 참조하십시오.

각 노드의 저장 볼륨 수

다음 표에는 각 호스트에 필요한 스토리지 볼륨(LUN) 수와 해당 호스트에 구축할 노드를 기준으로 각 LUN에 필요한 최소 크기가 나와 있습니다.

테스트된 최대 LUN 크기는 39TB입니다.



이러한 숫자는 전체 그리드가 아닌 각 호스트에 대한 것입니다.

LUN 사용 목적	스토리지 범주입니다	LUN 수입니다	최소 크기/LUN
컨테이너 엔진 스토리지 풀입니다	컨테이너 풀입니다	1	총 노드 수 × 100GB
/var/local 볼륨	시스템 데이터	이 호스트의 각 노드에 대해 1개	100GB

LUN 사용 목적	스토리지 범주입니다	LUN 수입니다	최소 크기/LUN
스토리지 노드	오브젝트 데이터	이 호스트의 각 스토리지 노드에 대해 3개 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 저장 볼륨을 권장합니다.	12TB(LUN당 4TB, 최소) 테스트된 최대 LUN 크기: 39TB. 보다 스토리지 노드의 스토리지 요구 사항 자세한 내용은.
스토리지 노드(메타데이터만)	오브젝트 메타데이터	1	최소 4TB/LUN 테스트된 최대 LUN 크기: 39TB. 보다 스토리지 노드의 스토리지 요구 사항 자세한 내용은. • 참고 *: 메타데이터 전용 스토리지 노드에는 하나의 rangedb만 필요합니다.
관리자 노드 감사 로그	시스템 데이터	이 호스트의 각 관리 노드에 대해 1개	200GB
관리자 노드 테이블	시스템 데이터	이 호스트의 각 관리 노드에 대해 1개	200GB



구성된 감사 수준, S3 개체 키 이름과 같은 사용자 입력 크기, 보존해야 하는 감사 로그 데이터 양에 따라 각 관리 노드에서 감사 로그 LUN 크기를 늘려야 할 수도 있습니다. 일반적으로 그리드는 S3 작업당 약 1KB의 감사 데이터를 생성합니다. 즉, 200GB LUN은 2~3일 동안 하루에 7,000만 개의 작업 또는 초당 800개의 작업을 지원할 수 있습니다.

호스트의 최소 스토리지 공간입니다

다음 표에는 각 노드 유형에 필요한 최소 스토리지 공간이 나와 있습니다. 이 표를 사용하여 각 스토리지 범주에서 호스트에 구축해야 하는 최소 스토리지 양을 해당 호스트에 구축될 노드를 기반으로 결정할 수 있습니다.



디스크 스냅샷을 사용하여 그리드 노드를 복원할 수 없습니다. 대신 "[그리드 노드 복구](#)" 각 노드 유형에 대한 절차를 참조하십시오.

각 노드 호스트에는 OS용 100GB LUN이 필요합니다.

노드 유형입니다	컨테이너 풀입니다	시스템 데이터	오브젝트 데이터
스토리지 노드	100GB	100GB	4,000GB
관리자 노드	100GB	500GB(3개 LUN)	_해당 사항 없음_
게이트웨이 노드	100GB	100GB	_해당 사항 없음_

예: 호스트 또는 가상 머신의 스토리지 요구 사항 계산

동일한 호스트나 가상 머신에 스토리지 노드 하나, 관리 노드 하나, 게이트웨이 노드 하나, 이렇게 세 개의 노드를 배포할 계획이라고 가정해 보겠습니다. 호스트에 최소 9개의 스토리지 볼륨을 제공해야 합니다. 노드 컨테이너의 경우 최소 300GB의 성능 계층 스토리지가 필요하고, 시스템 데이터와 트랜잭션 로그의 경우 700GB의 성능 계층 스토리지가 필요하며, 개체 데이터의 경우 12TB의 용량 계층 스토리지가 필요합니다.

Linux 호스트 예제

노드 유형입니다	LUN 사용 목적	LUN 수입니다	LUN 크기입니다
스토리지 노드	컨테이너 엔진 스토리지 풀입니다	1	300GB(100GB/노드)
스토리지 노드	/var/local 볼륨	1	100GB
스토리지 노드	오브젝트 데이터	3	12TB(4TB/LUN)
관리자 노드	/var/local 볼륨	1	100GB
관리자 노드	관리자 노드 감사 로그	1	200GB
관리자 노드	관리자 노드 테이블	1	200GB
게이트웨이 노드	/var/local 볼륨	1	100GB
• 합계 *		• 9 *	• 컨테이너 풀: * 300GB 시스템 데이터: 700GB • 오브젝트 데이터: * 12,000GB

VMware 가상 머신 예제

노드 유형입니다	LUN 사용 목적	LUN 수입니다	LUN 크기입니다
스토리지 노드	OS 볼륨	1	100GB
스토리지 노드	오브젝트 데이터	3	12TB(4TB/LUN)
관리자 노드	OS 볼륨	1	100GB
관리자 노드	관리자 노드 감사 로그	1	200GB
관리자 노드	관리자 노드 테이블	1	200GB
게이트웨이 노드	OS 볼륨	1	100GB
• 합계 *		8	시스템 데이터: 700GB • 오브젝트 데이터: * 12,000GB

스토리지 노드에 대한 특정 스토리지 요구 사항

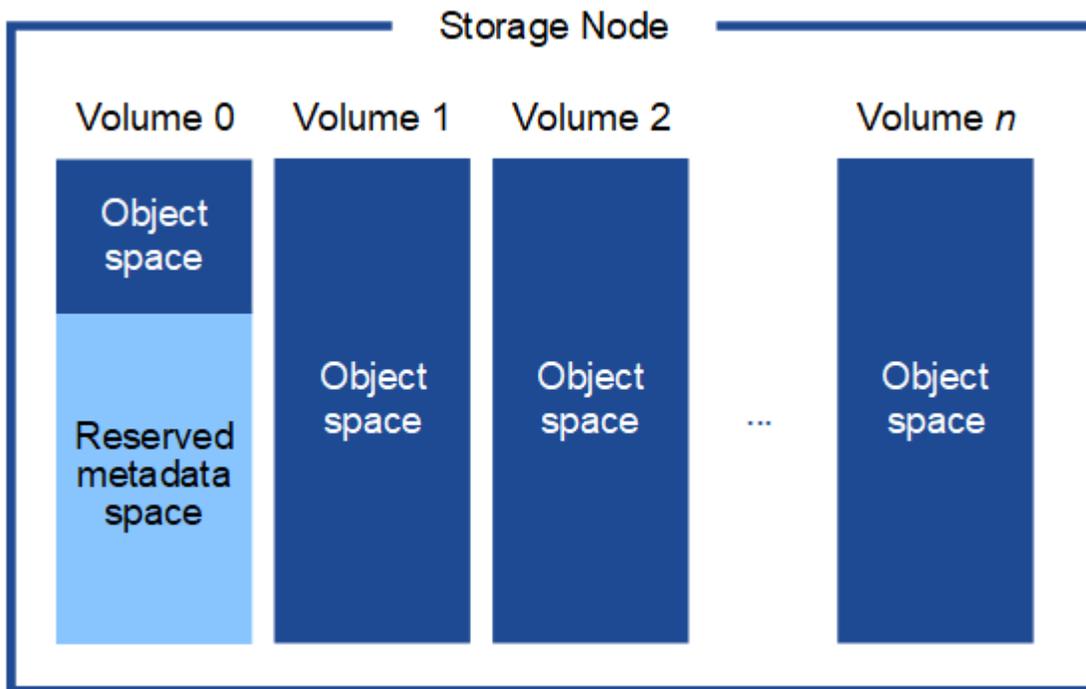
Linux와 VMware는 스토리지 노드에 대해 서로 다른 스토리지 요구 사항을 갖습니다.

- Linux 소프트웨어 기반 스토리지 노드는 1~48개의 스토리지 볼륨을 가질 수 있습니다.
- VMware 소프트웨어 기반 스토리지 노드는 1~16개의 스토리지 볼륨을 가질 수 있습니다.
- 3개 이상의 저장 볼륨을 권장합니다.
- 각 저장 볼륨은 4TB 이상이어야 합니다.



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

그림에 나와 있는 것처럼 StorageGRID는 각 스토리지 노드의 스토리지 볼륨 0에 객체 메타데이터를 위한 공간을 예약합니다. 스토리지 볼륨 0 및 스토리지 노드의 다른 스토리지 볼륨의 나머지 공간은 오브젝트 데이터에만 사용됩니다.



이중화를 제공하고 객체 메타데이터를 손실로부터 보호하기 위해 StorageGRID는 각 사이트의 시스템 모든 개체에 대한 메타데이터 복사본을 3개 저장합니다. 오브젝트 메타데이터의 복사본 3개는 각 사이트의 모든 스토리지 노드에 균등하게 분산됩니다.

메타데이터 전용 스토리지 노드가 있는 그리드를 설치할 경우 그리드에는 오브젝트 스토리지용 최소 노드 수도 있어야 합니다. 메타데이터 전용 스토리지 노드에 대한 자세한 내용은 을 "[스토리지 노드 유형](#)"참조하십시오.

- 단일 사이트 그리드의 경우 객체 및 메타데이터에 대해 2개 이상의 스토리지 노드가 구성됩니다.
- 다중 사이트 그리드의 경우 사이트당 하나 이상의 스토리지 노드가 객체 및 메타데이터에 대해 구성됩니다.

새 스토리지 노드의 볼륨 0에 공간을 할당하는 경우 모든 오브젝트 메타데이터의 해당 노드에 적절한 공간이 있는지 확인해야 합니다.

- 적어도 볼륨 0에 4TB 이상을 할당해야 합니다.



스토리지 노드에 대해 하나의 스토리지 볼륨만 사용하고 볼륨에 4TB 이하의 용량을 할당하면 스토리지 노드가 시작 시 스토리지 읽기 전용 상태로 전환되고 객체 메타데이터만 저장할 수 있습니다.



볼륨 0에 500GB 미만의 용량을 할당할 경우(비운영 전용) 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 사용 중인 경우 소프트웨어 기반 메타데이터 전용 노드가 다음과 같은 최소 요구사항을 충족해야 합니다.
 - 128GB RAM
 - 8코어 CPU
 - Cassandra 데이터베이스용 8TB SSD 또는 동급 스토리지(rangedb/0)
 - 기존 StorageGRID 사이트가 24GB RAM, 8코어 CPU, 3TB 또는 4TB의 메타데이터 스토리지를 갖춘 가상 스토리지 노드를 사용하는 경우, 소프트웨어 기반 메타데이터 전용 노드는 비슷한 리소스(24GB RAM, 8코어 CPU, 4TB의 메타데이터 스토리지(rangedb/0))를 사용해야 합니다.

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

- 새 시스템(StorageGRID 11.6 이상)을 설치하고 각 스토리지 노드에 128MB 이상의 RAM이 있는 경우 볼륨 0에 8TB 이상을 할당합니다. 볼륨 0에 더 큰 값을 사용하면 각 스토리지 노드에서 메타데이터에 허용되는 공간이 증가할 수 있습니다.
- 사이트에 대해 서로 다른 스토리지 노드를 구성할 때 가능하면 볼륨 0에 대해 동일한 설정을 사용합니다. 사이트에 크기가 다른 스토리지 노드가 있는 경우 볼륨이 0인 스토리지 노드가 해당 사이트의 메타데이터 용량을 결정합니다.

자세한 내용은 ["오브젝트 메타데이터 스토리지 관리"](#) 참조하십시오.

노드 컨테이너 마이그레이션 요구 사항(Linux)

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 배포를 의미합니다. 지원되는 버전 목록은 다음을 참조하세요. ["NetApp 상호 운용성 매트릭스 툴\(IMT\)"](#).

노드 마이그레이션을 통해 그리드 작업을 중단하지 않고 물리적 호스트 유지 관리를 수행할 수 있습니다. 물리적 호스트를 오프라인으로 전환하기 전에 한 번에 하나씩 모든 StorageGRID 노드를 다른 호스트로 이동합니다. 노드를 마이그레이션하려면 각 노드의 다운타임만 짧고 그리드 서비스의 운영 또는 가용성에 영향을 미치지 않아야 합니다.

StorageGRID 노드 마이그레이션 기능을 사용하려면 배포가 추가 요구 사항을 충족해야 합니다.

- 단일 물리적 데이터 센터의 호스트 전반에서 일관된 네트워크 인터페이스 이름
- 단일 물리적 데이터 센터의 모든 호스트에서 액세스할 수 있는 StorageGRID 메타데이터 및 오브젝트 저장소 볼륨을 위한 공유 스토리지입니다. 예를 들어, NetApp E-Series 스토리지 어레이를 사용할 수 있습니다.

가상 호스트를 사용 중이고 기본 하이퍼바이저 계층에서 VM 마이그레이션을 지원하는 경우 StorageGRID의 노드 마이그레이션 기능 대신 이 기능을 사용할 수 있습니다. 이 경우 이러한 추가 요구 사항을 무시할 수 있습니다.

마이그레이션 또는 하이퍼바이저 유지 보수를 수행하기 전에 노드를 정상적으로 종료합니다. 의 지침을 "[그리드 노드 종료](#)"참조하십시오.

VMware Live Migration은 지원되지 않습니다

VMware VM에서 베어 메탈 설치를 수행할 때 OpenStack Live Migration 및 VMware Live vMotion을 사용하면 가상 머신 클럭 시간이 증가하며 어떠한 유형의 그리드 노드에서도 지원되지 않습니다. 드물지만 잘못된 클럭 시간으로 인해 데이터 또는 구성 업데이트가 손실될 수 있습니다.

콜드 마이그레이션이 지원됩니다. 콜드 마이그레이션에서는 StorageGRID 노드를 호스트 간에 마이그레이션하기 전에 종료해야 합니다. 의 지침을 "[그리드 노드 종료](#)"참조하십시오.

일관된 네트워크 인터페이스 이름

한 호스트에서 다른 호스트로 노드를 이동하려면 StorageGRID 호스트 서비스가 노드가 현재 위치에 있는 외부 네트워크 연결을 새 위치에서 복제할 수 있다는 확신을 가져야 합니다. 호스트에서 일관된 네트워크 인터페이스 이름을 사용하면 이러한 자신감을 얻을 수 있습니다.

예를 들어 호스트 1에서 실행되는 StorageGRID NodeA가 다음과 같은 인터페이스 매핑으로 구성되었다고 가정합니다.

eth0 → bond0.1001

eth1 → bond0.1002

eth2 → bond0.1003

화살표의 왼쪽 면은 StorageGRID 컨테이너 내에서 보는 기존 인터페이스(즉, 그리드, 관리자 및 클라이언트 네트워크 인터페이스)에 해당합니다. 화살표의 오른쪽은 동일한 물리적 인터페이스 결합에 종속된 세 개의 VLAN 인터페이스인 이러한 네트워크를 제공하는 실제 호스트 인터페이스에 해당합니다.

이제 NodeA를 Host2로 마이그레이션한다고 가정해 보겠습니다. Host2에 bond0.1001, bond0.1002 및 bond0.1003이라는 인터페이스도 있는 경우 시스템은 Host1에서와 같이 같은 이름의 인터페이스가 Host2에서 동일한 연결을 제공한다고 가정하여 이동을 허용합니다. 호스트 2에 동일한 이름의 인터페이스가 없으면 이동이 허용되지 않습니다.

여러 호스트 간에 일관된 네트워크 인터페이스 이름을 지정하는 방법은 여러 가지가 있습니다. 몇 가지 예는 를 참조하십시오."[호스트 네트워크를 구성합니다](#)"

공유 스토리지

오버헤드가 낮은 노드를 신속하게 마이그레이션하기 위해 StorageGRID 노드 마이그레이션 기능은 노드 데이터를 물리적으로 이동하지 않습니다. 대신 노드 마이그레이션은 다음과 같이 한 쌍의 익스포트 및 임포트 작업으로 수행됩니다.

- "노드 내보내기" 작업 중에 HostA에서 실행 중인 노드 컨테이너에서 소량의 영구 상태 데이터가 추출되고 해당 노드의 시스템 데이터 볼륨에 캐시됩니다. 그런 다음 HostA의 노드 컨테이너가 인스턴스화됩니다.

- "노드 가져오기" 작업 중에 HostA에 적용된 동일한 네트워크 인터페이스와 블록 스토리지 매핑을 사용하는 HostB의 노드 컨테이너가 인스턴스화됩니다. 그런 다음 캐시된 영구 상태 데이터가 새 인스턴스에 삽입됩니다.

이 작업 모드가 주어지면 마이그레이션을 허용하고 작동하기 위해서는 노드의 모든 시스템 데이터와 객체 스토리지 볼륨을 HostA와 HostB에서 액세스할 수 있어야 합니다. 또한 HostA 및 HostB에서 동일한 LUN을 참조하도록 보장된 이름을 사용하여 노드에 매핑되어야 합니다.

다음 예에서는 StorageGRID 스토리지 노드에 대한 블록 디바이스 매핑 솔루션 중 하나를 보여 줍니다. 이 경우 DM 다중 경로가 호스트에서 사용되고 별칭 필드는 모든 호스트에서 사용할 수 있는 일관되고 알기 쉬운 블록 디바이스 이름을 제공하기 위해 예 /etc/multipath.conf 사용되었습니다.

```

/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 준비(Linux)

설치 중 호스트 전체 설정이 변경되는 방식(Linux)

베어 메탈 시스템에서 StorageGRID는 호스트 전체 설정을 일부 변경합니다 `sysctl`.



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

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 RTAM
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
```

Linux를 설치합니다

모든 Linux 그리드 호스트에 StorageGRID 설치해야 합니다. 지원되는 버전 목록을 보려면 NetApp 상호 운용성 매트릭스 도구를 사용하세요.

시작하기 전에

운영 체제가 아래 나열된 StorageGRID의 최소 커널 버전 요구 사항을 충족하는지 확인합니다. 명령을 사용하여 `uname -r` 운영 체제의 커널 버전을 가져오거나 OS 공급업체에 문의하십시오.



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

RHEL

RHEL 버전	최소 커널 버전	커널 패키지 이름입니다
8.8(폐기됨)	4.18.0-477.10.1.el8_x86_64	커널 - 4.18.0-477.10.1.el8_x86_64
8.10	4.18.0-553.el8_10.x86_64	kernel-4.18.0-553.el8_10.x86_64
9.0(폐기됨)	5.14.0-70.22.1.el9_0.x86_64	커널 - 5.14.0-70.22.1.el9_0.x86_64
9.2(사용되지 않음)	5.14.0-284.11.1.el9_2.x86_64	커널 - 5.14.0-284.11.1.el9_2.x86_64
9.4 를 참조하십시오	5.14.0-427.18.1.el9_4.x86_64	커널 - 5.14.0-427.18.1.el9_4.x86_64
9.6	5.14.0-570.18.1.el9_6.x86_64	커널-5.14.0-570.18.1.el9_6.x86_64

우분투

- 참고: * Ubuntu 버전 18.04 및 20.04에 대한 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다.

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

데비안

참고: 데비안 버전 11에 대한 지원은 더 이상 사용되지 않으며 향후 릴리스에서 제거될 예정입니다.

데비안 버전	최소 커널 버전	커널 패키지 이름입니다
11(폐기됨)	5.10.0-18-AMD64	Linux-image-5.10.0-18-AMD64/stable, 현재 5.10.150-1
12	6.1.0-9-AMD64	linux-image-6.1.0-9-amd64/stable, 현재 6.1.27-1

단계

1. 배포자의 지침 또는 표준 절차에 따라 모든 물리적 또는 가상 그리드 호스트에 Linux를 설치합니다.



그래픽 데스크톱 환경을 설치하지 마세요.

- RHEL을 설치할 때 표준 Linux 설치 프로그램을 사용하는 경우, 가능한 경우 "컴퓨트 노드" 소프트웨어 구성을

선택하거나 "최소 설치" 기본 환경을 선택합니다.

- Ubuntu를 설치할 때 *표준 시스템 유틸리티*를 선택해야 합니다. Ubuntu 호스트에 대한 ssh 액세스를 활성화하려면 *OpenSSH 서버*를 선택하는 것이 좋습니다. 다른 모든 옵션은 지워진 상태로 유지됩니다.

2. RHEL의 Extras 채널을 포함하여 모든 호스트가 패키지 저장소에 액세스할 수 있는지 확인하세요.

3. 스왑이 활성화된 경우:

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



스왑을 완전히 사용하지 않도록 설정하면 성능이 크게 저하될 수 있습니다.

AppArmor 프로파일 설치(Ubuntu 및 Debian) 이해

사용자가 자체 배포된 Ubuntu 환경에서 AppArmor 필수 액세스 제어 시스템을 사용하는 경우 기본 시스템에 설치하는 패키지와 관련된 AppArmor 프로파일은 StorageGRID와 함께 설치된 해당 패키지에 의해 차단될 수 있습니다.

기본적으로 AppArmor 프로파일은 기본 운영 체제에 설치하는 패키지에 설치됩니다. StorageGRID 시스템 컨테이너에서 이러한 패키지를 실행하면 AppArmor 프로파일은 차단됩니다. DHCP, MySQL, NTP 및 tcdump 기본 패키지가 AppArmor와 충돌하고 다른 기본 패키지도 충돌할 수 있습니다.

AppArmor 프로파일을 처리할 수 있는 두 가지 옵션이 있습니다.

- StorageGRID 시스템 컨테이너의 패키지와 겹치는 기본 시스템에 설치된 패키지의 개별 프로파일을 비활성화합니다. 개별 프로파일을 비활성화하면 AppArmor가 활성화되었음을 나타내는 항목이 StorageGRID 로그 파일에 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 모두 비활성화합니다. Ubuntu 9.10 이상의 경우 Ubuntu 온라인 커뮤니티의 지침을 따릅니다 ["AppArmor를 비활성화합니다"](#). 최신 Ubuntu 버전에서는 AppArmor를 완전히 비활성화할 수 없습니다.

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

호스트 네트워크 구성(Linux)

호스트에서 Linux 설치를 완료한 후 나중에 배포할 StorageGRID 노드에 매핑하는 데 적합한 네트워크 인터페이스 세트를 준비하기 위해 몇 가지 추가 구성을 수행해야 할 수 있습니다.



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

시작하기 전에

- 를 검토했습니다. "[StorageGRID 네트워킹 지침](#)"
- 에 대한 정보를 검토했습니다. "[노드 컨테이너 마이그레이션 요구사항](#)"
- 가상 호스트를 사용하는 경우 다음을 읽어야 합니다. [MAC 주소 복제에 대한 고려 사항 및 권장 사항](#) 호스트 네트워크를 구성하기 전에.



VM을 호스트로 사용하는 경우 가상 네트워크 어댑터로 VMXNET 3을 선택해야 합니다. VMware E1000 네트워크 어댑터로 인해 특정 Linux 배포판에 배포된 StorageGRID 컨테이너의 연결 문제가 발생했습니다.

이 작업에 대해

그리드 노드는 그리드 네트워크와 선택적으로 관리자 및 클라이언트 네트워크에 액세스할 수 있어야 합니다. 호스트의 물리적 인터페이스를 각 그리드 노드의 가상 인터페이스에 연결하는 매핑을 생성하여 이 액세스를 제공합니다. 호스트 인터페이스를 생성할 때 이름을 friendly 로 사용하여 모든 호스트에 쉽게 구축하고 마이그레이션을 설정할 수 있습니다.

호스트와 하나 이상의 노드 간에 동일한 인터페이스를 공유할 수 있습니다. 예를 들어, 호스트 액세스 및 노드 관리 네트워크 액세스에 동일한 인터페이스를 사용하여 호스트 및 노드 유지 관리를 용이하게 할 수 있습니다. 호스트와 개별 노드 간에 동일한 인터페이스를 공유할 수 있지만 모두 IP 주소가 서로 달라야 합니다. IP 주소는 노드 간 또는 호스트와 노드 간에 공유할 수 없습니다.

동일한 호스트 네트워크 인터페이스를 사용하여 호스트의 모든 StorageGRID 노드에 그리드 네트워크 인터페이스를 제공하거나, 각 노드에 대해 다른 호스트 네트워크 인터페이스를 사용하거나, 둘 사이에 작업을 수행할 수 있습니다. 그러나 일반적으로 단일 노드에 대한 Grid 및 Admin Network 인터페이스와 동일한 호스트 네트워크 인터페이스를 제공하거나 한 노드에 대한 Grid Network 인터페이스와 다른 노드에 대한 Client Network 인터페이스를 제공하지 않습니다.

이 작업은 여러 가지 방법으로 완료할 수 있습니다. 예를 들어, 호스트가 가상 머신이고 각 호스트에 대해 하나 또는 두 개의 StorageGRID 노드를 구축하는 경우 하이퍼바이저에서 올바른 수의 네트워크 인터페이스를 생성하고 일대일 매핑을 사용할 수 있습니다. 운영 용도로 베어 메탈 호스트에 여러 노드를 구축하는 경우 Linux 네트워킹 스택이 VLAN 및 LACP 지원을 활용하여 내결함성 및 대역폭 공유를 제공할 수 있습니다. 다음 섹션에서는 이러한 두 가지 예에 대해 자세히 설명합니다. 이러한 예제 중 하나를 사용할 필요가 없습니다. 필요에 맞는 방법을 사용할 수 있습니다.



Bond 또는 Bridge 장치를 컨테이너 네트워크 인터페이스로 직접 사용하지 마십시오. 이렇게 하면 컨테이너 네임스페이스의 연결 및 브리지 장치와 함께 MACVLAN을 사용하는 커널 문제로 인해 노드 시작이 방지될 수 있습니다. 대신 VLAN 또는 가상 이더넷(veth) 쌍과 같은 비연결 장치를 사용하십시오. 이 디바이스를 노드 구성 파일의 네트워크 인터페이스로 지정합니다.

MAC 주소 복제에 대한 고려 사항 및 권장 사항

MAC 주소 클로닝은 컨테이너가 호스트의 MAC 주소를 사용하고 호스트는 사용자가 지정한 주소나 임의로 생성된

주소의 MAC 주소를 사용하게 합니다. 무차별 모드 네트워크 구성을 사용하지 않으려면 MAC 주소 복제를 사용해야 합니다.

MAC 클론 생성 활성화

특정 환경에서는 관리 네트워크, 그리드 네트워크 및 클라이언트 네트워크에 전용 가상 NIC를 사용할 수 있으므로 MAC 주소 클로닝을 통해 보안을 강화할 수 있습니다. 컨테이너가 호스트에 있는 전용 NIC의 MAC 주소를 사용하도록 하면 무차별 모드 네트워크 구성을 사용하지 않도록 할 수 있습니다.



MAC 주소 복제는 가상 서버 설치에 사용하기 위한 것이며 모든 물리적 어플라이언스 구성에서 제대로 작동하지 않을 수 있습니다.



MAC 클론 대상 인터페이스가 사용 중이어서 노드가 시작되지 않는 경우 노드를 시작하기 전에 링크를 "다운"으로 설정해야 할 수 있습니다. 또한 링크가 작동 중일 때 가상 환경에서 네트워크 인터페이스에서 MAC 클로닝을 방지할 수 있습니다. 노드가 MAC 주소를 설정하지 못하고 사용 중인 인터페이스로 인해 시작되는 경우 노드를 시작하기 전에 링크를 "다운"으로 설정하면 문제가 해결될 수 있습니다.

MAC 주소 복제는 기본적으로 해제되어 있으며 노드 구성 키로 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

키를 "true"로 설정하면 컨테이너가 호스트 NIC의 MAC 주소를 사용하게 됩니다. 또한 호스트는 지정된 컨테이너 네트워크의 MAC 주소를 사용합니다. 기본적으로 컨테이너 주소는 무작위로 생성된 주소이지만 노드 구성 키를 사용하여 주소를 설정한 경우 `_NETWORK_MAC` 해당 주소가 대신 사용됩니다. 호스트와 컨테이너의 MAC 주소는 항상 다릅니다.



하이퍼바이저에서 무차별 모드를 설정하지 않고 가상 호스트에서 MAC 클로닝을 활성화하면 호스트의 인터페이스를 사용하는 Linux 호스트 네트워킹이 작동하지 않을 수 있습니다.

Mac 클론 복제 활용 사례

MAC 클로닝에는 다음 두 가지 사용 사례를 고려해야 합니다.

- MAC 클론 생성이 활성화되지 않음: `_CLONE_MAC` 노드 구성 파일의 키가 설정되지 않았거나 "false"로 설정되어 있지 않으면 호스트는 호스트 NIC MAC을 사용하고 컨테이너에 StorageGRID 생성 MAC을 갖게 됩니다. `_NETWORK_MAC` 키에 주소가 설정되어 있으면 `_NETWORK_MAC` 컨테이너에 키에 지정된 주소가 `_NETWORK_MAC` 지정됩니다. 이러한 키 구성을 위해서는 무차별 모드를 사용해야 합니다.
- MAC 클론 생성 활성화: 노드 구성 파일의 키가 "true"로 설정된 경우 `_CLONE_MAC` 컨테이너에서 호스트 NIC MAC을 사용하고, 키에 MAC이 지정되지 않은 경우 호스트는 StorageGRID에서 생성된 MAC을 사용합니다. `_NETWORK_MAC` 키에 주소가 설정된 경우 `_NETWORK_MAC` 호스트는 생성된 주소가 아닌 지정된 주소를 사용합니다. 이 키 구성에서 무차별 모드를 사용해서는 안 됩니다.



MAC 주소 클로닝을 사용하지 않고 하이퍼바이저에 의해 할당된 것이 아닌 MAC 주소에 대한 데이터를 모든 인터페이스에서 수신 및 전송하도록 허용하려면 가상 스위치 및 포트 그룹 수준의 보안 속성이 Promiscuous Mode, MAC Address 변경 및 Forged 전송에 대해 *Accept* 로 설정되어 있는지 확인합니다. 가상 스위치에 설정된 값은 포트 그룹 수준의 값으로 재정의할 수 있으므로 두 위치에서 설정이 동일한지 확인합니다.

MAC 복제를 활성화하려면 ["노드 구성 파일 생성 지침"](#)참조하십시오.

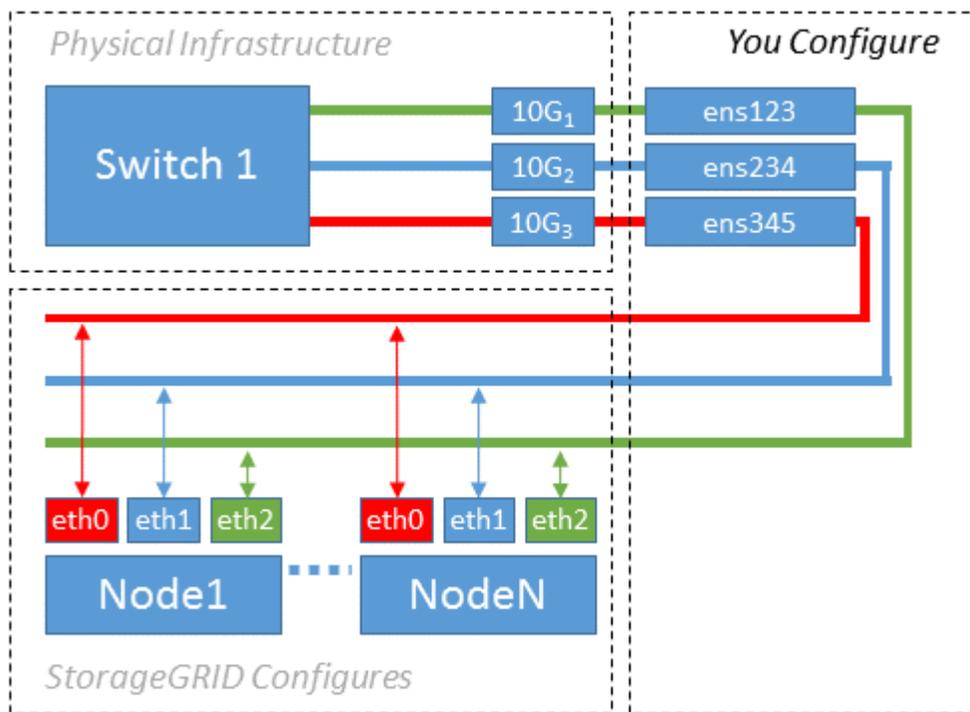
Mac 클론 복제의 예

인터페이스 ens256의 경우 MAC 주소가 11:22:33:44:55:66이고 노드 구성 파일의 경우 다음 키가 있는 호스트에서 활성화된 MAC 클론 복제의 예:

- ADMIN_NETWORK_TARGET = ens256
- ADMIN_NETWORK_MAC = b2:9c:02:c2:27:10
- ADMIN_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC = true
- 결과 *: en256의 호스트 MAC은 B2:9c:02:C2:27:10이고 관리 네트워크 MAC은 11:22:33:44:55:66입니다

예 1: 물리적 NIC 또는 가상 NIC에 1:1 대 1 매핑

예제 1에서는 호스트측 구성이 거의 또는 전혀 필요하지 않은 간단한 물리적 인터페이스 매핑에 대해 설명합니다.



Linux 운영 체제는 다음을 생성합니다. ensXYZ 설치 또는 부팅 중에 인터페이스가 자동으로 추가되거나 인터페이스가 핫 애드될 때 인터페이스가 자동으로 추가됩니다. 부팅 후 인터페이스가 자동으로 실행되도록 설정하는 것 외에는 다른 구성이 필요하지 않습니다. 나중에 구성 프로세스에서 올바른 매핑을 제공할 수 있도록 어떤 ensXYZ가 어떤 StorageGRID 네트워크(Grid, Admin 또는 Client)에 해당하는지 확인해야 합니다.

이 그림에서는 여러 StorageGRID 노드를 보여 줍니다. 그러나 일반적으로 단일 노드 VM에 이 구성을 사용합니다.

스위치 1이 물리적 스위치인 경우 액세스 모드에 대해 인터페이스 10G1 - 10G3에 연결된 포트를 구성하고 해당 VLAN에 배치해야 합니다.

예 2: VLAN을 전달하는 LACP 결합

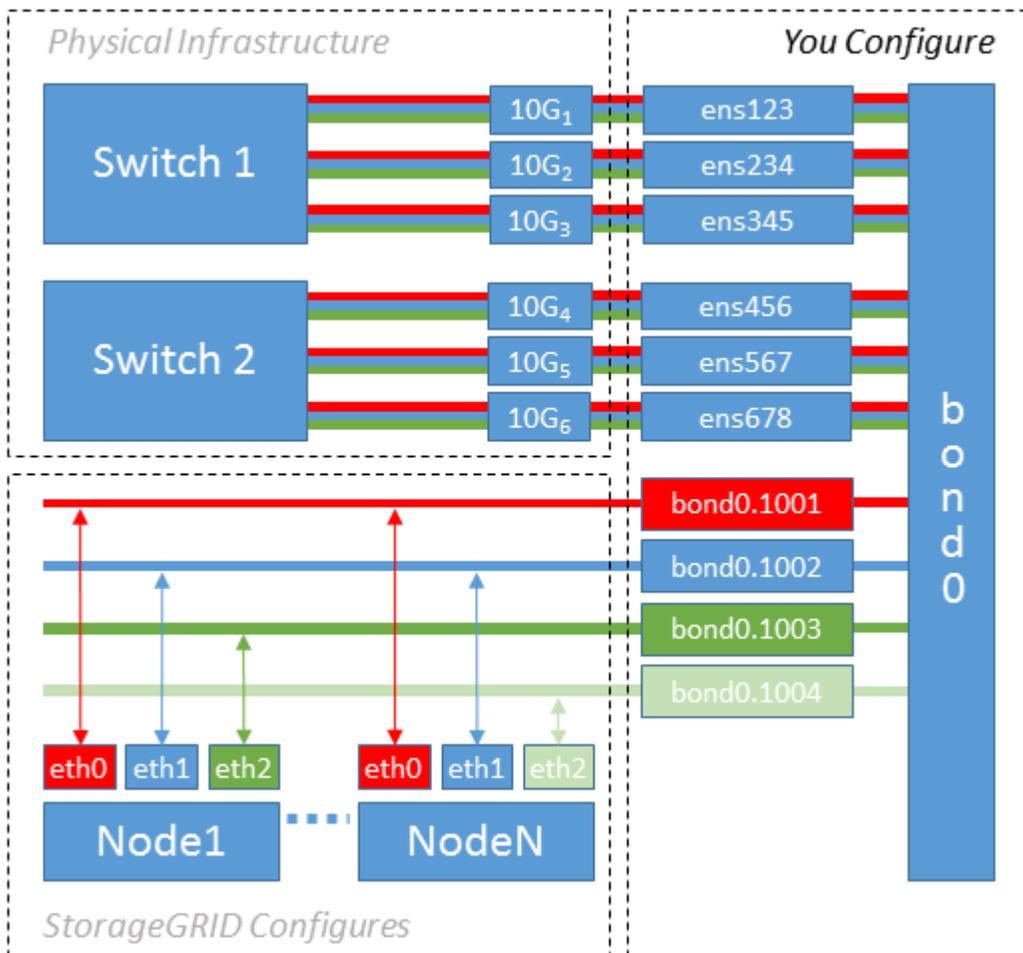
예제 2에서는 네트워크 인터페이스를 결합하거나 사용 중인 Linux 배포판에서 VLAN 인터페이스를 만드는 방법에 대해 잘 알고 있다고 가정합니다.

이 작업에 대해

예제 2에서는 단일 호스트의 모든 노드에서 사용 가능한 모든 네트워크 대역폭을 쉽게 공유할 수 있도록 지원하는 일반, 유연한 VLAN 기반 체계를 설명합니다. 이 예는 특히 베어 메탈 호스트에 적용할 수 있습니다.

이 예제를 이해하려면 각 데이터 센터에 그리드, 관리자 및 클라이언트 네트워크에 대한 세 개의 개별 서브넷이 있다고 가정합니다. 서브넷은 별도의 VLAN(1001, 1002 및 1003)에 있으며 LACP 결합 트렁크 포트(bond0)의 호스트에 제공됩니다. Bond.0.1001, bond0.1002 및 bond0.1003의 세 가지 VLAN 인터페이스를 구성합니다.

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



단계

1. StorageGRID 네트워크 연결에 사용할 모든 물리적 네트워크 인터페이스를 단일 LACP 결합으로 통합합니다.

예를 들어 모든 호스트의 본드에 동일한 이름을 사용합니다. bond0 .

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

1단계와 2단계는 네트워크 링크의 다른 끝을 종료하는 에지 스위치에 적절한 구성이 필요합니다. 에지 스위치 포트도 LACP 포트 채널로 집계되고 트렁크로 구성되어 필요한 모든 VLAN을 통과할 수 있도록 허용해야 합니다.

호스트별 네트워킹 구성 체계에 대한 샘플 인터페이스 구성 파일이 제공됩니다.

관련 정보

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

호스트 스토리지 구성(Linux)

각 Linux 호스트에 블록 스토리지 볼륨을 할당해야 합니다.

시작하기 전에

이 과제를 수행하는 데 필요한 정보를 제공하는 다음 주제를 검토했습니다.

- ["요구사항을 충족해야 합니다"](#)
- ["노드 컨테이너 마이그레이션 요구사항"](#)



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

이 작업에 대해

블록 스토리지 볼륨(LUN)을 호스트에 할당할 때 "스토리지 요구 사항"의 표를 사용하여 다음을 확인합니다.

- 각 호스트에 필요한 볼륨 수(해당 호스트에 구축할 노드 수 및 유형 기준)
- 각 볼륨의 스토리지 범주(즉, 시스템 데이터 또는 오브젝트 데이터)
- 각 볼륨의 크기입니다

호스트에 StorageGRID 노드를 배포할 때 이 정보와 Linux가 각 물리적 볼륨에 할당한 영구 이름을 사용합니다.



이러한 볼륨을 파티션, 포맷 또는 마운트할 필요가 없습니다. 호스트가 볼 수 있도록 해야 합니다.



메타데이터 전용 스토리지 노드에는 하나의 오브젝트 데이터 LUN만 필요합니다.

(`/dev/sdb`` 볼륨 이름 목록을 작성할 때 "raw" 특수 장치 파일을 사용하지 마십시오. 이러한 파일은 호스트의 재부팅 시 변경될 수 있으며, 이는 시스템의 올바른 작동에 영향을 줍니다. iSCSI LUN 및 Device Mapper Multipathing을 사용하는 경우, 특히 SAN 토폴로지에 공유 스토리지에 대한 중복 네트워크 경로가 포함되어 있는 경우 디렉토리에서 다중 경로 별칭을 사용하는 `/dev/mapper` 것이 좋습니다. 또는 영구 장치 이름에 대해 에서 시스템에서 만든 소프트링크를 사용할 수 `/dev/disk/by-path/` 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

```

각 설치 환경에 따라 결과가 달라집니다.

각 블록 스토리지 볼륨에 알기 쉬운 이름을 할당하여 초기 StorageGRID 설치 및 향후 유지 관리 절차를 간소화하십시오. 공유 스토리지 볼륨에 대한 중복 액세스를 위해 디바이스 매퍼 다중 경로 드라이버를 사용하는 경우 파일의 필드를 /etc/multipath.conf 사용할 수 alias 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
    }
    ...
}

```

이러한 방식으로 별칭 필드를 사용하면 별칭이 호스트의 디렉토리에 블록 디바이스로 나타나므로 구성 또는 유지 관리 작업에서 블록 /dev/mapper 스토리지 볼륨을 지정해야 할 때마다 쉽게 검증된 친숙한 이름을 지정할 수 있습니다.

StorageGRID 노드 마이그레이션을 지원하고 장치 매퍼 다중 경로를 사용하도록 공유 스토리지를 설정하는 경우 모든 공동 위치 호스트에 공통 을 생성하고 설치할 수 /etc/multipath.conf 있습니다. 각 호스트에서 다른 컨테이너 엔진 스토리지 볼륨을 사용해야 합니다. 별칭을 사용하고 각 컨테이너 엔진 스토리지 볼륨 LUN의 별칭에 타겟 호스트 이름을 포함시키면 기억하기 쉽고 권장됩니다.



소프트웨어 전용 배포를 위한 컨테이너 엔진으로 Docker에 대한 지원은 더 이상 사용되지 않습니다. Docker는 향후 릴리즈에서 다른 컨테이너 엔진으로 대체될 예정입니다.

관련 정보

- ["컨테이너 엔진 저장소 볼륨을 구성합니다"](#)
- ["요구사항을 충족해야 합니다"](#)

- "노드 컨테이너 마이그레이션 요구사항"

컨테이너 엔진 스토리지 볼륨 구성(Linux)

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



소프트웨어 전용 배포를 위한 컨테이너 엔진으로 Docker에 대한 지원은 더 이상 사용되지 않습니다. Docker는 향후 릴리즈에서 다른 컨테이너 엔진으로 대체될 예정입니다.



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

이 작업에 대해

Docker 또는 Podman 스토리지 볼륨에 루트 볼륨을 사용하려는 경우 다음이 포함된 호스트 파티션에 충분한 공간이 있는 경우 이 단계를 건너뛸 수 있습니다.

- 포드만: /var/lib/containers
- 도커: /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 저장소 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/fstab에 추가합니다.

- RHEL: 컨테이너-스토리지-볼륨-장치
- Ubuntu 또는 Debian: docker-storage-volume-device

이 단계를 수행하면 호스트가 재부팅된 후 스토리지 볼륨이 자동으로 다시 마운트됩니다.

Docker를 설치합니다

StorageGRID 시스템은 컨테이너 컬렉션으로 Linux에서 실행될 수 있습니다.

- Ubuntu 또는 Debian에 StorageGRID 설치하려면 먼저 Docker를 설치해야 합니다.
- Docker 컨테이너 엔진을 사용하기로 선택한 경우 다음 단계에 따라 Docker를 설치하세요. 그렇지 않으면, [Podman을 설치합니다](#).



소프트웨어 전용 배포를 위한 컨테이너 엔진으로 Docker에 대한 지원은 더 이상 사용되지 않습니다. Docker는 향후 릴리즈에서 다른 컨테이너 엔진으로 대체될 예정입니다.

단계

1. Linux 배포에 대한 지침에 따라 Docker를 설치합니다.



Docker가 Linux 배포판에 포함되어 있지 않은 경우 Docker 웹 사이트에서 다운로드할 수 있습니다.

2. 다음 두 명령을 실행하여 Docker를 활성화하고 시작했는지 확인합니다.

```
sudo systemctl enable docker
```

```
sudo systemctl start docker
```

3. 다음을 입력하여 예상 버전의 Docker를 설치했는지 확인합니다.

```
sudo docker version
```

클라이언트 및 서버 버전은 1.11.0 이상이어야 합니다.

Podman을 설치합니다

StorageGRID 시스템은 컨테이너의 컬렉션으로 실행됩니다. Podman 컨테이너 엔진을 사용하기로 선택한 경우 다음 단계에 따라 Podman을 설치하세요. 그렇지 않으면, [Docker를 설치합니다](#).

단계

1. Linux 배포에 대한 지침에 따라 Podman 및 Podman-Docker를 설치합니다.



Podman을 설치할 때 Podman-docker 패키지도 설치해야 합니다.

2. 다음을 입력하여 예상 버전의 Podman 및 Podman-Docker를 설치했는지 확인합니다.

```
sudo docker version
```



Podman-Docker 패키지를 사용하면 Docker 명령을 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 스토리지를 구성합니다"](#)

StorageGRID 호스트 서비스 설치(Linux)

StorageGRID 호스트 서비스를 설치하려면 운영 체제 유형에 맞는 StorageGRID 패키지를 사용합니다.



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

RHEL

StorageGRID RPM 패키지를 사용하여 StorageGRID 호스트 서비스를 설치합니다.

이 작업에 대해

다음 지침은 RPM 패키지에서 호스트 서비스를 설치하는 방법을 설명합니다. 또는 설치 아카이브에 포함된 DNF 리포지토리 메타데이터를 사용하여 RPM 패키지를 원격으로 설치할 수 있습니다. Linux 운영 체제에 대한 DNF 리포지토리 지침을 참조하십시오.

단계

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

예를 들어, /tmp 다음 단계에서 예제 명령을 사용할 수 있도록 디렉토리에 배치합니다.

2. 각 호스트에 루트로 로그인하거나 sudo 권한이 있는 계정을 사용하여 다음 명령을 지정된 순서대로 실행합니다.

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

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



먼저 이미지 패키지를 설치하고 서비스 패키지를 다시 설치해야 합니다.



패키지를 이외의 디렉토리에 배치한 경우 /tmp 사용한 경로를 반영하도록 명령을 수정합니다.

Ubuntu 또는 Debian

StorageGRID DEB 패키지를 사용하면 Ubuntu 또는 Debian용 StorageGRID 호스트 서비스를 설치할 수 있습니다.

이 작업에 대해

다음 지침은 DEB 패키지에서 호스트 서비스를 설치하는 방법을 설명합니다. 또는 설치 아카이브에 포함된 APT 리포지토리 메타데이터를 사용하여 DEB 패키지를 원격으로 설치할 수 있습니다. Linux 운영 체제에 대한 APT 리포지토리 지침을 참조하십시오.

단계

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

예를 들어, /tmp 다음 단계에서 예제 명령을 사용할 수 있도록 디렉토리에 배치합니다.

2. 각 호스트에 루트로 로그인하거나 sudo 권한이 있는 계정을 사용하여 다음 명령을 실행합니다.

먼저 패키지를 service 설치하고 패키지를 두 번째로 설치해야 images 합니다. 패키지를 이외의 디렉토리에 배치한 경우 /tmp 사용한 경로를 반영하도록 명령을 수정합니다.

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

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



StorageGRID 패키지를 설치하려면 먼저 Python 3이 설치되어 있어야 합니다. 그만큼 `sudo dpkg --install /tmp/storagegrid-webscale-images-version-SHA.deb` 그렇게 하지 않으면 명령은 실패할 것입니다.

소프트웨어 기반 노드 설치 자동화

설치 자동화(Linux)

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



이 작업에 대해

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

구축 자동화는 다음 경우에 유용할 수 있습니다.

- 이미 Ansible, Puppet 또는 Chef와 같은 표준 오케스트레이션 프레임워크를 사용하여 물리적 호스트 또는 가상 호스트를 구축 및 구성합니다.
- 여러 StorageGRID 인스턴스를 배포하려고 합니다.
- 크고 복잡한 StorageGRID 인스턴스를 구축하고 있습니다.

StorageGRID 호스트 서비스는 패키지에 의해 설치되며 구성 파일에 의해 구동됩니다. 다음 방법 중 하나를 사용하여 구성 파일을 만들 수 있습니다.

- "[구성 파일을 만듭니다](#)" 수동 설치 중에 대화형으로 작동합니다.
- 이 문서에 설명되어 있는 대로 표준 오케스트레이션 프레임워크를 사용하여 자동 설치를 수행할 수 있도록 구성 파일을 사전에(또는 프로그래밍 방식으로) 준비합니다.

StorageGRID는 StorageGRID 어플라이언스 및 전체 StorageGRID 시스템("그리드")의 구성을 자동화하기 위한 선택적 Python 스크립트를 제공합니다. 이러한 스크립트를 직접 사용하거나 직접 개발한 그리드 배포 및 구성 도구의 사용 방법을 알아보기 위해 해당 스크립트를 검사할 수 "[StorageGRID 설치 REST API](#)" 있습니다.

StorageGRID 호스트 서비스의 설치 및 구성을 자동화합니다

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

StorageGRID 호스트 서비스는 DEB(Ubuntu 또는 Debian) 또는 RPM(RHEL)으로 패키징되어 있으며, 미리(또는

프로그래밍 방식으로) 준비하여 자동 설치를 활성화할 수 있는 구성 파일에 의해 구동됩니다. 이미 표준 오케스트레이션 프레임워크를 사용하여 Linux 배포를 설치하고 구성하고 있다면 플레이북이나 레시피에 StorageGRID 추가하는 것은 간단할 것입니다.

호스트 준비 및 가상 그리드 노드 구축을 위한 모든 단계를 자동화할 수 있습니다.

Ansible 역할 및 플레이북 예

예제 Ansible 역할 및 플레이북은 폴더에 설치 아카이브와 함께 /extras 제공됩니다. Ansible 플레이북에서는 역할이 호스트를 준비하고 타겟 서버에 StorageGRID를 설치하는 방법을 보여줍니다 storagegrid. 필요에 따라 역할 또는 플레이북을 사용자 지정할 수 있습니다.



예제 플레이북에는 StorageGRID 호스트 서비스를 시작하기 전에 네트워크 디바이스를 생성하는 데 필요한 단계가 포함되어 있지 않습니다. 이 단계를 추가한 후 플레이북을 사용하여 작업을 완료합니다.

RHEL

RHEL의 경우 제공된 설치 작업은 다음과 같습니다. storagegrid 역할 예제 사용 ansible.builtin.dnf 로컬 RPM 파일이나 원격 Yum 저장소에서 설치를 수행하는 모듈입니다. 모듈을 사용할 수 없거나 지원되지 않는 경우 다음 파일에서 적절한 Ansible 작업을 편집하여 사용해야 할 수 있습니다. yum 또는 ansible.builtin.yum 기준 치수:

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

Ubuntu 또는 Debian

Ubuntu 또는 Debian의 경우 제공된 설치 작업 storagegrid 역할 예제 사용 ansible.builtin.apt 로컬 DEB 파일이나 원격 apt 저장소에서 설치를 수행하는 모듈입니다. 모듈을 사용할 수 없거나 지원되지 않는 경우 다음 파일에서 적절한 Ansible 작업을 편집하여 사용해야 할 수 있습니다. ansible.builtin.apt 기준 치수:

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

StorageGRID의 구성을 자동화합니다

그리드 노드를 구축한 후 StorageGRID 시스템 구성을 자동화할 수 있습니다.

시작하기 전에

- 설치 아카이브에서 다음 파일의 위치를 알고 있습니다.

파일 이름	설명
configure-storagegrid.py	구성을 자동화하는 데 사용되는 Python 스크립트입니다
configure -StorageGrid.sample.json	스크립트와 함께 사용할 예제 구성 파일
configure -StorageGrid.blank.json을 지정합니다	스크립트에 사용할 빈 구성 파일입니다

- `configure-storagegrid.json` 구성 파일을 만들었습니다. 이 파일을 작성하려면 예제 구성 파일 (`configure-storagegrid.sample.json`)이나 빈 구성 파일을 수정할 수 (`configure-storagegrid.blank.json`) 있습니다.



수정된 암호 섹션에서 관리 암호와 프로비저닝 암호를 저장합니다. `configure-storagegrid.json` 구성 파일을 안전한 위치에 보관하세요. 이러한 비밀번호는 설치, 확장 및 유지 관리 절차에 필요합니다. 수정된 내용도 백업해야 합니다. `configure-storagegrid.json` 구성 파일을 만들어 안전한 위치에 저장하세요.

이 작업에 대해

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



그리드 관리자 또는 설치 API를 사용하여 시스템을 구성할 수도 있습니다.

단계

1. Python 스크립트를 실행하기 위해 사용 중인 Linux 시스템에 로그인합니다.
2. 설치 아카이브를 추출한 디렉토리로 변경합니다.

예를 들면 다음과 같습니다.

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

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

3. Python 스크립트를 실행하고 생성한 구성 파일을 사용합니다.

예를 들면 다음과 같습니다.

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

결과

복구 패키지 `.zip` 파일은 구성 프로세스 중에 생성되며 설치 및 구성 프로세스를 실행 중인 디렉터리에 다운로드됩니다. 하나 이상의 그리드 노드에 장애가 발생할 경우 StorageGRID 시스템을 복구할 수 있도록 복구 패키지 파일을 백업해야 합니다. 예를 들어, 안전한 백업 네트워크 위치 및 안전한 클라우드 저장소 위치에 복사합니다.



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

임의의 암호를 생성하도록 지정한 경우 파일을 열고 `Passwords.txt` StorageGRID 시스템에 액세스하는 데 필요한 암호를 찾습니다.

```
#####
##### 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.           #####
#####
```

확인 메시지가 표시되면 StorageGRID 시스템이 설치 및 구성됩니다.

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

설치 자동화(VMware)

VMware OVF Tool을 사용하여 그리드 노드의 구축을 자동화할 수 있습니다. StorageGRID의 구성을 자동화할 수도 있습니다.

그리드 노드 구축을 자동화합니다

VMware OVF Tool을 사용하여 그리드 노드의 구축을 자동화합니다.

시작하기 전에

- Bash 3.2 이상이 설치된 Linux/Unix 시스템에 액세스할 수 있습니다.
- vCenter가 포함된 VMware vSphere를 사용하고 있습니다
- VMware OVF Tool을 설치하고 올바르게 구성했습니다.
- OVF 툴을 사용하여 VMware vSphere에 액세스하기 위한 사용자 이름과 암호를 알고 있습니다
- OVF 파일에서 VM을 배포하고 전원을 켤 수 있는 충분한 권한과 VM에 연결할 추가 볼륨을 생성할 수 있는 권한이 있습니다. `ovftool` 자세한 내용은 설명서를 참조하십시오.
- StorageGRID 가상 머신을 구축할 vSphere의 위치에 대한 VI(가상 인프라) URL을 알고 있습니다. 이 URL은 일반적으로 vApp 또는 리소스 풀이 됩니다. 예를 들면 다음과 같습니다.

```
vi://vcenter.example.com/vi/sgws
```



VMware 유틸리티를 사용하여 이 값을 확인할 수 ovftool 있습니다(자세한 내용은 설명서 참조 ovftool).



vApp에 구축할 경우 가상 머신이 처음 시작될 때 자동으로 시작되지 않으며 수동으로 전원을 설정해야 합니다.

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

파일 이름	설명
NetApp-SG-version-SHA.vmdk입니다	그리드 노드 가상 머신을 생성하기 위한 템플릿으로 사용되는 가상 머신 디스크 파일입니다. • 참고: * 이 파일은 및 .mf 파일과 동일한 폴더에 있어야 .ovf 합니다.
vsphere-primary-admin.ovf vsphere-primary-admin.mf	Open Virtualization Format 템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 기본 관리자 노드를 배포할 수 있습니다.
vsphere - non-primary-admin.ovf vsphere - non-primary-admin.mf	템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 비기본 관리 노드를 배포합니다.
vsphere-gateway.ovf vsphere-gateway.mf	템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 게이트웨이 노드를 배포할 수 있습니다.
vsphere-storage.ovf vsphere-storage.mf	템플릿 파일(.ovf) 및 매니페스트 파일(.mf)을 사용하여 가상 머신 기반 스토리지 노드를 구축합니다.
deploy-vsphere-ovftool.sh	가상 그리드 노드의 배포를 자동화하는 데 사용되는 Bash 셸 스크립트입니다.
deploy-vsphere-ovftool-sample.ini	스크립트와 함께 사용할 예제 구성 deploy-vsphere-ovftool.sh 파일입니다.

배포를 위한 구성 파일을 정의합니다

Bash 스크립트에서 사용하는 구성 파일에 StorageGRID용 가상 그리드 노드를 배포하는 데 필요한 정보를 지정합니다 deploy-vsphere-ovftool.sh. 예제 구성 파일을 수정하여 처음부터 새로 만들 필요가 없도록 할 수 있습니다.

단계

1. 예제 구성 파일의 복사본을 (deploy-vsphere-ovftool.sample.ini`만듭니다. 새 파일을 `deploy-vsphere-ovftool.ini 와 같은 디렉토리에 deploy-vsphere-ovftool.sh 저장합니다.
2. `deploy-vsphere-ovftool.ini`개방형:
3. VMware 가상 그리드 노드를 구축하는 데 필요한 모든 정보를 입력합니다.

자세한 내용은 [을 구성 파일 설정](#)참조하십시오.

4. 필요한 모든 정보를 입력하고 확인했으면 파일을 저장하고 닫습니다.

구성 파일 설정

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

구성 파일은 먼저 글로벌 매개 변수를 나열한 다음 노드 이름으로 정의된 섹션에 노드별 매개 변수를 나열합니다. 파일이 사용되는 경우:

- `_Global parameters_`는 모든 그리드 노드에 적용됩니다.
- `_노드별 parameters_override` 전역 매개 변수입니다.

전역 매개 변수

전역 매개 변수는 개별 섹션의 설정에 의해 재정의되지 않는 한 모든 그리드 노드에 적용됩니다. 전역 매개 변수 섹션에서 여러 노드에 적용되는 매개 변수를 배치한 다음 필요에 따라 개별 노드의 섹션에서 이러한 설정을 재정의합니다.

- `* OVFTOOL_Arguments *`: `OVFTOOL_Arguments`를 전역 설정으로 지정하거나 특정 노드에 인수를 개별적으로 적용할 수 있습니다. 예를 들면 다음과 같습니다.

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

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



노드를 서로 다른 데이터 저장소에 배포하고 전역이 아닌 각 노드에 대해 `OVFTOOL_MOUMENT`를 지정해야 합니다.

- `* 소스 *`: StorageGRID 가상 머신 템플릿의 경로(`.vmdk`) 파일 및 `.mf` 개별 그리드 노드의 파일과 `.ovf` 파일. 이 기본값은 현재 디렉터리입니다.

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

- `* 타겟 *`: StorageGRID를 구축할 위치의 VMware vSphere 가상 인프라(vi) URL입니다. 예를 들면 다음과 같습니다.

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

- `* GRID_NETWORK_CONFIG *`: 고정 또는 DHCP 중 IP 주소를 획득하는 데 사용되는 방법입니다. 기본값은 정적입니다. 모든 노드 또는 대부분의 노드가 동일한 방법으로 IP 주소를 획득하는 경우 여기에서 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 여기에서 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 마스크. 모든 노드 또는 대부분의 노드가 동일한 네트워크 마스크를 사용하는 경우 여기에서 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 게이트웨이. 모든 노드 또는 대부분의 노드가 동일한 네트워크 게이트웨이를 사용하는 경우 여기에서 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 사항. Grid Network의 MTU(Maximum Transmission Unit)입니다. 지정된 경우 값은 1280에서 9216 사이여야 합니다. 예를 들면 다음과 같습니다.

```
GRID_NETWORK_MTU = 9000
```

생략하면 1400이 사용됩니다.

점보 프레임을 사용하려면 MTU를 9000과 같은 점보 프레임에 적합한 값으로 설정합니다. 그렇지 않으면 기본값을 유지합니다.



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



최상의 네트워크 성능을 얻으려면 모든 노드를 그리드 네트워크 인터페이스에서 유사한 MTU 값으로 구성해야 합니다. 개별 노드의 그리드 네트워크에 대한 MTU 설정에 상당한 차이가 있을 경우 * Grid Network MTU mismatch * 경고가 트리거됩니다. MTU 값은 모든 네트워크 유형에 대해 같을 필요는 없습니다.

- *** admin_network_Config** *: IP 주소를 획득하는 데 사용되는 방법으로, 비활성화, 정적 또는 DHCP입니다. 기본값은 사용 안 함으로 설정되어 있습니다. 모든 노드 또는 대부분의 노드가 동일한 방법으로 IP 주소를 획득하는 경우 여기에서 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** *: 관리 네트워크에 사용할 기존 VMware 네트워크의 이름입니다. 이 설정은 관리 네트워크를 사용하지 않는 한 필요합니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 이름을 사용하는 경우 여기에서 지정할 수 있습니다. 그리드 네트워크와 달리 모든 노드는 동일한 관리 네트워크에 연결할 필요가 없습니다. 그런 다음 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` *: 관리 네트워크의 네트워크 마스크입니다. 이 설정은 고정 IP 주소 지정을 사용하는 경우에 필요합니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 마스크를 사용하는 경우 여기에서 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. 이 설정은 정적 IP 주소 지정을 사용하고 `admin_network_ESL` 설정에서 외부 서브넷을 지정하는 경우에 필요합니다. (즉, `admin_network_esl`이 비어 있으면 필요하지 않습니다.) 모든 노드 또는 대부분의 노드가 동일한 네트워크 게이트웨이를 사용하는 경우 여기에서 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` *: 심표로 구분된 CIDR 라우트 대상 목록으로 지정된 관리 네트워크의 외부 서브넷 목록 (라우트). 모든 노드 또는 대부분의 노드가 동일한 외부 서브넷 목록을 사용하는 경우 여기에서 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 사항. 관리자 네트워크의 MTU(Maximum Transmission Unit) `admin_network_Config=DHCP`인지 지정하지 마십시오. 지정된 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1400이 사용됩니다. 정보 프레임을 사용하려면 MTU를 9000과 같은 정보 프레임에 적합한 값으로 설정합니다. 그렇지 않으면 기본값을 유지합니다. 모든 노드 또는 대부분의 노드가 Admin Network에 대해 동일한 MTU를 사용하는 경우 여기에서 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 획득하는 데 사용되는 방법으로, 비활성화, 정적 또는 DHCP입니다. 기본값은 사용 안 함으로 설정되어 있습니다. 모든 노드 또는 대부분의 노드가 동일한 방법으로 IP 주소를 획득하는 경우 여기에서 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. 이 설정은 클라이언트 네트워크를 사용하지 않는 경우에만 필요합니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 이름을 사용하는 경우 여기에서 지정할 수 있습니다. 그리드 네트워크와 달리 모든 노드는 동일한 클라이언트

네트워크에 연결할 필요가 없습니다. 그런 다음 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. 이 설정은 고정 IP 주소 지정을 사용하는 경우에 필요합니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 마스크를 사용하는 경우 여기에서 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. 이 설정은 고정 IP 주소 지정을 사용하는 경우에 필요합니다. 모든 노드 또는 대부분의 노드가 동일한 네트워크 게이트웨이를 사용하는 경우 여기에서 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 사항. 클라이언트 네트워크의 MTU(Maximum Transmission Unit) client_network_Config = DHCP인지 지정하지 마십시오. 지정된 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1400이 사용됩니다. 점보 프레임을 사용하려면 MTU를 9000과 같은 점보 프레임에 적합한 값으로 설정합니다. 그렇지 않으면 기본값을 유지합니다. 모든 노드 또는 대부분의 노드가 클라이언트 네트워크에 동일한 MTU를 사용하는 경우 여기에서 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 *: 내부 그리드 노드 통신 또는 외부 통신을 위해 노드에서 사용하는 포트를 다시 매핑합니다. 엔터프라이즈 네트워킹 정책이 StorageGRID에서 사용하는 하나 이상의 포트를 제한하는 경우 포트를 다시 매핑해야 합니다. StorageGRID에서 사용하는 포트 목록은 의 내부 그리드 노드 통신 및 외부 통신 을 참조하십시오"[네트워킹 지침](#)".



로드 밸런서 엔드포인트를 구성하는 데 사용할 포트를 다시 매핑하지 마십시오.



port_remap 만 설정된 경우 지정하는 매핑이 인바운드 및 아웃바운드 통신 모두에 사용됩니다. port_remap_inbound 도 지정된 경우 port_remap 은 아웃바운드 통신에만 적용됩니다.

사용되는 형식은 다음과 같습니다 *network type/protocol/default port used by grid node/new port*. 여기서 네트워크 유형은 그리드, 관리자 또는 클라이언트이고 프로토콜은 TCP 또는 UDP입니다.

예를 들면 다음과 같습니다.

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

단독으로 사용하는 경우 이 예제 설정은 그리드 노드에 대한 인바운드 및 아웃바운드 통신을 포트 18082에서 포트 443으로 대칭적으로 매핑합니다. `port_remap_inbound` 와 함께 사용할 경우 이 예제 설정은 포트 18082에서 포트 443으로 아웃바운드 통신을 매핑합니다.

쉼표로 구분된 목록을 사용하여 여러 포트를 다시 매핑할 수도 있습니다.

예를 들면 다음과 같습니다.

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

- * `port_remap_inbound` *: 지정된 포트에 대한 인바운드 통신을 다시 매핑합니다. `port_remap_inbound` 를 지정하지만 `port_remap` 의 값을 지정하지 않으면 포트의 아웃바운드 통신이 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 그리드, 관리자 또는 클라이언트이고 프로토콜은 TCP 또는 UDP입니다.

예를 들면 다음과 같습니다.

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

이 예에서는 포트 443으로 전송된 트래픽을 내부 방화벽을 통과하여 그리드 노드가 S3 요청을 수신하는 포트 18082로 전달합니다.

쉼표로 구분된 목록을 사용하여 여러 인바운드 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 합류하기 전에 VM 콘솔이나 StorageGRID 설치 API에 액세스하거나 SSH를 사용할 때 사용되는 임시 설치 암호의 유형입니다.



모든 노드 또는 대부분의 노드가 동일한 유형의 임시 설치 암호를 사용하는 경우 전역 매개 변수 섹션에 형식을 지정합니다. 그런 다음 필요에 따라 개별 노드에 대해 다른 설정을 사용합니다. 예를 들어, * 사용자 정의 암호 사용 * 을 전역적으로 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.

- * 비밀번호 비활성화 * : 임시 설치 비밀번호가 사용되지 않습니다. 설치 문제를 디버깅하기 위해 VM에 액세스해야 하는 경우 를 참조하십시오"[설치 문제를 해결합니다](#)".
- * 사용자 정의 암호 사용 * : * custom_temporary_password=<password> * 에 제공된 값은 임시 설치 암호로 사용되며 VM 콘솔, StorageGRID 설치 API 및 SSH에 대한 액세스를 제공합니다.



필요한 경우 * Temporary_password_type * 매개 변수를 생략하고 * custom_Temporary_password=<password> * 만 지정할 수 있습니다.

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

노드별 매개 변수

각 노드는 구성 파일의 자체 섹션에 있습니다. 각 노드에는 다음과 같은 설정이 필요합니다.

- 섹션 헤드는 그리드 관리자에 표시될 노드 이름을 정의합니다. 노드에 대해 선택 사항인 node_name 매개 변수를 지정하여 이 값을 재정의할 수 있습니다.
- * node_type *: VM_Admin_Node, VM_Storage_Node 또는 VM_API_Gateway_Node
- * STORAGE_TYPE *: 조합, 데이터 또는 메타데이터. 스토리지 노드의 이 선택적 매개 변수는 지정되지 않은 경우 기본적으로 Combined(데이터 및 메타데이터)로 설정됩니다. 자세한 내용은 을 "[스토리지 노드 유형](#)"참조하십시오.
- * GRID_NETWORK_IP *: 그리드 네트워크의 노드에 대한 IP 주소입니다.
- * admin_network_ip *: 관리 네트워크의 노드에 대한 IP 주소입니다. 노드가 Admin Network에 연결되어 있고 admin_network_Config가 static으로 설정된 경우에만 필요합니다.
- * client_network_ip *: 클라이언트 네트워크의 노드에 대한 IP 주소입니다. 노드가 클라이언트 네트워크에 연결되어 있고 이 노드의 client_network_Config가 static으로 설정된 경우에만 필요합니다.
- * admin_IP *: 그리드 네트워크의 기본 관리 노드에 대한 IP 주소입니다. 기본 관리 노드에 대해 GRID_NETWORK_IP로 지정하는 값을 사용합니다. 이 매개 변수를 생략하면 노드가 mDNS를 사용하여 운영 관리 노드 IP를 검색합니다. 자세한 내용은 을 "[그리드 노드가 기본 관리자 노드를 검색하는 방법](#)"참조하십시오.



admin_ip 매개 변수는 기본 관리 노드에 대해 무시됩니다.

- 전역적으로 설정되지 않은 모든 매개변수. 예를 들어, 노드가 관리 네트워크에 연결되어 있고 admin_network 매개 변수를 전역적으로 지정하지 않은 경우 노드에 대해 이러한 매개 변수를 지정해야 합니다.

기본 관리자 노드

기본 관리 노드에는 다음과 같은 추가 설정이 필요합니다.

- * node_type *: vm_Admin_Node
- * admin_role *: 기본

이 예제 항목은 세 네트워크 모두에 있는 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
```

기본 관리 노드에는 다음과 같은 추가 설정이 선택 사항입니다.

- * 디스크 *: 기본적으로 감사 및 데이터베이스 사용을 위해 관리자 노드에 두 개의 추가 200GB 하드 디스크가 할당됩니다. disk 매개 변수를 사용하여 이러한 설정을 늘릴 수 있습니다. 예를 들면 다음과 같습니다.

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



관리 노드의 경우 인스턴스는 항상 2가 되어야 합니다.

스토리지 노드

스토리지 노드에는 다음과 같은 추가 설정이 필요합니다.

- * node_type *: vm_storage_Node

이 예제 항목은 그리드 및 관리 네트워크에 있지만 클라이언트 네트워크에 없는 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
```

이 두 번째 예제 항목은 고객의 엔터프라이즈 네트워킹 정책에서 포트 80 또는 443을 사용하여 스토리지 노드에 액세스할 수 있다고 명시하는 클라이언트 네트워크의 스토리지 노드에 대한 것입니다. 예제 구성 파일은 port_remap을 사용하여 스토리지 노드가 포트 443에서 S3 메시지를 보내고 받을 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 예에서는 포트 22에서 포트 3022로 ssh 트래픽에 대한 대칭 재매핑을 생성하지만 인바운드 및 아웃바운드 트래픽에 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 대한 다음 추가 설정은 선택 사항입니다.

- *** disk *:** 기본적으로 스토리지 노드에는 RangeDB 사용을 위해 3 개의 4TB 디스크가 할당됩니다. 디스크 매개 변수를 사용하여 이러한 설정을 늘릴 수 있습니다. 예를 들면 다음과 같습니다.

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

- *** STORAGE_TYPE *:** 기본적으로 모든 새 스토리지 노드는 오브젝트 데이터와 메타데이터를 모두 저장하도록 구성되어 있습니다. `storage_type` 매개 변수를 사용하여 데이터나 메타데이터만 저장하도록 스토리지 노드 유형을 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 예제 항목은 세 네트워크 모두에서 게이트웨이 노드의 예입니다. 이 예에서는 구성 파일의 전역 섹션에 클라이언트 네트워크 매개 변수가 지정되지 않아 노드에 대해 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
```

운영 관리자 노드가 아닌 노드

운영 관리자 노드가 아닌 경우 다음과 같은 추가 설정이 필요합니다.

- * node_type *: vm_Admin_Node
- * admin_role *: Non-Primary

이 예제 항목은 클라이언트 네트워크에 없는 비 기본 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 운영 관리자 노드가 아닌 경우 선택 사항입니다.

- * 디스크 *: 기본적으로 감사 및 데이터베이스 사용을 위해 관리자 노드에 두 개의 추가 200GB 하드 디스크가 할당됩니다. disk 매개 변수를 사용하여 이러한 설정을 늘릴 수 있습니다. 예를 들면 다음과 같습니다.

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



관리 노드의 경우 인스턴스는 항상 2가 되어야 합니다.

Bash 스크립트를 실행합니다

VMware vSphere에서 StorageGRID 노드의 배포를 자동화하기 위해 수정한 Bash 스크립트와 deploy-vsphere-ovftool.ini 구성 파일을 사용할 수 deploy-vsphere-ovftool.sh 있습니다.

시작하기 전에

사용자 환경에 대한 deploy-vsphere-ovftool.ini 구성 파일을 만들었습니다.

도움말 명령을 입력하여 Bash 스크립트에서 사용할 수 있는 도움말을 사용할 수 (`-h/--help` 있습니다.) 예를 들면 다음과 같습니다.

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

또는

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

단계

1. Bash 스크립트를 실행하기 위해 사용 중인 Linux 시스템에 로그인합니다.
2. 설치 아카이브를 추출한 디렉토리로 변경합니다.

예를 들면 다음과 같습니다.

```
cd StorageGRID-Webscale-version/vsphere
```

3. 모든 그리드 노드를 배포하려면 환경에 적합한 옵션을 사용하여 Bash 스크립트를 실행합니다.

예를 들면 다음과 같습니다.

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

4. 오류로 인해 그리드 노드를 배포하지 못한 경우 오류를 해결하고 해당 노드에 대해서만 Bash 스크립트를 다시 실행합니다.

예를 들면 다음과 같습니다.

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

각 노드의 상태가 "Passed"이면 배포가 완료됩니다.

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

StorageGRID의 구성을 자동화합니다

그리드 노드를 구축한 후 StorageGRID 시스템 구성을 자동화할 수 있습니다.

시작하기 전에

- 설치 아카이브에서 다음 파일의 위치를 알고 있습니다.

파일 이름	설명
configure-storagegrid.py	구성을 자동화하는 데 사용되는 Python 스크립트입니다
configure -StorageGrid.sample.json	스크립트와 함께 사용할 예제 구성 파일
configure -StorageGrid.blank.json을 지정합니다	스크립트에 사용할 빈 구성 파일입니다

- `configure-storagegrid.json`` 구성 파일을 만들었습니다. 이 파일을 작성하려면 예제 구성 파일 (`configure-storagegrid.sample.json`)이나 빈 구성 파일을 수정할 수 (`configure-storagegrid.blank.json`` 있습니다.



수정된 암호 섹션에서 관리 암호와 프로비저닝 암호를 저장합니다. `configure-storagegrid.json` 구성 파일을 안전한 위치에 보관하세요. 이러한 비밀번호는 설치, 확장 및 유지 관리 절차에 필요합니다. 수정된 내용도 백업해야 합니다. `configure-storagegrid.json` 구성 파일을 만들어 안전한 위치에 저장하세요.

이 작업에 대해

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



그리드 관리자 또는 설치 API를 사용하여 시스템을 구성할 수도 있습니다.

단계

1. Python 스크립트를 실행하기 위해 사용 중인 Linux 시스템에 로그인합니다.
2. 설치 아카이브를 추출한 디렉토리로 변경합니다.

예를 들면 다음과 같습니다.

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

여기서 platform 는 debs, rpms 또는 vSphere입니다.

3. Python 스크립트를 실행하고 생성한 구성 파일을 사용합니다.

예를 들면 다음과 같습니다.

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

결과

복구 패키지 .zip 파일은 구성 프로세스 중에 생성되며 설치 및 구성 프로세스를 실행 중인 디렉터리에 다운로드됩니다. 하나 이상의 그리드 노드에 장애가 발생할 경우 StorageGRID 시스템을 복구할 수 있도록 복구 패키지 파일을 백업해야 합니다. 예를 들어, 안전한 백업 네트워크 위치 및 안전한 클라우드 저장소 위치에 복사합니다.



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

임의의 암호를 생성하도록 지정한 경우 파일을 열고 Passwords.txt StorageGRID 시스템에 액세스하는 데 필요한 암호를 찾습니다.

```
#####
##### 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. #####
#####
```

확인 메시지가 표시되면 StorageGRID 시스템이 설치 및 구성됩니다.

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

관련 정보

- ["그리드 관리자로 이동합니다"](#)
- ["REST API 설치"](#)

가상 그리드 노드 배포

배포 환경(VMware)에 대한 정보 수집

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



일부 노드를 나중에 설치하는 대신 모든 노드를 한 번 설치하는 것이 더 효율적입니다.

VMware 정보입니다

배포 환경에 액세스하여 VMware 환경, 그리드, 관리자 및 클라이언트 네트워크용으로 생성된 네트워크, 스토리지 노드에 사용할 스토리지 볼륨 유형에 대한 정보를 수집해야 합니다.

다음은 포함하여 VMware 환경에 대한 정보를 수집해야 합니다.

- 구축을 완료할 수 있는 적절한 권한이 있는 VMware vSphere 계정의 사용자 이름 및 암호입니다.
- 각 StorageGRID 노드 가상 머신에 대한 호스트, 데이터 저장소 및 네트워크 구성 정보입니다.



VMware Live vMotion을 사용하면 가상 머신 클록 시간이 점프를 일으키며 모든 유형의 그리드 노드에서는 지원되지 않습니다. 드물지만 잘못된 클럭 시간으로 인해 데이터 또는 구성 업데이트가 손실될 수 있습니다.

그리드 네트워크 정보

StorageGRID 그리드 네트워크(필수)용으로 생성된 VMware 네트워크에 대한 정보를 수집해야 합니다. 여기에는 다음이 포함됩니다.

- 네트워크 이름입니다.
- 고정 또는 DHCP 중 IP 주소를 할당하는 데 사용되는 방법입니다.
 - 고정 IP 주소를 사용하는 경우 각 그리드 노드에 필요한 네트워킹 세부 정보(IP 주소, 게이트웨이, 네트워크 마스크)가 표시됩니다.
 - DHCP를 사용하는 경우 그리드 네트워크에 있는 기본 관리자 노드의 IP 주소입니다. 자세한 내용은 ["그리드 노드가 기본 관리자 노드를 검색하는 방법"](#) 참조하십시오.

관리자 네트워크 정보

선택적 StorageGRID 관리자 네트워크에 연결될 노드의 경우 이 네트워크에 대해 생성된 VMware 네트워크 관련 정보를 수집해야 합니다. 여기에는 다음이 포함됩니다.

- 네트워크 이름입니다.
- 고정 또는 DHCP 중 IP 주소를 할당하는 데 사용되는 방법입니다.
 - 고정 IP 주소를 사용하는 경우 각 그리드 노드에 필요한 네트워킹 세부 정보(IP 주소, 게이트웨이, 네트워크 마스크)가 표시됩니다.
 - DHCP를 사용하는 경우 그리드 네트워크에 있는 기본 관리자 노드의 IP 주소입니다. 자세한 내용은 ["그리드 노드가 기본 관리자 노드를 검색하는 방법"](#) 참조하십시오.
- 관리 네트워크의 외부 서브넷 목록(ESL).

클라이언트 네트워크 정보

선택적 StorageGRID 클라이언트 네트워크에 연결될 노드의 경우 이 네트워크에 대해 생성된 VMware 네트워크 관련 정보를 수집해야 합니다. 여기에는 다음이 포함됩니다.

- 네트워크 이름입니다.
- 고정 또는 DHCP 중 IP 주소를 할당하는 데 사용되는 방법입니다.
- 고정 IP 주소를 사용하는 경우 각 그리드 노드에 필요한 네트워킹 세부 정보(IP 주소, 게이트웨이, 네트워크 마스크)가 표시됩니다.

추가 인터페이스에 대한 정보입니다

노드를 설치한 후 vCenter에서 VM에 트렁크 또는 액세스 인터페이스를 선택적으로 추가할 수 있습니다. 예를 들어, 관리 또는 게이트웨이 노드에 트렁크 인터페이스를 추가하여 VLAN 인터페이스를 사용하여 다른 애플리케이션이나 테넌트에 속한 트래픽을 분리할 수 있습니다. 또는 고가용성(HA) 그룹에서 사용할 액세스 인터페이스를 추가할 수도 있습니다.

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

- 트렁크 인터페이스를 추가하는 경우 각각의 새 상위 인터페이스에 대해 하나 이상의 VLAN 인터페이스를 구성합니다. 을 ["VLAN 인터페이스를 구성합니다"](#)참조하십시오.
- 액세스 인터페이스를 추가할 경우 HA 그룹에 직접 추가해야 합니다. 을 ["고가용성 그룹을 구성합니다"](#)참조하십시오.

가상 스토리지 노드의 스토리지 볼륨

가상 머신 기반 스토리지 노드에 대한 다음 정보를 수집해야 합니다.

- 추가하려는 스토리지 볼륨(스토리지 LUN)의 수와 크기입니다. 을 참조하십시오. ["요구사항을 충족해야 합니다"](#)

그리드 구성 정보

그리드를 구성하려면 정보를 수집해야 합니다.

- 그리드 사용권
- NTP(Network Time Protocol) 서버 IP 주소입니다
- DNS 서버 IP 주소입니다

Linux 배포를 위한 노드 구성 파일 만들기

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



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

노드 구성 파일의 위치입니다

각 StorageGRID 노드의 구성 파일을 `/etc/storagegrid/nodes` 노드가 실행될 호스트의 디렉토리에 배치합니다. 예를 들어 HostA에서 관리자 노드 1개, 게이트웨이 노드 1개 및 스토리지 노드 1개를 실행하려면 HostA에 3개의 노드 구성 파일을 배치해야 `/etc/storagegrid/nodes` 합니다.

vim 또는 nano와 같은 텍스트 편집기를 사용하여 각 호스트에서 직접 구성 파일을 만들거나 다른 곳에서 구성 파일을 만들어 각 호스트로 이동할 수 있습니다.

노드 구성 파일 이름 지정

구성 파일의 이름이 중요합니다. 형식은 `입니디 node-name.conf`. 여기서 `node-name` 는 노드에 할당된 이름입니다. 이 이름은 StorageGRID Installer에 나타나며 노드 마이그레이션과 같은 노드 유지 관리 작업에 사용됩니다.

노드 이름은 다음 규칙을 따라야 합니다.

- 고유해야 합니다
- 문자로 시작해야 합니다
- A ~ Z 및 a ~ z 문자를 포함할 수 있습니다
- 0에서 9 사이의 숫자를 포함할 수 있습니다
- 하나 이상의 하이픈(-)을 포함할 수 있습니다.
- 확장자는 32자 이하여야 합니다 `.conf`

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

그리드에 대해 멀티 사이트 토폴로지를 계획한 경우 일반적인 노드 명명 규칙은 다음과 같습니다.

`site-nodetype-nodenummer.conf`

예를 들어, 데이터 센터 1의 첫 번째 관리자 노드와 `dc2-sn3.conf` 데이터 센터 2의 세 번째 스토리지 노드에 사용할 수 `dc1-adm1.conf` 있습니다. 그러나 모든 노드 이름이 명명 규칙을 따른다 하더라도 원하는 스키마를 사용할 수 있습니다.

노드 구성 파일의 내용입니다

구성 파일에는 키/값 쌍이 포함되어 있으며 한 줄에 하나의 키와 하나의 값이 있습니다. 각 키/값 쌍에 대해 다음 규칙을 따르십시오.

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

다음 표에서는 지원되는 모든 키의 값을 정의합니다. 각 키에는 다음 중 하나가 지정됩니다.

- * 필수 *: 모든 노드 또는 지정된 노드 유형에 필요합니다

- * 모범 사례 *: 선택 사항이지만 권장됨
- * 선택 사항 *: 모든 노드에 대해 선택 사항입니다

관리 네트워크 키

관리_IP

값	지정
<p>Linux 기반 노드를 설치하는 데 사용하려는 관리 노드의 그리드 네트워크 IPv4 주소입니다. 복구를 위해 기본 관리 노드의 IP가 있으면 이를 사용하고, 그렇지 않으면 기본이 아닌 관리 노드의 IP를 사용합니다. 이 매개변수를 생략하면 노드는 mDNS를 사용하여 기본 관리 노드를 검색하려고 시도합니다.</p> <p>"그리드 노드가 기본 관리자 노드를 검색하는 방법"</p> <ul style="list-style-type: none"> • 참고 *: 이 값은 기본 관리 노드에서 무시되고 금지될 수 있습니다. 	모범 사례

관리 네트워크 구성

값	지정
DHCP, 정적 또는 비활성	선택 사항

관리_네트워크_ESL

값	지정
<p>심표로 구분된 서브넷 목록으로, 이 노드가 Admin Network 게이트웨이를 사용하여 통신해야 하는 CIDR 표기법을 사용합니다.</p> <p>예: 172.16.0.0/21, 172.17.0.0/21</p>	선택 사항

Admin_network_Gateway를 선택합니다

값	지정
<p>이 노드에 대한 로컬 관리 네트워크 게이트웨이의 IPv4 주소입니다. admin_network_ip 및 admin_network_mask에 의해 정의된 서브넷에 있어야 합니다. DHCP 구성 네트워크에서는 이 값이 무시됩니다.</p> <p>예:</p> <p>1.1.1.1</p> <p>10.224.4.81</p>	<p>가 지정된 경우 ADMIN_NETWORK_ESL 필수입니다. 그렇지 않으면 선택 사항입니다.</p>

admin_network_ip을 선택합니다

값	지정
<p>관리 네트워크에서 이 노드의 IPv4 주소입니다. 이 키는 admin_network_Config=static인 경우에만 필요합니다. 다른 값에 대해서는 지정하지 마십시오.</p> <p>예:</p> <p>1.1.1.1</p> <p>10.224.4.81</p>	<p>admin_network_config = static 인 경우 필요합니다.</p> <p>그렇지 않으면 선택 사항입니다.</p>

admin_network_MAC입니다

값	지정
<p>컨테이너의 관리 네트워크 인터페이스에 대한 MAC 주소입니다.</p> <p>이 필드는 선택 사항입니다. 생략할 경우 MAC 주소가 자동으로 생성됩니다.</p> <p>콜론으로 구분된 6쌍의 16진수 숫자이어야 합니다.</p> <p>예: b2:9c:02:c2:27:10</p>	<p>선택 사항</p>

admin_network_mask를 선택합니다

값	지정
<p>이 노드의 IPv4 넷마스크는 관리자 네트워크에서 설정합니다. admin_network_config = static 인 경우 이 키를 지정하고 다른 값에 대해서는 이 키를 지정하지 마십시오.</p> <p>예:</p> <p>255.255.255.0</p> <p>255.255.248.0</p>	<p>admin_network_ip을 지정하고 admin_network_Config=static인 경우 필수입니다.</p> <p>그렇지 않으면 선택 사항입니다.</p>

admin_network_mtu

값	지정
<p>Admin Network의 이 노드에 대한 MTU(Maximum Transmission Unit)입니다. admin_network_Config=DHCP인지 지정하지 마십시오. 지정된 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1500이 사용됩니다.</p> <p>정보 프레임을 사용하려면 MTU를 9000과 같은 정보 프레임에 적합한 값으로 설정합니다. 그렇지 않으면 기본값을 유지합니다.</p> <ul style="list-style-type: none"> • 중요 *: 네트워크의 MTU 값은 노드가 연결된 스위치 포트에 구성된 값과 일치해야 합니다. 그렇지 않으면 네트워크 성능 문제 또는 패킷 손실이 발생할 수 있습니다. <p>예:</p> <p>1500</p> <p>8192</p>	선택 사항

admin_network_target 을 선택합니다

값	지정
<p>StorageGRID 노드에서 관리자 네트워크 액세스에 사용할 호스트 디바이스의 이름입니다. 네트워크 인터페이스 이름만 지원됩니다. 일반적으로 GRID_NETWORK_TARGET 또는 CLIENT_NETWORK_TARGET에 지정된 것과 다른 인터페이스 이름을 사용합니다.</p> <ul style="list-style-type: none"> • 참고 *: 네트워크 대상으로 연결 또는 브리지 장치를 사용하지 마십시오. 연결 디바이스 위에 VLAN(또는 기타 가상 인터페이스)을 구성하거나 브리지 및 가상 이더넷(veth) 쌍을 사용합니다. • 모범 사례 *: 이 노드에 처음에 관리 네트워크 IP 주소가 없을 경우에도 값을 지정하십시오. 그런 다음 나중에 호스트에서 노드를 다시 구성하지 않고도 관리 네트워크 IP 주소를 추가할 수 있습니다. <p>예:</p> <p>bond0.1002</p> <p>ens256</p>	모범 사례

admin_network_target_type입니다

값	지정
인터페이스(이 값만 지원됩니다.)	선택 사항

admin_network_target_type_interface_clone_MAC

값	지정
<p>참 또는 거짓</p> <p>StorageGRID 컨테이너가 관리자 네트워크에서 호스트 호스트 대상 인터페이스의 MAC 주소를 사용하도록 하려면 키를 "true"로 설정합니다.</p> <ul style="list-style-type: none"> 모범 사례: * promiscuous 모드가 필요한 네트워크에서는 admin_network_target_type_interface_clone_MAC 키를 대신 사용합니다. <p>Linux용 MAC 복제에 대한 자세한 내용은 다음을 참조하세요. "MAC 주소 복제에 대한 고려 사항 및 권장 사항"</p>	<p>모범 사례</p>

admin_role을 선택합니다

값	지정
<p>Primary 또는 Non-Primary</p> <p>이 키는 node_type=vm_Admin_Node인 경우에만 필요하며 다른 노드 유형에 대해서는 지정하지 않습니다.</p>	<p>node_type=vm_admin_Node인 경우 필요합니다</p> <p>그렇지 않으면 선택 사항입니다.</p>

장치 키를 차단합니다

Block_device_audit_logs

값	지정
<p>이 노드가 감사 로그의 영구 저장에 사용할 블록 디바이스 특수 파일의 경로 및 이름입니다.</p> <p>예:</p> <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-audit-logs</pre>	<p>node_type이 vm_admin_Node인 노드에 필요합니다. 다른 노드 유형에는 지정하지 마십시오.</p>

Block_device_RANGEDB_nnn을 선택합니다

값	지정
<p>이 노드가 영구 오브젝트 스토리지에 사용할 블록 디바이스 특수 파일의 경로 및 이름입니다. 이 키는 <code>node_type=vm_Storage_Node</code>인 노드에만 필요하며 다른 노드 유형에 대해서는 지정하지 않습니다.</p> <p><code>block_device_RANGEDB_000</code> 만 필요하며 나머지는 선택 사항입니다. <code>block_device_RANGEDB_000</code> 에 지정된 블록 디바이스는 4TB 이상이어야 하며 다른 블록 디바이스는 더 작을 수 있습니다.</p> <p>간격을 두지 마십시오. <code>BLOCK_DEVICE_RANGEDB_005</code>를 지정하는 경우 <code>BLOCK_DEVICE_RANGEDB_004</code>도 지정해야 합니다.</p> <ul style="list-style-type: none"> 참고 *: 기존 배포와의 호환성을 위해 업그레이드된 노드에 대해 2자리 키가 지원됩니다. 	<p>필수:</p> <p><code>BLOCK_DEVICE_RANGEDB_000</code></p> <p>선택 사항:</p> <p><code>BLOCK_DEVICE_RANGEDB_001</code></p> <p><code>BLOCK_DEVICE_RANGEDB_002</code> 를 참조하십시오</p> <p><code>Block_device_RANGEDB_003</code> 을 참조하십시오</p>
<p>예:</p> <pre> /dev/disk/by-path/pci-0000:03:00.0-scsi-0:0:0:0 /dev/disk/by-id/wwn- 0x600a09800059d6df000060d757b475fd /dev/mapper/sgws-sn1-rangedb-000 </pre>	<p><code>Block_device_RANGEDB_004</code> 를 참조하십시오</p> <p><code>Block_device_RANGEDB_005</code> 를 참조하십시오</p> <p><code>Block_device_RANGEDB_006</code></p> <p><code>Block_device_RANGEDB_007</code> 을 참조하십시오</p> <p><code>Block_device_RANGEDB_008</code> 을 참조하십시오</p> <p><code>Block_device_RANGEDB_009</code> 를 참조하십시오</p> <p><code>Block_device_RANGEDB_010</code></p> <p><code>Block_device_RANGEDB_011</code> 을 참조하십시오</p> <p><code>Block_device_RANGEDB_012</code> 를 참조하십시오</p> <p><code>Block_device_RANGEDB_013</code></p> <p><code>Block_device_RANGEDB_014</code></p> <p><code>Block_device_RANGEDB_015</code> 를 참조하십시오</p>

BLOCK_DEVICE_Tables

값	지정
<p>이 노드가 데이터베이스 테이블의 영구 저장에 사용할 블록 디바이스 특수 파일의 경로 및 이름입니다. 이 키는 <code>node_type=vm_Admin_Node</code>인 노드에만 필요합니다. 다른 노드 유형에 대해서는 지정하지 마십시오.</p> <p>예:</p> <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 요소입니다

BLOCK_DEVICE_VAR_LOCAL

값	지정
<p>이 노드가 영구 스토리지에 사용할 블록 디바이스 특수 파일의 경로 및 <code>/var/local</code> 이름입니다.</p> <p>예:</p> <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 요소입니다

클라이언트 네트워크 키

client_network_Config

값	지정
DHCP, 정적 또는 비활성	선택 사항

CLIENT_NETWORK_GATEWAY

값	지정

<p>client_network_ip 및 client_network_mask에 의해 정의된 서브넷에 있어야 하는 이 노드에 대한 로컬 클라이언트 네트워크 게이트웨이의 IPv4 주소입니다. DHCP 구성 네트워크에서는 이 값이 무시됩니다.</p> <p>예:</p> <p>1.1.1.1</p> <p>10.224.4.81</p>	<p>선택 사항</p>
--	--------------

client_network_ip

값	지정
<p>클라이언트 네트워크에서 이 노드의 IPv4 주소입니다.</p> <p>이 키는 client_network_Config = static 일 때만 필요합니다. 다른 값에 대해서는 지정하지 마십시오.</p> <p>예:</p> <p>1.1.1.1</p> <p>10.224.4.81</p>	<p>client_network_Config=static 인 경우 필요합니다</p> <p>그렇지 않으면 선택 사항입니다.</p>

client_network_MAC

값	지정
<p>컨테이너에 있는 클라이언트 네트워크 인터페이스의 MAC 주소입니다.</p> <p>이 필드는 선택 사항입니다. 생략할 경우 MAC 주소가 자동으로 생성됩니다.</p> <p>콜론으로 구분된 6쌍의 16진수 숫자이어야 합니다.</p> <p>예: b2:9c:02:c2:27:20</p>	<p>선택 사항</p>

client_network_mask.(클라이언트 네트워크 마스크

값	지정
<p>클라이언트 네트워크의 이 노드에 대한 IPv4 넷마스크입니다.</p> <p>client_network_config = static 인 경우 이 키를 지정하고 다른 값에는 이 키를 지정하지 마십시오.</p> <p>예:</p> <p>255.255.255.0</p> <p>255.255.248.0</p>	<p>client_network_ip을 지정하고 client_network_Config=static인 경우 필수입니다</p> <p>그렇지 않으면 선택 사항입니다.</p>

client_network_mtu

값	지정
<p>Client Network의 이 노드에 대한 MTU(Maximum Transmission Unit)입니다. client_network_Config = DHCP인지 지정하지 마십시오. 지정된 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1500이 사용됩니다.</p> <p>점보 프레임을 사용하려면 MTU를 9000과 같은 점보 프레임에 적합한 값으로 설정합니다. 그렇지 않으면 기본값을 유지합니다.</p> <ul style="list-style-type: none"> • 중요 *: 네트워크의 MTU 값은 노드가 연결된 스위치 포트에 구성된 값과 일치해야 합니다. 그렇지 않으면 네트워크 성능 문제 또는 패킷 손실이 발생할 수 있습니다. <p>예:</p> <p>1500</p> <p>8192</p>	<p>선택 사항</p>

client_network_target 을 선택합니다

값	지정
<p>StorageGRID 노드에서 클라이언트 네트워크 액세스에 사용할 호스트 디바이스의 이름입니다. 네트워크 인터페이스 이름만 지원됩니다. 일반적으로 GRID_NETWORK_TARGET 또는 ADMIN_NETWORK_TARGET에 지정된 것과 다른 인터페이스 이름을 사용합니다.</p> <ul style="list-style-type: none"> 참고 *: 네트워크 대상으로 연결 또는 브리지 장치를 사용하지 마십시오. 연결 디바이스 위에 VLAN(또는 기타 가상 인터페이스)을 구성하거나 브리지 및 가상 이더넷(veth) 쌍을 사용합니다. 모범 사례: * 이 노드에 클라이언트 네트워크 IP 주소가 없을 경우에도 값을 지정하십시오. 그런 다음 나중에 호스트에서 노드를 다시 구성하지 않고도 클라이언트 네트워크 IP 주소를 추가할 수 있습니다. <p>예:</p> <p>bond0.1003</p> <p>ens423</p>	<p>모범 사례</p>

CLIENT_NETWORK_TARGET_TYPE

값	지정
인터페이스(지원되는 값만 해당)	선택 사항

client_network_target_type_interface_clone_MAC

값	지정
<p>참 또는 거짓</p> <p>StorageGRID 컨테이너가 클라이언트 네트워크의 호스트 대상 인터페이스의 MAC 주소를 사용하도록 하려면 키를 "true"로 설정합니다.</p> <ul style="list-style-type: none"> 모범 사례: * promiscuous 모드가 필요한 네트워크에서는 대신 client_network_target_type_interface_clone_mac 키를 사용합니다. <p>Linux용 MAC 복제에 대한 자세한 내용은 다음을 참조하세요. "MAC 주소 복제에 대한 고려 사항 및 권장 사항"</p>	<p>모범 사례</p>

그리드 네트워크 키

GRID_NETWORK_CONFIG(그리드 네트워크 구성

값	지정
고정 또는 DHCP 지정하지 않으면 기본적으로 정적입니다.	모범 사례

GRID_NETWORK_Gateway를 참조하십시오

값	지정
GRID_NETWORK_IP 및 GRID_NETWORK_MASK로 정의된 서브넷에 있어야 하는 이 노드에 대한 로컬 Grid Network 게이트웨이의 IPv4 주소입니다. DHCP 구성 네트워크에서는 이 값이 무시됩니다. 그리드 네트워크가 게이트웨이가 없는 단일 서브넷인 경우, 서브넷(X.Y.Z.1)의 표준 게이트웨이 주소 또는 이 노드의 GRID_NETWORK_IP 값을 사용합니다. 두 값 중 하나를 사용하면 미래의 그리드 네트워크 확장이 단순화됩니다.	필수 요소입니다

GRID_NETWORK_IP입니다

값	지정
Grid Network에서 이 노드의 IPv4 주소입니다. 이 키는 GRID_NETWORK_CONFIG = static 일 때만 필요합니다. 다른 값에 대해서는 지정하지 마십시오. 예: 1.1.1.1 10.224.4.81	GRID_NETWORK_CONFIG = STATIC인 경우 필요합니다 그렇지 않으면 선택 사항입니다.

GRID_NETWORK_MAC을 선택합니다

값	지정
컨테이너의 그리드 네트워크 인터페이스에 대한 MAC 주소입니다. 콜론으로 구분된 6쌍의 16진수 숫자이어야 합니다. 예: b2:9c:02:c2:27:30	선택 사항 생략할 경우 MAC 주소가 자동으로 생성됩니다.

GRID_NETWORK_MASK 를 참조하십시오

값	지정
<p>그리드 네트워크에서 이 노드에 대한 IPv4 넷마스크입니다. GRID_NETWORK_CONFIG = STATIC인 경우 이 키를 지정하고 다른 값에는 이 키를 지정하지 마십시오.</p> <p>예:</p> <p>255.255.255.0</p> <p>255.255.248.0</p>	<p>GRID_NETWORK_IP를 지정하고 GRID_NETWORK_CONFIG=STATIC인 경우에 필요합니다.</p> <p>그렇지 않으면 선택 사항입니다.</p>

GRID_NETWORK_MTU 를 참조하십시오

값	지정
<p>Grid Network의 이 노드에 대한 MTU(Maximum Transmission Unit)입니다. GRID_NETWORK_CONFIG=DHCP인지 지정하지 마십시오. 지정된 경우 값은 1280에서 9216 사이여야 합니다. 생략하면 1500이 사용됩니다.</p> <p>점보 프레임을 사용하려면 MTU를 9000과 같은 점보 프레임에 적합한 값으로 설정합니다. 그렇지 않으면 기본값을 유지합니다.</p> <ul style="list-style-type: none"> • 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 얻으려면 모든 노드를 그리드 네트워크 인터페이스에서 유사한 MTU 값으로 구성해야 합니다. 개별 노드의 그리드 네트워크에 대한 MTU 설정에 상당한 차이가 있을 경우 * Grid Network MTU mismatch * 경고가 트리거됩니다. MTU 값은 모든 네트워크 유형에 대해 같을 필요는 없습니다. <p>예:</p> <p>1500</p> <p>8192</p>	<p>선택 사항</p>

GRID_NETWORK_TARGET

값	지정
<p>StorageGRID 노드에서 그리드 네트워크 액세스에 사용할 호스트 디바이스의 이름입니다. 네트워크 인터페이스 이름만 지원됩니다. 일반적으로 <code>admin_network_target</code> 또는 <code>client_network_target</code> 에 지정된 것과 다른 인터페이스 이름을 사용합니다.</p> <ul style="list-style-type: none"> 참고 *: 네트워크 대상으로 연결 또는 브리지 장치를 사용하지 마십시오. 연결 디바이스 위에 VLAN(또는 기타 가상 인터페이스)을 구성하거나 브리지 및 가상 이더넷(veth) 쌍을 사용합니다. <p>예:</p> <p><code>bond0.1001</code></p> <p><code>ens192</code></p>	필수 요소입니다

GRID_NETWORK_TARGET_TYPE

값	지정
인터페이스(이 값만 지원됩니다.)	선택 사항

GRID_NETWORK_TARGET_TYPE_INTERFACE_CLONE_MAC

값	지정
<p>참 또는 거짓</p> <p>StorageGRID 컨테이너가 그리드 네트워크에서 호스트 대상 인터페이스의 MAC 주소를 사용하도록 키 값을 "true"로 설정합니다.</p> <ul style="list-style-type: none"> 모범 사례: * promiscuous 모드가 필요한 네트워크에서는 <code>grid_network_target_type_interface_clone_mac</code> 키를 대신 사용합니다. <p>Linux용 MAC 복제에 대한 자세한 내용은 다음을 참조하세요. "MAC 주소 복제에 대한 고려 사항 및 권장 사항"</p>	모범 사례

설치 암호 키(임시)

사용자 지정_임시_암호_해시

값	지정
<p>기본 관리자 노드의 경우 설치 중에 StorageGRID 설치 API에 대한 기본 임시 암호를 설정합니다.</p> <ul style="list-style-type: none"> 참고 *: 기본 관리자 노드에서만 설치 암호를 설정합니다. 다른 노드 유형에 암호를 설정하려고 하면 노드 구성 파일의 유효성 검사가 실패합니다. <p>이 값을 설정해도 설치가 완료된 경우 아무런 영향이 없습니다.</p> <p>이 키를 생략하면 기본적으로 임시 암호가 설정되지 않습니다. 또는 StorageGRID 설치 API를 사용하여 임시 암호를 설정할 수 있습니다.</p> <p>8자 이상 32자 이하의 암호 형식을 가진 SHA-512 암호 \$6\$<salt>\$<password hash> 해시여야 crypt () 합니다.</p> <p>이 해시는 SHA-512 모드의 명령과 같은 CLI 툴을 사용하여 생성할 수 openssl passwd 있습니다.</p>	<p>모범 사례</p>

인터페이스 키입니다

interface_target_nnnn입니다

값	지정
<p>이 노드에 추가할 추가 인터페이스의 이름 및 선택적 설명입니다. 각 노드에 여러 개의 인터페이스를 추가할 수 있습니다.</p> <p>_nnnn_의 경우 추가할 각 interface_target 항목의 고유 번호를 지정합니다.</p> <p>값에 대해 베어 메탈 호스트의 물리적 인터페이스 이름을 지정합니다. 그런 다음 필요에 따라 심표를 추가하고 인터페이스에 대한 설명을 입력합니다. 이 설명은 VLAN 인터페이스 페이지와 HA 그룹 페이지에 표시됩니다.</p> <p>예: INTERFACE_TARGET_0001=ens256, Trunk</p> <p>트렁크 인터페이스를 추가하는 경우 StorageGRID에서 VLAN 인터페이스를 구성해야 합니다. 액세스 인터페이스를 추가할 경우 인터페이스를 HA 그룹에 직접 추가할 수 있으며, VLAN 인터페이스를 구성할 필요가 없습니다.</p>	<p>선택 사항</p>

최대 RAM 키

최대 RAM

값	지정
<p>이 노드가 사용할 수 있는 최대 RAM 양입니다. 이 키를 생략하면 노드의 메모리 제한 사항이 없게 됩니다. 운영 레벨 노드에 대해 이 필드를 설정할 때 총 시스템 RAM보다 최소 24GB 및 16 ~ 32GB 적은 값을 지정합니다.</p> <ul style="list-style-type: none"> 참고 *: RAM 값은 노드의 실제 메타데이터 예약 공간에 영향을 줍니다. 를 "메타데이터 예약된 공간에 대한 설명입니다"참조하십시오. <p>이 필드의 형식은 <i>numberunit</i>, WHERE <i>unit</i> b, k, m 또는 `g`입니다.</p> <p>예:</p> <p>24g</p> <p>38654705664b</p> <ul style="list-style-type: none"> 참고 *: 이 옵션을 사용하려면 메모리 cgroup에 대한 커널 지원을 활성화해야 합니다. 	선택 사항

노드 유형 키입니다

node_type입니다

값	지정
<p>노드 유형:</p> <ul style="list-style-type: none"> VM_Admin_Node VM_스토리지_노드 VM_Archive_Node VM_API_게이트웨이 	필수 요소입니다

스토리지 유형

값	지정
<p>스토리지 노드에 포함된 객체 유형을 정의합니다. 자세한 내용은 을 "스토리지 노드 유형"참조하십시오. 이 키는 <code>node_type=vm_Storage_Node</code>인 노드에만 필요하며 다른 노드 유형에 대해서는 지정하지 않습니다. 스토리지 유형:</p> <ul style="list-style-type: none"> 결합된 데이터 메타데이터 <ul style="list-style-type: none"> 참고 *: <code>storage_type</code>이 지정되지 않은 경우 스토리지 노드 유형은 기본적으로 결합(데이터 및 메타데이터)으로 설정됩니다. 	선택 사항



포트 리매핑 지원은 더 이상 제공되지 않으며 향후 릴리스에서 제거될 예정입니다. 재매핑된 포트를 제거하려면 다음을 참조하세요. "[베어 메탈 호스트에서 포트 재매핑을 제거합니다](#)".

port_remap 을 참조하십시오

값	지정
<p>노드에서 내부 그리드 노드 통신 또는 외부 통신을 위해 사용하는 모든 포트를 다시 매핑합니다. 엔터프라이즈 네트워킹 정책으로 StorageGRID에서 사용하는 하나 이상의 포트를 제한하는 경우 또는 예 설명된 대로 포트를 다시 매핑해야 "내부 그리드 노드 통신" "외부 통신"합니다.</p> <ul style="list-style-type: none"> • 중요 *: 로드 밸런서 엔드포인트를 구성하기 위해 사용하려는 포트를 다시 매핑하지 마십시오. • 참고 *: port_remap 만 설정된 경우 지정하는 매핑이 인바운드 및 아웃바운드 통신 모두에 사용됩니다. port_remap_inbound 도 지정된 경우 port_remap 은 아웃바운드 통신에만 적용됩니다. <p>사용되는 형식은 다음과 같습니다 <i>network type/protocol/default port used by grid node/new port</i>. 여기서 <i>network type</i> 그리드, 관리자 또는 클라이언트이고 <i>protocol</i> TCP 또는 UDP입니다.</p> <p>예: <code>PORT_REMAP = client/tcp/18082/443</code></p> <p>심표로 구분된 목록을 사용하여 여러 포트를 다시 매핑할 수도 있습니다.</p> <p>예: <code>PORT_REMAP = client/tcp/18082/443, client/tcp/18083/80</code></p>	<p>선택 사항</p>

port_remap_inbound 를 참조하십시오

값	지정
<p>인바운드 통신을 지정된 포트에 다시 매핑합니다. <code>port_remap_inbound</code> 를 지정하지만 <code>port_remap</code> 의 값을 지정하지 않으면 포트의 아웃바운드 통신이 변경되지 않습니다.</p> <ul style="list-style-type: none"> • 중요 *: 로드 밸런서 엔드포인트를 구성하기 위해 사용하려는 포트를 다시 매핑하지 마십시오. <p>사용되는 형식은 다음과 같습니다 <code>network type/protocol/remapped port/default port used by grid node</code>. 여기서 <code>network type</code> 그리드, 관리자 또는 클라이언트이고 <code>protocol</code> TCP 또는 UDP입니다.</p> <p>예: <code>PORT_REMAP_INBOUND = grid/tcp/3022/22</code></p> <p>쉼표로 구분된 목록을 사용하여 여러 인바운드 포트를 다시 매핑할 수도 있습니다.</p> <p>예: <code>PORT_REMAP_INBOUND = grid/tcp/3022/22, admin/tcp/3022/22</code></p>	<p>선택 사항</p>

그리드 노드가 기본 관리자 노드를 검색하는 방법

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

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

- `admin_ip` 매개 변수를 사용하여 기본 관리 노드의 IP 주소를 수동으로 입력할 수 있습니다.
- `admin_ip` 매개 변수를 생략하여 그리드 노드가 값을 자동으로 검색하도록 할 수 있습니다. 자동 검색은 그리드 네트워크가 DHCP를 사용하여 기본 관리 노드에 IP 주소를 할당할 때 특히 유용합니다.

운영 관리자 노드의 자동 검색은 mDNS(multicast domain name system)를 사용하여 수행됩니다. 운영 관리 노드가 처음 시작되면 mDNS를 사용하여 해당 IP 주소를 게시합니다. 그런 다음 동일한 서브넷에 있는 다른 노드에서 IP 주소를 쿼리하고 자동으로 가져올 수 있습니다. 그러나 멀티캐스트 IP 트래픽은 일반적으로 서브넷 간에 라우팅할 수 없기 때문에 다른 서브넷의 노드는 기본 관리 노드의 IP 주소를 직접 획득할 수 없습니다.

자동 검색을 사용하는 경우:



- 기본 관리 노드가 직접 연결되지 않은 서브넷에 있는 하나 이상의 그리드 노드에 대해 `admin_IP` 설정을 포함해야 합니다. 이 그리드 노드는 mDNS로 검색할 서브넷의 다른 노드에 대한 기본 관리 노드의 IP 주소를 게시합니다.
- 네트워크 인프라스트럭처가 서브넷 내의 다중 캐스트 IP 트래픽 전달을 지원하는지 확인합니다.

StorageGRID 노드를 가상 머신(VMware)으로 배포

VMware vSphere Web Client를 사용하여 각 그리드 노드를 가상 머신으로 구축합니다. 배포 중에는 각 그리드 노드가 하나 이상의 StorageGRID 네트워크에 생성되고 연결됩니다.

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

선택적으로 노드의 전원을 켜기 전에 노드 포트를 재매핑하거나 노드의 CPU 또는 메모리 설정을 늘릴 수 있습니다.

시작하기 전에

- "[설치 계획 및 준비](#)" 소프트웨어, CPU 및 RAM, 스토리지 및 성능에 대한 요구 사항을 검토하고 이해했습니다.
- VMware vSphere 하이퍼바이저에 대해 잘 알고 있으며 이 환경에서 가상 머신을 구축한 경험이 있습니다.



VMware 툴과 유사한 오픈 소스 구축 패키지인 `open-vm-tools` StorageGRID 가상 머신은 포함되어 있습니다. VMware Tools를 수동으로 설치할 필요가 없습니다.

- VMware용 StorageGRID 설치 아카이브의 올바른 버전을 다운로드하고 압축을 풀었습니다.



확장 또는 복구 작업의 일부로 새 노드를 구축하는 경우 현재 그리드에서 실행 중인 StorageGRID 버전을 사용해야 합니다.

- StorageGRID 가상 머신 디스크(`.vmdk`) 파일이 있는 경우:

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

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

파일 이름	설명
<code>vsphere-primary-admin.ovf</code> <code>vsphere-primary-admin.mf</code>	기본 관리 노드의 템플릿 파일 및 매니페스트 파일
<code>vsphere - non-primary-admin.ovf</code> <code>vsphere - non-primary-admin.mf</code>	비 기본 관리 노드에 대한 템플릿 파일 및 매니페스트 파일입니다.
<code>vsphere-storage.ovf</code> <code>vsphere-storage.mf</code>	스토리지 노드의 템플릿 파일 및 매니페스트 파일
<code>vsphere-gateway.ovf</code> <code>vsphere-gateway.mf</code>	게이트웨이 노드의 템플릿 파일 및 매니페스트 파일

- `.vmdk` `.ovf`, 및 `.mf` 파일은 모두 같은 디렉터리에 있습니다.
- 장애 도메인을 최소화할 계획이 있습니다. 예를 들어 단일 vSphere ESXi 호스트에 모든 게이트웨이 노드를 배포하면 안 됩니다.



운영 구축 환경에서는 단일 가상 머신에서 스토리지 노드를 두 개 이상 실행하지 마십시오. 허용할 수 없는 장애 도메인 문제가 발생할 경우 동일한 ESXi 호스트에서 여러 가상 머신을 실행하지 마십시오.

- 확장 또는 복구 작업의 일부로 노드를 구축하는 경우 또는 이 있는 "[StorageGRID 시스템 확장을 위한 지침](#)" 복구 및 유지 관리 지침" 것입니다.
- StorageGRID 노드를 NetApp ONTAP 시스템에서 할당된 스토리지가 있는 가상 머신으로 구축하는 경우 볼륨에

FabricPool 계층화 정책이 설정되어 있지 않은 것이 확인되었습니다. 예를 들어 StorageGRID 노드가 VMware 호스트에서 가상 머신으로 실행 중인 경우 노드의 데이터 저장소를 백업하는 볼륨에 FabricPool 계층화 정책이 설정되어 있지 않은지 확인합니다. StorageGRID 노드와 함께 사용되는 볼륨에 대해 FabricPool 계층화를 사용하지 않도록 설정하면 문제 해결과 스토리지 작업이 간소화됩니다.



FabricPool를 사용하여 StorageGRID 관련 데이터를 StorageGRID 자체로 계층화하지 마십시오. StorageGRID 데이터를 StorageGRID로 다시 계층화하면 문제 해결과 운영 복잡성이 늘어납니다.

이 작업에 대해

이 지침에 따라 초기에 VMware 노드를 배포하거나, 확장 시 새 VMware 노드를 추가하거나, 복구 작업의 일부로 VMware 노드를 대체하십시오. 단계에 설명된 경우를 제외하고 노드 구축 절차는 관리 노드, 스토리지 노드 및 게이트웨이 노드를 포함한 모든 노드 유형에 대해 동일합니다.

새 StorageGRID 시스템을 설치하는 경우:

- 노드는 원하는 순서로 배포할 수 있습니다.
- 각 가상 시스템이 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 네트워크를 통해 다른 모든 노드에 연결할 수 있는지 확인해야 합니다.

노드의 포트를 다시 매핑해야 하는 경우 포트 재매핑 구성이 완료될 때까지 새 노드의 전원을 켜서는 안 됩니다.



포트 리매핑 지원은 더 이상 제공되지 않으며 향후 릴리스에서 제거될 예정입니다. 재매핑된 포트를 제거하려면 다음을 참조하세요. "[베어 메탈 호스트에서 포트 재매핑을 제거합니다](#)".

단계

1. vCenter를 사용하여 OVF 템플릿을 구축합니다.

URL을 지정하는 경우 다음 파일이 포함된 폴더를 가리킵니다. 그렇지 않으면 로컬 디렉토리에서 각 파일을 선택합니다.

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

예를 들어 처음 구축하는 노드인 경우 다음 파일을 사용하여 StorageGRID 시스템의 기본 관리 노드를 배포합니다.

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

2. 가상 머신의 이름을 지정합니다.

표준 방법은 가상 머신과 그리드 노드 모두에 동일한 이름을 사용하는 것입니다.

- 가상 머신을 적절한 vApp 또는 리소스 풀에 배치합니다.
- 기본 관리자 노드를 배포하는 경우 최종 사용자 라이선스 계약을 읽고 동의합니다.

vCenter 버전에 따라 최종 사용자 라이선스 계약에 동의하고, 가상 머신의 이름을 지정하고, 데이터 저장소를 선택할 때 단계의 순서가 달라집니다.

- 가상 머신에 사용할 스토리지를 선택합니다.

복구 작업의 일부로 노드를 구축하는 경우 의 지침에 따라 **스토리지 복구 단계입니다** 새 가상 디스크를 추가하거나, 오류가 발생한 그리드 노드에서 가상 하드 디스크를 다시 연결하거나, 둘 다 수행합니다.

스토리지 노드를 구축할 때는 3개 이상의 스토리지 볼륨을 사용하고 각 스토리지 볼륨은 4TB 이상을 사용합니다. 볼륨 0에 4TB 이상을 할당해야 합니다.



Storage Node.ovf 파일은 스토리지를 위한 여러 VMDK를 정의합니다. 이러한 VMDK가 스토리지 요구 사항을 충족하지 않는 경우 노드의 전원을 켜기 전에 해당 VMDK를 제거하고 스토리지에 적절한 VMDK 또는 RDM을 할당해야 합니다. vmdks는 VMware 환경에서 일반적으로 사용되며 관리하기가 더 쉽습니다. 반면 RDM은 100MB 이상의 큰 개체 크기를 사용하는 워크로드에 더 나은 성능을 제공할 수 있습니다.



일부 StorageGRID 설치에서는 일반 가상화 워크로드보다 더 크고 사용 빈도가 높은 스토리지 볼륨을 사용할 수 있습니다. 최적의 성능을 얻으려면 등의 일부 하이퍼바이저 매개 변수를 조정해야 할 수 MaxAddressableSpaceTB 있습니다. 성능 저하가 발생하는 경우 가상화 지원 리소스에 문의하여 작업 부하별 구성 조정을 통해 해당 환경이 이점을 누릴 수 있는지 확인하십시오.

- 네트워크를 선택합니다.

각 소스 네트워크의 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, 모든 사이트와 서버넷에 걸쳐 연결을 제공합니다. 그리드 네트워크의 모든 노드는 다른 모든 노드와 통신할 수 있어야 합니다.
- 관리 네트워크를 사용하는 경우 vSphere 환경에서 다른 대상 네트워크를 선택합니다. 관리 네트워크를 사용하지 않는 경우 그리드 네트워크에 대해 선택한 것과 동일한 대상을 선택합니다.
- 클라이언트 네트워크를 사용하는 경우 vSphere 환경에서 다른 대상 네트워크를 선택합니다. 클라이언트 네트워크를 사용하지 않는 경우 그리드 네트워크에 대해 선택한 것과 동일한 대상을 선택합니다.
- Admin 또는 Client 네트워크를 사용하는 경우 노드가 동일한 Admin 또는 Client 네트워크에 있을 필요가 없습니다.

- 템플릿 사용자 정의 * 의 경우 필요한 StorageGRID 노드 속성을 구성합니다.

- a. 노드 이름 * 을 입력합니다.



그리드 노드를 복구하는 경우 복구할 노드의 이름을 입력해야 합니다.

- b. 임시 설치 암호를 지정하려면 * 임시 설치 암호 * 드롭다운을 사용하십시오. 그러면 새 노드가 그리드에 합류하기 전에 VM 콘솔이나 StorageGRID 설치 API에 액세스하거나 SSH를 사용할 수 있습니다.



임시 설치 암호는 노드 설치 중에만 사용됩니다. 노드가 그리드에 추가된 후 복구 패키지의 파일에 "노드 콘솔 암호" 나열된 를 사용하여 액세스할 수 있습니다. Passwords.txt

- * 노드 이름 사용 *: * 노드 이름 * 필드에 입력한 값이 임시 설치 암호로 사용됩니다.
- * 사용자 정의 암호 사용 *: 사용자 정의 암호는 임시 설치 암호로 사용됩니다.
- * 비밀번호 비활성화 *: 임시 설치 비밀번호가 사용되지 않습니다. 설치 문제를 디버깅하기 위해 VM에 액세스해야 하는 경우 를 참조하십시오 "설치 문제를 해결합니다".

c. 사용자 정의 암호 사용 * 을 선택한 경우 * 사용자 정의 암호 * 필드에 사용할 임시 설치 암호를 지정합니다.

d. Grid Network(eth0) * 섹션에서 * Grid 네트워크 IP 구성 * 에 대해 static 또는 DHCP를 선택합니다.

- 정자를 선택한 경우 * 그리드 네트워크 IP *, * 그리드 네트워크 마스크 *, * 그리드 네트워크 게이트웨이 * 및 * 그리드 네트워크 MTU * 를 입력합니다.
- DHCP를 선택하면 * 그리드 네트워크 IP *, * 그리드 네트워크 마스크 * 및 * 그리드 네트워크 게이트웨이 * 가 자동으로 할당됩니다.

e. Primary Admin IP * 필드에 Grid Network에 대한 기본 관리 노드의 IP 주소를 입력합니다.



구축하는 노드가 기본 관리 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. 그러나 여기서 기본 관리 노드 IP 주소를 설정하는 것이 좋습니다.

a. 관리 네트워크(eth1) * 섹션에서 * 관리 네트워크 IP 구성 * 에 대해 정적, DHCP 또는 비활성화를 선택합니다.

- 관리 네트워크를 사용하지 않으려면 비활성화를 선택하고 관리 네트워크 IP에 * 0.0.0.0 * 을 입력합니다. 다른 필드는 비워 둘 수 있습니다.
- static을 선택한 경우 * Admin network ip *, * Admin network mask *, * Admin network gateway *, * Admin network mtu * 를 입력합니다.
- static을 선택한 경우 * Admin network external subnet list * 를 입력합니다. 또한 게이트웨이를 구성해야 합니다.
- DHCP를 선택하면 * 관리 네트워크 IP *, * 관리 네트워크 마스크 * 및 * 관리 네트워크 게이트웨이 * 가 자동으로 할당됩니다.

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

- 클라이언트 네트워크를 사용하지 않으려면 비활성화를 선택하고 클라이언트 네트워크 IP에 * 0.0.0.0 * 을 입력합니다. 다른 필드는 비워 둘 수 있습니다.
- static을 선택한 경우 * Client network IP *, * Client network mask *, * Client network gateway *, * Client network mtu * 를 입력합니다.
- DHCP를 선택하면 * 클라이언트 네트워크 IP *, * 클라이언트 네트워크 마스크 * 및 * 클라이언트 네트워크 게이트웨이 * 가 자동으로 할당됩니다.

8. 가상 시스템 구성을 검토하고 필요한 사항을 변경합니다.

9. 완료할 준비가 되면 * 마침 * 을 선택하여 가상 머신 업로드를 시작합니다.

10. 이 노드를 복구 작업의 일부로 배포했으며 전체 노드 복구가 아닌 경우 구축이 완료된 후 다음 단계를 수행하십시오.

- a. 가상 컴퓨터를 마우스 오른쪽 단추로 클릭하고 * 설정 편집 * 을 선택합니다.
- b. 스토리지에 지정된 각 기본 가상 하드 디스크를 선택하고 * 제거 * 를 선택합니다.
- c. 데이터 복구 상황에 따라 저장소 요구 사항에 따라 새 가상 디스크를 추가하거나 이전에 제거된 장애 그리드 노드에서 보존된 가상 하드 디스크를 다시 연결하거나 두 디스크 모두를 다시 연결합니다.

다음 중요 지침을 참고하십시오.

- 새 디스크를 추가하는 경우 노드 복구 전에 사용한 것과 동일한 유형의 스토리지 디바이스를 사용해야 합니다.
- Storage Node.ovf 파일은 스토리지를 위한 여러 VMDK를 정의합니다. 이러한 VMDK가 스토리지 요구 사항을 충족하지 않는 경우 노드의 전원을 켜기 전에 해당 VMDK를 제거하고 스토리지에 적절한 VMDK 또는 RDM을 할당해야 합니다. vmdks는 VMware 환경에서 일반적으로 사용되며 관리하기가 더 쉽습니다. 반면 RDM은 100MB 이상의 큰 개체 크기를 사용하는 워크로드에 더 나은 성능을 제공할 수 있습니다.

11. 이 노드에서 사용하는 포트를 다시 매핑해야 하는 경우 다음 단계를 수행합니다.

엔터프라이즈 네트워킹 정책이 StorageGRID에서 사용하는 하나 이상의 포트에 대한 액세스를 제한하는 경우 포트를 다시 매핑해야 할 수 있습니다. StorageGRID에서 사용하는 포트는 를 "네트워킹 지침"참조하십시오.



로드 밸런서 끝점에 사용되는 포트를 다시 매핑하지 마십시오.

- a. 새 VM을 선택합니다.
- b. 구성 탭에서 * 설정 * > * vApp 옵션 * 을 선택합니다. vApp 옵션 * 의 위치는 vCenter 버전에 따라 다릅니다.
- c. Properties * 표에서 port_remap_inbound 및 port_remap을 찾습니다.
- d. 포트의 인바운드 및 아웃바운드 통신을 대칭적으로 매핑하려면 * port_remap * 을 선택합니다.



포트 리매핑 지원은 더 이상 제공되지 않으며 향후 릴리스에서 제거될 예정입니다. 재매핑된 포트를 제거하려면 다음을 참조하세요. "베어 메탈 호스트에서 포트 재매핑을 제거합니다" .



port_remap 만 설정된 경우 지정하는 매핑이 인바운드 및 아웃바운드 통신 모두에 적용됩니다. port_remap_inbound 도 지정된 경우 port_remap 은 아웃바운드 통신에만 적용됩니다.

- i. Set Value * 를 선택합니다.
- ii. 포트 매핑을 입력합니다.

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

<network type> 그리드, 관리자 또는 클라이언트이며 <protocol> TCP 또는 UDP입니다.

예를 들어 포트 22에서 포트 3022로 ssh 트래픽을 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. OK * 를 선택합니다.
- e. 노드에 대한 인바운드 통신에 사용되는 포트를 지정하려면 * port_remap_inbound * 를 선택합니다.



port_remap_inbound 를 지정하고 port_remap 의 값을 지정하지 않으면 포트의 아웃바운드 통신이 변경되지 않습니다.

- i. Set Value * 를 선택합니다.
- ii. 포트 매핑을 입력합니다.

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

<network type> 그리드, 관리자 또는 클라이언트이며 <protocol> TCP 또는 UDP입니다.

예를 들어, 포트 3022로 전송된 인바운드 SSH 트래픽을 그리드 노드가 포트 22에서 수신하도록 재매핑하려면 다음을 입력합니다.

```
client/tcp/3022/22
```

심표로 구분된 목록을 사용하여 여러 인바운드 포트를 다시 매핑할 수 있습니다.

예를 들면 다음과 같습니다.

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

- i. OK * 를 선택합니다

12. 노드의 CPU 또는 메모리를 기본 설정에서 늘리려면 다음을 수행합니다.

- a. 가상 컴퓨터를 마우스 오른쪽 단추로 클릭하고 * 설정 편집 * 을 선택합니다.
- b. 필요한 경우 CPU 수 또는 메모리 양을 변경합니다.

메모리 예약 * 을 가상 머신에 할당된 * 메모리 * 와 동일한 크기로 설정합니다.

- c. OK * 를 선택합니다.

13. 가상 머신의 전원을 켭니다.

작업을 마친 후

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

예제 노드 구성 파일(Linux)

예제 노드 구성 파일을 사용하여 StorageGRID 시스템의 노드 구성 파일을 설정할 수 있습니다. 이 예제에서는 모든 유형의 그리드 노드에 대한 노드 구성 파일을 보여 줍니다.



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

대부분의 노드의 경우 그리드 관리자 또는 설치 API를 사용하여 그리드를 구성할 때 관리 및 클라이언트 네트워크 주소

지정 정보(IP, 마스크, 게이트웨이 등)를 추가할 수 있습니다. 기본 관리 노드는 예외입니다. 그리드 네트워크가 라우팅되지 않는 등의 이유로 기본 관리 노드의 관리 네트워크 IP를 탐색하여 그리드 구성을 완료하려면 노드 구성 파일에서 기본 관리 노드에 대한 관리 네트워크 연결을 구성해야 합니다. 이 예제는 예 나와 있습니다.



이 예에서는 클라이언트 네트워크가 기본적으로 비활성화되어 있더라도 클라이언트 네트워크 타겟이 모범 사례로 구성되었습니다.

기본 관리자 노드의 예

- 파일 이름 예: `/etc/storagegrid/nodes/dc1-adm1.conf`
- 파일 내용 예: *

```
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 노드의 예

- 파일 이름 예: `*/etc/storagegrid/nodes/dc1-sn1.conf`
- 파일 내용 예: *

```
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 노드의 예

- 파일 이름 예: * /etc/storagegrid/nodes/dc1-gw1.conf
- 파일 내용 예: *

```
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
```

운영 관리자 노드가 아닌 노드의 예

- 파일 이름 예: * /etc/storagegrid/nodes/dc1-adm2.conf
- 파일 내용 예: *

```

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

```

StorageGRID 구성 검증(Linux)

각 StorageGRID 노드에 대해 에서 구성 파일을 만든 후에는 /etc/storagegrid/nodes 해당 파일의 내용을 확인해야 합니다.



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

구성 파일의 내용을 확인하려면 각 호스트에서 다음 명령을 실행합니다.

```
sudo storagegrid node validate all
```

파일이 올바른 경우, 예제에 표시된 대로 각 구성 파일에 대해 * Passed * 가 출력됩니다.



메타데이터 전용 노드에서 LUN을 하나만 사용하는 경우에는 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 명령의 또는 --quiet 옵션 storagegrid(예: storagegrid --quiet...)을 사용하여 이 출력을 표시하지 않을 수 있습니다 -q. 출력을 표시하지 않으면 구성 경고 또는 오류가 감지된 경우 명령에 0이 아닌 종료 값이 있는 것입니다.

구성 파일이 잘못된 경우, 이 예에서와 같이 문제가 * warning * 및 * error * 로 표시됩니다. 구성 오류가 발견되면 설치를 계속하기 전에 오류를 수정해야 합니다.

```
Checking for misnamed node configuration files...
WARNING: ignoring /etc/storagegrid/nodes/dc1-adml
WARNING: ignoring /etc/storagegrid/nodes/dc1-sn2.conf.keep
WARNING: ignoring /etc/storagegrid/nodes/my-file.txt
Checking configuration file for node dc1-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 dc1-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 dc1-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 dc1-sn2... PASSED
Checking configuration file for node dc1-sn3... PASSED
Checking for duplication of unique values between nodes...
ERROR: GRID_NETWORK_IP = 10.1.0.4
      dc1-sn2 and dc1-sn3 have the same GRID_NETWORK_IP
ERROR: BLOCK_DEVICE_VAR_LOCAL = /dev/mapper/sgws-sn2-var-local
      dc1-sn2 and dc1-sn3 have the same BLOCK_DEVICE_VAR_LOCAL
ERROR: BLOCK_DEVICE_RANGEDB_00 = /dev/mapper/sgws-sn2-rangedb-0
      dc1-sn2 and dc1-sn3 have the same BLOCK_DEVICE_RANGEDB_00
```

StorageGRID 호스트 서비스 시작(Linux)

StorageGRID 노드를 시작하고 호스트를 재부팅한 후 다시 시작하려면 StorageGRID 호스트 서비스를 설정하고 시작해야 합니다.



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

단계

1. 각 호스트에서 다음 명령을 실행합니다.

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

2. 다음 명령을 실행하여 구축이 진행되고 있는지 확인합니다.

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

3. 노드가 "not running" 또는 "stopped" 상태를 반환하는 경우 다음 명령을 실행합니다.

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

4. 이전에 StorageGRID 호스트 서비스를 설정 및 시작한 경우(또는 서비스가 활성화 및 시작되었는지 확실하지 않은 경우) 다음 명령을 실행합니다.

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

REST API 설치

StorageGRID는 설치 작업을 수행하기 위한 StorageGRID 설치 API를 제공합니다.

API는 Swagger 오픈 소스 API 플랫폼을 사용하여 API 문서를 제공합니다. swagger를 사용하면 개발자와 개발자가 아닌 사용자가 API가 매개 변수 및 옵션에 어떻게 응답하는지 보여주는 사용자 인터페이스에서 API와 상호 작용할 수 있습니다. 이 문서에서는 표준 웹 기술 및 JSON 데이터 형식에 대해 잘 알고 있다고 가정합니다.



API 문서 웹 페이지를 사용하여 수행하는 모든 API 작업은 라이브 작업입니다. 실수로 구성 데이터나 기타 데이터를 작성, 업데이트 또는 삭제하지 않도록 주의하십시오.

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

StorageGRID 설치 API

StorageGRID 설치 API는 StorageGRID 시스템을 처음 구성할 때와 기본 관리자 노드 복구를 수행해야 하는 경우에만 사용할 수 있습니다. 설치 API는 Grid Manager에서 HTTPS를 통해 액세스할 수 있습니다.

API 설명서에 액세스하려면 기본 관리 노드의 설치 웹 페이지로 이동하여 메뉴 모음에서 * 도움말 * > * API 설명서 * 를 선택합니다.

StorageGRID 설치 API에는 다음 섹션이 포함되어 있습니다.

- * config * — 제품 릴리스 및 API 버전과 관련된 작업. 제품 릴리스 버전과 해당 릴리스에서 지원하는 API의 주요

버전을 나열할 수 있습니다.

- * 그리드 * — 그리드 레벨 구성 작업. 그리드 세부 정보, 그리드 네트워크 서브넷, 그리드 암호, NTP 및 DNS 서버 IP 주소를 포함한 그리드 설정을 얻고 업데이트할 수 있습니다.
- * 노드 * — 노드 레벨 구성 작업. 그리드 노드 목록을 검색하고, 그리드 노드를 삭제하고, 그리드 노드를 구성하고, 그리드 노드를 보고, 그리드 노드의 구성을 재설정할 수 있습니다.
- * 프로비저닝 * — 프로비저닝 작업. 프로비저닝 작업을 시작하고 프로비저닝 작업의 상태를 볼 수 있습니다.
- * 복구 * — 기본 관리 노드 복구 작업 정보를 재설정하고, 복구 패키지를 업로드하고, 복구를 시작하고, 복구 작업의 상태를 볼 수 있습니다.
- * recovery-package * — 복구 패키지를 다운로드하기 위한 작업.
- * 사이트 * — 사이트 수준 구성 작업. 사이트를 만들고, 보고, 삭제하고, 수정할 수 있습니다.
- * Temporary-password * — 설치 중 mgmt-API를 보호하기 위한 임시 암호의 작동.

설치 문제를 해결합니다

StorageGRID 시스템을 설치하는 동안 문제가 발생하면 설치 로그 파일에 액세스할 수 있습니다. 기술 지원 부서에서는 설치 로그 파일을 사용하여 문제를 해결해야 할 수도 있습니다.



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

리눅스

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

- /var/local/log/install.log (모든 그리드 노드에 있음)
- /var/local/log/gdu-server.log (기본 관리자 노드에서 찾을 수 있음)

호스트에서 다음 설치 로그 파일을 사용할 수 있습니다.

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

로그 파일에 액세스하는 방법에 대한 자세한 내용은 ["로그 파일 및 시스템 데이터를 수집합니다"](#) 참조하십시오.

VMware

다음은 문제를 해결하기 위해 기술 지원 부서에서 필요로 하는 기본 설치 로그 파일입니다.

- /var/local/log/install.log (모든 그리드 노드에 있음)
- /var/local/log/gdu-server.log (기본 관리자 노드에서 찾을 수 있음)

가상 머신 리소스 예약에 조정이 필요합니다

OVF 파일에는 각 그리드 노드가 효율적으로 작동할 수 있는 충분한 RAM 및 CPU를 포함하도록 설계된 리소스 예약이 포함됩니다. VMware에 이러한 OVF 파일을 구축하여 가상 머신을 생성하는 경우 미리 정의된 리소스 수를 사용할 수 없으면 가상 머신이 시작되지 않습니다.

이 작업에 대해

VM 호스트에 각 그리드 노드에 대한 리소스가 충분하다고 확인하는 경우 각 가상 머신에 할당된 리소스를 수동으로 조정하면 다음 가상 머신을 시작합니다.

단계

1. VMware vSphere 하이퍼바이저 클라이언트 트리에서 시작되지 않은 가상 머신을 선택합니다.
2. 가상 머신을 마우스 오른쪽 버튼으로 클릭하고 * 설정 편집 * 을 선택합니다.
3. 가상 머신 속성 창에서 * 리소스 * 탭을 선택합니다.
4. 가상 머신에 할당된 리소스를 조정합니다.
 - a. CPU * 를 선택한 다음 예약 슬라이더를 사용하여 이 가상 머신에 예약된 MHz를 조정합니다.
 - b. Memory*를 선택한 다음 Reservation 슬라이더를 사용하여 이 가상 머신에 예약된 MB를 조정합니다.
5. 확인 * 을 클릭합니다.
6. 동일한 VM 호스트에서 호스팅되는 다른 가상 머신에 대해 필요에 따라 이 작업을 반복합니다.

임시 설치 비밀번호가 비활성화되었습니다

VMware 노드를 구축할 때 필요한 경우 임시 설치 암호를 지정할 수 있습니다. VM 콘솔에 액세스하거나 새 노드가 그리드에 합류하기 전에 SSH를 사용하려면 이 암호가 있어야 합니다.

임시 설치 암호를 사용하지 않도록 선택한 경우 설치 문제를 디버깅하기 위한 추가 단계를 수행해야 합니다.

다음 중 하나를 수행할 수 있습니다.

- VM을 재배포되 콘솔에 액세스하거나 SSH를 사용하여 설치 문제를 디버깅할 수 있도록 임시 설치 암호를 지정합니다.
- vCenter를 사용하여 암호를 설정합니다.
 - a. VM의 전원을 끕니다.
 - b. vm * 으로 이동하고 * Configure * 탭을 선택한 다음 * vApp Options * 를 선택합니다.
 - c. 설정할 임시 설치 암호 유형 지정:
 - 사용자 지정 임시 암호를 설정하려면 * custom_temporary_password * 를 선택하십시오.
 - 노드 이름을 임시 암호로 사용하려면 * Temporary_password_type * 을 선택하십시오.
 - d. Set Value * 를 선택합니다.
 - e. 임시 암호를 설정합니다.
 - custom_Temporary_password * 를 사용자 정의 암호 값으로 변경합니다.
 - use node name * 값으로 * Temporary_password_type * 을 업데이트합니다.
 - f. VM을 다시 시작하여 새 암호를 적용합니다.

관련 정보

- 로그 파일에 액세스하는 방법에 대한 자세한 내용은 ["로그 파일 참조"](#)참조하십시오.
- ["StorageGRID 시스템 문제를 해결합니다"](#)
- 추가 도움이 필요한 경우 ["NetApp 지원"](#)문의하십시오.

스크립트 예제

예제 /etc/sysconfig/network-scripts(RHEL)

예제 파일을 사용하여 4개의 Linux 물리적 인터페이스를 단일 LACP 결합으로 집계한 다음 StorageGRID 그리드, 관리 및 클라이언트 네트워크 인터페이스로 사용할 본드를 포함하는 3개의 VLAN 인터페이스를 설정할 수 있습니다.

물리적 인터페이스

링크의 다른 쪽 끝에 있는 스위치도 4개의 포트를 단일 LACP 트렁크 또는 포트 채널로 처리해야 하며, 태그가 있는 3개 이상의 참조된 VLAN을 통과해야 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
```

본드 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
```

예시 /etc/network/interfaces(Ubuntu 및 Debian)

이 /etc/network/interfaces 파일에는 물리적 인터페이스, 본드 인터페이스 및 VLAN 인터페이스를 정의하는 세 개의 섹션이 포함되어 있습니다. 3개의 예제 섹션을 단일 파일로 결합하여 4개의 Linux 물리적 인터페이스를 단일 LACP 결합으로 통합한 다음 StorageGRID 그리드, 관리 및 클라이언트 네트워크 인터페이스로 사용할 수 있는 결합을 형성하는 3개의 VLAN 인터페이스를 설정합니다.

물리적 인터페이스

링크의 다른 쪽 끝에 있는 스위치도 4개의 포트를 단일 LACP 트렁크 또는 포트 채널로 처리해야 하며, 태그가 있는 3개 이상의 참조된 VLAN을 통과해야 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
```

본드 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.