



AWS Snowcone

AWS Snowcone 사용자 가이드



AWS Snowcone 사용자 가이드: AWS Snowcone

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께 사용하여 고객에게 혼란을 초래하거나 Amazon을 폄하 또는 브랜드 이미지에 악영향을 끼치는 목적으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

무엇입니까 AWS Snowcone?	1
사용 사례	2
요금	2
AWS Snowcone 작동 방식	3
AWS Snowcone 워크플로우	3
가져오기 작업 작동 방식	5
Snowcone과 AWS 스토리지 서비스 간의 온라인 데이터 전송 DataSync	6
Snowcone과 Amazon S3 간에 오프라인 데이터 전송	6
가져오기 작업 S3 스토리지	7
컴퓨팅 작업 스토리지	7
디바이스 사양	8
요약	8
하드웨어 및 네트워크	9
상위 보기	10
후면 패널	10
전면 패널	11
AWS Snowcone Wi-Fi 사양	12
견고화 사양	12
Snowcone 전원 공급 장치 및 액세서리	13
디스크 및 CPU 성능	13
설정	14
AWS에 가입	14
주문하기 전에	15
로컬 환경에 대하여	15
특수 문자로 작업	15
Amazon EC2	17
Snow Family 디바이스상의 Amazon EC2와 Amazon EC2 호환 인스턴스 간 차이	18
Snowcone에서의 컴퓨팅 인스턴스 요금	18
사전 조건	18
AMI의 제품 코드 및 플랫폼 세부 정보 확인 AWS Marketplace	18
인스턴스에서 Linux AMI 생성	20
스냅샷에서 Linux AMI 생성	20
시작하기	24
Snow Family 디바이스 주문 작업 생성	24

1단계: 작업 유형 선택	25
2단계: 컴퓨팅 및 스토리지 옵션 선택	26
3단계: 기능 및 옵션 선택	28
4단계: 보안, 배송 및 알림 기본 설정 선택	29
5단계: 작업 요약 검토 및 작업 생성	31
다운로드 AWS OpsHub	32
Snow Family 디바이스 주문 작업 취소	33
Snow Family 디바이스에 액세스하기 위한 자격 증명 받기	33
Snow Family 디바이스 잠금 해제	34
Snow Family 디바이스 잠금 해제 문제 해결	36
Snow Family 디바이스 재부팅	36
장치 AWS OpsHub 관리에 사용	42
Snow 패밀리 AWS OpsHub 디바이스용 다운로드	43
디바이스 잠금 해제	43
로컬에서 디바이스 잠금 해제	44
원격으로 디바이스 잠금 해제	47
의 서명 확인 AWS OpsHub	51
AWS 서비스 관리	54
로컬에서 컴퓨팅 인스턴스 사용	55
NFS 인터페이스 관리	63
파일을 전송하는 AWS DataSync 데 사용 AWS	70
파일 전송	72
디바이스 관리	72
디바이스 재부팅	73
디바이스 종료하기	74
디바이스 별칭 편집	75
업데이트 받기	76
프로필 관리	77
디바이스의 NTP 시간 서버 설정	79
AWS Snowball Edge 클라이언트 사용	81
Snowball Edge 클라이언트 다운로드 및 설치	81
Snowball Edge 클라이언트에 대한 명령	81
Snowball Edge 클라이언트의 프로필 구성	82
NFC 검증에 대한 QR 코드 가져오기	84
장치 잠금 해제 AWS Snowcone	84
Snowcone 업데이트	85

자격 증명 받기	88
Snowcone 디바이스에서 서비스 시작하기	89
Snowcone 디바이스에서 서비스 중지하기	90
데이터를 전송하기 위해 인증서 가져오기	90
AWS Snowcone 로그	91
디바이스 상태 가져오기	93
서비스 상태 가져오기	95
AWS DataSync AMI 런칭	96
NFS 시작 및 액세스 제한	99
NFS 실행 시 NFS 공유에 대한 액세스 제한	100
Amazon S3 버킷의 내보내기 경로 가져오기	100
로컬 AWS 운영자 디버깅 활성화	101
로컬 운영자 디버깅 비활성화 AWS	101
다이렉트 네트워크 인터페이스 생성	101
다이렉트 네트워크 인터페이스에 대한 정보 가져오기	102
다이렉트 네트워크 인터페이스 업데이트	102
다이렉트 네트워크 인터페이스 삭제	102
기능 상태 확인	103
기능 상태 변경	103
타임 서버 설정	104
시간 소스 확인	105
Snow Device Management를 사용하여 디바이스 관리	108
Snow Family 디바이스를 주문할 때 Snow 디바이스 관리 상태 선택	108
Snow 디바이스 관리 활성화	110
Snow 디바이스 관리에 대한 권한을 IAM 역할에 추가	111
Snow Device Management CLI 명령	111
작업 생성	112
태스크 상태 확인	113
디바이스 정보 확인	114
Amazon EC2 호환 인스턴스 상태 확인	117
태스크 메타데이터 확인	119
태스크 취소	120
명령 및 구문 나열	121
원격 관리 가능 디바이스 나열	121
디바이스 전반의 태스크 상태 나열	122
사용 가능한 리소스 나열	124

디바이스 또는 태스크 태그 나열	125
상태별로 태스크 나열	126
태그 적용	127
태그 제거	127
AWS 서비스 사용	129
컴퓨팅에 Amazon EC2 사용	129
개요	130
사전 조건	131
컴퓨팅 인스턴스가 있는 작업 생성	132
컴퓨팅 인스턴스에 대한 네트워크 구성	136
SSH를 사용하여 Snowcone의 컴퓨팅 인스턴스에 연결	143
컴퓨팅 인스턴스에 대한 Snowcone 클라이언트 명령	143
Amazon EC2 호환 인스턴스에서 Snow용 인스턴스 메타데이터 서비스 사용	149
Amazon EC2 호환 엔드포인트 사용	158
시작 템플릿을 사용하여 Amazon EC2 호환 인스턴스 자동 시작	174
EC2 호환 인스턴스에 블록 스토리지 사용	175
Snow 디바이스의 보안 그룹	176
지원되는 인스턴스 메타데이터 및 사용자 데이터	177
Amazon EC2 문제 해결	179
DataSync를 사용하여 파일 전송	180
NFS 인터페이스 관리	182
Snow Family 디바이스의 NFS 구성	183
Amazon EC2 호환 인스턴스에서 AWS IoT Greengrass 사용	187
Amazon EC2 호환 인스턴스 설정	188
AWS 서비스를 사용하는 데 필요한 포트	191
Snowcone 디바이스 반납	192
Snowcone 디바이스 연결 해제	192
디바이스에서 데이터 보호	193
AWS Snowcone 보호	193
NFC 태그 검증	194
작업 상태 이해	196
알림	198
Snow가 Amazon SNS를 사용하는 방법	198
Snow 작업 상태 변경에 대한 SNS 주제 암호화	198
고객 관리형 KMS 키 정책 설정	199
SNS 알림 예시	200

주문 프로세스에 대한 이해	213
배송 프로세스에 대한 이해	213
Snowcone 디바이스 반납	213
AWS Management Console 사용	213
콘솔에서 Snowcone 주문	214
작업 관리 API 사용	214
JMAPI의 일반적인 용도	214
JMAPI 필수 문자열	214
JMAPI 엔드포인트	215
JMAPI CLI 명령	215
예시	216
EC2 작업	217
디바이스에서 시작된 컴퓨팅 인스턴스에 연결되도록 AMI를 구성하여 SSH 사용	217
콘솔을 사용하여 작업 생성	218
AWS CLI에서 작업 생성	219
배송 고려 사항	220
AWS Snowcone 디바이스 배송 준비	220
리전 기반 배송 제한	221
AWS Snowcone 디바이스 배송	221
배송 업체	222
Snowcone 디바이스 업데이트	229
필수 조건	230
업데이트 다운로드	230
업데이트 설치	232
SSL 인증서 업데이트	235
Snow Family 디바이스의 Amazon Linux 2 AMI 업데이트	236
모범 사례	237
보안	237
네트워크	237
리소스 관리	238
EC2 호환 인스턴스 관리	238
성능	239
Snowcone 할당량	240
컴퓨팅 리소스 할당량	240
Snowcone 디바이스 배송에 대한 제한	241
반환된 Snowcone 디바이스를 가져오기 위한 처리 제한	241

사용 가능 AWS 리전	242
문제 해결	243
컴퓨팅 문제 해결	243
네트워크 문제	243
IP 주소는 0.0.0.0입니다.	243
디바이스를 잠금 해제할 수 없습니다.	243
Datsync의 EC2 호환 인스턴스 문제	244
오류: 인스턴스 시작 실패	244
데이터 전송 문제	244
서버에 의한 액세스 거부	244
데이터 전송 중 연결 시간 초과	244
Showmount ENOENT 생성	245
Snow Family 디바이스 반품 문제 해결	245
API 참조	247
문서 기록	248
AWS 용어집	253
.....	ccliv

무엇입니까 AWS Snowcone?

AWS Snowcone 엣지 컴퓨팅 및 데이터 전송을 위한 견고하고 안전한 휴대용 장치입니다. Snowcone 디바이스를 사용하여 데이터를 수집, 처리 및 이동할 수 있습니다. 디바이스를 오프라인으로 배송하여 오프라인으로 전송하거나 AWS 사용하여 온라인으로 데이터를 이동할 수 있습니다. AWS 클라우드 AWS DataSync

엄격한 (데이터 센터가 아닌) 엣지 환경이나 일관된 네트워크 연결이 부족한 환경에서 애플리케이션을 실행하는 것은 어려울 수 있습니다. 이러한 위치에는 데이터 센터 IT 장비에 필요한 공간, 전력 및 냉각이 부족한 경우가 많습니다.

Snowcone은 두 가지 버전으로 제공됩니다.

- Snowcone - Snowcone은 vCPU 2개, 4GB 메모리, 8TB의 하드 디스크 드라이브(HDD) 기반 스토리지를 갖추고 있습니다.
- 스노우콘 SSD - Snowcone SSD는 두 개의 vCPU, 4GB의 메모리, 14TB의 솔리드 스테이트 드라이브(SSD) 기반 스토리지를 갖추고 있습니다.

두 개의 CPU와 테라바이트의 스토리지를 갖춘 Snowcone 디바이스는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스를 사용하는 엣지 컴퓨팅 워크로드를 실행하고 데이터를 안전하게 저장할 수 있습니다.

Snowcone 장치는 크기가 작기 때문에 (8.94인치 x 5.85인치 x 3.25인치/227mm x 148.6mm x 82.65mm) 공장의 기계 옆에 배치하여 데이터를 수집, 포맷하고 저장 및 분석을 위해 다시 전송할 수 있습니다. AWS Snowcone 디바이스의 무게는 약 2kg(4.5파운드)이므로 배낭에 넣고 다니면서 배터리 기반 작동 방식으로 사용하고 Wi-Fi 인터페이스를 사용하여 센서 데이터를 수집할 수 있습니다.

Note

Wi-Fi는 북미에서만 사용할 수 있습니다. AWS 리전

Snowcone 디바이스는 네트워크 파일 시스템 (NFS) 을 지원하는 인터페이스를 제공합니다.

Snowcone 디바이스는 NFS 인터페이스를 통해 온프레미스 Windows, Linux, macOS 서버 및 파일 기반 애플리케이션에서 데이터 전송을 지원합니다.

AWS Snowball마찬가지로 AWS Snowcone 에는 여러 계층의 보안 암호화 기능이 있습니다. 이러한 서비스 중 하나를 사용하여 데이터를 수집, 처리 및 전송하고 Amazon EC2 호환 인스턴스를 사용하는 엣

지 컴퓨팅 워크로드를 실행할 수 있습니다. AWS Snowcone은 최대 수십 테라바이트의 데이터 마이그레이션 요구 사항을 충족하도록 설계되었으며, Snowball Edge 디바이스가 맞지 않는 공간이 제한된 환경에서 사용할 수 있습니다.

사용 사례

다음과 같은 사용 사례에 AWS Snowcone 디바이스를 사용할 수 있습니다.

- 엣지 컴퓨팅 애플리케이션의 경우 데이터를 수집하고 처리하여 즉각적인 인사이트를 확보한 다음 온라인에서 AWS로 데이터를 전송합니다.
- 센서 또는 AWS 기계에서 지속적으로 생성되는 데이터를 온라인으로 공장이나 다른 엣지 로케이션으로 전송합니다.
- AWS 스토리지 서비스의 미디어, 과학 또는 기타 콘텐츠를 파트너 및 고객에게 배포합니다.
- 엣지 로케이션의 미디어, 과학 또는 기타 콘텐츠를 AWS로 전송하여 콘텐츠를 집계합니다.
- 데이터를 전송할 준비가 된 일회성 데이터 마이그레이션 시나리오의 경우, Snowcone은 장치를 다시 배송하여 최대 8TB 또는 14TB의 데이터를 빠르고 저렴하게 전송할 수 있는 AWS 클라우드 방법을 제공합니다. AWS

모바일 배포의 경우, Snowcone 디바이스는 지정된 배터리 전원으로 실행될 수 있습니다. CPU 사용량이 25%로 워크로드가 적은 경우 배터리로 최대 약 6시간 동안 디바이스를 실행할 수 있습니다. Snowcone 디바이스의 Wi-Fi 인터페이스를 사용하여 무선 센서에서 데이터를 수집할 수 있습니다. AWS Snowcone 장치는 저전력, 휴대용, 경량 및 진동 방지 기능을 갖추고 있어 다양한 원격 및 열악한 위치에서 사용할 수 있습니다.

Note

Wi-Fi는 AWS 리전 북미에서만 사용할 수 있습니다.

요금

Snowcone 디바이스를 주문하여 사용량에 따라 지불하면 디바이스를 최대 4년간 보유할 수 있습니다. AWS Snowcone 가격 및 수수료에 대한 자세한 내용은 [AWS Snowcone 가격을](#) 참조하십시오.

AWS Snowcone 작동 방식

AWS Snowcone 은 엣지 컴퓨팅 및 데이터 전송을 위한 휴대용 디바이스입니다. 시작하려면 전송해야 하는 데이터의 양과 필요한 컴퓨팅 성능에 AWS Management Console 따라 Snowcone 디바이스를 하나 이상 요청합니다. 선택한 Amazon Simple Storage Service(S3) 버킷, 데이터, Amazon Elastic Compute Cloud(Amazon EC2) Amazon Machine Image(AMI)는 자동으로 구성, 암호화되어 디바이스에 사전 설치됩니다. 또한 디바이스가 배송되기 전에 AWS DataSync 에이전트가 사전 설치됩니다.

디바이스가 도착하면 디바이스를 온프레미스 네트워크에 연결하고 Dynamic Host Configuration Protocol(DHCP)을 사용하여 수동 또는 자동으로 IP 주소를 설정합니다. Snowcone 장치를 관리하기 위한 그래픽 사용자 인터페이스 (GUI) 애플리케이션을 다운로드하여 AWS OpsHub for Snow Family 설치해야 합니다. 랩톱과 같은 Windows 또는 macOS 클라이언트 머신에 설치할 수 있습니다.

디바이스를 열고 잠금을 해제하면 AWS OpsHub 디바이스와 시스템 메트릭을 보여주는 대시보드가 표시됩니다. 그런 다음 AWS OpsHub에서 몇 번의 클릭만으로 인스턴스를 시작하여 엣지 애플리케이션을 배포하거나 데이터를 디바이스로 마이그레이션할 수 있습니다.

컴퓨팅 또는 데이터 전송 작업이 완료되고 디바이스를 반품할 준비가 되면 E Ink 배송 라벨이 자동으로 반송 주소를 업데이트하여 Snowcone 디바이스가 올바른 AWS 시설로 배송되도록 합니다. 디바이스가 배송되면 Amazon Simple Notification Service(SNS)에서 보낸 메시지, 생성된 문자 및 이메일을 통해 또는 콘솔에서 직접 추적 상태를 확인할 수 있습니다.

주제

- [AWS Snowcone 워크플로우](#)
- [가져오기 작업 작동 방식](#)
- [가져오기 작업 S3 스토리지](#)
- [컴퓨팅 작업 스토리지](#)

AWS Snowcone 워크플로우

세 가지 작업 유형을 생성할 수 있습니다. 사용 사례마다 작업 유형이 다르지만 모두 다음과 같은 디바이스 주문, 수령 및 반환 워크플로를 갖추고 있습니다.

워크플로

1. 작업 생성 - 작업 관리 API를 통해 AWS Snow 패밀리 관리 콘솔 또는 프로그래밍 방식으로 각 작업을 생성하고 사용 사례에 따라 디바이스 유형(Snowcone 또는 Snowcone SSD)을 선택합니다.

Snowcone API에서 AWS Management Console 또는 Snowcone API를 통해 작업 상태를 추적할 수 있습니다.

2. 장치가 작업에 맞게 준비되었습니다. 작업을 위해 AWS Snowcone 장치를 AWS 준비하면 작업 상태가 Snowcone 준비로 변경됩니다.
3. 리전의 운송업체가 디바이스를 배송 - 운송업체가 해당 위치에서 수거한 것으로, 현재 작업 상태는 배송 중으로 나타납니다. AWS Snow 패밀리 관리 콘솔 콘솔이나 작업 관리 API에서 추적 번호와 추적 웹 사이트 링크를 찾을 수 있습니다. 해당 리전의 운송업체에 대한 자세한 내용은 [AWS Snowcone에 대한 배송 고려 사항](#) 섹션을 참조하세요.
4. Snowcone 전원 공급 장치 구입 - 탄소 배출량을 최소화하기 위해 Snowcone 디바이스는 전원 공급 장치와 함께 배송되지 않습니다. Snowcone은 45와트 USB-C 연결 전원 공급 장치를 사용합니다. 휴대용 배터리로도 전원을 공급할 수 있습니다. 자세한 설명은 [AWS Snowcone 전원 공급 장치 및 액세스리](#) 섹션을 참조하세요.
5. 디바이스 수령 — 며칠 후 해당 지역의 배송사가 작업을 생성할 때 제공한 주소로 AWS Snowcone 디바이스를 배송합니다. 작업 상태가 배송됨으로 변경됩니다. 디바이스가 자체 배송 컨테이너이기 때문에 상자에 담겨 도착하지 않습니다.
6. 자격 증명을 받고 Snow Family 애플리케이션용 Snowball Edge 클라이언트를 다운로드하십시오. 자격 증명, 작업 매니페스트, 매니페스트의 잠금 해제 코드를 가져온 다음 Snowball Edge 클라이언트를 다운로드하여 데이터 전송을 시작할 준비를 하십시오. AWS OpsHub
 - 디바이스가 해당 위치의 온프레미스에 있을 때 콘솔이나 작업 관리 API에서 매니페스트를 가져옵니다. 매니페스트는 디바이스에 대한 액세스를 인증하는 데 사용됩니다. 매니페스트는 암호화되어 있기 때문에 잠금 해제 코드가 있어야만 해독할 수 있습니다.
 - 잠금 해제 코드는 매니페스트의 암호를 해제하는 데 사용되는 29자 코드입니다. 콘솔이나 작업 관리 API에서 잠금 해제 코드를 가져올 수 있습니다. 시설에 있는 동안 디바이스에 대한 무단 액세스를 방지하려면 잠금 해제 코드를 매니페스트 위치와 다른 안전한 장소에 보관하는 것이 좋습니다.
 - AWS OpsHub for Snow Family Snowcone을 포함한 Snow Family 디바이스를 관리하기 위한 애플리케이션입니다. AWS OpsHub for Snow Family GUI를 사용하면 Snowcone 디바이스를 설정하고 관리할 수 있으므로 엣지 컴퓨팅 워크로드를 신속하게 실행하고 데이터를 AWS 클라우드로 마이그레이션할 수 있습니다. 몇 번의 클릭만으로 Snowcone 장치를 잠금 해제 및 구성하고, 데이터를 드래그 앤 드롭하고, 애플리케이션을 시작하거나, 장치 지표를 모니터링하는 데 사용할 AWS OpsHub 수 있습니다. 랩톱과 같은 Windows 또는 macOS 클라이언트 머신에 설치할 수 있습니다. AWS OpsHub사용은 무료입니다.

AWS OpsHub [AWS Snowball 리소스에서](#) 다운로드하십시오. 에 대한 자세한 내용은 AWS OpsHub을 참조하십시오 [장치 AWS OpsHub for Snow Family 관리에 사용](#).

- Snowball Edge 클라이언트는 디바이스에서 온프레미스 데이터 대상 주소로 전송되는 데이터의 흐름을 관리하는 데 사용하는 도구입니다.
7. 하드웨어 배치 - 디바이스를 데이터 센터로 옮기고 케이스상의 지침에 따라 개봉합니다. 디바이스를 전원 공급 장치 및 로컬 네트워크에 연결합니다.
 8. 디바이스 전원 켜기 - 이제 LCD 디스플레이 위에 있는 전원 버튼을 눌러 디바이스를 켭니다. 몇 분 정도 기다리면 준비 완료 화면이 나타납니다.
 9. 디바이스의 IP 주소 가져오기 - LCD 디스플레이에는 연결 탭이 있습니다. 이 탭을 탭하면 AWS Snowcone 장치의 IP 주소를 확인할 수 있습니다.
 10. 사용 AWS OpsHub 장치 잠금 해제 - AWS Snowcone 장치를 잠금 해제하려면 장치의 IP 주소를 입력하고 매니페스트와 잠금 해제 코드를 업로드합니다. AWS OpsHub 매니페스트를 해독하고 이를 사용하여 디바이스에 대한 액세스를 인증합니다. 에 대한 자세한 내용은 [참조하십시오. AWS OpsHub 장치 AWS OpsHub for Snow Family 관리에 사용](#)
 11. 디바이스 사용 AWS OpsHub - 엣지 컴퓨팅 워크로드를 빠르게 실행하고 로 데이터를 전송할 수 있도록 AWS Snowcone 디바이스를 설정 및 관리하는 데 사용합니다. AWS 클라우드 몇 번의 클릭만으로 Snowcone 장치를 잠금 해제 및 구성하고, 데이터를 드래그 앤 드롭하고, 애플리케이션을 시작하거나, 장치 지표를 모니터링하는 데 사용할 AWS OpsHub 수 있습니다. 자세한 내용은 [장치 AWS OpsHub for Snow Family 관리에 사용](#) 섹션을 참조하세요.
 12. 이더넷 또는 전원 공급 장치 케이블 분리 금지 - 데이터 전송 또는 컴퓨팅 작업 중에는 이더넷 케이블이나 USB-C 전원 공급 장치 또는 배터리를 분리하면 안 됩니다. 데이터 전송 또는 컴퓨팅 작업이 완료된 후 Snowcone 디바이스를 끄려면 전원 버튼을 누릅니다.
 13. 왕복 여행을 위한 디바이스 준비 - 온프레미스 위치에서 디바이스 사용을 완료한 후 LCD 디스플레이 위의 전원 버튼을 눌러 디바이스의 전원을 끕니다. 디바이스의 플러그를 뽑고 전원 케이블을 디바이스 상단의 케이블 누크에 넣은 다음, 디바이스의 도어 세 개를 모두 닫습니다. 이제 디바이스는 반환 준비가 되었습니다.
 14. 해당 지역의 이동통신사에서 기기를 반납합니다. 이동 통신사에 AWS Snowcone 장치가 있으면 작업 상태가 '전송 중'으로 바뀝니다. AWS AWS

가져오기 작업 작동 방식

다음을 사용하여 Snowcone을 사용하여 장치와 AWS 스토리지 서비스 간에 온라인으로 데이터를 전송할 수 있습니다. AWS DataSync 또한 온프레미스 스토리지 디바이스에서 Snowcone 디바이스로 데이터를 오프라인으로 전송할 수 있습니다.

Snowcone과 AWS 스토리지 서비스 간의 온라인 데이터 전송 DataSync

[AWS DataSync](#) 인터넷이나 스토리지 서비스를 통해 대량의 데이터를 AWS 스토리지 서비스와 주고 받는 작업을 간소화, 자동화 및 가속화하는 온라인 데이터 전송 서비스입니다. AWS Direct Connect AWS DataSync 에이전트는 Snowcone 디바이스에 사전 설치되어 있으며, 에이전트와 Amazon S3 버킷, Amazon Elastic File System (Amazon EFS) 파일 시스템 및 Windows File Server용 Amazon FSx 간에 데이터를 전송하는 데 사용됩니다. DataSync 파일 및 객체 이동, 데이터 전송 예약, 전송 진행 상황 모니터링, 데이터 암호화, 데이터 전송 확인, 고객 문제 알림 등을 자동으로 처리합니다.

DataSync 에이전트는 Snowcone 작업 준비 중에 Amazon 머신 이미지 (AMI) 로 Snowcone 디바이스에 사전 설치됩니다. 온라인으로 데이터를 전송하려면 AWS Snowcone 디바이스를 외부 네트워크에 연결하고 AWS OpsHub for Snow Family 또는 AWS Command Line Interface (AWS CLI) 를 사용하여 에이전트 DataSync AMI를 시작합니다. AWS Management Console 또는 를 사용하여 DataSync 에이전트를 활성화합니다. AWS CLI그런 다음 AWS Snowcone 네트워크 파일 시스템 (NFS) 스토어와 Amazon S3, Amazon EFS 또는 Amazon FSx 간에 온라인 데이터 전송 작업을 설정합니다.

Snowcone에서 DataSync 실행하여 사용할 수 있는 용도는 다음과 같습니다.

- 엣지 컴퓨팅 애플리케이션의 경우 데이터를 수집하고 처리하여 즉각적인 인사이트를 확보한 다음 온라인에서 AWS로 데이터를 전송합니다.
- 센서 또는 머신에서 지속적으로 생성되는 데이터를 온라인으로 공장이나 다른 엣지 로케이션의 AWS 로 전송합니다.
- AWS 스토리지 서비스의 미디어, 과학 또는 기타 콘텐츠를 파트너와 고객에게 온라인으로 배포합니다.
- 미디어, 과학 또는 기타 콘텐츠를 엣지 로케이션에서 AWS로 온라인에서 전송하여 콘텐츠를 집계합니다.

일회성 엣지 컴퓨팅 또는 데이터 전송 워크플로우 또는 WAN (Wide Area Network) 링크가 없거나 WAN 대역폭이 충분하지 않은 엣지 로케이션의 Snowcone 워크플로의 경우 Snowcone 디바이스를 반송하여 데이터 전송을 AWS 완료하는 것이 좋습니다.

Snowcone과 Amazon S3 간에 오프라인 데이터 전송

오프라인 데이터 가져오기 작업의 경우, Snowcone 디바이스를 온-프레미스 네트워크에 연결한 다음 이를 AWS OpsHub 사용하여 디바이스를 잠금 해제합니다. [AWS Snowball 리소스](#) [AWS OpsHub](#) 페이지에서 다운로드하십시오. NFS 인터페이스를 통해 온프레미스 스토리지 디바이스의 데이터를 Snowcone 디바이스로 복사할 수 있습니다. 데이터를 Snowcone 디바이스로 복사한 후 디바이스의 E

Ink 배송 라벨은 디바이스가 자동으로 올바른 AWS 시설로 발송되도록 합니다. Amazon SNS에서 생성한 문자 메시지 또는 이메일과 콘솔을 사용하여 Snowcone 디바이스를 추적할 수 있습니다.

가져오기 작업 S3 스토리지

내부적으로 Snowcone 디바이스에는 8TB 또는 14TB의 디스크 스토리지가 포함되어 있습니다. 이 스토리지는 로컬 Amazon Elastic Block Store(Amazon EBS) 볼륨 프레젠테이션을 통해 내부 NFS(Network File System) 서비스 또는 로컬 Amazon EC2 호환 인스턴스와 함께 사용할 수 있습니다. NFS 스토리지 볼륨에는 8TB 또는 14TB를, 용량 최적화 HDD(sbg1) Amazon EBS 스토리지 볼륨에는 150GB를 사용할 수 있습니다.

컴퓨팅 작업 스토리지

작업 유형이 로컬 컴퓨팅인 경우 총 8TB 또는 14TB의 로컬 용량 최적화 HDD (sbg1) Amazon EBS 볼륨을 생성하여 Amazon EC2 호환 인스턴스에 연결할 수 있습니다. Amazon EBS 볼륨을 사용하면 루트 볼륨만 사용할 때보다 더 많은 로컬 용량에 로컬 Amazon EC2 호환 인스턴스가 액세스할 수 있습니다. 로컬 스토리지만 가능하기 때문에 Amazon S3로 가져올 수 AWS 없기 때문에 Amazon EBS 볼륨에 기록된 데이터는 디바이스가 반환될 때 손실됩니다.

Note

NFS 서버는 컴퓨팅 작업에 사용할 수 없습니다. AWS DataSync 에이전트를 실행하기 위해 OR 간에 데이터를 AWS 클라우드 가져오거나 내보내야 하는 경우, 주문할 때 로컬 컴퓨팅 작업 유형을 선택하지 마십시오.

AWS Snowcone 디바이스 사양

이 섹션에서는 하드웨어, 네트워크, Wi-Fi 및 전원 공급 장치의 AWS Snowcone 디바이스 사양 및 요구 사항에 대한 정보를 제공합니다.

주제

- [기능 및 사양 요약](#)
- [하드웨어 및 네트워크](#)
- [AWS Snowcone Wi-Fi 사양](#)
- [견고화 사양](#)
- [AWS Snowcone 전원 공급 장치 및 액세서리](#)
- [디스크 및 CPU 성능](#)

기능 및 사양 요약

다음 표에는 Snowcone 디바이스의 기능 및 사양이 요약되어 있습니다.

항목	사양
사용 시나리오	산업용 사물 인터넷(IoT), 운송, 의료 IoT, 콘텐츠 배포, 전술 엣지 컴퓨팅, 물류, 자율주행차, 데이터 마이그레이션
디바이스 크기	길이 9인치, 너비 6인치, 높이 3인치 (227mm x 148.6mm x 82.65mm)
디바이스 무게	스노우콘의 경우 4.5파운드 (2.04kg), 스노우콘 SSD의 경우 4.6파운드 (2.09kg)
스토리지 용량	Snowcone에는 8TB를, Snowcone SSD에는 14TB를 사용할 수 있습니다.
컴퓨팅 옵션 온보딩	Amazon EC2 Amazon Machine Image(AMI)
암호화(Encryption)	예, 256비트
NFS(Network File System)를 통한 전송	예

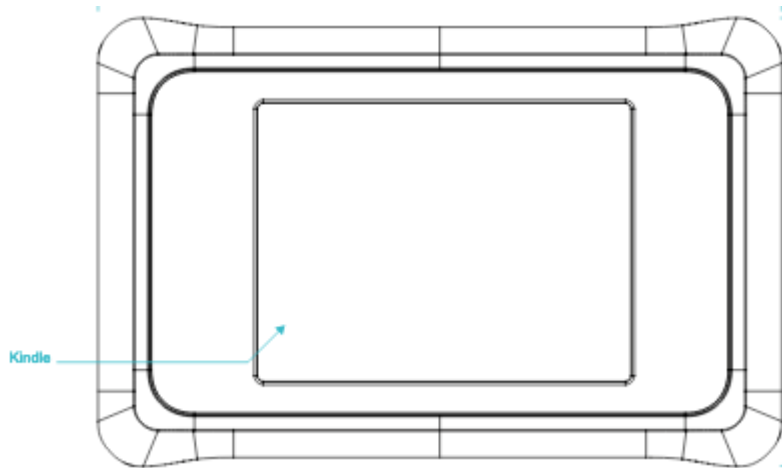
항목	사양
Amazon S3 API를 통한 전송	아니요
이동성	배터리 기반 작동
무선	Wi-Fi <div style="border: 1px solid #ccc; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Note Wi-Fi는 북미 지역 AWS 리전에서만 사용할 수 있습니다.</p> </div>
사용 가능한 vCPU 수	vCPU 2개
사용할 수 있는 메모리	4GB
네트워크 인터페이스	1/10기가비트(Gb) RJ45 2개
AWS DataSync 에이전트가 사전 설치됨	예
일반적인 작업 수명	오프라인 또는 온라인 데이터 전송: 며칠에서 몇 주 엣지 컴퓨팅: 몇 주에서 몇 달
최대 작업 길이	엣지 컴퓨팅 또는 지속적인 데이터 전송: 최대 360일

하드웨어 및 네트워크

Snowcone 디바이스는 8TB 또는 14TB의 사용 가능한 스토리지를 제공합니다. 애플리케이션과 AWS IoT Greengrass 기능을 지원하기 위해 사용 가능한 CPU 2개와 4GB의 사용 가능한 메모리가 있는 특정 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스를 실행합니다. 이 섹션에서는 디바이스의 다양한 뷰에 나타나는 인터페이스, 전원 버튼, 전원 요구 사항 등 물리적 디바이스에 대한 정보를 확인할 수 있습니다.

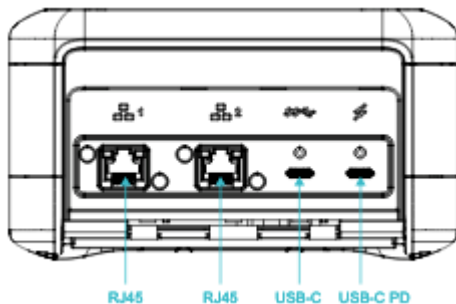
상위 보기

Snowcone 디바이스의 상단 표면에는 유선 및 무선 네트워킹을 모두 설정하기 위한 운영자 인터페이스로 사용되는 통합 E-Ink 터치 디스플레이가 있습니다. 또한 전자 배송 라벨의 디스플레이 역할도 합니다. 전자 배송 라벨에는 아웃바운드 및 인바운드 배송 라벨이 모두 사전 프로그래밍되어 있으며, 이 라벨은 디바이스의 전원을 처음 켜면 자동으로 변경됩니다.



후면 패널

모든 케이블을 후면 패널에서 연결합니다. 이 섹션에서는 각 커넥터에 대해 설명합니다.



Power

최소 45W를 공급할 수 있는 적절한 전원 어댑터를 사용하여 맨 오른쪽 USB-C 연결을 통해 디바이스에 전원이 공급됩니다.

Note

AWS Snowcone는 가능한 가장 작은 폼 팩터와 함께 제공되므로 전원 공급 장치는 포함되지 않습니다. 자세한 내용은 [AWS Snowcone 전원 공급 장치 및 액세서리](#) 섹션을 참조하십시오.

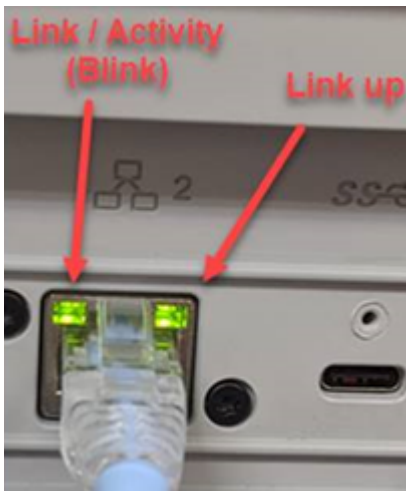
USB

첫 번째 USB-C 연결이 활성화되지 않았습니다.

이더넷 커넥터 1 및 2

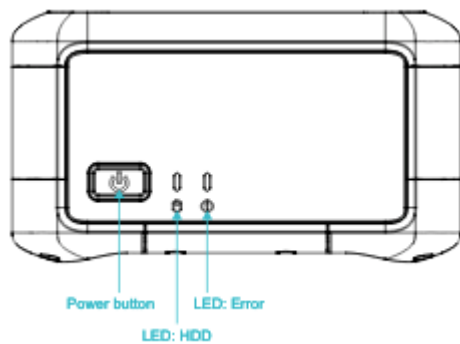
유선 네트워킹의 경우 Snowcone 디바이스는 1기가비트(Gb) 또는 10Gb 이더넷 네트워크에 대해 자동 협상되는 두 개의 포트를 제공합니다.

RJ45 10/1G Base-T 이더넷 포트 — 이 포트는 원거리 연결 기능에 따라 10Gb와 1Gb 사이에서 자동 협상됩니다. 1Gb 미만의 속도(예: 100Mb 또는 10Mb)는 지원하지 않습니다. 링크 LED는 각 커넥터의 오른쪽에 있고, 작동 LED는 왼쪽에 있습니다.



전면 패널

전면 패널에는 전원 버튼과 상태 LED 디스플레이가 있습니다.



전원 스위치

디바이스를 켜려면 전원 버튼을 잠시 누릅니다. 버튼이 켜지고 E-Ink 디스플레이가 진행률 표시줄로 바뀌고 기다려 주십시오라는 메시지가 표시됩니다.

디바이스를 끄려면 전원 스위치를 2초 동안 또는 E-Ink 디스플레이에 기다려 주십시오 표시될 때까지 길게 누릅니다.

Note

전원 버튼을 5초 이상 누르면 디바이스의 전원이 강제로 꺼집니다. 이렇게 하면 버퍼의 데이터가 손실될 수 있으므로 권장하지 않습니다. AC 전원이 끊긴 후 디바이스는 자동으로 전원을 마지막 작동 상태로 복원합니다.

상태 LED

전원 버튼 옆에는 두 개의 상태 LED가 있습니다. 디스크 작동과 함께 왼쪽 LED가 깜박이고 장애가 발생하면 오른쪽 LED가 켜집니다.

AWS Snowcone Wi-Fi 사양

AWS Snowcone는 IEEE 802.11ac 표준을 사용하는 기가비트 Wi-Fi 네트워크와 802.11abgn 네트워크를 지원합니다. Snowcone의 무선 처리량을 극대화하려면 2x2 802.11ac 160MHz 채널을 사용합니다. 이 채널은 기본 1x1 802.11bgn 네트워크보다 최대 10배 더 빠를 수 있습니다. Snowcone Wi-Fi 사양은 IEEE 802.11abgn+ac, 2x2, MIMO, 듀얼 밴드 2GHz 및 5G(160MHz)입니다. 작동 주파수는 2GHz와 5GHz입니다.

물리적 이더넷 케이블을 사용하여 Snowcone을 온프레미스 네트워크에 연결하거나 Wi-Fi를 사용하여 무선으로 연결할 수 있습니다. Wi-Fi 연결을 통해 AWS OpsHub를 사용하여 Snowcone 디바이스를 관리하고 Snowcone과 온프레미스 스토리지 디바이스 간에 데이터를 전송할 수 있습니다.

Note

Wi-Fi는 북미 지역 AWS 리전에서만 사용할 수 있습니다.

견고화 사양

AWS Snowcone 디바이스는 자유 낙하 충격, 작동 진동 등에 대한 ISTA-3A, ASTM D4169 및 MIL-STD-810G 등 엄격한 견고화 표준을 충족하도록 설계되었습니다. 최대 1.15미터(3.8피트)의 낙하를 견딜 수 있도록 설계되었습니다. 또한 IP65 국제 보호 마크 IEC 표준을 충족합니다. 즉, 방진(밀봉 시 인클로저 내부에 먼지가 묻지 않음) 및 방수(사방의 물 분출 방지 포함) 기능을 모두 갖추고 있습니다.

이 디바이스는 영하(섭씨 0도 또는 화씨 32도)부터 사막과 같은 환경(Snowcone: 섭씨 38도 또는 화씨 100도, Snowcone SSD: 섭씨 45도 또는 화씨 113도)에 이르기까지 작동 온도 범위가 넓습니다. 보관 또는 배송 시 Snowcone 디바이스는 혹독한 온도(섭씨 -32도 또는 화씨 -25.6도에서 섭씨 63도 또는 화씨 145.4도까지)에도 견딜 수 있습니다.

AWS Snowcone 전원 공급 장치 및 액세서리

AWS Snowcone 디바이스는 가능한 가장 작은 폼 팩터와 함께 제공되므로 전원 공급 장치 또는 이더넷 케이블(RJ45)은 포함되지 않습니다. 플러그인 전원 또는 배터리를 통해 Snowcone 디바이스를 실행할 수 있습니다. 전원 공급 장치 및 이더넷 케이블 주문 시 참고할 수 있는 세부 정보는 다음과 같습니다.

USB-C 전원 어댑터 - 전원 연결 또는 고정식(비모바일) 작동 환경에서는 Snowcone 디바이스에 USB-C 전원 어댑터를 사용합니다. Snowcone 디바이스에 전원을 공급하려면 AWS 테스트를 거친 다음 USB-C 전원 어댑터 중 하나를 구입하면 됩니다.

- [Apple 61W USB-C 전원 어댑터](#)
- [Lenovo USB-C 65W 표준 AC 어댑터](#)

또는 45W+ 정격의 USB-C 전원 어댑터와 주변 온도를 사용할 수 있습니다.

USB-C 배터리 - USB-C 배터리를 사용하여 모바일 또는 휴대용 운영 환경에서 Snowcone 디바이스에 전원을 공급합니다. Snowcone 디바이스에 전원을 공급하려면 AWS 테스트를 거친 다음 USB-C 배터리 중 하나를 구입하면 됩니다.

- [Dell 노트북 파워 뱅크 플러스 - USB C, 65Wh - PW7018LC](#)
- [옴니 20c+](#)

또는 45W+ 등급의 모든 USB-C 배터리를 사용할 수 있습니다.

이더넷 케이블(RJ45) - Snowcone 디바이스를 로컬 네트워크에 연결하려면 이더넷 케이블(RJ45)을 사용합니다. 아직 없는 경우 구매해야 합니다.

디스크 및 CPU 성능

AWS Snowcone는 특수 목적의 엣지 컴퓨팅/데이터 전송 디바이스입니다. 디스크 및 CPU 성능은 다양한 요인에 따라 달라집니다. 전송을 위한 몇 가지 샘플 성능 수치는 [Snowcone](#) 성능에 있습니다.

AWS Snowcone에 대해 AWS 설정

첫 번째 AWS Snowcone 작업을 생성하기 전에 다음 지침에 따라 환경을 적절하게 준비해야 합니다.

주제

- [AWS에 가입](#)

AWS에 가입

Amazon Web Services(AWS)에 가입하면 모든 AWS 서비스에 대해 계정이 자동으로 등록됩니다. AWS에서는 사용한 서비스에 대해서만 비용을 청구합니다. 계정을 설정한 후에는 AWS Snow 패밀리 관리 콘솔을 통해 AWS Snowcone 디바이스를 주문, 구성 및 관리할 수 있습니다. Snowcone의 요금 및 수수료에 대한 자세한 내용은 [AWS Snowcone 요금](#)을 참조하십시오.

이미 AWS 계정이 있다면 AWS 계정 번호를 적어 두십시오. AWS 계정 계정이 없는 경우, 다음 절차에 따라 진행합니다.

AWS 계정 생성

1. <https://portal.aws.amazon.com/billing/signup>을 엽니다.
2. 온라인 지시 사항을 따릅니다.

등록 절차 중 전화를 받고 전화 키패드로 확인 코드를 입력하는 과정이 있습니다.

AWS 계정 루트 사용자에게 가입하면 AWS 계정 루트 사용자가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스에 액세스하는 권한이 있습니다. 보안 모범 사례는 [관리 사용자에게 관리자 액세스 권한을 할당하고](#), 루트 사용자만 [루트 사용자 액세스 권한이 필요한 태스크](#)를 수행하는 것입니다.

다음 단계에서 필요한 AWS 계정 번호를 기록해 둡니다.

Snowcone 디바이스를 주문하기 전에

AWS Snowcone 는 지역별 서비스이므로 작업을 계획하기 전에 해당 지역에서 서비스를 이용할 수 있는지 확인하십시오. 위치 및 Amazon S3 버킷이 동일한 위치에 있는지 확인하십시오. 이는 디바이스 주문 능력에 영향을 AWS 리전 미치기 때문입니다. Snowcone 디바이스를 리전의 국경 밖으로 배송하는 데에는 제한이 있습니다. 자세한 설명은 [리전 기반 배송 제한](#) 섹션을 참조하세요.

특정 지역 (예: 미국, 멕시코) 의 일부 위치에 AWS Snow 패밀리 관리 콘솔 대해 허용 목록에 등록된 고객의 경우 해당 지역 외 지역으로의 해외 배송이 AWS 리전 지원됩니다. 고객 담당 팀과 이러한 요청을 수용할 수 있는 대상 목적지, 비용 및 시기를 논의해야 합니다.

주문 프로세스의 일환으로 AWS Identity and Access Management (IAM) 역할과 AWS Key Management Service () 키를 생성합니다. AWS KMS KMS에서는 각 디바이스에서 데이터를 보호하는데 사용되는 암호화 키를 보호합니다. 자세한 내용은 [Snow Family 디바이스 주문을 위한 작업 생성](#)을 참조하십시오.

주제

- [로컬 환경에 대한 질문](#)
- [특수 문자가 있는 파일 작업](#)
- [Snowcone에서 Amazon EC2 사용](#)

로컬 환경에 대한 질문

데이터 세트와 로컬 환경 설정 방법을 이해하면 데이터 전송을 완료하는 데 도움이 됩니다. 주문하기 전에 다음 사항을 고려하세요.

전송 중에 데이터에 액세스할 수 있습니까?

데이터 손상을 방지하려면 데이터를 전송하는 동안 Snowcone 디바이스의 연결을 해제하거나 네트워크 설정을 변경하면 안 됩니다. 파일은 디바이스에 기록되는 동안 정적 상태에 있어야 합니다. 파일이 디바이스에 쓰여지는 동안 수정된 파일은 읽기/쓰기 충돌을 일으킬 수 있습니다.

특수 문자가 있는 파일 작업

파일에 특수 문자가 포함된 경우 오류가 발생할 수 있다는 점에 유의해야 합니다. Amazon S3에서는 특수 문자를 허용하지만 다음 문자는 사용하지 않는 것이 좋습니다.

- 백슬래시('\')
- 왼쪽 중괄호('{')
- 오른쪽 중괄호('}')
- 왼쪽 대괄호('[')
- 오른쪽 대괄호(']')
- '보다 작음' 기호("<")
- '보다 큼' 기호(">")
- 인쇄되지 않는 ASCII 문자(128~255 사이의 10진수)
- 캐럿('^')
- 백분율 문자('%')
- 억음 악센트 기호("´")
- 인용 부호
- 물결표('~')
- '파운드' 문자('#')
- 세로 막대/파이프('|')

파일에 이러한 문자가 하나 이상 있는 경우 AWS Snowcone 장치에 복사하기 전에 이름을 바꾸십시오. 파일 이름에 공백이 있는 Windows 사용자는 개별 객체를 복사하거나 재귀 명령을 실행할 때 주의해야 합니다. 이름에 공백이 포함된 개별 객체는 다음표로 묶습니다. 이러한 파일의 예는 다음과 같습니다.

운영 체제	파일 이름: test file.txt
Windows	'C:\Users\ <username>\desktop\test file.txt'</username>
Mac	/Users/<username>/test\ file.txt
Linux	/home/<username>/test\ file.txt

Note

객체 메타데이터 중 객체 이름과 크기만이 유일하게 전송됩니다. 하지만 ACL (액세스 제어 목록) AWS DataSync 은 보존됩니다. 자세한 내용은 사용 AWS DataSync 설명서의 [메타데이터 및 특수 파일 DataSync 처리 방법을](#) 참조하십시오.

Snowcone에서 Amazon EC2 사용

이 섹션에서는 AWS Snowcone 디바이스에서 Amazon EC2 호환 컴퓨팅 인스턴스를 사용하는 방법을 간략하게 설명합니다.

가상 머신(Amazon 머신 이미지 또는 AMI)으로 관리 및 배포되는 엣지에서 실행되는 애플리케이션이 있는 경우 Amazon EC2 호환 인스턴스를 사용해야 합니다. Snowcone은 snc1.micro(CPU 1개와 RAM 1GB), snc1.small(CPU 1개 및 RAM 2GB), snc1.medium(CPU 2개 및 RAM 4GB) 등 세 개의 인스턴스로 구성된 SNC1 인스턴스 유형을 지원합니다. Snowcone에서 EC2 호환 인스턴스를 지원하므로 Amazon EC2에서 애플리케이션을 구축하고 테스트할 수 있습니다. AWS Snowcone 작업을 생성하는 동안 AWS Management Console, AWS Snowball SDK 또는 중 하나를 사용하여 EC2 호환 AMI를 활성화하고 프로비저닝할 수 있습니다. AWS CLI

지원되는 Amazon EC2 인스턴스 유형

컴퓨팅 작업에 다음 Amazon EC2 인스턴스 유형을 사용하세요.

snc1.micro - CPU 코어 1개, RAM 1GB

snc1.small - CPU 코어 1개, RAM 2GB

snc1.medium - CPU 코어 2개, RAM 4GB

Snowcone에서 인스턴스를 관리하는 AWS OpsHub 데 사용합니다. [AWS Snowball 리소스](#) 웹 AWS OpsHub 사이트에서 다운로드하십시오. 를 사용하여 AWS OpsHub디바이스를 잠금 해제한 후 Amazon EC2 페이지로 이동합니다. 인스턴스 생성을 선택하여 작업을 생성할 때 디바이스에 사전 로드한 AMI를 기반으로 EC2 호환 인스턴스를 생성합니다. 그런 다음 인스턴스에 연결하여 에지 애플리케이션을 실행할 수 있습니다. AWS OpsHub 또한 EC2 호환 인스턴스를 시작, 중지, 종료 및 재부팅할 수 있는 원클릭 버튼을 제공합니다. 자세한 설명은 [장치 AWS OpsHub for Snow Family 관리에 사용](#) 섹션을 참조하세요.

디바이스 사용을 마치면 디바이스를 로 반납하십시오. AWS디바이스가 가져오기 작업에 사용되었다면 파일 인터페이스를 사용해 전송된 데이터는 Snowcone NFS 인터페이스를 통해 Amazon S3 로 가져오기 됩니다. 그렇지 않으면 기기가 반품될 때 기기가 완전히 삭제됩니다. AWS이 삭제는 NIST(National Institute of Standards and Technology) 800-88 표준에 따른 것입니다.

Important

Snowcone에서 실행 중인 컴퓨팅 인스턴스의 데이터는 AWS로 가져오기 되지 않습니다.

Snow Family 디바이스상의 Amazon EC2와 Amazon EC2 호환 인스턴스 간 차이

AWS Snow Family EC2 호환 인스턴스를 사용하면 고객이 EC2 API의 하위 집합과 AMI의 하위 집합을 사용하여 Amazon EC2 호환 인스턴스를 사용하고 관리할 수 있습니다.

Snowcone에서의 컴퓨팅 인스턴스 요금

컴퓨팅 인스턴스 사용 시에는 추가 비용이 발생합니다. 자세한 내용은 [AWS Snowcone 요금](#)을 참조하십시오.

사전 조건

작업을 생성하기 전에 다음 정보를 유념하세요.

- 작업에 AMI를 추가하려면 먼저 AWS 계정에 AMI가 있어야 하며 지원되는 이미지 유형이어야 합니다. 현재 지원되는 AMI는 [Amazon Linux 2](#), [CentOS 7\(x86_64\) - HVM 업데이트 포함](#) 또는 [Ubuntu 16.04 LTS - Xenial\(HVM\)](#) 이미지에 기반을 두고 있습니다. 이 이미지는 [AWS Marketplace](#)에서 얻을 수 있습니다.

AMI를 작업 요청에 추가하기 전에 지원되는 AMI가 AWS 계정에 하나 이상 있는지 확인합니다. Marketplace에서 AMI를 선택할 때는 AMI에 지원되는 제품 코드 및 플랫폼이 있는지 확인하십시오. 자세한 설명은 [AMI의 제품 코드 및 플랫폼 세부 정보 확인 AWS Marketplace](#) 섹션을 참조하십시오.


- 모든 AMI는 Amazon Elastic Block Store(Amazon EBS)를 기반으로 하며 볼륨이 하나여야 합니다.
- Snowcone에서 실행 중인 컴퓨팅 인스턴스에 연결하려는 경우 Secure Shell(SSH)을 사용해야 합니다. 이렇게 하려면 먼저 키 페어를 추가합니다.

AMI의 제품 코드 및 플랫폼 세부 정보 확인 AWS Marketplace

Snow Family 디바이스에 AMI를 추가하는 프로세스를 시작하기 전에 AMI의 제품 코드 및 플랫폼 세부 정보가 해당 디바이스에서 지원되는지 확인하십시오 AWS 리전. AWS Marketplace

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 표시줄에서 인스턴스를 시작하고 Snow Family 디바이스를 주문하기 위한 작업을 생성할 지역을 선택합니다. 위치와 상관없이 사용 가능한 모든 지역을 선택할 수 있습니다.
- 탐색 창에서 AMI를 선택합니다.

4. 필터 및 검색 옵션을 사용하여 표시된 AMI 목록의 범위를 지정하여 기준에 맞는 AMI만 볼 수 있습니다. 예를 들어 에서 제공하는 AMI의 AWS Marketplace 경우 공개 이미지를 선택합니다. 그런 다음 검색 옵션을 사용하여 표시된 AMI 목록의 범위를 더 자세히 지정하십시오.
 - (새 콘솔) 검색 창을 선택하고 메뉴에서 소유자 별칭, = 연산자, 아마존 값을 차례로 선택합니다.
 - (기존 콘솔) 검색(Search) 창을 선택하고 메뉴에서 소유자(Owner), 값 Amazon 이미지(Amazon images)를 차례로 선택합니다.

 Note

AMI의 AMI는 AWS Marketplace 소스 옆에 AWS-마켓플레이스를 포함합니다.

5. AMI ID 옆에서 AMI의 AMI ID를 선택합니다.
6. AMI의 이미지 요약에서 제품 코드가 해당 지역에서 지원되는지 확인하십시오. 자세한 내용은 아래 표를 참조하십시오.

지원되는 AWS Marketplace AMI 제품 코드

AMI 운영 체제	제품 코드
Ubuntu Server 14.04 LTS	b3dl4415quatdnd14qa6kcu45
CentOS 7(x86_64)	aw0evgkw8e5c1q413zgy5picce
Ubuntu 16.04 LTS	csv6h7oyg29b7epjzg7qdr7no
Amazon Linux 2	avyfzznywektkgl5qv5f57ska
Ubuntu 20.04 LTS	a8jyynf4hjutohctm41o2z18m
Ubuntu 22.04 LTS	47xbqns9xujfkkjt189a13aqe

7. 또한 플랫폼 세부 정보에 아래 목록의 항목 중 하나가 포함되어 있는지 확인하십시오.
 - 아마존 리눅스, 우분투 또는 데비안
 - 레드햇 리눅스 bring-your-own-license
 - 오라클용 아마존 RDS bring-your-own-license
 - 윈도우 bring-your-own-license

인스턴스에서 Linux AMI 생성

콘솔 또는 명령줄을 사용하여 AMI를 생성할 수 있습니다. 기존 AMI로 인스턴스를 시작한 다음 사용자 지정하고 해당 인스턴스에서 새 AMI를 생성합니다. 마지막으로 새 AMI의 인스턴스를 실행합니다.

콘솔을 사용하여 인스턴스에서 AMI 생성

1. 새 AMI의 시작점으로 사용하기에 적절한 EBS 지원 AMI를 선택하고 시작하기 전에 필요에 따라 구성합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 방법](#) 섹션을 참조하세요.
2. 시작을 선택하여 선택한 EBS 지원 AMI의 인스턴스를 시작합니다. 나머지 기본값을 그대로 두고 마법사를 계속 진행합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 방법](#) 섹션을 참조하세요.
3. 인스턴스가 실행 중일 때 인스턴스에 연결합니다. 인스턴스에서 다음과 같은 작업을 수행하여 인스턴스를 원하는 대로 사용자 지정할 수 있습니다.
 - 소프트웨어 및 애플리케이션 설치
 - 데이터 복사
 - 임시 파일 삭제, 하드 드라이브 조각 모음, 여유 공간 제로 클리어를 통한 시작 속도 향상
 - 추가 Amazon EBS 볼륨 연결
4. (선택 사항) 인스턴스에 연결한 모든 볼륨의 스냅샷을 생성합니다. 스냅샷 생성에 대한 자세한 내용은 [Amazon EBS 스냅샷 생성](#)을 참조하세요.
5. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다. 작업, 이미지, 이미지 생성을 선택합니다.

Tip

이 옵션이 비활성화되어 있다면 Amazon EBS 지원 인스턴스가 아님을 의미합니다.

6. 이미지 생성 대화 상자에서 다음 정보를 지정한 후 이미지 생성을 선택합니다.
 - 이미지 이름 - 이미지의 고유한 이름입니다.
 - 이미지 설명 - 이미지에 대한 선택적 설명으로, 최대 255자까지 입력할 수 있습니다.
 - 재부팅 안 함 - 이 옵션은 기본적으로 선택되지 않습니다. Amazon EC2는 인스턴스를 종료하고, 연결된 볼륨의 스냅샷을 캡처하고, AMI를 생성하여 등록된 다음 인스턴스를 재부팅합니다. 인스턴스가 종료되지 않도록 하려면 재부팅 안 함을 선택합니다.

⚠ Warning

재부팅 안 함을 선택하는 경우 생성된 이미지의 파일 시스템 무결성을 보장할 수 없습니다.

- 인스턴스 볼륨 - 이 섹션의 필드에서는 루트 볼륨을 수정하고 다른 Amazon EBS 및 인스턴스 스토어 볼륨을 추가할 수 있습니다. 각 필드에 대한 자세한 내용을 보려면 각 필드 옆에 있는 **i** 아이콘에서 일시 중지하여 필드 도구 설명을 표시합니다. 몇 가지 중요한 사항은 아래에 나열되어 있습니다.
 - 루트 볼륨 크기를 변경하려면 볼륨 유형 옆의 루트를 찾습니다. 크기(GiB)에 필요한 값을 입력합니다.
 - 종료 시 삭제를 선택할 경우 이 AMI에서 생성된 인스턴스를 종료하면 Amazon EBS 볼륨이 삭제됩니다. 종료 시 삭제를 선택 취소할 경우 인스턴스를 종료하면 Amazon EBS 볼륨이 삭제되지 않습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 종료 시 Amazon EBS 볼륨 보존](#) 섹션을 참조하세요.
 - Amazon EBS 볼륨을 추가하려면 새 볼륨 추가를 선택합니다(새 행이 추가됨). 볼륨 유형에서 EBS를 선택하고 행의 필드를 작성합니다. 새 AMI에서 인스턴스를 시작하면 추가 볼륨이 인스턴스에 자동으로 연결됩니다. 빈 볼륨은 반드시 포맷하고 탑재해야 합니다. 스냅샷 기반 볼륨을 반드시 탑재해야 합니다.
 - 인스턴스 스토어 볼륨을 추가하려면 [AMI에 인스턴스 스토어 볼륨 추가](#) 섹션을 참조하세요. 새 AMI에서 인스턴스를 시작하면 추가 볼륨이 자동으로 시작되어 탑재됩니다. 이러한 볼륨에는 AMI를 기반으로 하는 실행 중인 인스턴스에 대한 인스턴스 스토어 볼륨의 데이터가 포함되어 있지 않습니다.
- 7. 생성 중인 AMI의 상태를 보려면 탐색 창에서 AMI를 선택합니다. 처음에는 상태가 보류 중이지만 몇 분 후에 사용 가능으로 변경되어야 합니다.

(선택 사항) 새 AMI에 대해 생성된 스냅샷을 보려면 스냅샷을 선택합니다. 이 AMI에서 인스턴스를 시작할 때 이 스냅샷을 사용하여 루트 디바이스 볼륨을 생성합니다.
- 8. 새 AMI에서 인스턴스를 시작합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 방법](#) 섹션을 참조하세요.
- 9. 실행 중인 새 인스턴스에는 이전 단계에서 적용한 모든 사용자 지정 사항이 포함되어 있습니다.

명령줄을 사용하여 인스턴스에서 AMI 생성

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 Amazon EC2 액세스 섹션을 참조하세요.

- [create-image](#)(AWS CLI)
- [New-EC2Image](#)(AWS Tools for Windows PowerShell)

스냅샷에서 Linux AMI 생성

인스턴스의 루트 디바이스 볼륨 스냅샷이 있는 경우 AWS Management Console 또는 명령줄을 사용하여 이 스냅샷에서 AMI를 생성할 수 있습니다.

콘솔을 사용하여 스냅샷에서 AMI 생성

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Elastic Block Store에서 스냅샷을 선택합니다.
3. 스냅샷을 선택하고 작업, 이미지 생성을 선택합니다.
4. EBS 스냅샷에서 이미지 생성 대화 상자에서 AMI 생성에 필요한 필드 정보를 모두 입력하고 생성을 선택합니다. 상위 인스턴스를 다시 생성하는 경우에는 상위 인스턴스와 동일한 옵션을 선택합니다.
 - 아키텍처: 32비트의 경우 i386을 선택하고 64비트의 경우 x86_64를 선택합니다.
 - 루트 디바이스 이름: 루트 볼륨에 적절한 이름을 입력합니다. 자세한 내용은 [Linux 인스턴스의 디바이스 명명](#)을 참조하세요.
 - 가상화 유형: 이 AMI에서 실행된 인스턴스가 반가상화(PV)를 사용하는지 또는 하드웨어 가상 머신(HVM) 가상화를 사용하는지 선택합니다. 자세한 내용은 [Linux AMI 가상화 유형](#)을 참조하세요.
 - (PV 가상화 유형에만 해당) 커널 ID 및 RAM 디스크 ID: 목록에서 AKI 및 ARI를 선택합니다. 기본 AKI를 선택하거나 AKI를 선택하지 않으면 이 AMI를 사용하여 인스턴스를 시작할 때마다 AKI를 지정해야 합니다. 또한 기본 AKI가 인스턴스와 호환되지 않는 경우, 상태 확인 시 인스턴스 오류가 발생할 수 있습니다.
 - (선택 사항) 블록 디바이스 매핑: 볼륨을 추가하거나 AMI에 대한 루트 볼륨의 기본 크기를 확장합니다. 대용량 볼륨 인스턴스의 파일 시스템 크기 조정에 대한 자세한 내용은 [볼륨 크기 조정 후 Linux 파일 시스템 확장](#) 섹션을 참조하세요.

명령줄을 사용하여 스냅샷에서 AMI 생성

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2 액세스 섹션](#)을 참조하세요.

- [register-image](#)(AWS CLI)
- [Register-EC2Image](#)(AWS Tools for Windows PowerShell)

시작하기

이 섹션에서는 에서 첫 번째 AWS Snowcone 작업을 생성하고 완료하는 데 필요한 일반 지침을 제공합니다. AWS Snow 패밀리 관리 콘솔. AWS Snowcone 장치 개요는 를 참조하십시오. [AWS Snowcone 작동 방식](#).

이 시작 설명서에서는 사용자가 작업을 생성하는 [AWS Snow 패밀리 관리 콘솔](#) 데 를 사용하고, Snowball Edge 클라이언트 또는 AWS OpsHub for Snow Family 애플리케이션을 사용하여 디바이스를 잠금 AWS Snowcone 해제한다고 가정합니다. 생성 중인 작업에 대해 더 많은 옵션을 사용하여 프로그래밍 방식으로 작업을 생성할 경우 작업 관리 API를 사용할 수 있습니다. 자세한 내용은 [AWS Snowcone API 참조](#)를 참조하세요.

시작하려면 먼저 AWS Identity and Access Management (IAM) 에서 AWS 계정 및 관리자 사용자를 생성해야 합니다. 자세한 설명은 [AWS Snowcone에 대해 AWS 설정](#) 섹션을 참조하세요.

시작하려면 AWS Snowcone [Snow Family 디바이스 주문을 위한 작업 생성](#)을 참조하십시오.

주제

- [Snow Family 디바이스 주문 작업 생성](#)
- [를 통한 작업 취소 AWS Snow 패밀리 관리 콘솔](#)
- [Snow Family 디바이스에 액세스하기 위한 자격 증명 받기](#)
- [Snow Family 디바이스 잠금 해제](#)
- [Snow Family 디바이스 재부팅](#)

Snow Family 디바이스 주문 작업 생성

Snow Family 디바이스를 주문하려면 에서 Snow Family 디바이스를 주문하는 작업을 생성해야 합니다. AWS Snow 패밀리 관리 콘솔. 직업이란 고객이 AWS Snow Family 디바이스를 사용하는 라이프사이클을 설명하는 데 사용하는 용어입니다. 작업은 디바이스를 주문할 때 시작되고, 디바이스를 AWS 준비하여 배송하여 사용자가 사용하고 있는 시점까지 계속되며, 디바이스를 AWS 수령하고 반품한 후에 처리가 완료됩니다. 작업은 내보내기, 가져오기, 로컬 컴퓨팅 및 스토리지와 같은 유형별로 분류됩니다. 자세한 내용은 [AWS Snowball Edge 작업 이해](#)를 참조하십시오.

디바이스를 주문하기 위한 작업을 생성한 후에는 를 사용하여 작업 상태를 보고 주문한 디바이스의 진행 상황을 모니터링하면서 디바이스를 배송할 AWS 준비를 하고 반품한 후의 진행 상황을 모니터링할 수 있습니다. AWS Snow 패밀리 관리 콘솔 자세한 내용은 [Job Statuses](#)를 참조하십시오. 에서 디바이

스를 반환하고 처리한 AWS후에는 를 통해 작업 완료 보고서에 액세스하고 로그인할 수 있습니다. AWS Snow 패밀리 관리 콘솔자세한 내용은 [콘솔에서 작업 완료 보고서 및 로그 가져오기](#)를 참조하십시오.

또한 작업 관리 API를 사용하여 작업을 생성하고 관리할 수도 있습니다. 자세한 내용은 [AWS Snowball API 참조](#)를 참조하십시오.

주제

- [1단계: 작업 유형 선택](#)
- [2단계: 컴퓨팅 및 스토리지 옵션 선택](#)
- [3단계: 기능 및 옵션 선택](#)
- [4단계: 보안, 배송 및 알림 기본 설정 선택](#)
- [5단계: 작업 요약 검토 및 작업 생성](#)
- [다운로드 AWS OpsHub](#)

1단계: 작업 유형 선택

작업 생성의 첫 번째 단계는 필요한 작업 유형을 결정하고 AWS Snow 패밀리 관리 콘솔을 사용하여 작업 계획을 시작하는 단계입니다.

작업 유형 선택

1. 에 AWS Management Console로그인하고 를 엽니다 [AWS Snow 패밀리 관리 콘솔](#). 이 AWS 리전 분야에서 처음으로 일 자리를 창출하는 경우 AWS Snow Family 페이지가 표시됩니다. 그렇지 않으면 기존 작업 목록이 표시됩니다.
2. 처음 하는 일이라면 AWS Snow Family 기기 주문을 선택하세요. 500TB가 넘는 데이터를 마이그레이션하는 작업이 여러 개 있을 것으로 예상되면 500TB를 초과하는 라지 데이터 마이그레이션 계획 생성을 선택합니다. 그렇지 않으면 왼쪽 탐색 표시줄에서 작업 생성을 선택합니다. 다음 단계를 선택하여 작업 계획 페이지를 엽니다.
3. 작업 이름 섹션의 작업 이름 상자에 작업 이름을 입력합니다.
4. 필요에 따라 다음 작업 유형 중 하나를 선택합니다.
 - Amazon S3로 가져오기 - 빈 Snowcone 디바이스를 AWS 가 배송합니다. 장치를 로컬 네트워크에 연결하고 를 사용하여 장치를 구성합니다 OpsHub. NFS 공유를 사용하여 데이터를 디바이스에 복사하고 AWS, 다시 전송하면 데이터가 Amazon S3에 업로드됩니다.
 - 로컬 컴퓨팅 및 스토리지 전용 - 데이터를 전송하지 않고 디바이스에서 컴퓨팅 및 스토리지 워크로드를 수행할 수 있습니다.

Choose a job type

Import into Amazon S3 [Info](#)

AWS will ship an empty device to you for storage and compute workloads. You'll transfer your data onto it, and ship it back. After AWS gets it, your data will be moved.

Export from Amazon S3 [Info](#)

Choose what data you want to export from your S3 buckets for storage and compute workloads. AWS will load that data onto a device and ship it to you. When you're done ship the device back for erasing.

Local compute and storage only [Info](#)

Perform local compute and storage workloads without transferring data. You can order multiple devices in a cluster for increased durability and storage capacity. Includes rugged and rack-mountable devices.

5. 다음을 선택하여 계속 진행합니다.

2단계: 컴퓨팅 및 스토리지 옵션 선택

Snow Family 디바이스의 하드웨어 사양, 여기에 포함할 Amazon EC2 호환 인스턴스, 데이터 저장 방법 및 요금을 선택합니다.

디바이스의 컴퓨팅 및 스토리지 옵션 선택

1. Snow 디바이스 섹션에서 주문할 Snow Family 디바이스를 선택합니다.

Note

주문 대상 및 선택한 작업 유형에 따라 일부 AWS 리전 Snow Family 장치를 사용하지 못할 수도 있습니다.

Snow devices Info					
	Name	Compute	Memory	Storage (HDD)	Storage (SSD)
<input checked="" type="radio"/>	Snowcone	2 vCPUs	4 GB	8 TB	-
<input type="radio"/>	Snowcone SSD	2 vCPUs	4 GB	-	14 TB
<input type="radio"/>	Snowball Edge Compute Optimized	52 vCPUs	208 GB	39.5 TB	7.68 TB
<input type="radio"/>	Snowball Edge Compute Optimized with GPU	52 vCPUs, GPU	208 GB	39.5 TB	7.68 TB
<input type="radio"/>	Snowball Edge Compute Optimized	104 vCPUs	416 GB	-	28 TB

2. Snowcone 전원 공급 장치 섹션에서 자체 전원 공급 장치 및 이더넷 케이블을 제공하겠습니다를 선택합니다. 전원 공급 장치에 대한 자세한 내용은 [AWS Snowcone 전원 공급 장치 및 액세서리](#)를 참조하십시오.
3. 요금 옵션 선택 섹션의 요금 옵션 선택 메뉴에서 이 작업에 적용할 요금 유형을 선택합니다. 디바이스 요금에 대해서는 [AWS Snowcone 요금](#) 섹션을 참조하십시오.
4. 스토리지 유형 선택 섹션에서 필요에 따라 선택하세요.
 - NFS 기반 데이터 전송: NFS(Network File System) 기반 데이터 전송을 사용하여 컴퓨터의 파일을 Snow Family 디바이스의 Amazon S3 버킷으로 끌어다 놓습니다.

Warning

NFS 기반 데이터 전송은 S3 어댑터를 지원하지 않습니다. NFS 기반 데이터 전송을 진행하는 경우 NFS 공유를 탑재하여 객체를 전송해야 합니다. 를 AWS CLI 사용하여 객체를 전송하면 실패합니다.

자세한 내용은 [사용 AWS Snowcone 설명서의 오프라인 데이터 전송을 위한 NFS 사용](#)을 참조하십시오.

5. 스토리지 유형으로 NFS 기반 데이터 전송을 선택한 경우, S3 버킷 선택 섹션에서 다음 중 하나 이상을 수행하여 하나 이상의 S3 버킷을 선택합니다.
 - a. S3 버킷 이름 목록에서 사용할 S3 버킷을 선택합니다.

- b. 항목 검색 필드에 버킷 이름 전체 또는 일부를 입력하여 항목의 사용 가능한 버킷 목록을 필터링한 다음 버킷을 선택합니다.
- c. 새 S3 버킷을 만들려면 새 S3 버킷 생성을 선택합니다. 새 버킷 이름이 버킷 이름 목록에 표시됩니다. 이를 선택합니다.

하나 이상의 S3 버킷을 포함할 수 있습니다. 이러한 버킷은 디바이스에 로컬 S3 버킷으로 표시됩니다.

Choose your NFS storage

These S3 buckets will appear on directories on your device. You can transfer data onto these buckets using NFS.

ⓘ Only data stored in these directories will be ingested to your S3 buckets in the cloud.

The NFS storage limit is 80 TB

[Create a new S3 bucket](#)

	S3 bucket name	Date created
<input type="checkbox"/>	this-unique-bucket-name	6/14/2023, 12:20:08 PM EDT

6. EC2 호환 인스턴스를 사용한 컴퓨팅 - 선택 사항 섹션에서 디바이스에 포함할 계정의 Amazon EC2 호환 AMI를 선택합니다. 검색 필드에 버킷 이름 전체 또는 일부를 입력하여 항목의 사용 가능한 AMI 목록을 필터링한 다음 AMI를 선택합니다.

자세한 내용은 이 설명서의 [컴퓨팅 인스턴스를 사용한 작업 생성](#)을 참조하십시오.

이 기능은 추가 요금이 부과됩니다. 자세한 내용은 [AWS Snowball Edge 요금](#)을 참조하십시오.

7. 다음 버튼을 선택합니다.

3단계: 기능 및 옵션 선택

Amazon EKS Anywhere for AWS Snow, 인스턴스, 원격 디바이스 관리 기능을 포함하여 Snow Family 디바이스 작업에 포함할 기능과 옵션을 선택하십시오. AWS IoT Greengrass

기능 및 옵션 선택

1. Snowcone 디바이스에서 무선 네트워킹을 활성화하려면 Snowcone에서 무선 활성화를 선택합니다.

2. 또는 Snowball Edge 클라이언트를 통해 Snow Family 디바이스를 원격으로 관리할 수 있도록 하려면 AWS OpsHub 또는 Snowball 클라이언트를 사용하여 AWS OpsHub Snow 디바이스를 원격으로 관리하기를 선택합니다.
3. 다음 버튼을 선택합니다.

4단계: 보안, 배송 및 알림 기본 설정 선택

주제

- [보안 기본 설정 선택](#)
- [배송 기본 설정 선택](#)
- [알림 기본 설정 선택](#)

보안 기본 설정 선택

보안을 설정하면 AWS Snow Family 디바이스 작업에 대한 권한 및 암호화 설정이 추가되어 전송 중에 데이터를 보호할 수 있습니다.

작업 보안 설정

1. 암호화 섹션에서 사용하려는 KMS 키를 선택합니다.
 - 기본 AWS Key Management Service (AWS KMS) 키를 사용하려면 AWS/importexport (기본값)를 선택하십시오. 이 키는 다른 키가 정의되지 않은 경우 가져오기 및 내보내기 작업을 보호하는 기본 키입니다.
 - 자체 AWS KMS 키를 제공하려면 키 ARN 입력을 선택하고 키 ARN 상자에 Amazon 리소스 이름 (ARN)을 입력한 다음 이 KMS 키 사용을 선택합니다. 키 ARN이 목록에 추가됩니다.
2. 서비스 액세스 유형 선택 섹션에서 다음 중 하나를 수행합니다.
 - Snow 콘솔을 선택하면 서비스 연결 역할을 생성하고 사용하여 사용자를 대신하여 리소스에 액세스할 수 있습니다. AWS 귀하를 대신하여 AWS Snow Family에 Amazon S3 및 Amazon Simple Service (Amazon SNS)를 사용할 수 있는 권한을 부여합니다. 이 역할은 Snow 서비스에 AWS 보안 토큰 서비스 (AWS STS)의 AssumeRole 신뢰를 부여합니다.
 - 사용할 기존 서비스 역할 추가를 선택하여 원하는 역할 ARN을 지정하거나 기본 역할을 사용할 수 있습니다.
3. 다음을 선택합니다.

배송 기본 설정 선택

Snow Family 디바이스를 수령하고 반환하려면 디바이스 이동이 이루어져야 하므로 정확한 배송 정보를 제공하는 것이 중요합니다.

배송 세부 정보 제공

1. 배송 주소 섹션에서 기존 주소를 선택하거나 새 주소를 추가합니다.
 - 최근 주소 사용을 선택하면 파일에 있는 주소가 표시됩니다. 목록에서 원하는 주소를 신중하게 선택합니다.
 - 새 주소 추가를 선택한 경우 요청된 주소 정보를 입력합니다. 새 배송 정보가 AWS Snow 패밀리 관리 콘솔 저장됩니다.

Note

입력한 해당 주소의 국가는 디바이스의 목적지 국가와 일치해야 하며 해당 국가에서 유효해야 합니다.

2. 배송 속도 섹션에서 작업의 배송 속도를 선택합니다. 이 속도는 목적지 간에 디바이스가 얼마나 빨리 배송되는지를 나타내며 오늘 날짜 이후에 도착하는 시간은 반영하지 않습니다. 선택할 수 있는 배송 속도는 다음과 같습니다.
 - 1일 배송(영업일 기준 1일)
 - 2일 배송(영업일 기준 2일)
 - [배송 업체](#)를 참조하십시오.

알림 기본 설정 선택

알림을 통해 AWS Snow Family 디바이스 작업의 최신 상태를 확인할 수 있습니다. 작업 상태가 변경되면 Amazon Simple Notification Service(SNS)에서 이메일을 받습니다.

알림 설정

- 알림 설정 섹션에서 다음 중 하나를 수행합니다.
 - 기존 SNS 주제를 사용하려면 기존 SNS 주제 사용을 선택하고 목록에서 Amazon 리소스 이름(ARN) 주제를 선택합니다.

- 새 SNS 주제를 생성하려면 새 SNS 주제 생성을 선택합니다. 주제의 이름을 입력하고 이메일 주소를 제공합니다.

알림은 작업의 다음 상태 중 하나와 관련이 있습니다.

- 작업 생성됨
- 디바이스 준비 중
- 배송 준비 중
- 배송 중
- 사용자에게 배달됨
- 다음 주소로 이동 중 AWS
- 분류 시설에 있음
- ... 에서 AWS
- 가져오기
- 완료됨
- 취소됨

작업 상태 변경 알림 및 암호화된 SNS 주제에 대한 자세한 내용은 이 가이드의 [Snow Family 디바이스에 대한 알림](#)을 참조하십시오.

다음을 선택합니다.

5단계: 작업 요약 검토 및 작업 생성

AWS Snow Family 디바이스 작업에 필요한 모든 정보를 제공한 후 작업을 검토하고 생성하십시오. 작업을 생성하고 나면 Snow Family 디바이스를 배송할 준비를 시작합니다. AWS

작업에는 특정 국가의 수출 통제법이 적용되며 수출 허가가 필요할 수 있습니다. 미국 수출 및 재수출 법도 적용됩니다. 해당 국가 및 미국의 법률 및 규정을 위반하는 행위는 금지됩니다.

1. 작업을 생성하기 전에 Job summary 페이지에서 모든 섹션을 검토하십시오. 변경하려면 해당 섹션에서 편집을 선택하고 관련 정보를 편집합니다.
2. 검토 및 편집이 끝나면 작업 생성을 선택합니다.

Note

Snow Family 디바이스를 주문하기 위해 작업을 생성한 후에는 요금이 발생하지 않고 Job create 상태에 있는 동안 작업을 취소할 수 있습니다. 자세한 내용은 를 통한 [작업 취소를](#) 참조하십시오. AWS Snow 패밀리 관리 콘솔

Note

Snowcone 디바이스에는 전원 코드가 제공되지 않으며, 전원 코드는 별도로 제공됩니다. 자세한 내용은 [AWS Snowcone 전원 공급 장치 및 액세서리를](#) 참조하십시오.

작업이 생성된 후 작업 상태 섹션에서 작업 상태를 확인할 수 있습니다. 작업 상태에 대한 자세한 내용은 [작업 상태](#) 섹션을 참조하십시오.

다운로드 AWS OpsHub

AWS Snow Family 장치는 장치 및 로컬 AWS 서비스장치를 관리하는 데 사용할 수 있는 사용자 친화적인 도구를 제공합니다. AWS OpsHub for Snow Family

클라이언트 컴퓨터에 AWS OpsHub 설치하면 다음과 같은 작업을 수행할 수 있습니다.

- 단일 또는 클러스터링된 디바이스의 잠금 해제 및 구성
- 파일 전송
- Snow Family 디바이스에서 실행되는 인스턴스 시작 및 관리

자세한 설명은 [장치 AWS OpsHub for Snow Family 관리에 사용](#) 섹션을 참조하세요.

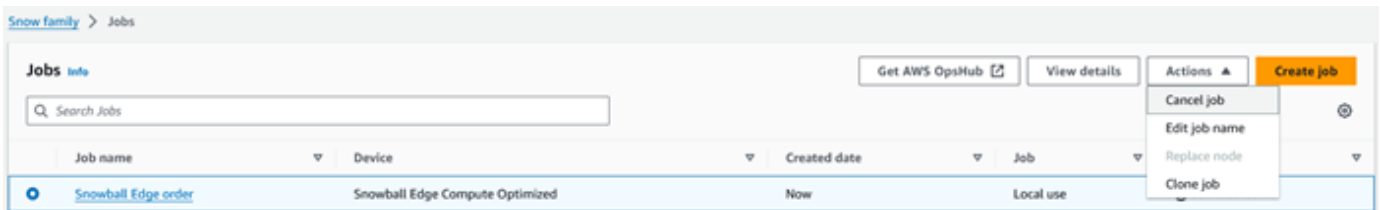
다운로드 및 설치하기 AWS OpsHub for Snow Family

1. [AWS Snowball 리소스에서](#) 을 클릭합니다 AWS OpsHub. 다운로드 링크가 있는 AWS OpsHub 섹션에서 운영 AWS OpsHub 체제에 설치할 적절한 다운로드 링크를 선택합니다.
2. AWS OpsHub 섹션에서 운영 체제에 맞는 다운로드를 선택하고 설치 단계를 따릅니다. 마쳤으면 [Next]를 선택합니다.

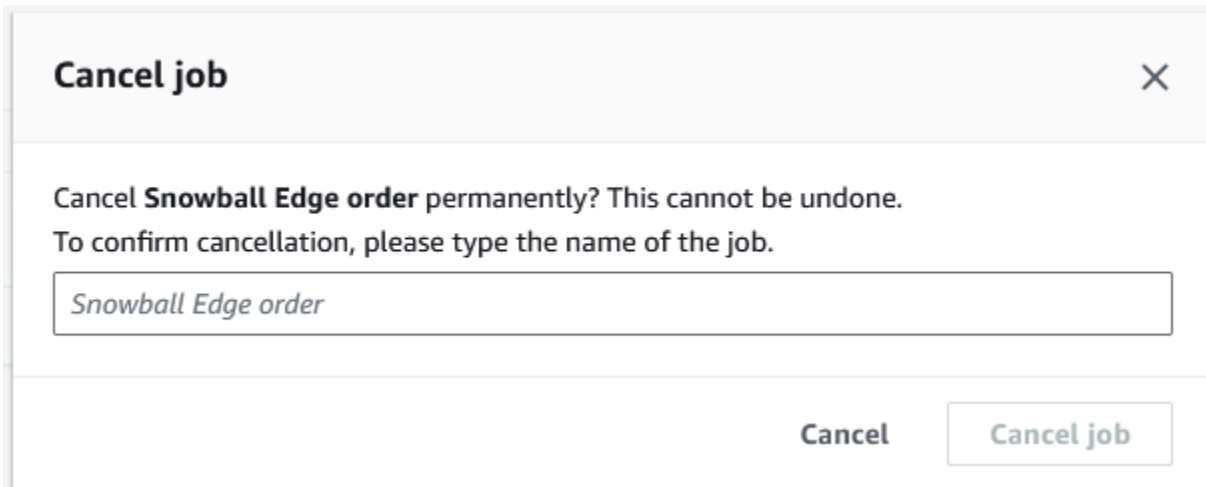
를 통한 작업 취소 AWS Snow 패밀리 관리 콘솔

Snow Family 디바이스를 주문하기 위한 작업을 생성한 후 를 통해 작업을 취소할 수 AWS Snow 패밀리 관리 콘솔 있습니다. 작업을 취소하면 주문한 디바이스를 받을 수 없습니다. 작업 상태가 Job Create인 경우에만 작업을 취소할 수 있습니다. 작업이 이 상태를 지나면 작업을 취소할 수 없습니다. 자세한 내용은 [Job Statuses](#)를 참조하십시오.

1. [AWS Snow 패밀리 관리 콘솔](#)에 로그인합니다.
2. 취소할 작업을 선택합니다.
3. 작업을 선택합니다. 나타나는 메뉴에서 작업 취소를 선택합니다.



4. 작업 취소 창이 나타납니다. 작업 취소를 확인하려면 를 **job name** 입력하고 작업 취소를 선택합니다. 작업 목록에서 상태 열에 취소됨이 표시됩니다.



Snow Family 디바이스에 액세스하기 위한 자격 증명 받기

각 작업에는 Snow Family 디바이스에 대한 액세스를 인증하기 위해 AWS Snow 패밀리 관리 콘솔 또는 작업 관리 API에서 받아야 하는 자격 증명 세트가 있습니다. 이러한 자격 증명은 암호화된 매니페스트 파일과 관련 잠금 해제 코드입니다. 매니페스트 파일에는 작업 및 작업에 연결된 권한에 대한 중요한 정보가 포함되어 있습니다.

Note

보안 인증 정보는 디바이스가 전송된 후에 받게 됩니다. AWS Snow 패밀리 관리 콘솔에서 작업 상태를 확인할 수 있습니다. 자세한 설명은 [AWS Snowcone 작업 상태 이해](#) 섹션을 참조하세요.

콘솔을 사용하여 보안 인증 정보 받기

1. [AWS Management Console](#) 로그인하고 [콘솔](#) 을 엽니다. [AWS Snow 패밀리 관리 콘솔](#)
2. 콘솔에서 작업 매니페스트를 다운로드할 특정 작업에 대한 테이블을 검색한 다음 해당 작업을 선택합니다.
3. 해당 작업 상태 창을 확장하고 작업 세부 정보 보기를 선택합니다.
4. 세부 정보 창이 표시되면 보안 인증 정보를 확장하고 다음을 수행합니다.
 - 장치 잠금을 해제하려면 29자를 모두 입력해야 하므로 잠금 해제 코드 (하이픈 포함) 를 기록해 두십시오.
 - 대화 상자에서 매니페스트 다운로드를 선택하고 지침에 따라 컴퓨터에 작업 매니페스트 파일을 다운로드합니다. 매니페스트 파일의 이름에는 [Job ID]가 포함됩니다.

Note

컴퓨터의 해당 작업에 대한 매니페스트와 같은 위치에 잠금 해제 코드의 사본을 저장하지 않는 것이 좋습니다. 자세한 설명은 [AWS Snowcone 디바이스의 모범 사례](#) 섹션을 참조하세요.

자격 증명이 준비되었으므로 다음 단계는 디바이스 잠금을 해제하는 데 사용되는 Snowball Edge 클라이언트를 다운로드하는 것입니다. AWS Snowball Edge

다음: [AWS Snowball Edge 클라이언트 사용](#)

Snow Family 디바이스 잠금 해제

이 섹션에서는 Snowball Edge CLI를 사용하여 Snow 패밀리 디바이스를 잠금 해제하는 방법에 대해 설명합니다. [Snow Family 디바이스용 그래픽 사용자 인터페이스 \(GUI\) 도구를 사용하여 AWS OpsHub 디바이스를 잠금 해제하려면 디바이스 참조하십시오.](#)

Snow Family 장치 장치를 사용하여 데이터를 전송하거나 에지 컴퓨팅 작업을 수행하기 전에 먼저 장치의 잠금을 해제해야 합니다. 디바이스를 잠금 해제할 때 29자리 잠금 해제 코드와 매니페스트 파일이라는 두 가지 형태의 자격 증명을 제공하여 액세스 권한을 인증합니다. 디바이스를 잠금 해제하면 디바이스를 추가로 구성하고, 디바이스로 또는 디바이스에서 데이터를 이동하고, Amazon EC2 호환 인스턴스를 설정 및 사용하는 등의 작업을 수행할 수 있습니다.

디바이스를 잠금 해제하기 전에 디바이스를 전원과 네트워크에 연결하고 전원을 켜고 IP 주소를 할당해야 합니다. [AWS Snowcone 장치](#) 사양을 참조하십시오. Snow Family 디바이스에 대한 다음 정보가 필요합니다.

- Snowball Edge 클라이언트를 다운로드 및 설치합니다. 자세한 설명은 [AWS Snowball Edge 클라이언트 사용](#) 섹션을 참조하세요.
- 에서 자격 증명을 가져오십시오 AWS Snow 패밀리 관리 콘솔. 하나 이상의 독립형 장치의 경우 각 Snow Family 장치의 잠금 해제 코드 및 매니페스트 파일. 자격 증명 다운로드에 대한 자세한 내용은 [Snow Family 디바이스에 액세스하기 위한 자격 증명 받기](#) 을 참조하십시오.
- 각 장치의 전원을 켜고 네트워크에 연결합니다. 자세한 설명은 [AWS Snowcone 디바이스 사양](#) 섹션을 참조하세요.

Snowball Edge 클라이언트를 사용하여 독립형 디바이스를 잠금 해제하려면

1. 장치의 LCD 디스플레이에서 Snowcone의 IP 주소를 찾으십시오. 이 IP 주소를 기록해 둡니다.
2. `unlock-device` 명령을 사용하여 다음과 같이 Snow Family 디바이스의 IP 주소 및 자격 증명으로 Snow Family 디바이스에 대한 액세스를 인증합니다.

```
snowballEdge unlock-device --endpoint https://ip-address-of-device --manifest-file /Path/to/manifest/file.bin --unlock-code 29-character-unlock-code
```

디바이스는 다음 메시지와 함께 성공적으로 잠금 해제되었음을 나타냅니다.

```
Your Snowball Edge device is unlocking. You may determine the unlock state of your device using the describe-device command. Your Snowball Edge device will be available for use when it is in the UNLOCKED state.
```

명령이 `connection refused` 반환되면 [Snow Family 디바이스 잠금 해제 문제 해결](#) 을 참조하십시오.

Example **unlock-device** 명령의

이 예제에서 디바이스의 IP 주소는 이고 192.0.2.0 매니페스트 파일 이름은 이며 JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin 29자 잠금 해제 코드는 입니다. 12345-abcde-12345-ABCDE-12345

```
snowballEdge unlock-device --endpoint https://192.0.2.0 --manifest-file /
Downloads/JID2EXAMPLE-0c40-49a7-9f53-916aEXAMPLE81-manifest.bin /
--unlock-code 12345-abcde-12345-ABCDE-12345
```

Snow Family 디바이스 잠금 해제 문제 해결

명령이 connection refused 반환되면 unlock-device 명령 구문을 잘못 입력했거나 컴퓨터 또는 네트워크 구성으로 인해 명령이 Snow 장치에 도달하지 못할 수 있습니다. 다음 조치를 취하여 상황을 해결하십시오.

1. 명령이 올바르게 입력되었는지 확인하십시오.
 - a. 장치의 LCD 화면을 사용하여 명령에 사용된 IP 주소가 올바른지 확인합니다.
 - b. 파일 이름을 포함하여 명령에 사용된 매니페스트 파일의 경로가 올바른지 확인하십시오.
 - c. [AWS Snow 패밀리 관리 콘솔](#)를 사용하여 명령에 사용된 잠금 해제 코드가 올바른지 확인하십시오.
2. 사용 중인 컴퓨터가 Snow 디바이스와 동일한 네트워크 및 서브넷에 있는지 확인하십시오.
3. 사용 중인 컴퓨터와 네트워크가 Snow 장치에 액세스할 수 있도록 구성되어 있는지 확인하십시오. 운영 체제의 ping 명령을 사용하여 컴퓨터가 네트워크를 통해 Snow 장치에 연결할 수 있는지 확인합니다. 바이러스 백신 소프트웨어 구성, 방화벽 구성, VPN (가상 사설망) 또는 컴퓨터와 네트워크의 기타 구성을 확인하십시오.

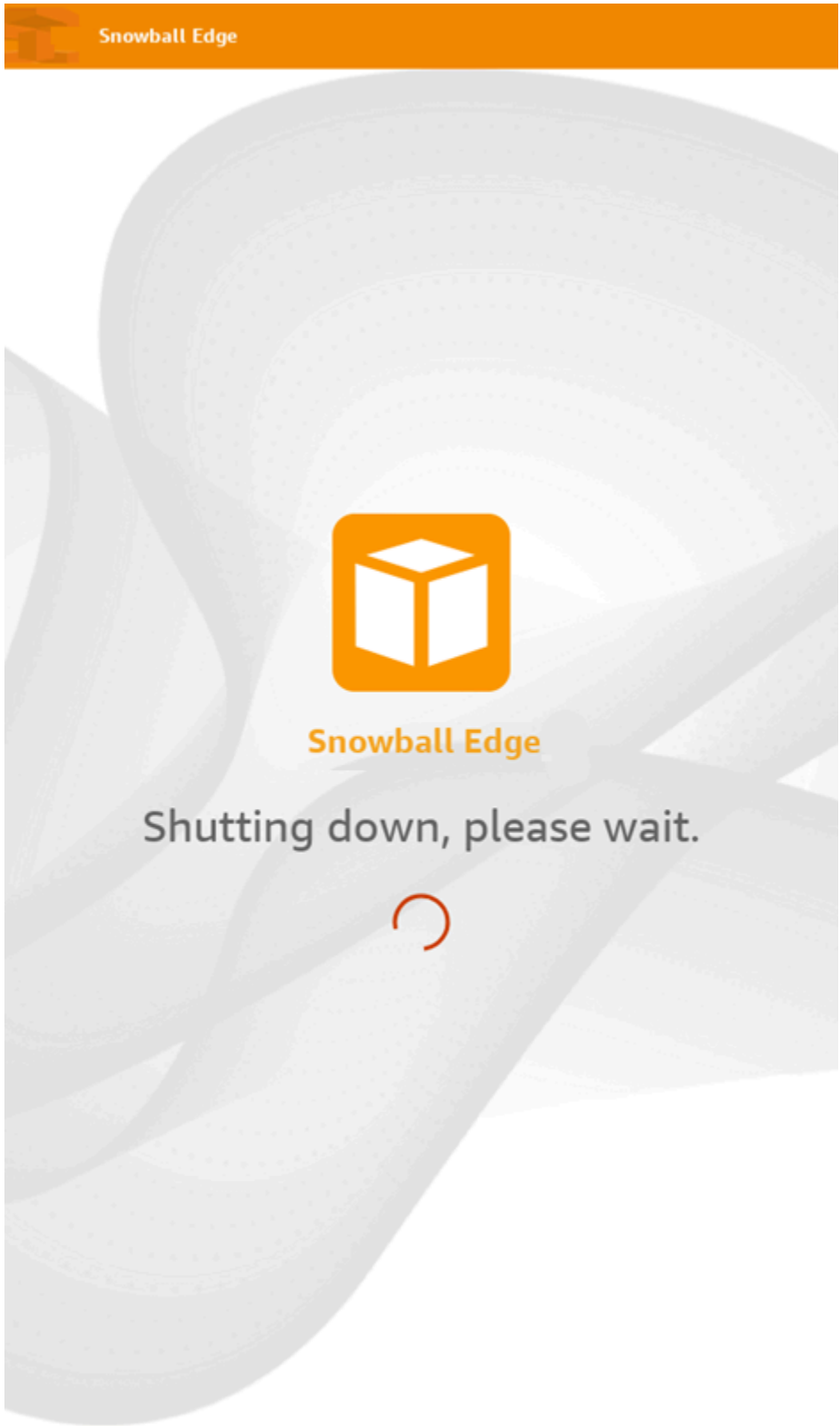
이제 Snow Family 디바이스 사용을 시작할 수 있습니다.

Snow Family 디바이스 재부팅

Snow Family 디바이스를 재부팅하기 전에 디바이스로의 모든 데이터 전송이 중단되었는지 확인합니다. NFS 인터페이스를 사용하여 데이터를 전송하는 경우 디바이스의 전원을 끄기 전에 비활성화하십시오. 자세한 설명은 [NFS 인터페이스 중지](#) 섹션을 참조하세요.

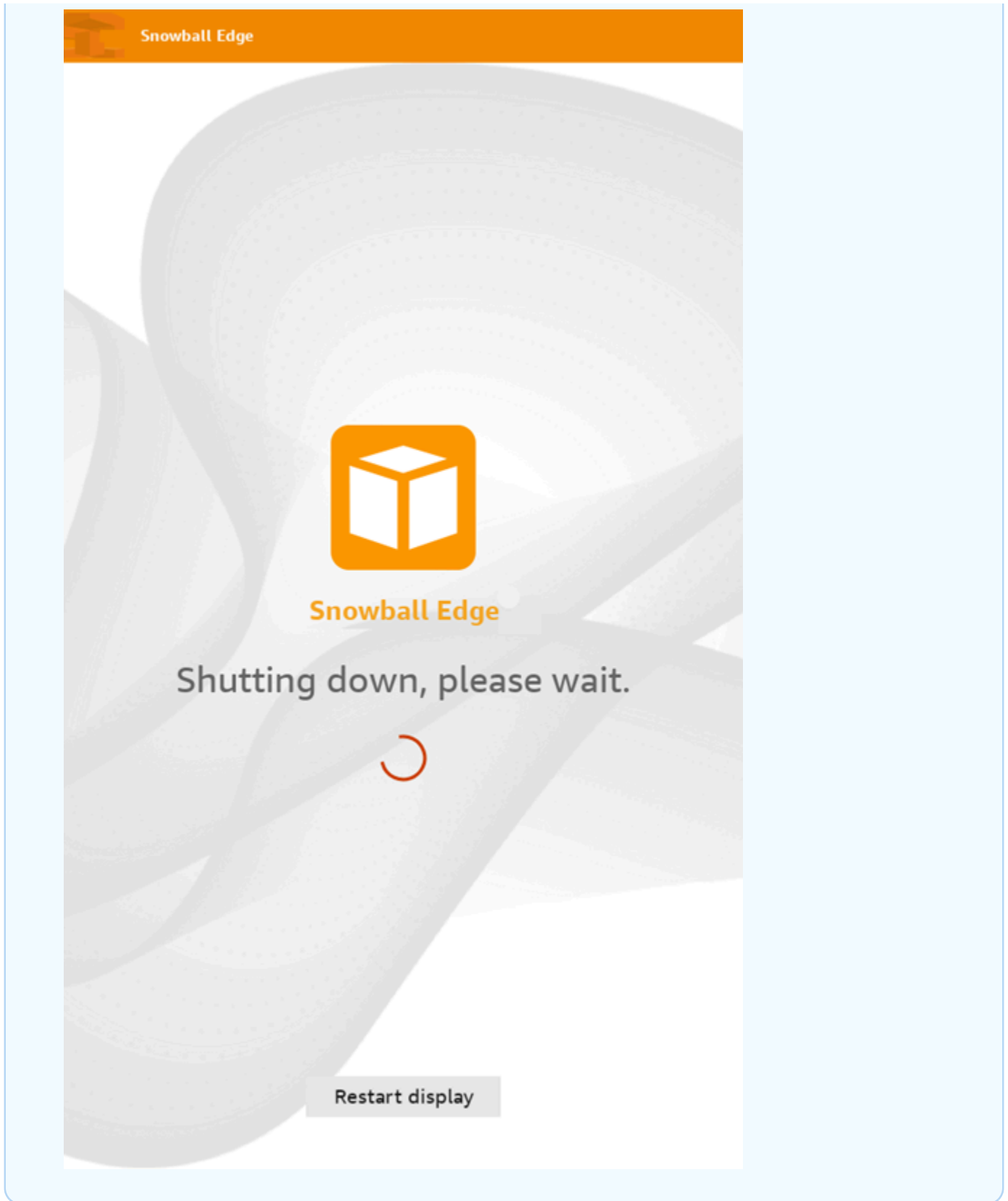
전원 버튼을 사용하여 디바이스를 재부팅하려면:

1. 디바이스와의 모든 통신이 종료된 후 디바이스 전면의 전원 버튼을 눌러서 전원을 끕니다. 디바이스를 종료하는 데 20초 정도 걸립니다. 디바이스가 종료되는 동안 LCD 화면에 디바이스가 종료되었음을 알리는 메시지가 표시됩니다.



Note

디바이스가 실제로 종료되지 않았는데도 LCD 화면에 종료 메시지가 표시되는 경우 화면의 디스플레이 재시작 버튼을 눌러 화면을 정상 작동 상태로 되돌립니다.



2. 전원 버튼을 누릅니다. 디바이스가 준비되면 디바이스가 시작 준비를 하는 동안 LCD 디스플레이에 짧은 비디오가 재생됩니다. 약 10분 후 디바이스는 잠금 해제할 준비가 됩니다.

3. 디바이스의 잠금을 해제합니다. [장치 잠금 해제 AWS Snowcone](#) 섹션을 참조하세요.

Snowball Edge 클라이언트를 사용하여 디바이스를 재부팅하려면:

1. 디바이스와의 모든 통신이 종료되면 `reboot-device` 명령을 사용하여 디바이스를 재부팅합니다. 디바이스가 준비되면 디바이스가 시작 준비를 하는 동안 LCD 디스플레이에 짧은 비디오가 재생됩니다. 약 10분 후 디바이스는 잠금 해제할 준비가 됩니다.

```
snowballEdge reboot-device
```

2. 디바이스의 잠금을 해제합니다. [장치 잠금 해제 AWS Snowcone](#) 섹션을 참조하세요.

장치 AWS OpsHub for Snow Family 관리에 사용

Snow Family 장치는 이제 장치 및 로컬 AWS 서비스를 관리하는 데 사용할 수 있는 사용자 친화적인 도구를 제공합니다. AWS OpsHub for Snow Family 클라이언트 AWS OpsHub 컴퓨터에서 사용하여 단일 또는 클러스터 디바이스의 잠금 해제 및 구성, 파일 전송, Snow Family 디바이스에서 실행되는 인스턴스 시작 및 관리와 같은 작업을 수행합니다. 를 AWS OpsHub 사용하여 스토리지 최적화 및 컴퓨팅 최적화 Snow 디바이스 유형을 모두 관리할 수 있습니다. AWS OpsHub 애플리케이션은 추가 비용 없이 사용할 수 있습니다.

AWS OpsHub Snowball API에서 사용할 수 있는 기존 작업을 모두 가져와서 그래픽 사용자 인터페이스로 표시합니다. 이 인터페이스를 사용하면 Snow Family 디바이스로 데이터를 빠르게 AWS 클라우드 마이그레이션하고 엣지 컴퓨팅 애플리케이션을 Snow Family 디바이스에 배포할 수 있습니다.

AWS OpsHub Snow Family 디바이스에서 실행되는 AWS 서비스에 대한 통합 보기를 제공하고 운영 작업을 AWS Systems Manager 자동화합니다. 를 사용하면 AWS OpsHub 기술 전문 지식 수준이 서로 다른 사용자가 대량의 Snow Family 장치를 관리할 수 있습니다. 몇 번의 클릭만으로 디바이스의 잠금을 해제하고, 파일을 전송하고, Amazon EC2 호환 인스턴스를 관리하고, 디바이스 지표를 모니터링할 수 있습니다.

Snow 디바이스가 사이트에 도착하면 노트북과 같은 클라이언트 시스템에서 AWS OpsHub 애플리케이션을 다운로드, 설치 및 시작합니다. 설치 후에는 장치 잠금을 해제하고 관리를 시작하고 로컬에서 지원되는 AWS 서비스를 사용할 수 있습니다. AWS OpsHub 디바이스의 스토리지 용량 및 활성 인스턴스와 같은 주요 지표를 요약하는 대시보드를 제공합니다. 또한 Snow Family 디바이스에서 지원되는 다양한 AWS 서비스를 제공합니다. 몇 분 안에 파일을 디바이스로 전송할 수 있습니다.

[AWS OpsHub 애플리케이션을 다운로드하여](#) 클라이언트 컴퓨터에 설치한 후에는 AWS Snowcone 장치가 Wi-Fi를 통해 연결되어 있든 물리적 케이블을 통해 연결되어 있든 관계없이 동일한 네트워크에 있는 장치에 연결할 AWS OpsHub 수 있습니다. 그런 다음 장치를 AWS OpsHub 열고 잠금을 해제합니다. 그러면 디바이스 및 시스템 지표를 보여주는 대시보드가 표시됩니다. 그런 다음 엣지 애플리케이션 배포를 시작하거나 데이터를 디바이스로 마이그레이션할 수 있습니다. AWS OpsHub drag-and-drop 파일 또는 폴더를 장치에 저장할 수 있도록 하여 Snowcone 장치로 데이터를 간단하게 전송할 수 있습니다. AWS OpsHub를 사용하면 장치에 저장된 내용을 쉽게 확인할 수도 있습니다.

주제

- [Snow 패밀리 AWS OpsHub 디바이스용 다운로드](#)
- [디바이스 잠금 해제](#)
- [의 PGP 서명 확인 \(선택 사항\) AWS OpsHub](#)

- [장치의 AWS 서비스 관리](#)
- [파일을 전송하는 DataSync 데 사용 AWS](#)
- [디바이스 관리](#)
- [디바이스의 NTP 시간 서버 설정](#)

Snow 패밀리 AWS OpsHub 디바이스용 다운로드

다운로드하려면 AWS OpsHub

1. [AWS Snowball 리소스 웹 사이트](#)로 이동합니다.

OpsHub

OpsHub is a graphical user interface you can use to manage Snowball devices. OpsHub makes it easy to setup and manage Snowball devices enabling you to rapidly deploy edge computing workloads and simplify data migration to the cloud. With just a few clicks in OpsHub, you have the full functionality of the Snow Family of devices at your fingertips; you can unlock and configure devices, drag-and-drop data to devices, launch applications, and monitor device metrics.

- [OpsHub documentation](#)

	OpsHub
Windows 7 or higher	Download
Mac OS X 10.10 or higher	Download
Linux (Ubuntu version 14 or higher, and Fedora version 24 or higher)	Download
	(Signature)

2. AWS OpsHub 섹션에서 운영 체제에 맞는 다운로드를 선택하고 설치 단계를 따릅니다.

디바이스 잠금 해제

디바이스가 사이트에 도착하면 첫 번째 단계는 디바이스를 연결하고 잠금을 해제하는 것입니다. AWS OpsHub 를 통해 다음 방법으로 디바이스에 로그인하고 잠금을 해제하고 디바이스를 관리할 수 있습니다.

- 로컬 - 디바이스에 로컬로 로그인하려면 디바이스의 전원을 켜고 로컬 네트워크에 연결해야 합니다. 그런 다음 잠금 해제 코드와 매니페스트 파일을 제공합니다.

- 원격 - 디바이스에 원격으로 로그인하려면 디바이스의 전원을 켜고 네트워크를 통해 **device-order-region**.amazonaws.com에 연결할 수 있는지 확인해야 합니다. 그런 다음 장치에 연결된 자격 증명 AWS Identity and Access Management (IAM) 자격 증명 (액세스 키 및 비밀 키) 을 제공하십시오. AWS 계정

원격 관리를 활성화하고 관련 계정을 생성하는 방법에 대한 자세한 내용은 [Snow 디바이스 관리 활성화](#)를 참조하십시오.

주제

- [로컬에서 디바이스 잠금 해제](#)
- [원격으로 디바이스 잠금 해제](#)

로컬에서 디바이스 잠금 해제

로컬에서 디바이스를 연결하고 잠금을 해제하려면

1. 디바이스의 덮개를 열고 전원 코드를 찾아 디바이스를 전원에 연결합니다.
2. 네트워크 케이블 (일반적으로 Ethernet RJ45 케이블) 을 사용하여 장치를 네트워크에 연결한 다음 전면 패널을 열고 장치의 전원을 켭니다.
3. 애플리케이션을 엽니다. AWS OpsHub 처음 사용하는 사용자인 경우 언어를 선택하라는 메시지가 표시됩니다. 다음을 선택합니다.
4. 시작하기 OpsHub 페이지에서 로컬 장치에 로그인을 선택한 다음 로그인을 선택합니다.



Get started with OpsHub

Sign into local devices
You'll need an unlock code and manifest file

Sign into remote devices
You'll need an access key & secret key

Sign in

5. 로컬 디바이스 로그인 페이지에서 Snow Family 디바이스 유형을 선택한 다음 로그인을 선택합니다.
6. 로그인 페이지에서 디바이스 IP 주소와 잠금 해제 코드를 입력합니다. 디바이스 매니페스트를 선택하려면 파일 선택을 선택한 다음 로그인을 선택합니다.



Sign into your Snowball Edge

Sign in with an unlock code and manifest file


Device IP address

Eg 12.34.45.678

Unlock code

7c0e1-bab84-f7675-0a2b6-bfcc3

Manifest file

 Choose file

No file chosen

Back

Sign in

- 선택적으로 디바이스의 자격 증명을 프로필로 저장할 수 있습니다. 프로필 이름을 지정하고 프로필 이름 저장을 선택합니다. 프로필에 대한 자세한 내용은 [프로필 관리](#) 섹션을 참조하십시오.
- 로컬 장치 탭에서 장치를 선택하여 장치에서 실행 중인 네트워크 인터페이스 및 AWS 서비스와 같은 세부 정보를 확인합니다. 또한 이 탭에서 클러스터의 세부 정보를 보거나 AWS Command Line Interface (AWS CLI) 를 사용하는 것처럼 장치를 관리할 수 있습니다. 자세한 설명은 [장치의 AWS 서비스 관리](#) 섹션을 참조하세요.

Note

NFS 서비스가 시작되기 전까지는 Snowcone 디바이스의 사용 가능한 스토리지 공간이 정확하지 않습니다. [NFS 인터페이스 관리](#) 섹션을 참조하십시오.

AWS Snow Device Management 설치된 장치의 경우 원격 관리 활성화를 선택하여 기능을 켤 수 있습니다. 자세한 설명은 [AWS Snow Device Management를 사용하여 디바이스 관리](#) 섹션을 참조하세요.

원격으로 디바이스 잠금 해제

Snow Family 기기를 잠금 해제하려면 다음을 수행하십시오.

원격으로 디바이스를 연결하고 잠금을 해제하려면

1. 디바이스의 덮개를 열고 전원 코드를 찾아 디바이스를 전원에 연결합니다.
2. 이더넷 케이블(일반적으로 RJ45 케이블)을 사용하여 디바이스를 네트워크에 연결하고 전면 패널을 연 다음 디바이스 전원을 켭니다.

Note

원격으로 잠금을 해제하려면 디바이스가 `device-order-region.amazonaws.com`에 연결할 수 있어야 합니다.

3. AWS OpsHub 애플리케이션을 엽니다. 처음 사용하는 사용자인 경우 언어를 선택하라는 메시지가 표시됩니다. 다음을 선택합니다.
4. 시작하기 OpsHub 페이지에서 원격 장치에 로그인을 선택한 다음 로그인을 선택합니다.



Get started with OpsHub

Sign into local devices
You'll need an unlock code and manifest file

Sign into remote devices
You'll need an access key & secret key

Sign in

5. 원격 디바이스에 로그인 페이지에서 디바이스에 연결된 AWS 계정 의 AWS Identity and Access Management (IAM) 자격 증명(액세스 키 및 비밀 키)을 입력한 다음 로그인을 선택합니다.



Sign into remote devices

Sign in with an access key and secret key

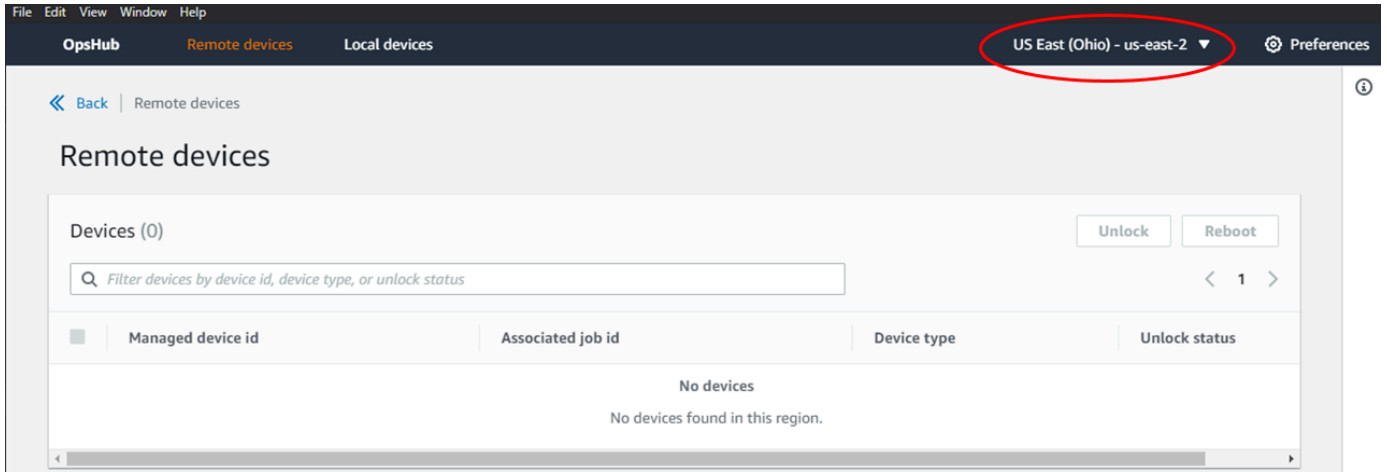
Access key

Secret key

Back

Sign in

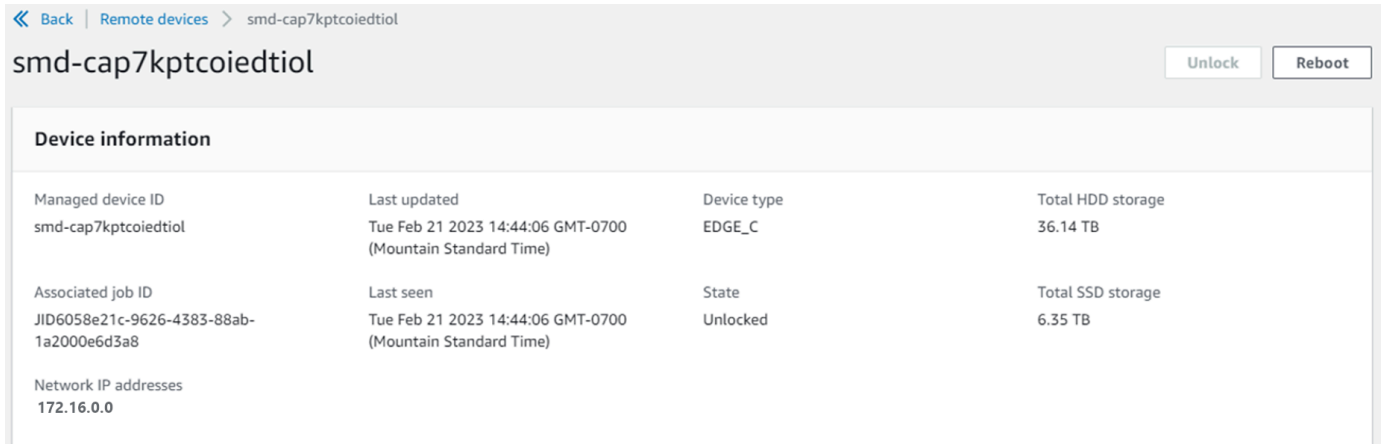
- 원격 디바이스 탭 상단에서 원격으로 잠금을 해제할 Snow 디바이스의 리전을 선택합니다.



- 원격 디바이스 탭에서 디바이스를 선택하면 상태 및 네트워크 인터페이스와 같은 세부 정보를 볼 수 있습니다. 그런 다음 잠금 해제를 선택하여 디바이스의 잠금을 해제합니다.

Note

NFS 서비스가 시작되기 전까지는 Snowcone 디바이스의 사용 가능한 스토리지 공간이 정확하지 않습니다. [NFS 인터페이스 관리](#) 섹션을 참조하십시오.



원격 장치의 세부 정보 페이지에서 AWS Command Line Interface (AWS CLI) 를 사용하는 것처럼 장치를 재부팅하고 관리할 수도 있습니다. 다른 AWS 리전위치에 있는 원격 장치를 보려면 탐색 표시줄에서 현재 지역을 선택한 다음 보려는 지역을 선택합니다. 자세한 설명은 [장치의 AWS 서비스 관리](#) 섹션을 참조하세요.

의 PGP 서명 확인 (선택 사항) AWS OpsHub

Linux 운영 체제용 AWS OpsHub 애플리케이션 설치 프로그램 패키지는 암호로 서명되어 있습니다. 퍼블릭 키를 사용하여 설치 관리자 패키지가 원본이며 수정되지 않았는지 확인할 수 있습니다. 파일에 손상이나 변경이 있을 경우 확인에 실패합니다. GPG(GNU Privacy Guard)를 사용하여 설치 관리자 패키지의 서명을 확인할 수 있습니다. 이 확인은 선택 사항입니다. 애플리케이션의 서명을 확인하기로 선택한 경우 언제든지 수행할 수 있습니다.

[AWS Snowcone 리소스 또는 Snowball Edge 리소스에서](#) Linux 운영 체제 설치 프로그램을 위한 SIGNATURE 파일을 다운로드할 수 있습니다.

Linux 운영 체제용 AWS OpsHub 설치 패키지를 확인하려면

1. 다음 공개 키를 복사하여 파일에 저장하고 파일 이름을 지정합니다. 예를 들어 opshub-public-key.pgp입니다.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
xsFNBF/hGf8BEAC9HCDV8uLjDX02Jxspi6kmPu4xqf4ZZLQsSqJcHU61oL/c
/zAN+mUqJT9aJ1rr0QFGVD1bMogecUPf1TW1DkEEpG8ZbX5P8vR+EE10/rW/
WtqizSudy6qy59ZRK+YVSDx7DZyuJmI07j00UADCL+95ZQN9vqwHNjBHsgfQ
l/1Tqhy81ozTZXCi/+u+99YLaugJIP6ZYIEdfpxnghqyVtaappBFTAyfg67Y
N/5mea1VqJzd8liFpIFQnl+X7U2x6emDbM01yJWV3aMmPwhtQ7iBdt5a4x82
EF5bZJ8HSRMvANDILD/9VTN8VfUQGKFjFY2GdX9ERwvfTb47bbv9Z28V1284
4lw2w1B1007Fo02v/Y0ukrN3VHCpmJQS1IiqZbYRa0DVK6UR5QNvUlj5fwWs
4qW9UDPhT/HDuaMrMFCejEn/7wvRUrGVtzCT9F56A1/dwRSxBejQQEb1AC8j
uuyi7gJaPdyNntR0EFTD7i02L6X2jB4YLfvGxP7Xeq1Y37t8NKF8CYTp0ry/
Wvw0iKZFbo4AkiI0aLyBCk9HBXhUKA9x06g0nhh1UFQrPGrk60RPQKqL76HA
E2ewzGDa90wLRBUAt2nRQpyNYjoASBvz/cAr3e0nuWsIzopZienrxI5ffcjY
f6UWA/OK3ITHTYHewVhseDyEqTQ4MUIWQS4NAwARAQABzTlBV1MgT3BzSHVi
IGZvciBTbm93IEZhbWlseSA8YXdzLW9wc2h1Yi1zaWduZXJAYW1hem9uLmNv
bT7CwY0EEAEIACAFA1/hGf8GCwkHCAMCBBUICgIEFgIBAAIZAQIBAwIeAQAh
CRAHgc9adPNF8RYhBDcvpe1IaY930b0vqiGBz1p080XxGbcP+gPZX7LzKc1Y
w9CT3UHgkAIawOSXYktujzoYVxAz8/j3jEkCY0dKnfyqvWZDiJAXnzmXWwBg
cxg1g0GXNXCM4lAd68CmbA0LoLTaWSQX30ZbswzhbtX2ADAlOpV8RLBik7fm
bS9FyuubDRhfYRQq0fpjUGXFiEgwg6aMFxsrGLlv4QD7t+6ftFie/mxLbjR4
iMgtr8FIPXbgn05YYY/LeF4NIgX4iLEqRbAnfWjPzqQ1spFWAotIzDmZqby+
WdWThrH4K1rwtYM8sDhqRnMnqJrGFZzk7aDhVPwF+FOVMmPeEN5JRazEeUr1
VZaSw6mu0n4FMGSXuwGgdvmkqnMe6I5/xLdU4IOPNhp0UmakDW0q/a1dREDE
ZLMQDMINphmeQno4inGmwbRo63gitD4ZNR5sWwfuwty251o8Ekv7jkkp3mSv
pdxn5tptttnPaSPcSIX/4ED119Tu0i7aup+v30t7eikYDSZG6g9+jHB3Va9e
/VWShFSgy8Jm2+qq/ujUQDAGTCfSuY9jg1ITsog6ayEza/2upDJ1m+40HK4p
8DrEzP/3jTahT8q5ofFWSRD17d31TSU+JBmPE3mz311FNXgi08w+taY320z
```

```

+irHtb3iSiiukbjS8s0maVgzszRqS9mhaEn4LL0zoqrUicmXgTyFB7n2LuYv
07vxM05xxhGQwsF2BBABCAAJBQJf4RoCAhsDACEJEBFVzT/tDi5FiEEi+09
V+UAYN9Gnw36EVm/NP+00LnnEQ/+J4C0Mn8j0AebXrwBiFs83sQo2q+WHL1S
MRc1g5gRFDXs6h1Gv+TGXRen7j1oeaddWvg0tUBxqmC0jr+8AKH00tiBWSu0
lsS8JU5rindEsKUrkTwcG2wyZFoe1z1E8xPkLRSRN5ZbbgKsTz16l1HgCCId
Do+WJdDkWGwXmtDvzjM32EI/PVBd108ga9aPwXdhLw0dKAjZ4JrJXLUQJjRI
IVDSyM0bEH0UM6a/+mWNZazNfo0LsGwqGva6Xn5WJWlwR1S78vPNf03BQYu0
YRjaVQR+kPtB9aSAZni5sWfk6N1rRNd1Q78d067uhhejsjRt7Mja2fEL4Kb1X
nK4U/ps7X103o/VjblneZ0hJK6kAKU172tnPJTJ31Jb0xX73wsMWDYZRZVcK
9X9+GFrpwKHwKkpjpm0t/FRxNepvqRl72TkgBPqGH2TM0FdB1f/uQprivqge
PBbS0JrmBIH9/anIqgtMdtcNQB/0erLdCDqI5af0uD10LcLwdJwG9/bSrfwT
TVEE3WbXmJ8pZgMzlHUizE6V2DSadV/YItk50I0jJR0VH0Hv1FMwGCEAIFzf
9P/pNi8hpEm1RphRi0VVcdQ30bH0M0gPHU5V9f1IhyCL1zU3LjYTHkq0yJD5
YDA1x01MYq3DcSM5130VBbLmuVS2GpcsTCYqlgQA6h/zzMwz+/70wU0EX+EZ
/wEQA0AY8ULmcJIQWIr14V0jy1pJeD3qwj7wd+QsBzJ+mOp0B/3ZFAhQiN01
9yCDLHeiZeAmWYX90IXrNiIdcHy+WTAp4G+NaMpqE52qhbDjz+IbvLp11yDH
bYEHpjnTHXEy21bvKAJ0Kkw/2RcQ0i4dodGnq5icyYj+9gcuHvnVwbrQ96Ia
0D7c+b5T+bzFqk90nIcztrMRuhDLJnJpi70jpvQwfq/TkkZA+mzupxfSkq/Y
N9qXNEToT/VI2gn/LS0X4Ar112KxBjzNEsQkwGSiWSYtMA5J+Tj5ED0uZ/qe
omNblA1D4bm7Na8NAoLxCtAiDq/f3To9Xb18lHsnd0mfLCb/BVgP4edQKTii
C/OZH9QJ1fmN0aq7JVLQAuvQNEL88RKW6YZBqkPd3P6zdc7sWDLTMXM0d3I
e6NUvU7pW0E9NyRfUF+oT4s9wAJhAodinAi8Zi9rEfhK1VCJ76j7bcQyZe0
jXD3Ij7T+X2XA8M/BmypwMw0Soljzhwh044RAasr/fAzpKNPB318JwcQunIz
u2N3CeJ+zrsomjcPxzehwsSVq1lzaL2ureJBL0KkBgYxUJYXpbS01ax1TsFG
091dAN0s9Ej8CND37GsNnuygj0gWXB6MNgbvPs3H3zi/AbMunQ1VB1w07JX
zdM1hBQZh6w+NeiEsK1T6wHi7IhxABEBAAHcWXYEGAEIAAKFAL/hGf8CGwwA
IQkQIYHPwnTzRfEWIQQ3L6XpSGmPd9Gzr6ohgc9adPNF8TMBD/9TbU/+PVbF
ywKvwi3GL0lpY7BXn8lQaHyunMGUavm080faRR0ynkH0ZqLHCp6bIajF0fvF
b7c0Jamzx8Hg+SIId16yRpRY+fa4RQ6PNnmT93ZgWW3EbjPyJG1m0/rt03SR
+0yn4/ldlg2KfBX4ppMoPCMKUdWxGmDETXsGihwZ0gmCZqXe81K122PYkSN
JQQ+L1fjKvCaxfPKEjXYTbIbfyyhCR6NzA0VZxCrzSz2xDrYwP/V002K1xda
0ix6r2aEHf+xEU0aBt80HY5nXTuRReCVU789MUVtCMqD2u6amdo4BR0kWA
QNg4yavKwV+LVtyYh2Iju9VSyv4xL1Q4xKHvcAUrSH73bHG7b7jkUJckD0f4
twhjJk/Lfwe6RdnVo2WoeTvE93w+NAq2FXmvbiG7eltl0XfQecvQU3QNbRvH
U8B96W0w8UXJdvTKg4f0NbjSw7iJ3x5naixQ+rA8hLV8x0gn2LX6wvxT/SEu
mn20KX+fPtJELK7v/NheFLX1jsKLXYo4jHrkfIXNsNUhg/x2E71kAjbeT3s+
t9kCtxt2iXDDZvpIbmG04QkvLFvoR0aSmN6+8fupe3e+e2yN0e6xGTuE60gX
I2+X1p1g9IduDYTp0I20XleHyyMqGeeIb4g0iisloTp5oi3EuAYRGflXuqAT
VA19bKnpkBsJ0A==
=tD2T
-----END PGP PUBLIC KEY BLOCK-----

```

2. 퍼블릭 키를 키링으로 가져오고 반환된 키 값을 기록해 둡니다.

GPG

```
gpg --import opshub-public-key.gpg
```

출력 예시

```
gpg: key 1655BBDE2B770256: public key "AWS OpsHub for Snow Family <aws-opshub-signer@amazon.com>" imported
gpg: Total number processed: 1
gpg:             imported: 1
```

3. 지문을 확인합니다. *key-value*을 이전 단계의 값으로 바꿔야 합니다. GPG를 사용하여 지문을 확인하는 것이 좋습니다.

```
gpg --fingerprint key-value
```

다음과 비슷한 출력이 반환됩니다.

```
pub  rsa4096 2020-12-21 [SC]
     372F A5E9 4869 8F77 D1B3  AFAA 2181 CF5A 74F3 45F1
uid  [ unknown] AWS OpsHub for Snow Family <aws-opshub-signer@amazon.com>
sub  rsa4096 2020-12-21 [E]
```

지문은 다음과 일치해야 합니다.

```
372F A5E9 4869 8F77 D1B3 AFAA 2181 CF5A 74F3 45F1
```

지문이 일치하지 않으면 AWS OpsHub 애플리케이션을 설치하지 마세요. AWS Support에 문의하십시오.

4. 설치 관리자 패키지를 확인하고, 아직 다운로드하지 않은 경우 인스턴스의 아키텍처 및 운영 체제에 따라 SIGNATURE 파일을 다운로드합니다.
5. 설치 관리자 패키지 서명을 확인합니다. 반드시 *signature-filename* 및 *OpsHub-download-filename*를 SIGNATURE 파일 및 AWS OpsHub 애플리케이션을 다운로드할 때 지정한 값으로 바꿉니다.

GPG

```
gpg --verify signature-filename OpsHub-download-filename
```

다음과 비슷한 출력이 반환됩니다.

GPG

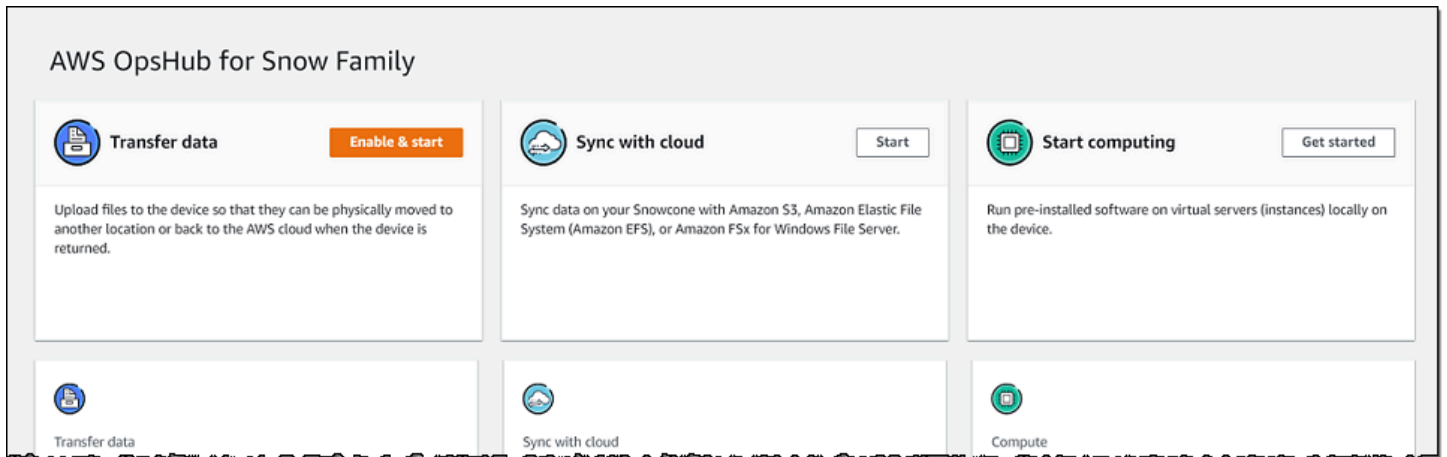
```
gpg: Signature made Mon Dec 21 13:44:47 2020 PST
gpg:          using RSA key 1655BBDE2B770256
gpg: Good signature from "AWS OpsHub for Snow Family <aws-opshub-
signer@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 9C93 4C3B 61F8 C434 9F94 5CA0 1655 BBDE 2B77 0256
```

GPG 사용 시, 출력에 BAD signature 문구가 포함된 경우 절차를 올바르게 수행했는지 확인합니다. 이 응답을 계속 받으면 에이전트를 설치하지 말고 AWS Support 문의하세요. 신뢰에 대한 경고 메시지는 서명이 유효하지 않다는 의미가 아니라 퍼블릭 키를 확인하지 않았다는 의미입니다. 사용자 또는 사용자가 신뢰하는 사람이 서명한 키만 신뢰됩니다.

장치의 AWS 서비스 관리

를 AWS OpsHub 사용하면 Snow Family 디바이스에서 AWS 서비스를 사용하고 관리할 수 있습니다. 현재, 다음 리소스를 AWS OpsHub 지원합니다.

- Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스 - Amazon EC2 호환 인스턴스를 사용하면 처리를 위해 AWS 클라우드에 소프트웨어를 보내지 않고도 가상 서버에 설치된 소프트웨어를 실행할 수 있습니다.
- AWS DataSync—온프레미스 스토리지와 파일 시스템 또는 Amazon S3와 같은 다른 AWS 클라우드 위치 간에 대량의 파일을 전송합니다.
- NFS(Network File System) - 파일 공유를 사용하여 데이터를 디바이스로 이동합니다. 디바이스를 배송하여 데이터를 AWS로 전송하거나 다른 AWS 클라우드 위치로 AWS 클라우드 전송하는 DataSync 데 사용할 수 있습니다.



주제

- [로컬에서 Amazon EC2 호환 컴퓨팅 인스턴스 사용](#)
- [NFS 인터페이스 관리](#)

로컬에서 Amazon EC2 호환 컴퓨팅 인스턴스 사용

를 AWS OpsHub 사용하여 디바이스의 로컬 가상 서버 (인스턴스) 에서 사전 설치된 소프트웨어를 실행하고 디바이스의 Amazon EC2 인스턴스를 관리할 수 있습니다.

주제

- [Amazon EC2 호환 인스턴스 시작하기](#)
- [Amazon EC2 호환 인스턴스 중지](#)
- [Amazon EC2 호환 인스턴스 시작](#)
- [키 페어 작업](#)
- [Amazon EC2 호환 인스턴스 종료](#)
- [로컬에서 스토리지 볼륨 사용](#)

Amazon EC2 호환 인스턴스 시작하기

를 사용하여 Amazon EC2 호환 인스턴스를 시작하려면 다음 단계를 따르십시오. AWS OpsHub

Amazon EC2 호환 인스턴스를 시작하려면

1. 애플리케이션을 엽니다. AWS OpsHub

2. 대시보드의 Start computing(컴퓨팅 시작) 섹션에서 Get started(시작하기)를 선택합니다. 또는 상단의 Services(서비스) 메뉴를 선택한 다음 Compute(EC2)(컴퓨팅(EC2))를 선택하여 Compute(컴퓨팅) 페이지를 엽니다. 모든 컴퓨팅 리소스가 Resources(리소스) 섹션에 표시됩니다.
3. 디바이스에서 Amazon EC2 호환 인스턴스를 실행 중인 경우 인스턴스 아래의 인스턴스 이름 옆에 해당 인스턴스가 표시됩니다. 이 페이지에서 각 인스턴스의 세부 정보를 볼 수 있습니다.
4. 인스턴스 시작을 선택합니다. 인스턴스 시작 마법사가 열립니다.
5. 디바이스에서 Amazon EC2 호환 인스턴스를 시작할 Snow 디바이스를 선택합니다.

Launch instance ✕

Device

192.0.2.0 ▼

Image (AMI)

snow-al2-test-ami-1.0.2 ▼

Instance type

sbe-c.small ▼

Create public IP address (VNI) Use existing IP address (VNI) Do not attach IP address

Physical network interface

SFP+:a.bc-1d2ef456gg678gi9j ▼

IP Address assignment

DHCP ▼

Key pair

Create key pair Use existing key pair Do not attach key pair

Name

test-instance-key-pair

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Create key pair

Cancel **Launch**

6. Image(AMI)(이미지(AMI))에서 Amazon 머신 이미지(AMI)를 목록에서 선택합니다. 이 AMI가 인스턴스를 시작하는 데 사용됩니다.
7. Instance type(인스턴스 유형)에서 인스턴스 유형을 목록에서 선택합니다.
8. 인스턴스에 IP 주소를 연결하는 방법을 선택합니다. 다음과 같은 옵션이 있습니다:
 - 퍼블릭 IP 주소 생성(VNI) - 물리적 네트워크 인터페이스를 사용하여 새 IP 주소를 생성하려면 이 옵션을 선택합니다. 물리적 네트워크 인터페이스와 IP 주소 할당을 선택합니다.
 - 기존 IP 주소 사용(VNI) - 기존 IP 주소를 사용하고 기존 가상 네트워크 인터페이스를 사용하려면 이 옵션을 선택합니다. 물리적 네트워크 인터페이스와 가상 네트워크 인터페이스를 선택합니다.
 - IP 주소 연결 안 함 - IP 주소를 연결하지 않으려면 이 옵션을 선택합니다.
9. 인스턴스에 키 페어를 연결하는 방법을 선택합니다. 다음과 같은 옵션이 있습니다:

키 페어 생성 - 새 키 페어를 생성하고 이 키 페어로 새 인스턴스를 시작하려면 이 옵션을 선택합니다.

기존 키 페어 사용 - 기존 키 페어를 사용하여 인스턴스를 시작하려면 이 옵션을 선택합니다.

IP 주소 연결 안 함 - 키 페어를 연결하지 않으려면 이 옵션을 선택합니다. 이 AMI에 내장된 암호를 이미 알고 있지 않다면 이 인스턴스에 연결할 수 없다는 점을 인정해야 합니다.

자세한 내용은 [키 페어 작업](#) 섹션을 참조하세요.
10. 시작을 선택합니다. Compute instances(컴퓨팅 인스턴스) 섹션에서 인스턴스가 시작되는 것을 볼 수 있습니다. State(상태)가 Pending(대기 중)으로 나타나며 완료되면 Running(실행 중)으로 변경됩니다.

Amazon EC2 호환 인스턴스 중지

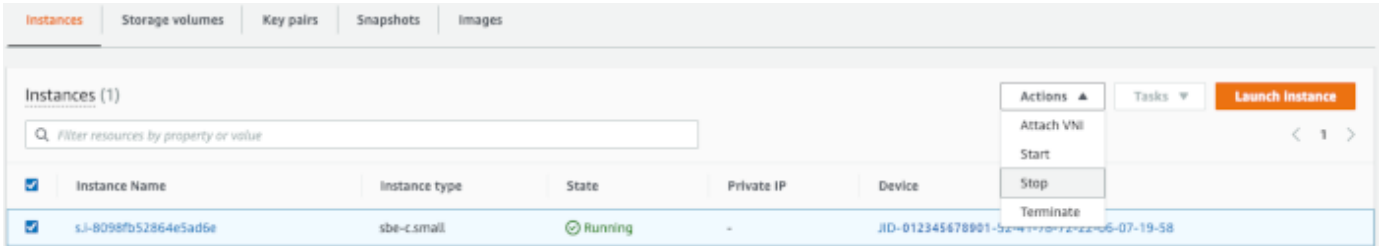
다음 단계를 사용하여 Amazon EC2 호환 인스턴스를 AWS OpsHub 중지할 수 있습니다.

Amazon EC2 호환 인스턴스를 중지하려면

1. 애플리케이션을 엽니다. AWS OpsHub
2. 대시보드의 Start computing(컴퓨팅 시작) 섹션에서 Get started(시작하기)를 선택합니다. 또는 상단의 Services(서비스) 메뉴를 선택한 다음 Compute(EC2)(컴퓨팅(EC2))를 선택하여 Compute(컴퓨팅) 페이지를 엽니다.

모든 컴퓨팅 리소스가 Resources(리소스) 섹션에 표시됩니다.

3. 디바이스에서 Amazon EC2 호환 인스턴스를 실행 중인 경우 인스턴스 아래의 인스턴스 이름 옆에 해당 인스턴스가 표시됩니다.
4. 중지할 인스턴스를 선택하고 작업 메뉴를 선택한 다음 중지를 선택합니다. State(상태)가 Stopping(중지 중)으로 변경되고 완료되면 Stopped(중지됨)로 변경됩니다.



Amazon EC2 호환 인스턴스 시작

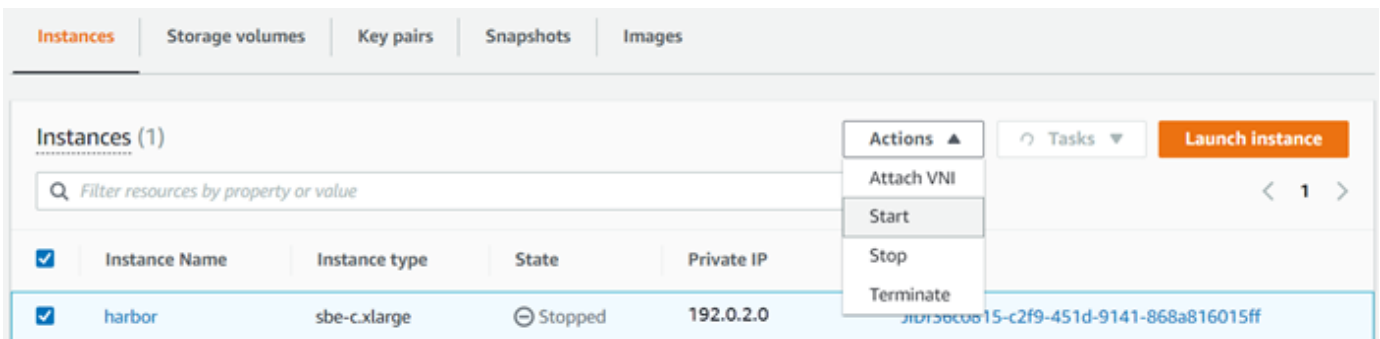
다음 단계를 사용하여 Amazon EC2 호환 인스턴스를 시작할 수 있습니다. AWS OpsHub

Amazon EC2 호환 인스턴스를 시작하려면

1. 애플리케이션을 엽니다. AWS OpsHub
2. 대시보드의 Start computing(컴퓨팅 시작) 섹션에서 Get started(시작하기)를 선택합니다. 또는 상단의 Services(서비스) 메뉴를 선택한 다음 Compute(EC2)(컴퓨팅(EC2))를 선택하여 Compute(컴퓨팅) 페이지를 엽니다.

컴퓨팅 리소스가 Resources(리소스) 섹션에 표시됩니다.

3. 인스턴스 이름 옆의 인스턴스에서 시작하려는 인스턴스를 찾습니다.
4. 인스턴스를 선택한 후 Start(시작)를 선택합니다. State(상태)가 Pending(대기 중)으로 변경되며 완료되면 Running(실행 중)으로 변경됩니다.



키 페어 작업

Amazon EC2 호환 인스턴스를 시작하고 SSH를 사용하여 연결하려는 경우 키 페어를 제공해야 합니다. Amazon EC2를 사용하여 새 키 페어를 생성하거나 기존 키 페어를 가져오거나 키 페어를 관리할 수 있습니다.

키 페어를 생성하거나 가져오거나 관리하려면

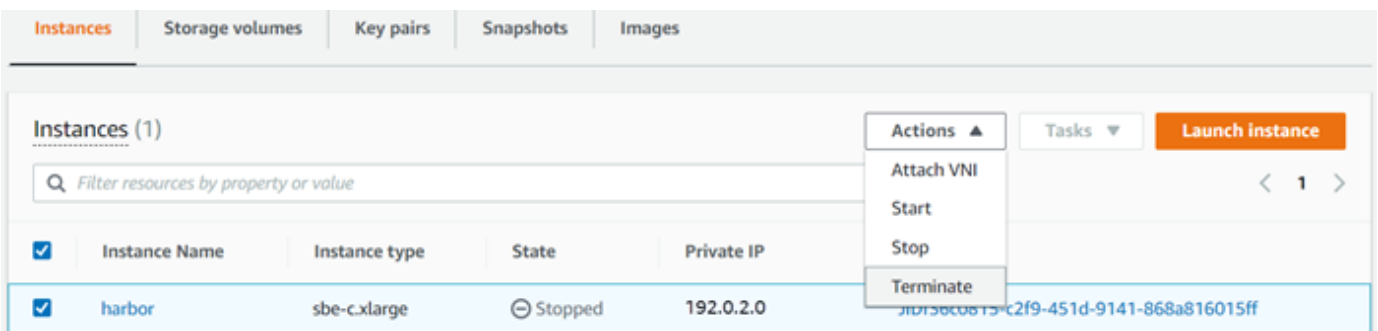
1. AWS OpsHub 대시보드에서 Compute를 엽니다.
2. 탐색 창에서 컴퓨팅(EC2) 페이지를 선택한 다음 키 페어 탭을 선택합니다. 키 페어를 생성하거나 가져오거나 관리할 수 있는 Amazon EC2 콘솔로 리디렉션됩니다.
3. 키 페어를 생성하거나 가져오는 방법에 대한 자세한 내용은 Amazon EC2 Linux 인스턴스용 사용 설명서의 [Amazon EC2 키 페어 및 Linux 인스턴스](#)를 참조하십시오.

Amazon EC2 호환 인스턴스 종료

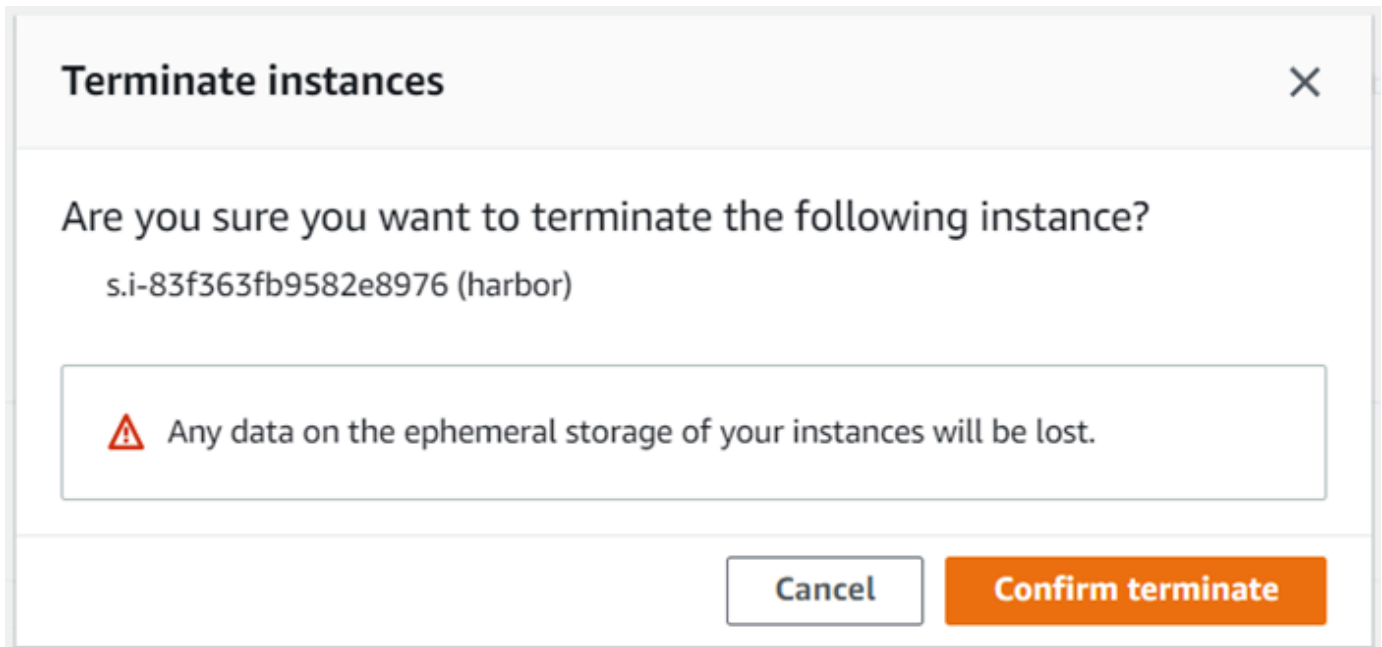
Amazon EC2 호환 인스턴스를 종료한 후에는 인스턴스를 다시 시작할 수 없습니다.

Amazon EC2 호환 인스턴스를 종료하려면

1. AWS OpsHub 애플리케이션을 엽니다.
2. 대시보드의 Start computing(컴퓨팅 시작) 섹션에서 Get started(시작하기)를 선택합니다. 또는 상단의 Services(서비스) 메뉴를 선택한 다음 Compute(EC2)(컴퓨팅(EC2))를 선택하여 Compute(컴퓨팅) 페이지를 엽니다. Resources(리소스) 섹션에서 모든 컴퓨팅 리소스를 볼 수 있습니다.
3. 인스턴스 이름 옆의 인스턴스에서 종료하려는 인스턴스를 찾습니다.
4. 인스턴스를 선택하고 작업 메뉴를 선택합니다. 작업 메뉴에서 종료를 선택합니다.



5. 인스턴스 종료 창에서 종료 확인을 선택합니다.



Note

인스턴스가 종료된 후에는 다시 시작할 수 없습니다.

State(상태)가 Terminating(종료 중)으로 변경되고 완료되면 Terminated(종료됨)으로 변경됩니다.

로컬에서 스토리지 볼륨 사용

Amazon EC2 호환 인스턴스는 Amazon EBS 볼륨을 스토리지로 사용합니다. 이 절차에서는 볼륨을 사용하여 스토리지 볼륨을 생성하고 인스턴스에 연결합니다 AWS OpsHub.

스토리지 볼륨을 생성하려면

1. AWS OpsHub 애플리케이션을 엽니다.
2. 대시보드의 Start computing(컴퓨팅 시작) 섹션에서 Get started(시작하기)를 선택합니다. 또는 상단의 Services(서비스) 메뉴를 선택한 다음 Compute(EC2)(컴퓨팅(EC2))를 선택하여 Compute(컴퓨팅) 페이지를 엽니다.
3. Storage volumes(스토리지 볼륨) 탭을 선택합니다. 디바이스에 스토리지 볼륨이 있는 경우 볼륨에 대한 세부 정보가 Storage volumes(스토리지 볼륨) 아래에 표시됩니다.
4. Create volume(볼륨 생성)을 선택하여 Create volume(볼륨 생성) 페이지를 엽니다.

5. 볼륨을 만들려는 장치를 선택하고 만들려는 크기 (인치 GiBs) 를 입력한 다음 볼륨 유형을 선택합니다.
6. 제출을 선택합니다. State(상태)가 Creating(생성 중)으로 나타나며 완료되면 Available(사용 가능)로 변경됩니다. Volumes(볼륨) 탭에서 볼륨 및 세부 정보를 볼 수 있습니다.

스토리지 볼륨을 인스턴스에 연결하려면

1. 생성한 볼륨을 선택한 다음 Attach volume(볼륨 연결)을 선택합니다.

2. 컴퓨팅 인스턴스 ID에서 볼륨을 연결할 인스턴스를 선택합니다.
3. Volume Device Name(볼륨 디바이스 이름)에 볼륨의 디바이스 이름(예: **/dev/sdh** 또는 **xvdh**)을 입력합니다.
4. 연결을 선택합니다.

볼륨이 더 이상 필요하지 않으면 인스턴스에서 볼륨을 분리한 다음 삭제할 수 있습니다.

NFS 인터페이스 관리

네트워크 파일 시스템 (NFS) 인터페이스를 사용하여 마치 장치가 운영 체제의 로컬 스토리지인 것처럼 Snow Family 장치에 파일을 업로드할 수 있습니다. 이렇게 하면 파일 복사, 드래그 앤 드롭 또는 기타 그래픽 사용자 인터페이스 기능과 같은 운영 체제 기능을 사용할 수 있으므로 보다 사용자 친화적인 방식으로 데이터를 전송할 수 있습니다. 디바이스의 각 S3 버킷은 NFS 인터페이스 엔드포인트로 사용할 수 있으며 데이터를 복사하기 위해 마운트할 수 있습니다. NFS 인터페이스는 가져오기 작업에 사용할 수 있습니다.

시작 시 NFS 인터페이스는 1GB의 메모리와 1개의 CPU를 사용합니다. 이로 인해 Snow Family 디바이스에서 실행되는 다른 서비스의 수 또는 실행할 수 있는 EC2 호환 인스턴스의 수가 제한될 수 있습니다.

NFS 인터페이스를 통해 전송된 데이터는 전송 중에 암호화되지 않습니다. NFS 인터페이스를 구성할 때 CIDR 블록을 제공할 수 있으며 Snow Family 디바이스는 해당 블록에 주소가 있는 클라이언트 컴퓨터에서 NFS 인터페이스에 대한 액세스를 제한합니다.

Amazon S3로 반환되면 디바이스에 있는 파일이 Amazon S3로 전송됩니다 AWS. 자세한 내용은 [Snowcone 작동 방식을](#) 참조하십시오.

컴퓨터 운영 체제에서 NFS를 사용하는 방법에 대한 자세한 내용은 운영 체제 설명서를 참조하십시오.

NFS 인터페이스를 사용할 때는 다음 세부 정보를 염두에 두십시오.

- 파일 이름은 Snow Family 디바이스의 로컬 S3 버킷에 있는 객체 키입니다. 키 이름은 최대 1,024바이트 길이와 UTF-8 인코딩을 사용하는 유니코드 문자의 시퀀스입니다. 가능한 경우 NFSv4.1을 사용하고 파일 이름을 유니코드 UTF-8 형식으로 인코딩하여 데이터를 성공적으로 가져오는 것이 좋습니다. UTF-8 인코딩으로 인코딩되지 않은 파일 이름은 사용하는 NFS 인코딩에 따라 S3에 업로드되지 않거나 다른 파일 이름으로 S3에 업로드될 수 있습니다.
- 파일 경로의 최대 길이는 1,024자 미만이어야 합니다. Snow Family 디바이스는 1,024자를 초과하는 파일 경로를 지원하지 않습니다. 이 파일 경로 길이를 초과하면 파일 가져오기 오류가 발생합니다.
- 자세한 내용은 Amazon 심플 스토리지 서비스 사용 설명서의 [객체 키](#)를 참조하십시오.
- NFS 기반 전송의 경우 Snow Family 디바이스에서 Amazon S3로 객체를 가져올 때 표준 POSIX 스타일 메타데이터가 객체에 추가됩니다. 또한 NFS 옵션을 사용한 Snow Family 디바이스 가져오기를 위해 현재 Amazon S3로 내부 가져오기 메커니즘의 AWS DataSync 일부로 사용하고 있는 메타데이터 "x-amz-meta-user-agent aws-datasync"가 표시됩니다.

Note

의 디바이스 세부 정보 페이지에서 Snowcone 디바이스의 AWS OpsHub 사용 가능한 스토리지 공간은 NFS 인터페이스가 시작될 때까지 정확하지 않습니다.

또한 CLI (명령줄 인터페이스) 도구인 Snowball Edge 클라이언트를 사용하여 NFS 인터페이스를 구성하고 관리할 수 있습니다. 자세한 내용은 NFS 인터페이스 [관리를 참조하십시오](#).

주제

- [Windows 운영 체제에서 NFS 서비스 시작](#)
- [NFS 인터페이스 자동 구성](#)
- [NFS 인터페이스를 수동으로 구성](#)
- [Snow Family 디바이스의 NFS 엔드포인트 관리](#)
- [클라이언트 컴퓨터에 NFS 엔드포인트 탑재](#)
- [NFS 인터페이스 중지](#)

Windows 운영 체제에서 NFS 서비스 시작

클라이언트 컴퓨터에서 Windows 10 Enterprise 또는 Windows 7 Enterprise 운영 체제를 사용하는 경우 애플리케이션에서 NFS를 구성하기 전에 클라이언트 컴퓨터에서 NFS 서비스를 시작하십시오.
AWS OpsHub

1. 클라이언트 컴퓨터에서 시작을 열고 제어판을 선택한 다음 프로그램을 선택합니다.
2. Windows 기능 사용/사용 안 함을 선택합니다.

Note

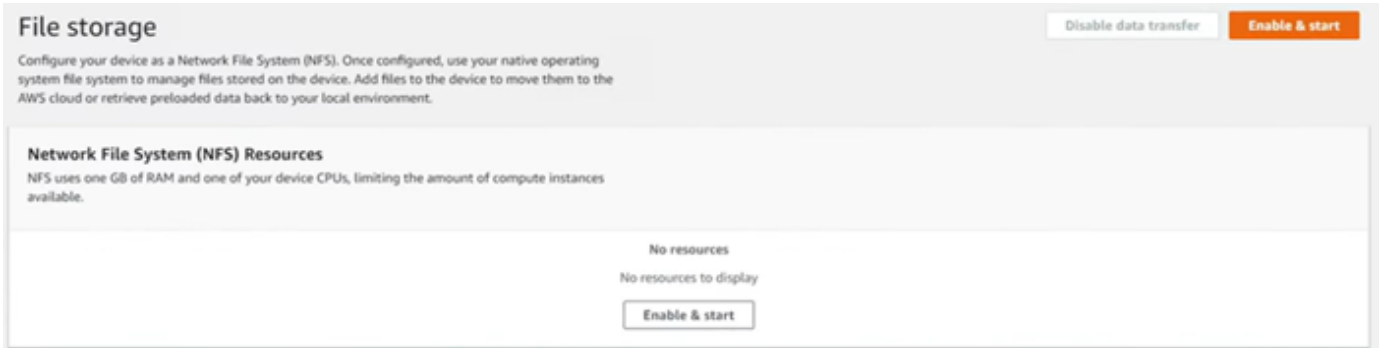
Windows 기능을 켜려면 컴퓨터의 관리자 이름과 암호를 제공해야 할 수 있습니다.

3. Services for NFS(NFS용 서비스)에서 Client for NFS(NFS용 클라이언트)를 선택하고 확인을 선택합니다.

NFS 인터페이스 자동 구성

NFS 인터페이스는 기본적으로 Snow Family 디바이스에서 실행되지 않으므로 디바이스에서 데이터 전송을 활성화하려면 인터페이스를 시작해야 합니다. Snow Family 디바이스는 몇 번의 클릭만으로 NFS 인터페이스를 빠르고 자동으로 구성할 수 있습니다. NFS 인터페이스를 직접 구성할 수도 있습니다. 자세한 설명은 [NFS 인터페이스를 수동으로 구성](#) 섹션을 참조하세요.

1. 대시보드의 데이터 전송 섹션에서 활성화 및 시작을 선택합니다. 이 작업을 완료하는 데 1~2분이 걸릴 수 있습니다.



2. NFS 서비스가 시작되면 NFS 인터페이스의 IP 주소가 대시보드에 표시되고 데이터 전송 섹션에는 서비스가 활성 상태라고 표시됩니다.
3. 탐색기에서 열기 (Windows 또는 Linux 운영 체제를 사용하는 경우) 를 선택하여 운영 체제의 파일 브라우저에서 파일 공유를 열고 Snow Family 디바이스로 파일 전송을 시작합니다. 클라이언트 컴퓨터의 파일을 복사하여 파일 공유에 붙여넣거나 끌어서 놓을 수 있습니다. Windows 운영 체제에서 파일 공유는 다음과 같습니다 `buckets(\\12.123.45.679)(Z:)`.

Note

Linux 운영 체제에서 NFS 엔드포인트를 마운트하려면 루트 권한이 필요합니다.

NFS 인터페이스를 수동으로 구성

NFS 인터페이스는 기본적으로 Snow Family 디바이스에서 실행되지 않으므로 디바이스에서 데이터 전송을 활성화하려면 인터페이스를 시작해야 합니다. 필요한 경우 Snow Family 디바이스에서 실행되는 가상 네트워크 인터페이스 (VNI) 의 IP 주소를 제공하고 파일 공유에 대한 액세스를 제한하여 NFS 인터페이스를 수동으로 구성할 수 있습니다. NFS 인터페이스를 수동으로 구성하기 전에 Snow Family 디바이스에 가상 네트워크 인터페이스 (VNI) 를 설정하십시오. 자세한 내용은 [컴퓨팅 인스턴스를 위한 네트워크 구성](#)을 참조하세요.

Snow Family 디바이스에서 NFS 인터페이스를 자동으로 구성하도록 할 수도 있습니다. 자세한 설명은 [NFS 인터페이스 자동 구성](#) 섹션을 참조하세요.

1. 대시보드의 데이터 전송 섹션 하단에서 수동 구성을 선택합니다.

2. 활성화 및 시작을 선택하여 NFS 시작 마법사를 엽니다. 물리적 네트워크 인터페이스 필드가 채워집니다.

Start NFS ✕

Physical network interface

RJ45: s.ni-8459d6c7273eed333 ▼

Create IP address (VNI)

Use existing IP address (VNI)

IP Address assignment

DHCP ▼

Restrict NFS to allowed hosts

Allow all hosts

Allowed hosts
Provide a set of CIDR blocks allowed to connect to the NFS service.

192.0.2.0/24 ✕

0.0.0.0/0 ✕

Add allowed hosts

Allow instances on this device to access NFS

Enable

Cancel
Start NFS

3. IP 주소(VNI) 생성을 선택하거나 기존 IP 주소 사용을 선택합니다.

4. IP 주소 생성(VNI)을 선택한 경우 IP 주소 할당 목록 상자에서 DHCP 또는 고정 IP를 선택합니다.

Important

DHCP 네트워크를 사용하는 경우 DHCP 서버가 NFS 인터페이스의 IP 주소를 재할당할 수 있습니다. 이는 디바이스 연결이 끊기고 IP 주소가 재할용된 후에 발생할 수 있습니다. 허용된 호스트 범위를 설정하고 클라이언트 주소가 변경되면 다른 클라이언트가 해당 주소를 선택할 수 있습니다. 이 경우 새로운 클라이언트가 공유에 액세스할 수 있습니다. 이를 방지하려면 DHCP 예약 또는 고정 IP 주소를 사용하십시오.

기존 IP 주소 사용을 선택한 경우 가상 네트워크 인터페이스 목록 상자에서 가상 네트워크 인터페이스를 선택합니다.

5. NFS 인터페이스에 대한 액세스를 제한하고 허용된 네트워크 주소 블록을 제공하거나 네트워크의 모든 디바이스가 Snow Family 디바이스의 NFS 인터페이스에 액세스하도록 허용하도록 선택합니다.
 - Snow Family 디바이스의 NFS 인터페이스에 대한 액세스를 제한하려면 NFS를 허용된 호스트로 제한을 선택합니다. 허용된 호스트에서 CIDR 블록 세트를 입력합니다. 둘 이상의 CIDR 블록에 대한 액세스를 허용하려면 다른 블록 세트를 입력하십시오. 블록 세트를 제거하려면 블록이 포함된 필드 옆의 X를 선택합니다. 허용된 호스트 추가를 선택합니다.

Note

NFS를 허용된 호스트로 제한하고 허용된 CIDR 블록을 제공하지 않는 경우 Snow Family 디바이스는 NFS 인터페이스를 마운트하기 위한 모든 요청을 거부합니다.

- 네트워크의 모든 디바이스가 NFS 인터페이스에 액세스할 수 있도록 허용하려면 Allow all host (Allow all host) 를 선택합니다.
6. Snow Family 디바이스에서 실행되는 EC2 호환 인스턴스가 NFS 어댑터에 액세스할 수 있도록 허용하려면 [Enable] 을 선택합니다.
 7. NFS 시작을 선택합니다. 시작하는 데 1~2분 정도 걸릴 수 있습니다.

Important

NFS 인터페이스가 시작되는 동안에는 Snow Family 디바이스를 끄지 마십시오.

네트워크 파일 시스템 (NFS) 리소스 섹션에서 NFS 인터페이스 상태가 활성으로 표시됩니다. 인터페이스를 클라이언트 컴퓨터에 로컬 스토리지로 마운트하려면 나열된 IP 주소가 필요합니다.

Snow Family 디바이스의 NFS 엔드포인트 관리

Snow Family 디바이스의 각 S3 버킷은 엔드포인트로 표시되며 마운트 경로에 나열됩니다. NFS 인터페이스가 시작된 후 엔드포인트를 마운트하여 해당 엔드포인트로 또는 엔드포인트에서 파일을 전송합니다. 한 번에 하나의 엔드포인트만 마운트할 수 있습니다. 다른 엔드포인트를 마운트하려면 먼저 현재 엔드포인트를 마운트 해제하십시오.

엔드포인트를 마운트하려면

1. 마운트 경로 섹션에서 다음 중 하나를 수행하여 엔드포인트를 선택합니다.
 - Filter endpoints 필드에 항목의 사용 가능한 엔드포인트 목록을 필터링할 버킷 이름 전체 또는 일부를 입력한 다음 엔드포인트를 선택합니다.
 - 마운트 경로 목록에서 마운트할 엔드포인트를 선택합니다.
2. 마운트 NFS 엔드포인트를 선택합니다. Snow Family 디바이스는 엔드포인트를 마운트하여 사용합니다.

엔드포인트를 마운트 해제하려면

1. 마운트 경로 섹션에서 마운트 해제할 엔드포인트를 선택합니다.
2. 언마운트 엔드포인트를 선택합니다. Snow Family 디바이스는 엔드포인트를 마운트 해제하므로 더 이상 사용할 수 없습니다.

Note

엔드포인트를 마운트 해제하기 전에 엔드포인트에서 데이터를 복사하거나 엔드포인트로 데이터가 복사되지 않도록 하십시오.

클라이언트 컴퓨터에 NFS 엔드포인트 탑재

NFS 인터페이스가 시작되고 엔드포인트가 마운트되면 엔드포인트를 클라이언트 컴퓨터에 로컬 스토리지로 마운트합니다.

1. 마운트 경로에서 마운트할 엔드포인트의 복사 아이콘을 선택합니다. 엔드포인트를 마운트할 때 운영 체제에 붙여넣으십시오.
2. 다음은 윈도우, 리눅스, macOS 운영 체제의 기본 마운트 명령입니다.

- Windows:

```
mount -o nolock rsize=128 wsize=128 mtype=hard nfs-interface-ip-address:/
buckets/BucketName *
```

- Linux:

```
mount -t nfs nfs-interface-ip-address:/buckets/BucketName mount_point
```

- macOS:

```
mount -t nfs -o vers=3,rsize=131072,wsize=131072,nolocks,hard,retrans=2 nfs-
interface-ip-address:/buckets/$bucketname mount_point
```

NFS 인터페이스 중지

Snow Family 장치와의 파일 전송을 완료하면 Snow Family 장치에서 NFS 인터페이스를 중지하십시오.

1. 대시보드에서 서비스를 선택한 다음 파일 스토리지를 선택합니다.
2. 파일 스토리지 페이지에서 데이터 전송 비활성화를 선택합니다. 대개 NFS 엔드포인트가 대시보드에서 사라지는 데 최대 2분이 걸립니다.

파일을 전송하는 DataSync 데 사용 AWS

를 AWS OpsHub 사용하여 Snowcone 디바이스에서 AWS DataSync 에이전트를 생성할 수 있습니다. 이를 사용하여 AWS 클라우드에서 디바이스와 Amazon S3, Amazon Elastic File System(Amazon EFS) 또는 Windows File Server용 FSx 간에 파일을 전송할 수 있습니다.

AWS DataSync AWS 스토리지 서비스와의 대량 데이터 복제를 간소화, 자동화 및 가속화하도록 설계된 온라인 데이터 전송 서비스입니다. DataSync 인터넷을 통해 데이터를 복사하거나 AWS Direct Connect. 완전 관리형 서비스이므로 DataSync 애플리케이션을 수정하거나 스크립트를 개발하거나 인프라를 관리할 필요가 거의 없습니다.

DataSync 네트워크 파일 시스템 (NFS) 과 Windows File Server용 Amazon EFS, Amazon S3 또는 Amazon FSx 간의 데이터 전송을 지원합니다.

에서 지원하는 AWS DataSync원본 및 대상 위치 조합에 대한 자세한 내용은 사용 설명서의 [AWS DataSync 위치 작업을](#) 참조하십시오.

Snowcone은 온프레미스 스토리지 시스템에서 데이터를 읽거나 쓰는 데 사용되는 가상 머신 (VM) 인 DataSync 에이전트와 함께 제공됩니다. 사용하려면 DataSync 먼저 에이전트를 시작한 다음 DataSync 콘솔로 이동하여 활성화해야 합니다. 에 대한 자세한 DataSync 내용은 [시작하기](#)를 참조하십시오 AWS DataSync.

DataSync 에이전트를 시작하려면

1. DataSync 에이전트를 시작하기 전에 Snowcone 디바이스에서 NFS를 활성화하십시오. [NFS 인터넷 페이지 자동 구성 및 NFS 시작 및 액세스 제한](#)를 참조합니다.
2. AWS OpsHub 대시보드의 클라우드와 동기화 섹션에서 시작을 선택하여 DataSync 에이전트 시작 마법사를 엽니다. DataSync 에이전트 시작 양식은 디바이스 IP 주소 및 물리적 네트워크 인터페이스 필드로 채워집니다.
3. IP 주소(VNI) 생성을 선택하여 가상 IP 주소를 생성하거나 기존 IP 주소 사용을 선택합니다.
4. IP 주소 생성(VNI)을 선택한 경우 IP 주소 할당 목록 상자에서 DHCP 또는 고정 IP를 선택합니다.

기존 IP 주소 사용을 선택한 경우 가상 네트워크 인터페이스 목록 상자에서 가상 인터페이스를 선택합니다.

5. Start agent(에이전트 시작)을 선택합니다. DataSync 리소스 페이지로 리디렉션됩니다. 에이전트의 IP 주소가 표시되는 데 최대 5분이 걸릴 수 있습니다.
6. 복사 아이콘을 사용하여 에이전트 IP 주소 파일에서 에이전트의 IP 주소 값을 복사한 다음 DataSync 콘솔 열기를 선택합니다.

그러면 DataSync 콘솔이 열리고 에이전트를 활성화하고 파일을 전송할 수 있습니다. 나머지 설정은 AWS DataSync 콘솔에서 수행됩니다.

를 사용하여 파일을 전송합니다. DataSync

AWS Snowcone 이미 에이전트를 만들었으므로 에이전트를 활성화하고, 소스 및 대상 위치를 구성하고, 작업을 생성하고, 작업을 시작하기만 하면 됩니다.

DataSync 에이전트를 활성화하고 DataSync 서비스를 사용하려면

1. <https://console.aws.amazon.com/datasync/> 에서 AWS DataSync 콘솔을 엽니다.
2. 에이전트 생성 페이지의 활성화 섹션에서 복사한 IP 주소를 에이전트 주소 상자에 붙여넣고 키 가져오기를 선택합니다. 브라우저가 IP 주소에 연결하고 에이전트의 고유한 정품 인증 키를 가져옵니다.
3. 에이전트가 활성화되면 Snowcone 디바이스에서 실행되는 NFS를 소스 위치로 구성합니다. DataSync 자세한 내용은 AWS DataSync 사용 설명서에서 [소스 위치 구성](#)을 참조하십시오.

Note

Snowcone 디바이스에서 실행되는 DataSync 에이전트는 네트워크에서 연결할 수 있는 위치로 파일을 주고받을 수 있습니다.

4. 대상 구성 페이지에서 파일을 전송할 대상을 선택하고 구성합니다. 자세한 지침은 AWS DataSync 사용 설명서의 [대상 위치 구성](#)을 참조하십시오.
5. 작업 설정을 구성합니다. 자세한 내용은 AWS DataSync 사용 설명서의 [태스크 설정 구성](#)을 참조하십시오.
6. 설정을 검토하고 작업을 생성합니다. 자세한 지침은 AWS DataSync 사용 설명서의 [설정 검토 및 태스크 생성](#)을 참조하십시오.
7. 작업을 시작하고 파일이 전송될 때까지 기다립니다. 지침을 보려면 AWS DataSync 사용 설명서의 [태스크 시작하기](#)를 참조하십시오.

디바이스 관리

를 사용하여 Snow AWS OpsHub Family 장치를 관리할 수 있습니다. 장치 세부 정보 페이지에서 장치 별칭 변경, 장치 재부팅 AWS CLI, 업데이트 확인 등 를 사용하여 수행하는 것과 동일한 작업을 수행할 수 있습니다.

주제

- [디바이스 재부팅](#)

- [디바이스 종료하기](#)
- [디바이스 별칭 편집](#)
- [장치 및 AWS OpsHub 애플리케이션에 대한 업데이트 받기](#)
- [프로필 관리](#)

디바이스 재부팅

다음 단계에 따라 Snow AWS OpsHub 장치를 재부팅하십시오.

Important

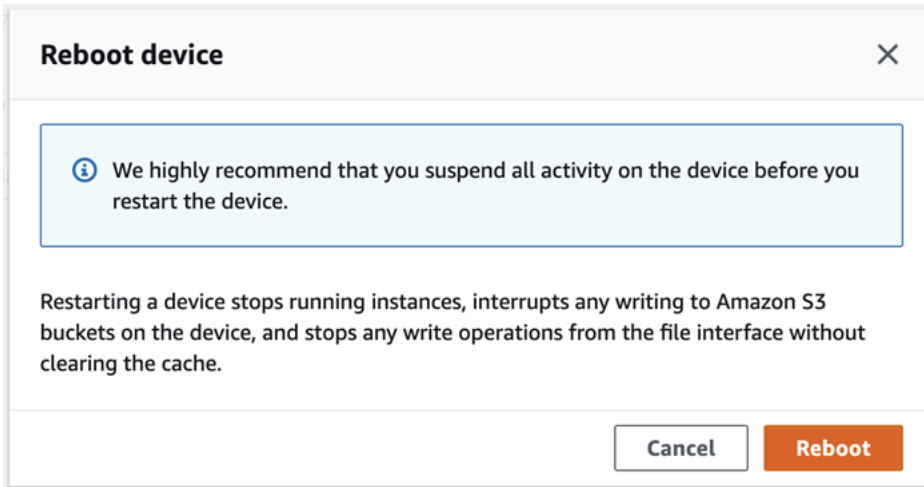
디바이스 재부팅 전에 디바이스의 모든 활동을 일시 중지하는 것이 좋습니다. 디바이스를 재부팅하면 인스턴스 실행이 중지되고 디바이스의 Amazon S3 버킷에 대한 모든 쓰기가 중단됩니다.

디바이스를 재부팅하려면

1. AWS OpsHub 대시보드의 Devices 아래에서 디바이스를 찾으십시오. 그런 다음 디바이스를 선택하여 디바이스 세부 정보 페이지를 엽니다.
2. 디바이스 전원 메뉴를 선택한 다음 재부팅을 선택합니다. 대화 상자가 나타납니다.



3. 대화 상자에서 재부팅을 선택합니다. 디바이스가 재부팅되기 시작합니다.



디바이스 종료하기

다음 단계에 따라 Snow AWS OpsHub 디바이스를 종료하세요.

⚠ Important

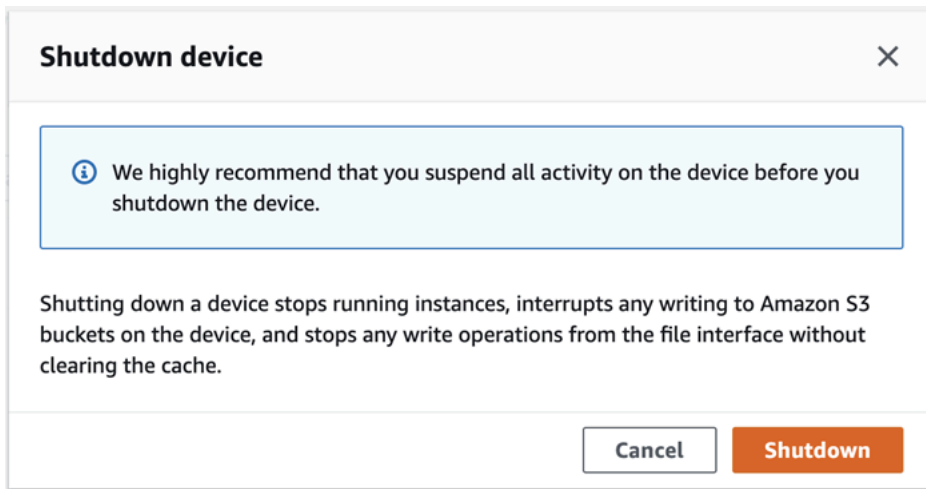
디바이스 종료 전에 디바이스의 모든 활동을 일시 중지하는 것이 좋습니다. 디바이스를 종료하면 인스턴스 실행이 중단되고 디바이스의 Amazon S3 버킷에 대한 모든 쓰기가 중단됩니다.

디바이스를 종료하려면

1. AWS OpsHub 대시보드의 Devices 아래에서 디바이스를 찾으십시오. 그런 다음 디바이스를 선택하여 디바이스 세부 정보 페이지를 엽니다.
2. 디바이스 전원 메뉴를 선택한 다음 종료를 선택합니다. 대화 상자가 나타납니다.



3. 대화 상자에서 종료를 선택합니다. 디바이스가 종료되기 시작합니다.

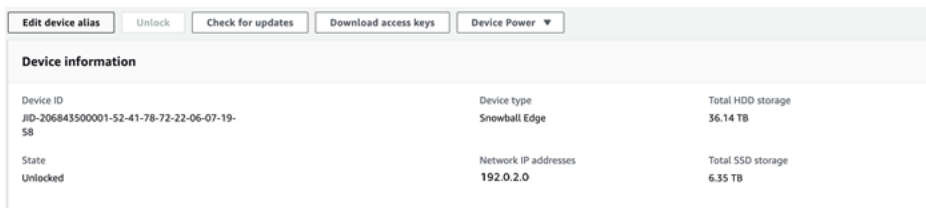


디바이스 별칭 편집

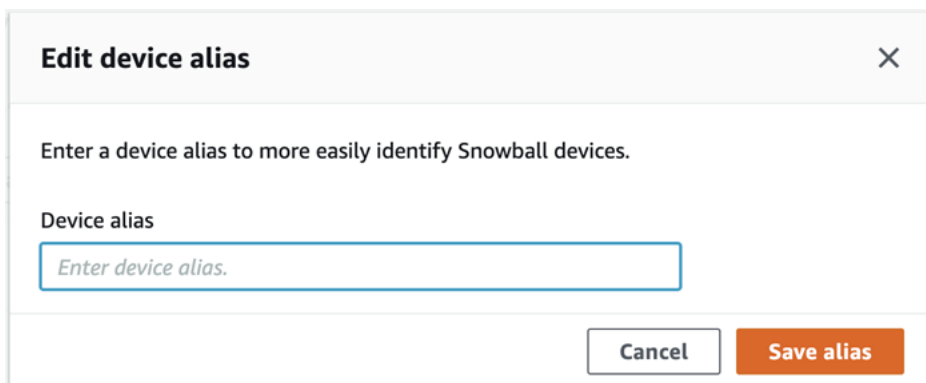
다음 단계를 사용하여 장치 별칭을 편집하십시오. AWS OpsHub

단말기의 별칭을 편집하려면

1. AWS OpsHub 대시보드의 장치에서 장치를 찾으십시오. 디바이스를 선택하여 디바이스 세부 정보 페이지를 엽니다.
2. Edit device alias(디바이스 별칭 편집) 탭을 선택합니다.



3. Device alias(디바이스 별칭)에 새 이름을 입력하고 Save alias(별칭 저장)를 선택합니다.



장치 및 AWS OpsHub 애플리케이션에 대한 업데이트 받기

디바이스의 업데이트를 확인하고 설치할 수 있습니다. 애플리케이션을 최신 버전으로 자동 AWS OpsHub 업데이트하도록 구성할 수도 있습니다.

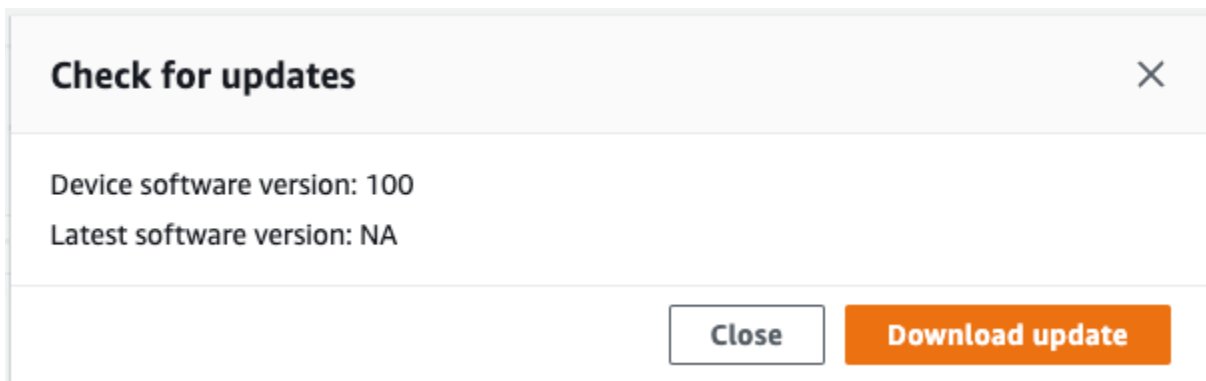
디바이스 업데이트

다음 단계에 따라 Snow AWS OpsHub 장치를 업데이트하십시오.

디바이스를 업데이트하려면

1. AWS OpsHub 대시보드의 디바이스에서 디바이스를 찾으세요. 디바이스를 선택하여 디바이스 세부 정보 페이지를 엽니다.
2. Check for updates(업데이트 확인) 탭을 선택합니다.

Check for updates(업데이트 확인) 페이지에는 디바이스의 현재 소프트웨어 버전과 최신 소프트웨어 버전(있는 경우)이 표시됩니다.



3. 업데이트가 있는 경우 업데이트 다운로드를 선택합니다. 그렇지 않으면 [Close]를 선택합니다.

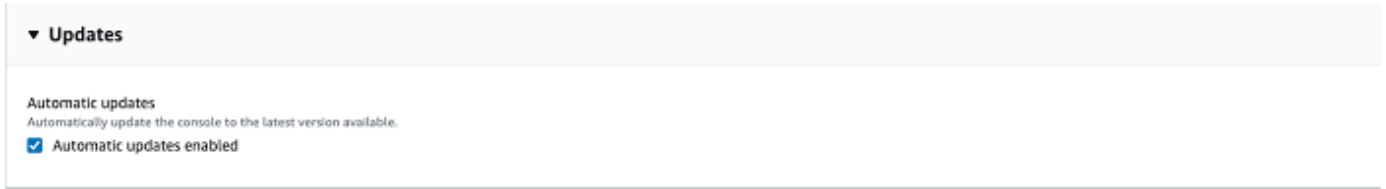
AWS OpsHub 애플리케이션 업데이트

AWS OpsHub 애플리케이션을 최신 버전으로 자동 업데이트합니다. 다음 단계에 따라 자동 업데이트가 활성화되어 있는지 확인합니다.

다음에 대해 자동 업데이트가 활성화되어 있는지 확인하려면 AWS OpsHub

1. AWS OpsHub 대시보드에서 환경설정을 선택합니다.
2. 업데이트 탭을 엽니다.

3. 자동 업데이트 활성화가 선택되어 있는지 확인합니다. 자동 업데이트는 기본적으로 활성화되어 있습니다.



자동 업데이트 활성화가 선택되지 않은 경우 AWS OpsHub 애플리케이션의 최신 버전을 가져올 수 없습니다.

프로필 관리

로컬 파일 시스템에 있는 자격 증명의 영구 스토리지에 대한 프로필을 생성할 수 있습니다. 를 사용하면 장치 IP 주소 AWS OpsHub, 잠금 해제 코드 및 매니페스트 파일을 사용하여 장치를 잠금 해제할 때마다 새 프로필을 만들 수 있습니다.

Snowball Edge 클라이언트를 사용하여 언제든지 프로필을 생성할 수도 있습니다. [Snowball Edge 클라이언트의 프로필 구성](#)을 참조하십시오.

프로필을 편집하거나 삭제하려면 텍스트 편집기에서 프로필 파일을 편집합니다.

Example 예시 `snowball-edge.config` 파일

이 예시에서는 세 개의 프로필(SnowDevice1profile, SnowDevice2profile, SnowDevice3profile)이 포함된 프로필 파일을 보여 줍니다.

```
{
  "version": 1,
  "profiles": [
    {
      "SnowDevice1profile": {
        "name": "SnowDevice1profile",
        "jobId": "JID12345678-136f-45b4-b5c2-847db8adc749",
        "unlockCode": "db223-12345-dbe46-44557-c7cc2",
        "manifestPath": "C:\\Users\\Administrator\\.aws\\ops-hub\\manifest\\JID12345678-136f-45b4-b5c2-847db8adc749_manifest-1670622989203.bin",
        "defaultEndpoint": "https://10.16.0.1",
        "isCluster": false,
        "deviceIps": []
      }
    }
  ]
}
```

```

    },
    "SnowDevice2profile":
    {
        "name": "SnowDevice2profile",
        "jobId": "JID12345678-fdb2-436a-a4ff-7c510dec1bae",
        "unlockCode": "b893b-54321-0f65c-6c5e1-7f748",
        "manifestPath": "C:\\Users\\Administrator\\.aws\\ops-hub\\manifest\\JID12345678-
fdb2-436a-a4ff-7c510dec1bae_manifest-1670623746908.bin",
        "defaultEndpoint": "https://10.16.0.2",
        "isCluster": false,
        "deviceIps": []
    },
    "SnowDevice3profile":
    {
        "name": "SnowDevice3profile",
        "jobId": "JID12345678-c384-4a5e-becd-ab5f38888463",
        "unlockCode": "64c89-13524-4d054-13d93-c1b80",
        "manifestPath": "C:\\Users\\Administrator\\.aws\\ops-hub\\manifest\\JID12345678-
c384-4a5e-becd-ab5f38888463_manifest-1670623999136.bin",
        "defaultEndpoint": "https://10.16.0.3",
        "isCluster": false,
        "deviceIps": []
    }
}

```

프로필을 생성하려면

1. 로컬에서 단말기를 잠금 해제하고 [디바이스 잠금 해제](#)의 지침에 따라 로그인합니다.
2. 프로필 이름을 지정하고 프로필 이름 저장을 선택합니다.

프로필을 편집하려면

1. 텍스트 편집기에서 home directory\.aws\snowball\config에 있는 snowball-edge.config를 엽니다.
2. 필요에 따라 파일을 편집합니다. 예를 들어 프로필에서 디바이스의 IP 주소를 변경하려면 defaultEndpoint 항목을 변경하십시오.
3. 파일을 저장하고 닫습니다.

프로필을 삭제하려면

1. 텍스트 편집기를 사용하여, `home directory\.aws\snowball\config`에서 `snowball-edge.config`을 엽니다.
2. 프로필 이름이 들어 있는 줄, 프로필 이름 뒤에 오는 중괄호 { }, 해당 괄호 안의 내용을 삭제합니다.
3. 파일을 저장하고 닫습니다.

디바이스의 NTP 시간 서버 설정

다음 단계에 따라 디바이스에서 시간을 동기화해야 하는 시간 서버를 확인하고 업데이트하십시오.

시간 소스를 확인하려면

1. AWS OpsHub 대시보드의 장치에서 장치를 찾으십시오. 디바이스를 선택하여 디바이스 세부 정보 페이지를 엽니다.
2. 디바이스가 시간을 동기화하고 있는 시간 소스 목록이 시간 소스 테이블에 표시됩니다.

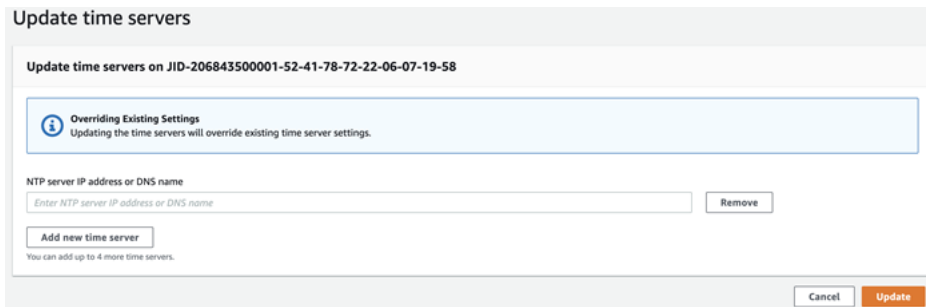
시간 소스 테이블에는 다음과 같은 네 개의 열이 있습니다.

- 주소: 시간 원본의 DNS 이름/IP 주소
- 상태: 디바이스와 해당 시간 소스 간의 현재 연결 상태로, 가능한 상태는 다음과 같이 5가지입니다.
 - 현재: 현재 시간 소스는 시간을 동기화하는 데 사용되고 있습니다.
 - 결합: 시간 소스가 현재 소스와 결합됩니다.
 - 제외: 결합 알고리즘에 의해 시간 소스가 제외됩니다.
 - 분실됨: 시간 소스와의 연결이 끊어졌습니다.
 - 사용할 수 없음: 결합 알고리즘이 잘못된 티커로 간주되거나 변동성이 너무 큰 잘못된 시간 소스입니다.
- 유형: NTP(Network Time Protocol) 소스는 서버 또는 피어일 수 있습니다. 서버는 사용자가 `update-time-server` 명령을 사용하여 설정할 수 있지만, 피어는 클러스터의 다른 Snowball Edge 디바이스로만 설정할 수 있으며 클러스터가 연결되면 자동으로 설정됩니다.

- 계층: 소스의 계층입니다. 계층 1은 로컬로 연결된 기존 클록이 있는 소스를 나타냅니다. 계층 1 소스와 동기화된 소스는 계층 2에 있습니다. 계층 2 소스와 동기화된 소스는 계층 3에 설정되는 방식으로 계속됩니다.

타임 서버를 업데이트하려면

1. AWS OpsHub 대시보드의 기기에서 기기를 찾으세요. 디바이스를 선택하여 디바이스 세부 정보 페이지를 엽니다.
2. 디바이스가 시간을 동기화하고 있는 시간 소스 목록이 시간 소스 테이블에 표시됩니다.
3. 시간 소스 테이블에서 시간 서버 업데이트를 선택합니다.
4. 디바이스에서 시간을 동기화할 시간 서버의 DNS 이름 또는 IP 주소를 입력하고 업데이트를 선택합니다.



Update time servers

Update time servers on JID-206843500001-52-41-78-72-22-06-07-19-58

ⓘ Overriding Existing Settings
Updating the time servers will override existing time server settings.

NTP server IP address or DNS name
Enter NTP server IP address or DNS name

Remove

Add new time server
You can add up to 4 more time servers.

Cancel Update

AWS Snowball Edge 클라이언트 사용

Snowball Edge 클라이언트는 로컬 서버에서 실행하여 AWS Snowcone 디바이스를 잠금 해제하고 자격 증명, 로그 및 상태 정보를 가져오는 독립 실행형 터미널 애플리케이션입니다. Snowball Edge 클라이언트를 사용하는 동안 `snowballEdge help` 명령을 실행하여 추가 지원 정보를 얻을 수 있습니다.

AWS Snowcone 디바이스에 데이터를 읽고 쓸 때는 NFS 인터페이스를 사용합니다. 또한 AWS OpsHub for Snow Family 애플리케이션을 사용하여 Snowcone 장치를 포함한 Snow Family 장치를 관리할 수 있습니다. 자세한 설명은 [장치 AWS OpsHub for Snow Family 관리에 사용](#) 섹션을 참조하세요.

Snowball Edge 클라이언트 다운로드 및 설치

Snowball Edge 클라이언트는 [AWS Snowball 리소스](#)에서 다운로드하여 설치할 수 있습니다. 해당 페이지에서 사용 중인 운영 체제에 대한 설치 패키지를 찾은 다음 지침에 따라 Snowball Edge 클라이언트를 설치할 수 있습니다. 워크스테이션의 터미널에서 Snowball Edge 클라이언트를 실행하려면 운영 체제에 따라 특정 경로를 사용해야 할 수 있습니다.

- Microsoft Windows - 클라이언트를 설치한 후 별도의 준비 없이 모든 디렉터리에서 클라이언트를 실행할 수 있습니다.
- Linux - Snowball Edge 클라이언트는 `~/snowball-client-linux-build_number/bin/` 디렉터리에서 실행해야 합니다. Snowball Edge 클라이언트는 64비트 Linux 배포판에서만 지원됩니다.
- MacOS - `install.sh` 스크립트는 폴더를 Snowball Edge 클라이언트 `.tar` 파일에서 `/usr/local/bin/snowball` 디렉터리로 복사합니다. 이 스크립트를 실행하면 `/usr/local/bin`이 `bash_profile`의 경로라면 어느 디렉터리에서든 Snowball Edge 클라이언트를 실행할 수 있습니다. `echo $PATH` 명령을 사용하여 경로를 확인할 수 있습니다.

Snowball Edge 클라이언트에 대한 명령

아래에서 사용 예와 샘플 출력을 비롯하여 Snowball Edge 클라이언트 명령에 대한 정보를 확인할 수 있습니다.

Note

AWS Snowcone 디바이스는 동일한 Snowball Edge CLI 명령을 사용하지만 클러스터링에 적용되는 명령은 지원하지 않습니다.

주제

- [Snowball Edge 클라이언트의 프로필 구성](#)
- [NFC 검증에 대한 QR 코드 가져오기](#)
- [장치 잠금 해제 AWS Snowcone](#)
- [Snowcone 업데이트](#)
- [자격 증명 받기](#)
- [Snowcone 디바이스에서 서비스 시작하기](#)
- [Snowcone 디바이스에서 서비스 중지하기](#)
- [데이터를 전송하기 위해 인증서 가져오기](#)
- [AWS Snowcone 로그](#)
- [디바이스 상태 가져오기](#)
- [서비스 상태 가져오기](#)
- [AWS DataSync AMI 런칭](#)
- [NFS 시작 및 액세스 제한](#)
- [NFS 실행 시 NFS 공유에 대한 액세스 제한](#)
- [Amazon S3 버킷의 내보내기 경로 가져오기](#)
- [로컬 AWS 운영자 디버깅 활성화](#)
- [로컬 운영자 디버깅 비활성화 AWS](#)
- [다이렉트 네트워크 인터페이스 생성](#)
- [다이렉트 네트워크 인터페이스에 대한 정보 가져오기](#)
- [다이렉트 네트워크 인터페이스 업데이트](#)
- [다이렉트 네트워크 인터페이스 삭제](#)
- [기능 상태 확인](#)
- [기능 상태 변경](#)
- [타임 서버 설정](#)
- [시간 소스 확인](#)

Snowball Edge 클라이언트의 프로필 구성

Snowball Edge 클라이언트에 대해 명령을 실행할 때마다 매니페스트 파일, 잠금 해제 코드 및 IP 주소를 입력합니다. 이 중 처음 두 개는 AWS Snow 패밀리 관리 콘솔 또는 작업 관리 API에서 가져올 수 있

습니다. 매니페스트 및 잠금 해제 코드 가져오기에 대한 자세한 내용은 [자격 증명 받기](#) 섹션을 참조하십시오.

snowballEdge configure 명령을 사용하여 엔드포인트, 29자 잠금 해제 코드 및 매니페스트 경로를 프로필로 저장할 수 있습니다. 구성 후 특정 작업에 대해 이러한 값을 수동으로 입력하지 않고도 다른 Snowball Edge 클라이언트 명령을 사용할 수 있습니다. Snowball Edge 클라이언트를 구성하면 해당 정보가 일반 텍스트 JSON 형식으로 *home directory*/.aws/snowball/config/snowball-.config에 저장됩니다.

엔드포인트는 https://가 추가된 IP 주소입니다. 장치 LCD 디스플레이에서 AWS Snowcone AWS Snowcone 장치의 IP 주소를 찾을 수 있습니다. AWS Snowcone 디바이스가 처음으로 네트워크에 연결되면 자동으로 DHCP IP 주소를 가져옵니다(DHCP 서버를 사용할 수 있는 경우). 다른 IP 주소를 사용하고 싶은 경우 LCD 디스플레이에서 변경할 수 있습니다. 자세한 설명은 [AWS Snowcone에서 AWS 서비스 사용](#) 섹션을 참조하십시오.

Important

구성 파일에 액세스할 수 있는 사람은 누구나 Snowcone 디바이스의 데이터에 액세스할 수 있습니다. 이 파일에 대한 로컬 액세스 제어를 관리하는 것은 관리자 책임 중 하나입니다.

사용량

이 명령을 '인라인' 또는 '프롬프트 시'라는 두 가지 방법으로 사용할 수 있습니다. 이 사용 예시는 프롬프트된 방법을 보여 줍니다.

```
snowballEdge configure
```

Example 출력 예

```
Configuration will be stored at home directory\.aws\snowball\config\snowball-.config
Snowcone Manifest Path: Path/to/manifest/file
Unlock Code: 29 character unlock code
Default Endpoint: https://192.0.2.0
```

한 번에 여러 작업이 있는 경우 프로필을 여러 개 가질 수 있습니다. 여러 AWS CLI 프로필에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 [명명된 프로필](#)을 참조하십시오.

NFC 검증에 대한 QR 코드 가져오기

이 명령을 사용하여 AWS Snowcone Verification 앱에 사용할 디바이스 고유의 QR 코드를 생성할 수 있습니다. 이 앱은 Apple App Store 또는 Google Play 스토어에서 다운로드할 수 있습니다. NFC 검증에 대한 자세한 내용은 [NFC 태그 검증](#) 섹션을 참조하십시오.

사용량

```
snowballEdge get-app-qr-code --output-file ~/downloads/snowball-qr-code.png
```

Example 출력 예

```
QR code is saved to ~/downloads/snowball-qr-code.png
```

장치 잠금 해제 AWS Snowcone

독립형 AWS Snowcone 장치의 잠금을 해제하려면 명령을 실행합니다. snowballEdge unlock-device 이 명령은 장치에 대한 액세스를 인증합니다. AWS Snowcone

이러한 잠금 해제 명령 중 하나를 실행하면 매니페스트 파일의 경로, 29자의 잠금 해제 코드, 독립형 디바이스의 IP 주소를 수동으로 입력할 수 있습니다. 이 프로세스는 지루할 수 있으므로 대신 Snowball Edge 클라이언트를 구성하는 것이 좋습니다. Snowball Edge 클라이언트를 이미 구성한 경우에는 매니페스트 경로, 잠금 해제 코드 또는 IP 주소 없이 명령 자체만 입력하면 됩니다.

Note

작업과 연결된 디바이스를 잠금 해제하려면 디바이스가 현장에 있고 전원 및 네트워크에 연결되어 켜진 상태여야 합니다. 또한 장치 전면의 LCD 디스플레이에 AWS Snowcone 장치를 사용할 준비가 되었음을 나타내야 합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge unlock-device
```

Example

Example 출력 잠금 해제

```
Your AWS Snowcone device is unlocking. You may determine the unlock state of your
device using the describe-device command. Your AWS Snowcone device will be available
for use when it is in the UNLOCKED state.
```

Snowcone 업데이트

다음 명령은 Snowcone 디바이스에 대한 업데이트를 다운로드 및 설치하는 데 사용할 수 있습니다. 이 명령을 사용하는 절차는 [Snowcone 업데이트](#) 섹션을 참조하십시오.

`snowballEdge check-for-updates` - 클라우드에서 사용 가능한 Snowball 소프트웨어 관련 버전 정보와 디바이스에 설치된 현재 버전을 반환합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge check-for-updates
```

Example 출력 예

```
Latest version: 102
Installed version: 101
```

`snowballEdge describe-device-software` - 디바이스에 대한 현재 소프트웨어 버전을 반환합니다. 추가로 업데이트가 다운로드 중인 경우 다운로드 상태 또한 표시됩니다. 소프트웨어 업데이트가 진행 중인 경우 업데이트의 버전 매니페스트와 설치 상태 또한 표시됩니다. 다음은 가능한 출력 목록입니다.

- `NA` - 현재 진행 중인 소프트웨어 업데이트가 없습니다.
- `Downloading` - 새 소프트웨어가 다운로드되는 중입니다.
- `Installing` - 새 소프트웨어가 설치되는 중입니다.
- `Requires Reboot` - 새 소프트웨어가 설치되었고, 디바이스를 재시작해야 합니다.

⚠ Warning

디바이스 재시작 전에 디바이스의 모든 활동을 일시 중지하는 것이 좋습니다. 디바이스를 재시작하면 인스턴스 실행이 중지되고 디바이스의 Amazon S3 버킷에 대한 모든 쓰기가 중단됩니다. 이는 모두 데이터 손실로 이어질 수 있습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge describe-device-software
```

Example 출력 예

```
Installed version: 101
Installing version: 102
Install State: Downloading
```

`snowballEdge download-updates` - Snowcone 디바이스에 대한 최신 소프트웨어 업데이트 다운로드를 시작합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge download-updates
```

Example 출력 예

```
Download started. Run describe-device-software API for additional information.
```

`snowballEdge install-updates` - 이미 다운로드한 Snowcone 디바이스에 대한 최신 소프트웨어 업데이트 설치를 시작합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge install-updates
```

Example 출력 예

```
Installation started.
```

`snowballEdge reboot-device` - 디바이스를 재부팅합니다.

Warning

디바이스 재시작 전에 디바이스의 모든 활동을 일시 중지하는 것이 좋습니다. 디바이스를 재시작하면 인스턴스 실행이 중지되고 디바이스의 Amazon S3 버킷에 대한 모든 쓰기가 중단됩니다. 이는 모두 데이터 손실로 이어질 수 있습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge reboot-device
```

Example 출력 예

```
Rebooting device now.
```

`snowballEdge configure-auto-update-strategies` - 자동 업데이트 방식을 구성합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge configure-auto-update-strategy --auto-check autoCheck [--auto-check-  
frequency  
autoCheckFreq] --auto-download autoDownload  
[--auto-download-frequency autoDownloadFreq]  
--auto-install autoInstall  
[--auto-install-frequency autoInstallFreq]  
--auto-reboot autoReboot [--endpoint  
endpoint]
```

Example 출력 예

```
Successfully configured auto update strategy. Run describe-auto-update-strategies for  
additional information.
```

`snowballEdge describe-auto-update-strategies` - 현재 구성된 자동 업데이트 방식을 반환합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge describe-auto-update-strategies
```

Example 출력 예

```
auto-update-strategy {[
auto-check:true,
auto-check-frequency: "0 0 * * FRI", // CRON Expression String, Every Friday at
midnight
auto-download:true,
auto-download-frequency: "0 0 * * SAT", // CRON Expression String, Every Saturday at
midnight
auto-install:true,
auto-install-frequency: "0 13 * * Sun", // CRON Expression String, Every Saturday at
midnight
auto-reboot: false;
]}
```

자격 증명 받기

snowballEdge list-access-keys 및 snowballEdge get-secret-access-key 명령을 사용하여 로컬 자격 증명을 가져올 수 있습니다. 또는 SDK를 사용할 때 이를 사용하여 요청을 인증합니다. AWS CLI AWS 이러한 자격 증명은 Snowcone에 대한 개별 작업에만 연결되고, 디바이스에서만 사용할 수 있습니다. 디바이스에는 AWS 클라우드에서의 AWS Identity and Access Management (IAM) 권한이 없습니다.

자세한 내용은 [AWS 자격 증명](#) 및 [Amazon EC2 엔드포인트 사용](#)을 참조하십시오.

Note

AWS CLI Snowball과 함께 사용하는 경우 CLI를 구성할 때 이러한 자격 증명을 사용해야 합니다. CLI의 자격 증명을 구성하는 방법에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 [빠른 구성](#)을 참조하십시오.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge list-access-keys
```


Example 출력 예

```
{
  "AccessKeyIds" : [ "AKIAIOSFODNN7EXAMPLE" ]
}
```

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge get-secret-access-key --access-key-id Access Key
```

Example 출력 예

```
[snowballEdge]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

Snowcone 디바이스에서 서비스 시작하기

Snowcone은 컴퓨팅 인스턴스, NFS 파일 인터페이스, Amazon EC2 등을 포함한 여러 서비스를 지원합니다. AWS DataSyncsnowballEdge `start-service` 명령으로 이러한 서비스를 시작할 수 있습니다. 각 서비스의 서비스 ID를 가져오려면 `snowballEdge list-services` 명령을 사용하면 됩니다.

이 명령을 실행하기 전에 가상 네트워크 인터페이스를 하나 생성하여 시작하려는 서비스에 바인딩합니다. 자세한 설명은 [가상 네트워크 인터페이스 생성](#) 섹션을 참조하세요.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge start-service --service-id service_id --virtual-network-interface-arns virtual-network-interface-arn
```

Example 출력 예

```
Starting the AWS service on your Snowball Edge . You can determine the status of the AWS service using the describe-service command.
```

Snowcone 디바이스에서 서비스 중지하기

Snowcone 디바이스에서 실행 중인 서비스를 중지하려면 `snowballEdge stop-service` 명령을 사용할 수 있습니다. Amazon EC2 서비스는 중지할 수 없습니다.

Warning

남은 버퍼 상태 데이터가 디바이스에 쓰여지기 전에 파일 인터페이스가 중지되면 데이터가 손실될 수 있습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge stop-service --service-id service_id
```

Example 출력 예

```
Stopping the AWS service on your Snowball . You can determine the status of the AWS service using the describe-service command.
```

데이터를 전송하기 위해 인증서 가져오기

Snowcone 디바이스로 데이터를 전송하려면 NFS 인터페이스 또는 를 사용하십시오. AWS DataSync 다른 IP 주소를 사용하는 Snowcone 디바이스를 잠금 해제하면, 새 인증서가 생성되고 이전 인증서는 해당 엔드포인트에서 더 이상 사용할 수 없게 됩니다. `get-certificate` 명령을 사용하면 업데이트된 새 인증서를 Snowcone 디바이스에서 다시 가져올 수 있습니다.

다음 명령을 사용하여 Snowcone 디바이스에서 이러한 인증서를 나열하고 다운로드할 수 있습니다.

- `list-certificates` - 사용할 수 있는 인증서의 ARN(Amazon 리소스 이름)을 나열합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge list-certificates
```

Example 출력 예

```
{
```

```
"Certificates" : [ {
  "CertificateArn" : "arn:aws:snowball-
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7",
  "SubjectAlternativeNames" : [ "192.0.2.0" ]
} ]
}
```

- `get-certificate` - 제공된 ARN을 기반으로 특정 인증서를 가져옵니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge get-certificate --certificate-arn arn:aws:snowball-
device:::certificate/78EXAMPLE516EXAMPLEf538EXAMPLEa7
```

Example 출력 예

```
-----BEGIN CERTIFICATE-----
Certificate
-----END CERTIFICATE-----
```

AWS Snowcone 로그

온프레미스 데이터 센터와 Snowcone 디바이스 간에 데이터를 전송하면 로그가 자동으로 생성됩니다. 디바이스에 데이터를 전송하는 동안 예기치 않은 오류가 발생할 경우 다음 명령을 사용하여 로그 사본을 로컬 서버에 저장할 수 있습니다.

다음은 로그와 관련된 세 가지 명령입니다.

- `list-logs` - 로그 목록을 JSON 형식으로 반환합니다. 이 목록은 로그 크기(바이트), 로그의 ARN, 로그의 서비스 ID, 로그 유형을 보고합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge list-logs
```

Example 출력 예

```
{
  "Logs" : [ {
```

```

    "LogArn" : "arn:aws:snowball-device::log/s3-storage-JIEXAMPLE2f-1234-4953-a7c4-
dfEXAMPLE709",
    "LogType" : "SUPPORT",
    "ServiceId" : "datasync",
    "EstimatedSizeBytes" : 53132614
  }, {
    "LogArn" : "arn:aws:snowball-device::log/fileinterface-JIDEXAMPLEf-1234-4953-
a7c4-dfEXAMPLE709",
    "LogType" : "CUSTOMER",
    "ServiceId" : "nfs",
    "EstimatedSizeBytes" : 4446
  }
}

```

- `get-log`— Snowcone 장치에서 지정된 경로의 서버로 특정 로그의 사본을 다운로드합니다. CUSTOMER로그는 다음 .zip 형식으로 저장되며 이 유형의 로그를 추출하여 내용을 볼 수 있습니다. SUPPORT로그는 암호화되며 AWS Support 엔지니어만 읽을 수 있습니다. 로그의 이름과 경로를 지정할 수 있습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge get-log --log-arn arn:aws:snowball-device::log/fileinterface-
JIDEXAMPLEf-1234-4953-a7c4-dfEXAMPLE709
```

Example 출력 예

```
Logs are being saved to download/path/snowball--logs-1515EXAMPLE88.bin
```

- `get-support-logs` - 모든 SUPPORT 유형 로그의 사본을 Snowcone 디바이스에서 지정된 경로의 서비스로 다운로드합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge get-support-logs
```

Example 출력 예

```
Logs are being saved to download/path/snowball--logs-1515716135711.bin
```

⚠ Important

CUSTOMER 로그에는 자체 데이터에 대한 민감한 정보가 포함되었을 수 있습니다. 민감할 수 있는 이 정보를 보호하려면 작업을 마친 후 이러한 로그를 삭제하는 것이 좋습니다.

디바이스 상태 가져오기

다음 Snowball Edge 클라이언트 명령을 사용하여 Snowcone 디바이스의 상태 및 일반 상태를 확인할 수 있습니다.

- describe-device

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge describe-device
```

Example 출력 예

```
{
  "DeviceId" : "JIDbEXAMPLE-7eed-1234-ABCD-7EXAMPLE123c",
  "UnlockStatus" : {
    "State" : "UNLOCKED"
  },
  "ActiveNetworkInterface" : {
    "IpAddress" : "192.168.1.2"
  },
  "PhysicalNetworkInterfaces" : [ {
    "PhysicalNetworkInterfaceId" : "s.ni-8bEXAMPLE5EXAMPLE",
    "PhysicalConnectorType" : "RJ45",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.168.1.13",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.168.1.1",
    "MacAddress" : "EX:AM:PL:E0:12:34"
  }, {
    "PhysicalNetworkInterfaceId" : "s.ni-84EXAMPLE3EXAMPLE",
    "PhysicalConnectorType" : "RJ45_2",
    "IpAddressAssignment" : "STATIC",
    "IpAddress" : "0.0.0.0",
    "Netmask" : "0.0.0.0",
```

```
"DefaultGateway" : "192.168.1.1",
"MacAddress" : "EX:AM:PL:E0:12:34"
}, {
  "PhysicalNetworkInterfaceId" : "s.ni-87EXAMPLE5EXAMPLE",
  "PhysicalConnectorType" : "WIFI",
  "IpAddressAssignment" : "STATIC",
  "IpAddress" : "0.0.0.0",
  "Netmask" : "0.0.0.0",
  "DefaultGateway" : "192.168.1.1",
  "MacAddress" : "EX:AM:PL:E0:12:34"
} ],
"DeviceCapacities" : [ {
  "Name" : "HDD Storage",
  "Unit" : "Byte",
  "Total" : 157242114048,
  "Used" : 81604378624,
  "Available" : 75637735424
}, {
  "Name" : "SSD Storage",
  "Unit" : "Byte",
  "Total" : 0,
  "Used" : 0,
  "Available" : 0
}, {
  "Name" : "vCPU",
  "Unit" : "Number",
  "Total" : 3,
  "Used" : 3,
  "Available" : 0
}, {
  "Name" : "Memory",
  "Unit" : "Byte",
  "Total" : 5368709120,
  "Used" : 5368709120,
  "Available" : 0
}, {
  "Name" : "GPU",
  "Unit" : "Number",
  "Total" : 0,
  "Used" : 0,
  "Available" : 0
} ],
"DeviceType" : "SNC1_HDD"
```

}

서비스 상태 가져오기

`describe-service` 명령을 사용하여 Snowcone 디바이스에서 실행되는 서비스의 상태 및 일반 상태를 확인할 수 있습니다. 먼저 `list-services` 명령을 실행하여 실행 중인 서비스를 볼 수 있습니다.

- `list-services`

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge list-services
```

Example 출력 예

```
{
  "ServiceIds" : [ "nfs", "datasync", "ec2" ]
}
```

- `describe-service`

이 명령은 서비스의 상태 값을 반환합니다. 반환되는 값에는 서비스 사용 시 발생하는 문제를 해결하는 데 도움이 될 수 있는 상태 정보도 포함되어 있습니다. 이러한 상태는 다음과 같습니다.

- ACTIVE - 서비스가 실행 중이며 사용 가능합니다.
- ACTIVATING - 서비스가 시작되는 중이지만 아직 사용할 수는 없습니다.
- DEACTIVATING - 서비스가 종료되는 중입니다.
- INACTIVE - 서비스가 실행되고 있지 않으므로 사용할 수 없습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge describe-service --service-id service-id
```

Example 출력 예

```
{
  "ServiceId" : "ec2",
  "Status" : {
    "State" : "ACTIVE"
```

```

    },
    "Storage" : {
      "TotalSpaceBytes" : 99608745492480,
      "FreeSpaceBytes" : 99608744468480
    },
    "Endpoints" : [ {
      "Protocol" : "http",
      "Port" : 8080,
      "Host" : "192.0.2.0"
    }, {
      "Protocol" : "https",
      "Port" : 8443,
      "Host" : "192.0.2.0",
      "CertificateAssociation" : {
        "CertificateArn" : "arn:aws:snowball-
device:::certificate/6d955EXAMPLEdb71798146EXAMPLE3f0"
      }
    } ]
  }
}

```

AWS DataSync AMI 런칭

AWS DataSync Snowcone에서 AMI를 실행합니다.

사용량(구성된 Snowball Edge 클라이언트)

AWS DataSync snc1.medium 인스턴스 유형으로 시작해야 합니다. 다른 인스턴스 유형으로 시작하면 DataSync 운영이 불안정해지고 데이터 손실이 발생할 수 있습니다. `describe-images` 명령을 사용하여 인스턴스를 시작할 이미지를 찾을 수 있습니다. 출력은 다음과 같습니다.

```

{
  "ImageId": "s.ami-0c046f119de4f752f",
  "Public": false,
  "State": "AVAILABLE",
  "BlockDeviceMappings": [
    {
      "DeviceName": "/dev/sda",
      "Ebs": {
        "DeleteOnTermination": true,
        "Iops": 0,
        "SnapshotId": "s.snap-0d7558ce444ab09bf",
        "VolumeSize": 20,

```



```

    "VolumeType": "sbp1"
  }
}
],
"Description": "AWS DataSync AMI for online data transfer",
"EnaSupport": false,
"Name": "scn-datasync-ami",
"RootDeviceName": "/dev/sda"
}

```

```
aws ec2 describe-instances --endpoint http://${snowcone_ip}:8008
```

Example 출력 예

```

{
  "Reservations": [
    {
      "Instances": [
        {
          "AmiLaunchIndex": 0,
          "ImageId": "s.image id",
          "InstanceId": "s.instance id",
          "InstanceType": "snc1.medium",
          "LaunchTime": "2020-03-06T18:58:36.609Z",
          "PrivateIpAddress": "ip address",
          "State": {
            "Code": 16,
            "Name": "running"
          },
          "BlockDeviceMappings": [
            {
              "DeviceName": "/dev/sda",
              "Ebs": {
                "AttachTime": "2020-03-06T19:14:21.336Z",
                "DeleteOnTermination": true,
                "Status": "attached",
                "VolumeId": "s.volume id"
              }
            }
          ],
          "EbsOptimized": false,
          "EnaSupport": false,
          "RootDeviceName": "/dev/sda",

```

```

        "SecurityGroups": [
            {
                "GroupName": "default",
                "GroupId": "s.security group id"
            }
        ],
        "SourceDestCheck": false,
        "CpuOptions": {
            "CoreCount": 2,
            "ThreadsPerCore": 1
        }
    },
    "ReservationId": "s.r-80c8ee6b041b29eb4"
},
]
}

```

인스턴스를 실행합니다.

```

aws ec2 run-instances --image-id s.ami id \
  --instance-type snc1.medium --endpoint
  http://${snowcone_ip}:8008

```

Example 출력 예

```

{
  "Instances": [
    {
      "AmiLaunchIndex": 0,
      "ImageId": "s.ami-0623310b494365cc5",
      "InstanceId": "s.i-80c8ee6b041b29eb4",
      "InstanceType": "snc1.medium",
      "State": {
        "Code": 0,
        "Name": "pending"
      },
      "EbsOptimized": false,
      "EnaSupport": false,
      "RootDeviceName": "/dev/sda",
      "SecurityGroups": [
        {
          "GroupName": "default",
          "GroupId": "s.sg-80c8ee6b041b29eb4"
        }
      ]
    }
  ]
}

```

```

    }
  ],
  "SourceDestCheck": false,
  "CpuOptions": {
    "CoreCount": 2,
    "ThreadsPerCore": 1
  }
}
],
"ReservationId": "s.r-80c8ee6b041b29eb4"
}

```

NFS 시작 및 액세스 제한

Important

Amazon Elastic Block Store(Amazon EBS)를 사용하려는 경우에는 NFS 서비스를 시작하면 안 됩니다. NFS를 처음 시작하면 모든 스토리지가 NFS에 할당됩니다. NFS 서비스가 중지된 경우에도 NFS 스토리지를 Amazon EBS에 재할당할 수 없습니다.

Note

NFS 서비스가 시작되기 전까지는 Snowcone 디바이스의 사용 가능한 스토리지 공간이 정확하지 않습니다.

디바이스에 의해 노출된 NFS 공유를 탑재하도록 허용된 IP 주소 범위에 대한 CIDR 블록을 제공할 수 있습니다. 예를 들어 10.0.0.0/16입니다. 허용된 CIDR 블록을 제공하지 않으면 모든 탑재 요청이 거부됩니다.

NFS를 통해 전송되는 데이터는 전송 중에 암호화되지 않습니다.

Snowcone에서는 CIDR 블록에서 허용된 호스트 이외의 NFS 공유에 대한 인증 또는 권한 부여 메커니즘을 제공하지 않습니다.

snowballEdge start-service 명령으로 NFS를 시작합니다. NFS 서비스의 서비스 ID를 가져오려면 snowballEdge list-services 명령을 사용하면 됩니다.

이 명령을 실행하기 전에 가상 네트워크 인터페이스를 하나 생성하여 시작하려는 서비스에 바인딩합니다. 자세한 설명은 [가상 네트워크 인터페이스 생성](#) 섹션을 참조하세요. Amazon S3 버킷의 파일 공유 및 데이터에 대한 액세스를 제한하고 현재 적용되는 제한 사항을 확인할 수 있습니다. NFS 서비스

를 시작할 때 파일 공유 및 S3 버킷에 액세스할 수 있는 허용된 호스트에 CIDR 블록을 할당하면 됩니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge start-service --service-id nfs --virtual-network-interface-arns
arn:aws:snowball-device:::interface/s.ni-12345fgh45678j --service-configuration
AllowedHosts=ip address-1/32,ip address-2/24
```

Example 출력 예

Starting the service on your Snowball Edge. You can determine the status of the service using the describe-service command.

NFS 실행 시 NFS 공유에 대한 액세스 제한

NFS를 시작한 후 Amazon S3 버킷의 파일 공유 및 데이터에 대한 액세스를 제한할 수 있습니다. 현재 어떤 제한이 적용되고 있는지 확인하고 각 버킷에 다른 액세스 제한을 부여할 수 있습니다. NFS 서비스를 시작할 때 파일 공유 및 S3 버킷에 액세스할 수 있는 호스트에 CIDR 블록을 할당하면 됩니다. 다음은 예시 명령입니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge start-service \
--service-id nfs \
--virtual-network-interface-arns virtual-network-interface-arn --service-
configuration AllowedHosts=ip-address-1/32,ip-address-1/24
```

현재 제한 사항을 보려면 describe-service 명령을 사용하십시오.

```
snowballEdge describe-service --service-id nfs
```

Amazon S3 버킷의 내보내기 경로 가져오기

Amazon S3 버킷의 내보내기 경로를 가져오기 위한 특정 Snowcone 명령은 없습니다. 내보내기 경로의 형식은 다음과 같습니다.

```
/buckets/bucket-name.
```

로컬 AWS 운영자 디버깅 활성화

- `enable-local-aws-operator-debugging`— SSH 포트 22를 열어 로컬 AWS 운영자 디버깅을 위해 디바이스를 활성화합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge enable-local-aws-operator-debugging
```

로컬 운영자 디버깅 비활성화 AWS

- `disable-local-aws-operator-debugging` – SSH 포트 22를 닫아 로컬 AWS 운영자 디버깅을 위해 디바이스를 비활성화합니다. 기본적으로 SSH 포트 22는 닫힙니다. Snowcone 장치를 끄거나 전원을 껐다가 다시 켜면 로컬 AWS 운영자 디버깅이 비활성화됩니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge disable-local-aws-operator-debugging
```

다이렉트 네트워크 인터페이스 생성

- `create-direct-network-interface` - [다이렉트 네트워크 인터페이스\(DNI\)를 생성합니다](#). 디바이스의 Amazon EC2 컴퓨팅 인스턴스와 함께 사용할 다이렉트 네트워크 인터페이스를 생성합니다. `describe-direct-network-interfaces` 명령을 사용하여 디바이스에서 사용할 수 있는 다이렉트 네트워크 인터페이스를 찾을 수 있습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
create-direct-network-interface [--endpoint endpoint] [--instance-id instanceId] [--mac macAddress]
                                [--manifest-file manifestFile] [--physical-network-
interface-id physicalNetworkInterfaceId]
                                [--profile profile] [--unlock-code unlockCode] [--vlan vlanId]
```

다이렉트 네트워크 인터페이스에 대한 정보 가져오기

- `describe-direct-network-interface` - 디바이스의 다이렉트 네트워크 인터페이스를 가져옵니다. 다이렉트 네트워크 인터페이스를 사용하여 디바이스의 Amazon EC2 컴퓨팅 인스턴스 및 서비스에 대한 네트워킹을 구성할 수 있습니다. `create-direct-network-interface` 명령을 사용하여 새 다이렉트 네트워크 인터페이스를 생성할 수 있습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
describe-direct-network-interfaces [--endpoint endpoint] [--manifest-file manifestFile]
[--profile profile] [--unlock-code unlockCode]
```

다이렉트 네트워크 인터페이스 업데이트

- `update-direct-network-interface` - 다이렉트 네트워크 인터페이스를 업데이트합니다. 이 명령을 사용하여 디바이스의 Amazon EC2 컴퓨팅 인스턴스와 함께 사용할 다이렉트 네트워크 인터페이스를 업데이트하십시오. `describe-direct-network-interfaces` 명령을 사용하여 디바이스에서 사용할 수 있는 다이렉트 네트워크 인터페이스를 찾을 수 있습니다. Amazon EC2 인스턴스에 연결된 네트워크 인터페이스를 수정하면 인터페이스가 먼저 분리됩니다.

사용량(구성된 Snowball Edge 클라이언트)

```
update-direct-network-interface [--direct-network-interface-arn directNetworkInterfaceArn]
[--endpoint endpoint]
[--mac macAddress]
[--manifest-file manifestFile] [--profile profile]
[--unlock-code unlockCode]
[--vlan vlanId] [--attach-instance-id instanceId | --detach]
```

다이렉트 네트워크 인터페이스 삭제

- `delete-direct-network-interface` - 더 이상 사용하지 않는 다이렉트 네트워크 인터페이스를 삭제합니다. Amazon EC2 컴퓨팅 인스턴스에 연결된 다이렉트 네트워크 인터페이스를 삭제하려면 먼저 인스턴스에서 다이렉트 네트워크 인터페이스를 분리해야 합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
delete-direct-network-interface [--direct-network-interface-arn directNetworkInterfaceArn] [--endpoint endpoint]
                                [--manifest-file manifestFile] [--profile profile] [--unlock-code unlockCode]
```

기능 상태 확인

Snowcone 장치 및 로컬 AWS 서비스를 원격으로 관리할 수 있는 기능을 포함하여 AWS Snow Device Management 장치에서 사용할 수 있는 기능의 상태를 나열하려면 명령을 사용하십시오. `describe-features`

`RemoteManagementState`는 Snow 디바이스 관리의 상태를 나타내며 다음 상태 중 하나를 반환합니다.

- `INSTALLED_ONLY` - 기능이 설치되었지만 활성화되지 않았습니다.
- `INSTALLED_AUTOSTART`— 기능이 활성화되고 전원이 AWS 리전 켜지면 장치가 연결을 시도합니다.
- `NOT_INSTALLED` - 디바이스가 기능을 지원하지 않거나 출시 전에 이미 현장에 있었습니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge describe-features
--manifest-file manifest.bin path
--unlock-code unlock-code
--endpoint https://device-local-ip:9091
```

출력 예

```
{
  "RemoteManagementState" : String
}
```

기능 상태 변경

AWS Snowcone 기기에서 사용할 수 있는 기능의 상태를 변경하려면 `set-features` 명령을 사용합니다. Snowcone 장치 및 로컬 AWS 서비스를 원격으로 관리할 수 있는 기능을 활성화하거나 비활성화

하려면 AWS Snow Device Management 파라미터를 사용하십시오. `--remote-management-state` 이 명령을 실행하기 전에 디바이스를 잠금 해제해야 합니다.

Snow 디바이스 관리를 다음 상태로 설정할 수 있습니다.

- `INSTALLED_ONLY` - 기능이 설치되었지만 활성화되지 않았습니다.
- `INSTALLED_AUTOSTART`— 이 기능이 활성화되고 전원이 AWS 리전 켜지면 장치가 해당 기능에 연결을 시도합니다.

Note

이 `NOT_INSTALLED` 상태는 Snow Device Management를 지원하지 않거나 출시 전에 이미 현장에 있었던 디바이스를 식별하기 위한 용도로만 존재합니다. 이미 배포된 디바이스에는 이 기능을 설치하거나 제거할 수 없습니다. Snow 디바이스 관리를 사용하려면 기능이 사전 설치된 새 디바이스를 주문해야 합니다.

사용량(구성된 Snowball Edge 클라이언트)

```
snowballEdge set-features
--remote-management-state INSTALLED_AUTOSTART
--manifest-file ./JID2bf11d5a-fict-414a-b5b1-3bf7e6a6e83d_manifest.bin
--unlock-code 73bb0-f8ke1-69a4a-f4288-4f88d
--endpoint https://10.0.0.25
```

출력 예

```
{
  "RemoteManagementState" : "INSTALLED_AUTOSTART"
}
```

타임 서버 설정

외부 NTP(Network Time Protocol) 서버를 설정할 수 있습니다. 디바이스가 잠긴 상태와 잠금 해제된 상태 모두에 있을 때 NTP CLI 명령을 사용할 수 있습니다. 매니페스트 및 잠금 해제 코드가 필요합니다. 이러한 설정은 `snowballEdge configure` 명령을 사용하거나 `--manifest-file` 및 `--`

unlock-code 옵션을 사용하여 설정할 수 있습니다. 참고로 snowballEdge CLI는 Edge와 AWS Snowcone Edge 모두에서 사용할 수 있습니다. AWS Snowcone

안전한 NTP 시간 서버를 제공하는 것은 사용자의 책임입니다. 디바이스가 연결되는 NTP 시간 서버를 설정하려면 update-time-servers CLI 명령을 사용합니다.

Note

이 update-time-servers 명령은 이전 NTP 시간 서버 설정을 재정의합니다.

사용량

```
snowballEdge update-time-servers time.google.com
```

Example 출력 예

```
Updating time servers now.
```

시간 소스 확인

디바이스가 현재 연결되어 있는 NTP 시간 소스를 확인하려면 describe-time-sources Snowball Edge CLI 명령을 사용하세요.

사용량

```
snowballEdge describe-time-sources
```

Example 출력 예

```
{
  "Sources" : [ {
    "Address" : "172.31.2.71",
    "State" : "LOST",
    "Type" : "PEER",
    "Stratum" : 10
  }, {
    "Address" : "172.31.3.203",
    "State" : "LOST",
```

```

    "Type" : "PEER",
    "Stratum" : 10
  }, {
    "Address" : "172.31.0.178",
    "State" : "LOST",
    "Type" : "PEER",
    "Stratum" : 10
  }, {
    "Address" : "172.31.3.178",
    "State" : "LOST",
    "Type" : "PEER",
    "Stratum" : 10
  }, {
    "Address" : "216.239.35.12",
    "State" : "CURRENT",
    "Type" : "SERVER",
    "Stratum" : 1
  } ]
}

```

이 `describe-time-sources` 명령은 시간 소스 상태 목록을 반환합니다. 각 시간 소스 상태에는 `Address`, `State`, `Type`, `Stratum` 필드가 포함됩니다. 이러한 필드의 의미는 다음과 같습니다.

- `Address` - 시간 소스의 DNS 이름/IP 주소입니다.
- `State` - 디바이스와 해당 시간 소스 간의 현재 연결 상태입니다. 다음과 같은 다섯 가지 상태가 있을 수 있습니다.
 - `CURRENT` - 시간 소스가 현재 시간을 동기화하는 데 사용되고 있습니다.
 - `COMBINED` - 시간 소스가 현재 소스와 결합됩니다.
 - `EXCLUDED` - 시간 소스가 결합 알고리즘에 의해 제외됩니다.
 - `LOST` - 시간 소스와의 연결이 끊어졌습니다.
 - `UNACCEPTABLE` - 결합 알고리즘이 잘못된 티커로 간주되거나 변동성이 너무 큰 잘못된 시간 소스입니다.
- `Type` - NTP 시간 소스는 서버 또는 피어일 수 있습니다. `update-time-servers` 명령으로 서버를 설정할 수 있습니다. 피어는 클러스터의 다른 Snowball Edge 디바이스로만 사용할 수 있으며 클러스터가 연결되면 자동으로 설정됩니다.
- `Stratum` - 이 필드에는 소스의 계층이 표시됩니다. 계층 1은 로컬로 연결된 기준 클럭이 있는 소스를 나타냅니다. 계층 1 소스와 동기화된 소스는 계층 2에 있습니다. 계층 2 소스와 동기화된 소스는 계층 3에 있는 방식입니다.

NTP 시간 소스는 서버 또는 피어일 수 있습니다. 서버는 사용자가 `update-time-servers` 명령으로 설정할 수 있지만, 피어는 클러스터의 다른 Snowball Edge 디바이스만 설정할 수 있습니다. 예시 출력에서는 클러스터가 5개인 Snowball Edge에서 `describe-time-sources`가 호출됩니다. 출력에는 피어 4개와 서버 1개가 포함됩니다. 피어의 계층은 10이고 서버의 계층은 1이므로 서버가 현재 시간 소스로 선택됩니다.

AWS Snow Device Management를 사용하여 디바이스 관리

AWS Snow Device Management를 통해 Snow Family 디바이스 및 로컬 AWS 서비스를 원격으로 관리할 수 있습니다. 모든 Snow Family 장치는 Snow 장치 관리를 지원하며 Snow Family 장치를 사용할 수 있는 AWS 리전 있는 대부분의 지역에서 새 장치에 설치됩니다.

Snow 디바이스 관리를 사용하면 다음 작업을 수행할 수 있습니다.

- 작업 생성
- 태스크 상태 확인
- 태스크 메타데이터 확인
- 태스크 취소
- 디바이스 정보 확인
- Amazon EC2 호환 인스턴스 상태 확인
- 명령 및 구문 나열
- 원격 관리 가능 디바이스 나열
- 디바이스 전반의 태스크 상태 나열
- 사용 가능한 리소스 나열
- 상태별로 태스크 나열
- 디바이스 또는 태스크 태그 나열
- 태그 적용
- 태그 제거

주제

- [Snow Family 디바이스를 주문할 때 Snow 디바이스 관리 상태 선택](#)
- [Snow 디바이스 관리 활성화](#)
- [Snow 디바이스 관리에 대한 권한을 IAM 역할에 추가](#)
- [Snow Device Management CLI 명령](#)

Snow Family 디바이스를 주문할 때 Snow 디바이스 관리 상태 선택

Snow 장치를 주문하기 위한 작업을 생성할 때 장치를 받았을 때 Snow Device Management의 상태 (설치되었지만 활성화되지 않았거나 설치 및 활성화되지 않음) 를 선택할 수 있습니다. 설치되었지만

활성화되지 않은 경우 사용하기 전에 AWS OpsHub 또는 Snowball Edge 클라이언트를 사용하여 활성화해야 합니다. Snow Device Management를 설치하여 활성화한 경우, 디바이스를 수신하여 로컬 네트워크에 연결한 후 Snow 디바이스 관리를 사용할 수 있습니다. 작업을 생성할 때 Snow 디바이스 관리 상태를 선택하여, Snowball Edge 클라이언트 AWS Snow 패밀리 관리 콘솔 AWS CLI, 또는 Snow 작업 관리 API를 통해 디바이스를 주문할 수 있습니다.

Snow 디바이스 관리 상태를 선택하려면 AWS Snow 패밀리 관리 콘솔

1. Snow 디바이스 관리를 설치하고 활성화하려면 AWS OpsHub 또는 Snowball 클라이언트를 사용하여 Snow 디바이스를 원격으로 관리하기를 선택합니다.
2. Snow 디바이스 관리를 설치하되 활성화하지 않도록 선택하려면, AWS OpsHub 또는 Snowball 클라이언트를 사용하여 Snow 디바이스 원격 관리를 선택하지 마십시오.

자세한 내용은 [3단계: 이 가이드의 기능 및 옵션 선택](#)을 참조하십시오.

AWS CLI, Snowball Edge 클라이언트 또는 Snow 작업 관리 API에서 Snow 디바이스 관리 상태를 선택하려면:

- `remote-management` 파라미터를 사용하여 Snow 디바이스 관리 상태를 지정합니다. 매개변수 `INSTALLED_ONLY` 값은 Snow 장치 관리가 설치되었지만 활성화되지 않았음을 의미합니다. 파라미터 `INSTALLED_AUTOSTART` 값은 Snow 디바이스 관리가 설치되고 활성화되었음을 의미합니다. 이 파라미터의 값을 지정하지 않는 경우 이 `INSTALLED_ONLY` 기본값입니다.

Example `create-job` 명령 `remote-management` 매개 변수의 구문

```
aws snowball create-job \
  --job-type IMPORT \
  --remote-management INSTALLED_AUTOSTART
  --device-configuration '{"SnowconeDeviceConfiguration": {"WirelessConnection":
{"IsWifiEnabled": false} } }' \
  --resources '{"S3Resources":[{"BucketArn":"arn:aws:s3:::bucket-name"}]}' \
  --description "Description here" \
  --address-id ADID00000000-0000-0000-0000-000000000000 \
  --kms-key-arn arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \
  --role-arn arn:aws:iam::000000000000:role/SnowconeImportGamma \
  --snowball-capacity-preference T8 \
  --shipping-option NEXT_DAY \
```

```
--snowball-type SNC1_HDD \  
--region us-west-2 \  

```

자세한 내용은 [API 참조의 Job Management AWS Snowball API 참조](#)를 참조하십시오.

Snow 디바이스 관리 활성화

다음 절차에 따라 Snowball Edge 클라이언트를 사용하여 Snow 디바이스 관리를 활성화하십시오.

이 절차를 사용하기 전에 다음을 수행하십시오.

- 최신 버전의 Snowball Edge 클라이언트를 다운로드하고 설치합니다. 자세한 내용은 [Snowball 클라이언트 다운로드 및 설치](#)를 참조하십시오.
- 매니페스트 파일을 다운로드하고 Snow Family 디바이스의 잠금 해제 코드를 받으십시오. 자세한 내용은 [자격 증명 및 도구 받기](#)를 참조하십시오.
- Snow Family 장치를 로컬 네트워크에 연결합니다. 자세한 내용은 섹션을 참조하십시오.
- Snow Family 디바이스의 잠금을 해제하세요. 자세한 내용은 [로컬 디바이스 잠금 해제를](#) 참조하십시오.

```
snowballEdge set-features /  
--remote-management-state INSTALLED_AUTOSTART /  
--manifest-file JID1717d8cc-2dc9-4e68-aa46-63a3ad7927d2_manifest.bin /  
--unlock-code 7c0e1-bab84-f7675-0a2b6-f8k33 /  
--endpoint https://192.0.2.0:9091
```

Snowball Edge 클라이언트는 명령이 성공하면 다음을 반환합니다.

```
{  
  "RemoteManagementState" : "INSTALLED_AUTOSTART"  
}
```

Snow 디바이스 관리에 대한 권한을 IAM 역할에 추가

디바이스를 주문하는 AWS 계정에서 AWS Identity and Access Management(IAM) 역할을 생성하고 역할에 다음 정책을 추가합니다. 그런 다음 Snow Device Management로 디바이스를 원격으로 관리할 수 있도록 로그인할 IAM 사용자에게 역할을 할당합니다. 자세한 내용은 [IAM 역할 생성](#) 및 [AWS 계정에서 IAM 사용자 생성](#)을 참조하세요.

정책

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "snow-device-management:ListDevices",
        "snow-device-management:DescribeDevice",
        "snow-device-management:DescribeDeviceEc2Instances",
        "snow-device-management:ListDeviceResources",
        "snow-device-management:CreateTask",
        "snow-device-management:ListTasks",
        "snow-device-management:DescribeTask",
        "snow-device-management:CancelTask",
        "snow-device-management:DescribeExecution",
        "snow-device-management:ListExecutions",
        "snow-device-management:ListTagsForResource",
        "snow-device-management:TagResource",
        "snow-device-management:UntagResource"
      ],
      "Resource": "*"
    }
  ]
}
```

Snow Device Management CLI 명령

이 섹션에서는 Snow Device Management를 통해 Snow Family 디바이스를 원격으로 관리하는 데 사용할 수 있는 AWS CLI 명령을 설명합니다. AWS OpsHub for Snow Family를 사용하여 일부 원격 관리 작업을 수행할 수도 있습니다. 자세한 내용은 장치의 [AWS서비스 관리 장치의 AWS](#) 참조하십시오.

Note

디바이스를 관리하기 전에는 디바이스가 켜져 있고 네트워크에 연결되어 있으며 프로비저닝된 AWS 리전에 연결할 수 있는지 확인하세요.

주제

- [작업 생성](#)
- [태스크 상태 확인](#)
- [디바이스 정보 확인](#)
- [Amazon EC2 호환 인스턴스 상태 확인](#)
- [태스크 메타데이터 확인](#)
- [태스크 취소](#)
- [명령 및 구문 나열](#)
- [원격 관리 가능 디바이스 나열](#)
- [디바이스 전반의 태스크 상태 나열](#)
- [사용 가능한 리소스 나열](#)
- [디바이스 또는 태스크 태그 나열](#)
- [상태별로 태스크 나열](#)
- [태그 적용](#)
- [태그 제거](#)

작업 생성

하나 이상의 대상 디바이스에 잠금 해제 또는 재부팅과 같은 작업을 수행하도록 지시하려면 `create-task`를 사용합니다. `--targets` 파라미터와 함께 관리 대상 디바이스 ID 목록을 제공하여 대상 디바이스를 지정하고, `--command` 파라미터로 수행할 작업을 지정합니다. 디바이스에서 한 번에 하나의 명령을 실행할 수 있습니다.

지원되는 명령:

- `unlock`(인수 없음)
- `reboot`(인수 없음)

대상 디바이스에서 실행할 작업을 생성하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management create-task
--targets smd-fictbgr3rbcjeqa5
--command reboot={}
```

예외

```
ValidationException
ResourceNotFoundException
InternalServerError
ThrottlingException
AccessDeniedException
ServiceQuotaExceededException
```

출력

```
{
  "taskId": "st-ficthmqoc2pht111",
  "taskArn": "arn:aws:snow-device-management:us-west-2:000000000000:task/st-
cjkwhmqoc2pht111"
}
```

태스크 상태 확인

하나 이상의 대상 디바이스에서 실행 중인 원격 태스크의 상태를 확인하려면 `describe-execution` 명령을 사용합니다.

태스크는 다음 중 하나의 상태를 가질 수 있습니다.

- QUEUED
- IN_PROGRESS

- CANCELED
- FAILED
- COMPLETED
- REJECTED
- TIMED_OUT

태스크 상태를 확인하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management describe-execution \
--taskId st-ficthmqoc2phtlef \
--managed-device-id smd-fictqic6gcldf111
```

출력

```
{
  "executionId": "1",
  "lastUpdatedAt": "2021-07-22T15:29:44.110000+00:00",
  "managedDeviceId": "smd-fictqic6gcldf111",
  "startedAt": "2021-07-22T15:28:53.947000+00:00",
  "state": "SUCCEEDED",
  "taskId": "st-ficthmqoc2pht111"
}
```

디바이스 정보 확인

디바이스 유형, 소프트웨어 버전, IP 주소, 잠금 상태 등 디바이스별 정보를 확인하려면 `describe-device` 명령을 사용합니다. 출력에는 다음이 포함됩니다.

- `lastReachedOutAt` - 디바이스가 AWS 클라우드에 마지막으로 연락한 시점입니다. 디바이스가 온 라인 상태임을 나타냅니다.
- `lastUpdatedAt` - 디바이스에서 데이터가 마지막으로 업데이트된 시점. 디바이스 캐시가 새로 고쳐진 시간을 나타냅니다.

다음 명령을 사용하여 디바이스를 확인합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management describe-device \  
--managed-device-id smd-fictqic6gcldf111
```

예외

```
ValidationException  
ResourceNotFoundException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

출력

```
{  
  "associatedWithJob": "JID2bf11d5a-ea1e-414a-b5b1-3bf7e6a6e111",  
  "deviceCapacities": [  
    {  
      "available": 158892032000,  
      "name": "HDD Storage",  
      "total": 158892032000,  
      "unit": "Byte",  
      "used": 0  
    },  
    {  
      "available": 0,  
      "name": "SSD Storage",  
      "total": 0,  
      "unit": "Byte",  
      "used": 0  
    },  
    {  
      "available": 3,  
      "name": "vCPU",
```

```
        "total": 3,
        "unit": "Number",
        "used": 0
    },
    {
        "available": 5368709120,
        "name": "Memory",
        "total": 5368709120,
        "unit": "Byte",
        "used": 0
    },
    {
        "available": 0,
        "name": "GPU",
        "total": 0,
        "unit": "Number",
        "used": 0
    }
],
"deviceState": "UNLOCKED",
"deviceType": "SNC1_HDD",
"lastReachedOutAt": "2021-07-23T21:21:56.120000+00:00",
"lastUpdatedAt": "2021-07-23T21:21:56.120000+00:00",
"managedDeviceId": "smd-fictqic6gcldf111",
"managedDeviceArn": "arn:aws:snow-device-management:us-west-2:000000000000:managed-device/smd-fictqic6gcldf111"
"physicalNetworkInterfaces": [
    {
        "defaultGateway": "10.0.0.1",
        "ipAddress": "10.0.0.2",
        "ipAddressAssignment": "DHCP",
        "macAddress": "ab:cd:ef:12:34:56",
        "netmask": "255.255.252.0",
        "physicalConnectorType": "RJ45",
        "physicalNetworkInterfaceId": "s.ni-530f866d526d4b111"
    },
    {
        "defaultGateway": "10.0.0.1",
        "ipAddress": "0.0.0.0",
        "ipAddressAssignment": "STATIC",
        "macAddress": "ab:cd:ef:12:34:57",
        "netmask": "0.0.0.0",
        "physicalConnectorType": "RJ45",
        "physicalNetworkInterfaceId": "s.ni-8abc787f0a6750111"
    }
]
```

```

    }
  ],
  "software": {
    "installState": "NA",
    "installedVersion": "122",
    "installingVersion": "NA"
  },
  "tags": {
    "Project": "PrototypeA"
  }
}

```

Amazon EC2 호환 인스턴스 상태 확인

Amazon EC2 인스턴스의 현재 상태를 확인하려면 `describe-ec2-instances` 명령을 사용합니다. 출력은 `describe-device` 명령의 출력과 비슷하지만, 결과는 AWS 클라우드의 디바이스 캐시에서 가져온 것이며 사용 가능한 필드의 하위 집합을 포함합니다.

Amazon EC2 호환 인스턴스의 현재 상태를 확인하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```

aws snow-device-management describe-device-ec2-instances \
--managed-device-id smd-fictbgr3rbcje111 \
--instance-ids s.i-84fa8a27d3e15e111

```

예외

```

ValidationException
ResourceNotFoundException
InternalServerError
ThrottlingException
AccessDeniedException

```

출력

```
{
  "instances": [
    {
      "instance": {
        "amiLaunchIndex": 0,
        "blockDeviceMappings": [
          {
            "deviceName": "/dev/sda",
            "ebs": {
              "attachTime": "2021-07-23T15:25:38.719000-07:00",
              "deleteOnTermination": true,
              "status": "ATTACHED",
              "volumeId": "s.vol-84fa8a27d3e15e111"
            }
          }
        ],
        "cpuOptions": {
          "coreCount": 1,
          "threadsPerCore": 1
        },
        "createdAt": "2021-07-23T15:23:22.858000-07:00",
        "imageId": "s.ami-03f976c3cadaa6111",
        "instanceId": "s.i-84fa8a27d3e15e111",
        "state": {
          "name": "RUNNING"
        },
        "instanceType": "snc1.micro",
        "privateIpAddress": "34.223.14.193",
        "publicIpAddress": "10.111.60.160",
        "rootDeviceName": "/dev/sda",
        "securityGroups": [
          {
            "groupId": "s.sg-890b6b4008bdb3111",
            "groupName": "default"
          }
        ],
        "updatedAt": "2021-07-23T15:29:42.163000-07:00"
      },
      "lastUpdatedAt": "2021-07-23T15:29:58.071000-07:00"
    }
  ]
}
```

```
}
```

태스크 메타데이터 확인

디바이스에서 특정 태스크의 메타데이터를 확인하려면 `describe-task` 명령을 사용합니다. 태스크의 메타데이터에는 다음 항목이 포함됩니다.

- 대상 디바이스
- 태스크 상태
- 태스크 생성 시점
- 디바이스에서 데이터가 마지막으로 업데이트된 시점
- 태스크 완료 시점
- 태스크가 생성될 때 제공된 설명(있는 경우)

태스크 메타데이터를 확인하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management describe-task \  
--task-id st-ficthmqoc2pht111
```

예외

```
ValidationException  
ResourceNotFoundException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

출력

```
{
  "completedAt": "2021-07-22T15:29:46.758000+00:00",
  "createdAt": "2021-07-22T15:28:42.613000+00:00",
  "lastUpdatedAt": "2021-07-22T15:29:46.758000+00:00",
  "state": "COMPLETED",
  "tags": {},
  "targets": [
    "smd-fictbgr3rbcje111"
  ],
  "taskId": "st-ficthmqoc2pht111",
  "taskArn": "arn:aws:snow-device-management:us-west-2:000000000000:task/st-ficthmqoc2pht111"
}
```

태스크 취소

특정 태스크의 취소 요청을 보내려면 `cancel-task` 명령을 사용합니다. 아직 실행되지 않은 QUEUED 상태의 태스크만 취소할 수 있습니다. 이미 실행 중인 태스크는 취소할 수 없습니다.

Note

취소하려는 태스크가 `cancel-task` 명령으로 태스크 상태를 변경하기 전에 대기열에서 처리 되면 계속 실행될 수 있습니다.

태스크를 취소하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management cancel-task \
--task-id st-ficthmqoc2pht111
```

예외

```
ValidationException
ResourceNotFoundException
```



```
InternalServerException
ThrottlingException
AccessDeniedException
```

출력

```
{
  "taskId": "st-ficthmqoc2pht111"
}
```

명령 및 구문 나열

Snow Device Management API에 지원되는 모든 명령 목록을 반환하려면 `help` 명령을 사용합니다. 또한 `help` 명령을 사용하여 해당 명령에 대한 세부 정보와 구문을 반환할 수 있습니다.

지원되는 명령을 모두 나열하려면 다음 명령을 사용합니다.

명령

```
aws snow-device-management help
```

명령의 구체적인 정보 및 구문을 반환하려면 다음 명령을 사용합니다. *command*을 관심 있는 명령의 이름으로 바꿉니다.

명령

```
aws snow-device-management command help
```

원격 관리 가능 디바이스 나열

명령이 실행된 AWS 리전에서 Snow Device Management가 활성화된 계정의 모든 디바이스 목록을 반환하려면 `list-devices` 명령을 사용합니다. `--max-results` 및 `--next-token`은 선택 사항입니다. 자세한 내용은 'AWS Command Line Interface 사용 설명서'의 [AWS CLI 페이지 매김 옵션 사용](#) 섹션을 참조하세요.

원격 관리 가능 디바이스의 목록을 보려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management list-devices \
--max-results 10
```

예외

```
ValidationException
InternalServerError
ThrottlingException
AccessDeniedException
```

출력

```
{
  "devices": [
    {
      "associatedWithJob": "ID2bf11d5a-ea1e-414a-b5b1-3bf7e6a6e111",
      "managedDeviceId": "smd-fictbgr3rbcjeqa5",
      "managedDeviceArn": "arn:aws:snow-device-management:us-west-2:000000000000:managed-device/smd-fictbgr3rbcje111"
      "tags": {}
    }
  ]
}
```

디바이스 전반의 태스크 상태 나열

하나 이상의 대상 디바이스에 대한 태스크 상태를 반환하려면 `list-executions` 명령을 사용합니다. 반환 목록을 필터링하여 현재 하나의 특정 상태에 있는 작업을 표시하려면 `--state` 파라미터를 사용합니다. `--max-results` 및 `--next-token`은 선택 사항입니다. 자세한 내용은 'AWS Command Line Interface 사용 설명서'의 [AWS CLI 페이지 매김 옵션 사용](#) 섹션을 참조하세요.

태스크는 다음 중 하나의 상태를 가질 수 있습니다.

- QUEUED
- IN_PROGRESS
- CANCELED
- FAILED
- COMPLETED
- REJECTED
- TIMED_OUT

여러 디바이스의 태스크 상태를 나열하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management list-executions \  
--taskId st-ficthmqoc2phtlef \  
--state SUCCEDED \  
--max-results 10
```

예외

```
ValidationException  
InternalServerError  
ThrottlingException  
AccessDeniedException
```

출력

```
{  
  "executions": [  
    {  
      "executionId": "1",
```

```

        "managedDeviceId": "smd-fictbgr3rbcje111",
        "state": "SUCCEEDED",
        "taskId": "st-ficthmqoc2pht111"
    }
]
}

```

사용 가능한 리소스 나열

디바이스에 사용할 수 있는 AWS 리소스 목록을 반환하려면 `list-device-resources` 명령을 사용합니다. 특정 유형의 리소스별로 목록을 필터링하려면 `--type` 파라미터를 사용합니다. 현재 지원되는 리소스 유형은 Amazon EC2 호환 인스턴스뿐입니다. `--max-results` 및 `--next-token`은 선택 사항입니다. 자세한 내용은 'AWS Command Line Interface 사용 설명서'의 [AWS CLI 페이지 매김 옵션 사용](#) 섹션을 참조하세요.

디바이스의 사용 가능한 리소스를 나열하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```

aws snow-device-management list-device-resources \
--managed-device-id smd-fictbgr3rbcje111 \
--type AWS::EC2::Instance
--next-
token YAQGPwAT9L3wVKaGYjt4yS34MiQLWvzcShe9oIeDJr05AT4rXSprqcqQhhBEYRfcerAp0YYbJmRT=
--max-results 10

```

예외

```

ValidationException
InternalServerError
ThrottlingException
AccessDeniedException

```

출력

```
{
  "resources": [
    {
      "id": "s.i-84fa8a27d3e15e111",
      "resourceType": "AWS::EC2::Instance"
    }
  ]
}
```

디바이스 또는 태스크 태그 나열

관리 대상 디바이스 또는 태스크의 태그 목록을 반환하려면 `list-tags-for-resource` 명령을 사용합니다.

디바이스 태그를 나열하려면 다음 명령을 사용합니다. Amazon 리소스 이름(ARN)을 디바이스의 ARN으로 바꿉니다.

명령

```
aws snow-device-management list-tags-for-resource
--resource-arn arn:aws:snow-device-management:us-west-2:123456789012:managed-device/smd-fictbgr3rbcjeqa5
```

예외

```
AccessDeniedException
InternalServerError
ResourceNotFoundException
ThrottlingException
```

출력

```
{
  "tags": {
    "Project": "PrototypeA"
  }
}
```

상태별로 태스크 나열

`list-tasks` 명령을 사용하면 명령이 실행되는 AWS 리전의 디바이스에서 태스크 목록을 반환할 수 있습니다. `IN_PROGRESS`, `COMPLETED` 또는 `CANCELED` 상태별로 결과를 필터링하려면 `--state` 파라미터를 사용합니다. `--max-results` 및 `--next-token`은 선택 사항입니다. 자세한 내용은 'AWS Command Line Interface 사용 설명서'의 [AWS CLI 페이지 매김 옵션 사용](#) 섹션을 참조하세요.

상태별로 태스크를 나열하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management list-tasks \
--state IN_PROGRESS \
--next-token K8VAMqKiP2Cf4xGkmH8GMyZrg0F8FUb+d10KTP9+P4pUb+8PhW+6MiXh4= \
--max-results 10
```

예외

```
ValidationException
InternalServerError
ThrottlingException
AccessDeniedException
```

출력

```
{
  "tasks": [
    {
      "state": "IN_PROGRESS",
      "tags": {},
      "taskId": "st-ficthmqoc2phtlef",
      "taskArn": "arn:aws:snow-device-management:us-west-2:000000000000:task/st-ficthmqoc2phtlef"
    }
  ]
}
```

태그 적용

디바이스 또는 디바이스상의 태스크에 태그를 추가하거나 바꾸려면 `tag-resource` 명령을 사용합니다. `--tags` 파라미터는 쉼표로 구분된 Key=Value 페어 목록을 허용합니다.

디바이스 태그를 적용하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management tag-resource \
--resource-arn arn:aws:snow-device-management:us-west-2:123456789012:managed-device/
smd-fictbgr3rbcjeqa5 \
--tags Project=PrototypeA
```

예외

```
AccessDeniedException
InternalServerError
ResourceNotFoundException
ThrottlingException
```

태그 제거

디바이스 또는 디바이스상의 태스크에서 태그를 제거하려면 `untag-resources` 명령을 사용합니다.

디바이스에서 태그를 제거하려면 다음 명령을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

명령

```
aws snow-device-management untag-resources \
--resource-arn arn:aws:snow-device-management:us-west-2:123456789012:managed-device/
smd-fictbgr3rbcjeqa5 \
--tag-keys Project
```

예외

AccessDeniedException
InternalServerError
ResourceNotFoundException
ThrottlingException

AWS Snowcone에서 AWS 서비스 사용

다음은 AWS Snowcone 디바이스의 개요를 기술한 것입니다. AWS Snowcone는 물리적으로 견고한 디바이스로서, 로컬 스토리지와 컴퓨팅 또는 온프레미스 서버와 Amazon Simple Storage Service(S3) 간 데이터 전송에 사용하는 AWS Key Management Service(AWS KMS)로 보호됩니다.

AWS Snowcone 디바이스 잠금 해제에 대한 자세한 내용은 [AWS Snowball Edge 클라이언트 사용](#) 섹션을 참조하세요.

디바이스가 처음 도착하면 손상되거나 명백히 변경된 부분이 있는지 검사합니다.

Warning

디바이스에서 의심이 가는 부분이 있으면 내부 네트워크에 연결하지 마세요. 그 대신에 [AWS Support](#)에 연락하면 새 디바이스를 받을 수 있습니다.

디바이스가 배송되고 전원이 켜지면 사용할 준비가 된 것입니다.

주제

- [Amazon EC2 호환 컴퓨팅 인스턴스 사용](#)
- [AWS DataSync를 사용하여 파일 전송](#)
- [NFS 인터페이스 관리](#)
- [AWS IoT Greengrass를 사용하여 Amazon EC2 호환 인스턴스에서 사전 설치된 소프트웨어 실행](#)
- [AWS Snowcone 디바이스에서 AWS 서비스를 사용하는 데 필요한 포트](#)

Amazon EC2 호환 컴퓨팅 인스턴스 사용

이 주제에서는 Amazon Elastic Compute Cloud(Amazon EC2) 컴퓨팅 인스턴스를 AWS Snowcone 디바이스에서 사용하는 방법에 대한 개요를 찾을 수 있습니다. 이 주제에는 개념 정보, 절차, 예시가 포함됩니다.

Note

아시아 태평양 (뭄바이)에서는 이러한 기능이 지원되지 않습니다. AWS 리전

개요

지원되는 EC2 호환 인스턴스 유형을 사용하여 Snowcone에서 호스팅되는 Amazon EC2 호환 컴퓨팅 인스턴스를 실행할 수 있습니다. 이에 상응하는 클라우드 기반 인스턴스와 마찬가지로 이들 인스턴스 역시 시작하려면 Amazon Machine Image(AMI)가 필요합니다. Snowcone 작업을 생성하기 전에 AMI를 클라우드에서 인스턴스의 기본 이미지로 선택할 수 있습니다. 지원되는 인스턴스 유형에 대한 자세한 내용은 [Snowcone에서 Amazon EC2 사용](#) 섹션을 참조하십시오.

작업 유형이 로컬 컴퓨팅인 경우 총 8TiB 로컬 EBS 볼륨을 생성하여 Amazon EC2 호환 인스턴스에 연결할 수 있습니다. 이렇게 하면 로컬 EC2 호환 인스턴스가 루트 볼륨만 사용할 때보다 더 많은 로컬 용량에 액세스할 수 있습니다. 이는 로컬 스토리지만 가능하므로 Amazon S3로 가져올 수 없으므로 디바이스를 반환하면 EBS 볼륨에 AWS 기록된 데이터가 손실됩니다.

Note

NFS 서버는 컴퓨팅 작업에 사용할 수 없습니다. 에서 데이터를 가져오거나 에서 데이터를 보내야 하는 경우 AWS 클라우드, 주문할 때 로컬 컴퓨팅 작업 유형을 선택하지 마십시오.

Snowcone에서 컴퓨팅 인스턴스를 사용하려면 Snow Family 디바이스를 주문하는 작업을 생성하고 AMI를 지정하십시오. SDK를 사용하거나 SDK [AWS Snow 패밀리 관리 콘솔](#) 중 하나를 사용하여 이 작업을 수행할 수 있습니다. AWS CLI AWS 일반적으로, 인스턴스를 사용하기 위해 작업을 생성하기 전에 수행해야 하는 정리 작업 사전 조건이 있습니다.

디바이스가 도착한 후 AMI 및 인스턴스 관리를 시작할 수 있습니다. Amazon EC2 호환 엔드포인트를 통해 Snowcone에서 컴퓨팅 인스턴스를 관리할 수 있습니다. 이 유형의 엔드포인트는 SDK에 대한 많은 Amazon EC2 CLI 명령 및 작업을 지원합니다. AWS AWS OpsHub for Snow Family 도구를 사용하여 AMI, 컴퓨팅 인스턴스 및 서비스를 관리합니다. AWS 자세한 설명은 [장치 AWS OpsHub for Snow Family 관리에 사용](#) 섹션을 참조하세요.

기기 사용을 마치면 기기에 반납하세요. AWS디바이스를 가져오기 작업에 사용한 경우 NFS 인터페이스를 통해 전송된 데이터를 Amazon S3로 가져옵니다. 아닌 경우에는 AWS로 디바이스가 반환됐을 때 디바이스에서 데이터를 완전히 삭제합니다. 이 삭제는 NIST(National Institute of Standards and Technology) 800-88 표준에 따른 것입니다.

Important

- Snowcone 디바이스에서 암호화된 AMI를 사용하는 것은 지원되지 않습니다.

- Snowcone에서 실행 중인 컴퓨팅 인스턴스의 데이터는 AWS로 가져오기 되지 않습니다.

Snowcone에서의 컴퓨팅 인스턴스 요금

컴퓨팅 인스턴스 사용 시에는 추가 비용이 발생합니다. 자세한 내용은 [AWS Snowcone 요금](#)을 참조하십시오.

사전 조건

작업을 생성하기 전에 다음 정보를 유념하세요.

- 작업에 AMI를 추가하려면 먼저 AMI가 있어야 AWS 계정하며 지원되는 이미지 유형이어야 합니다. 현재 지원되는 AMI는 다음 운영 체제를 기반으로 합니다.
 - [Amazon Linux 2](#)

Note

이 AMI의 최신 버전은 Snow Family 디바이스의 배송 준비 시점에 제공됩니다 AWS. AMI를 수신했을 때 디바이스에서 이 AMI의 버전을 확인하려면 [참조하십시오 스노우 패밀리 리용 아마존 리눅스 2 AMI 버전 결정.](#)

- [CentOS 7 \(x86_64\) - HVM 업데이트 포함](#)
- [우분투 16.04 LTS - 제니얼 \(HVM\), 우분투 20.04 LTS - 포컬 또는 우분투 22.04 LTS - 재미](#)

Note

Ubuntu 16.04 LTS - 제니얼 (HVM) 이미지는 더 이상 에서 지원되지 않지만 Amazon EC2 VM 가져오기/내보내기를 통해 Snow Family 디바이스에서 사용할 수 있으며 AMI에서 로컬로 실행되도록 계속 지원됩니다. AWS Marketplace

이 이미지는 [AWS Marketplace](#)에서 얻을 수 있습니다.

AMI를 작업 생성 요청에 추가하기 전에 AWS 계정에 지원되는 AMI가 하나 이상 있는지 확인합니다.

- 모든 AMI는 단일 볼륨의 Amazon Elastic Block Store(Amazon EBS)를 기반으로 해야 합니다.
- Snowcone에서 실행 중인 컴퓨팅 인스턴스에 연결하려는 경우 Secure Shell(SSH)을 사용해야 합니다. 이렇게 하려면 먼저 키 페어를 추가합니다.

컴퓨팅 인스턴스가 있는 작업 생성

이 단원에서는 첫 번째 컴퓨팅 인스턴스 작업을 생성합니다.

⚠ Important

작업을 생성하기 전에 다음 사항에 유의해야 합니다.

- 의 AWS Marketplace AMI를 사용하려는 경우 AMI에 지원되는 제품 코드 및 사용 작업 코드가 있는지 확인하십시오. 자세한 설명은 [AMI용 제품 및 사용 옵션 코드 확인 AWS Marketplace](#) 섹션을 참조하세요.
- AMI와 관련된 vCPU, 메모리 및 스토리지 값이 생성하고자 하는 인스턴스의 유형과 일치하는지 확인합니다.
- Snowcone에서 인스턴스를 시작한 후 SSH를 사용하여 인스턴스에 연결하려는 경우 먼저 다음 절차를 수행해야 합니다.
- AWS Snowcone 디바이스에서 암호화된 AMI 또는 암호화된 Amazon EBS 볼륨을 사용하는 것은 지원되지 않습니다.

AMI용 제품 및 사용 옵션 코드 확인 AWS Marketplace

Snow Family 디바이스에 AMI를 추가하는 프로세스를 시작하기 전에 AMI의 제품 및 사용 코드가 해당 디바이스에서 지원되는지 확인하십시오 AWS 리전. AWS Marketplace

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 표시줄에서 인스턴스를 시작하고 Snow Family 디바이스를 주문하기 위한 작업을 생성할 지역을 선택합니다. 현재 위치와 관계없이 사용자가 고를 수 있는 리전을 임의로 선택합니다.
3. 탐색 창에서 AMI를 선택합니다.
4. 필터 및 검색 옵션을 사용하여 표시된 AMI 목록의 범위를 지정하여 기준에 맞는 AMI만 볼 수 있습니다. 예를 들어 에서 제공하는 AWS모든 Linux AMI를 나열하려면 공개 이미지를 선택합니다. 그런 다음 검색 옵션을 사용하여 표시되는 AMI 목록의 범위를 추가로 지정할 수 있습니다.

(새 콘솔) 검색 창을 선택하고 메뉴에서 소유자 별칭, = 연산자, 아마존 값을 차례로 선택합니다. 검색 창을 다시 선택하고 플랫폼(Platform)에서 = 연산자를 선택한 다음 제공된 목록에서 운영 체제를 선택합니다.

(기존 콘솔) 검색(Search) 창을 선택하고 메뉴에서 소유자(Owner), 값 Amazon 이미지(Amazon images)를 차례로 선택합니다. 검색(Search) 창을 다시 선택하여 플랫폼(Platform)을 선택한 다음 제공된 목록에서 운영 체제를 선택합니다.

Note

AMI는 소스 열에 AWS-마켓플레이스를 AWS Marketplace 포함합니다.

5. AMI ID 열에서 AMI의 AMI ID를 선택합니다.
6. AMI의 이미지 요약에서 제품 코드가 해당 지역에서 지원되는지 확인하십시오. 자세한 내용은 아래 표를 참조하십시오.

Note

제품 코드 avyffzzywekkg15qv5f57ska는 모든 지역에서 지원됩니다.

지원되는 AWS Marketplace AMI 제품 코드

AMI 운영 체제	제품 코드
Ubuntu Server 14.04 LTS	b3dl4415quatdnd14qa6kcu45
CentOS 7(x86_64)	aw0evgkw8e5c1q413zgy5picce
Ubuntu 16.04 LTS	csv6h7oyg29b7epjzg7qdr7no
Amazon Linux 2	avyffzzywekkg15qv5f57ska
Ubuntu 20.04 LTS	a8jyyf4hjutohctm41o2z18m
Ubuntu 22.04 LTS	47xbqns9xujfkkjt189a13aqe

7. 모든 지역의 사용 작업 코드가 인지 확인하십시오. RunInstances

스노우 패밀리용 아마존 리눅스 2 AMI 버전 결정

Snow Family 디바이스에서 Snow Family용 Amazon Linux 2 AMI의 버전을 확인하려면 다음 절차를 따르십시오. AWS CLI 계속하기 전에 의 최신 버전을 설치하십시오. 자세한 내용은 AWS Command Line Interface 사용 [설명서의 최신 버전 설치 또는 업데이트](#)를 참조하십시오. AWS CLI

- describe-images AWS CLI 명령을 사용하여 AMI의 설명을 확인합니다. 버전은 설명에 포함되어 있습니다. 이전 단계의 공개 키 인증서를 제공하십시오. 자세한 내용은 명령 참조의 [describe-images](#)를 AWS CLI 참조하십시오.

```
aws ec2 describe-images --endpoint http://snow-device-ip:8008 --region snow
```

Example 명령 출력의 describe-images

```
{
  "Images": [
    {
      "CreationDate": "2024-02-12T23:24:45.705Z",
      "ImageId": "s.ami-02ba84cb87224e16e",
      "Public": false,
      "ProductCodes": [
        {
          "ProductCodeId": "avyfzzywekktkg15qv5f57ska",
          "ProductCodeType": "marketplace"
        }
      ],
      "State": "AVAILABLE",
      "BlockDeviceMappings": [
        {
          "DeviceName": "/dev/xvda",
          "Ebs": {
            "DeleteOnTermination": true,
            "Iops": 0,
            "SnapshotId": "s.snap-0efb49f2f726fde63",
            "VolumeSize": 8,
            "VolumeType": "sbp1"
          }
        }
      ]
    }
  ]
}
```

```

    ],
    "Description": "Snow Family Amazon Linux 2 AMI 2.0.20240131.0 x86_64
HVM gp2",
    "EnaSupport": false,
    "Name": "amzn2-ami-snow-family-hvm-2.0.20240131.0-x86_64-gp2-
b7e7f8d2-1b9e-4774-a374-120e0cd85d5a",
    "RootDeviceName": "/dev/xvda"
  }
]
}

```

이 예제에서 스노우 패밀리용 Amazon Linux 2 AMI 버전은 다음과 같습니다 **2.0.20240131.0**. `Description` 이름의 값에서 찾을 수 있습니다.

디바이스에서 시작된 컴퓨팅 인스턴스에 연결되도록 AMI를 구성하여 SSH 사용

Secure Shell(SSH)을 사용하여 Snowcone 디바이스의 컴퓨팅 인스턴스에 연결하려면 작업을 생성하기 전에 다음 절차를 수행하여야 합니다. 이 절차는 작업을 생성하기 전 AMI에 SSH 키를 추가합니다. 또한, 이 절차를 사용하여 작업의 AMI로 사용하려는 인스턴스에 애플리케이션을 설정하는 것이 좋습니다.

SSH 키를 AMI에 넣기

1. [스노우 패밀리용 아마존 리눅스 2, CentOS 7 \(x86_64\) AWS 클라우드 기반의 업데이트 HVM 또는 우분투 16.04 LTS - 제니얼 \(HVM\) 이미지로 새 인스턴스를 시작합니다.](#)

인스턴스 시작 시 인스턴스에 할당하는 스토리지 크기가 Snowcone에서 나중에 사용할 용도로 적합한지 확인합니다. Amazon EC2 콘솔에서는 4단계: 스토리지 추가에서 이 작업을 수행합니다. Snowcone에서 컴퓨팅 인스턴스 스토리지 볼륨에 지원되는 크기의 목록은 [AWS Snowcone 할당량](#) 섹션을 참조하십시오.

2. Snowcone에서 실행하고 싶은 애플리케이션을 설치 및 구성한 후 예상대로 작동하는지 테스트합니다.
3. 이 인스턴스를 생성하기 위해 SSH 키 페어에 사용한 PEM/PPK 파일의 사본을 만듭니다. 이 파일을 Snowcone과 통신하기 위해 사용하려는 서버에 저장합니다. 이 파일은 디바이스에서 시작된 인스턴스에 SSH를 사용하여 연결할 때 필요하므로 이 파일의 경로를 적어둡니다.
4. 인스턴스를 AMI로 저장합니다. 자세한 정보는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EBS 지원 Linux AMI 생성](#)을 참조하십시오.

- SSH를 사용하여 연결하려는 각 인스턴스에 대해 이 절차를 반복합니다. 서로 다른 SSH 키 페어의 사본이 있는지 확인하고 관련된 AMI에 유의합니다.

콘솔에서 작업 생성

다음 단계는 Snow Family 디바이스를 주문하기 위한 작업을 생성하는 것입니다. 작업의 유형은 클러스터를 포함해 어떤 유형이어도 됩니다. [AWS Snow 패밀리 관리 콘솔](#)을 사용하려면 [시작하기](#)의 지침을 따르십시오.

에서 Job 생성하기 AWS CLI

또는 AWS CLI를 사용하여 작업을 생성할 수도 있습니다. 이렇게 하려면 터미널을 열고 다음 명령을 실행하여 빨간색으로 표시된 텍스트를 실제 값으로 교체합니다.

```
aws snowballEdge create-job --job-type IMPORT --resources '{"S3Resources": [{"BucketArn": "arn:aws:s3:::bucket-name"}], "Ec2AmiResources": [{"AmiId": "ami-12345678"}]}' --description Example --address-id ADIEXAMPLE60-1234-1234-5678-41fEXAMPLE57 --kms-key-arn arn:aws:kms:us-west-2:012345678901:key/eEXAMPLE-1234-1234-5678-5b4EXAMPLE8e --role-arn arn:aws:iam::012345678901:role/snowball-local-s3-lambda-us-west-2-role --snowball-capacity-preference T100 --shipping-option SECOND_DAY --snowball-type SNOWCONE
```

디바이스가 도착하면 잠금을 해제한 후 Snowball Edge 클라이언트를 사용하여 로컬 자격 증명을 얻습니다. 자세한 설명은 [자격 증명 받기](#) 섹션을 참조하세요.

컴퓨팅 인스턴스에 대한 네트워크 구성

Snow Family 디바이스에서 컴퓨팅 인스턴스를 시작한 후에는 네트워크 인터페이스를 생성하여 IP 주소를 제공해야 합니다. Snow Family 디바이스는 가상 네트워크 인터페이스와 직접 네트워크 인터페이스라는 두 종류의 네트워크 인터페이스를 지원합니다.

가상 네트워크 인터페이스(VNI)

가상 네트워크 인터페이스는 Snow Family 디바이스의 EC2 호환 인스턴스에 연결하기 위한 표준 네트워크 인터페이스입니다. 직접 네트워크 인터페이스 사용 여부에 관계없이 EC2 호환 인스턴스 각각에 대해 VNI를 생성해야 합니다. VNI를 통과하는 트래픽은 설정한 보안 그룹에 의해 보호됩니다. Snow Family 디바이스를 제어하는 데 사용하는 물리적 네트워크 포트에만 VNI를 연결할 수 있습니다.

Note

VNI는 Snow Family 디바이스를 관리하는 데 사용되는 것과 동일한 물리적 인터페이스(RJ45, SFP+ 또는 QSFP)를 사용합니다. 디바이스 관리에 사용되는 인터페이스와 다른 물리적 인터페이스에 VNI를 만들면 예상치 못한 결과가 발생할 수 있습니다.

다이렉트 네트워크 인터페이스(DNI)

다이렉트 네트워크 인터페이스(DNI)는 멀티캐스트 스트림, 전환 라우팅, 로드 밸런싱과 같은 사용 사례를 지원하는 고급 네트워크 기능입니다. 중개 변환이나 필터링 없이 계층 2 네트워크 액세스 권한을 인스턴스에 제공하면 Snow Family 디바이스의 네트워크 구성에서 유연성을 높이고 네트워크 성능을 개선할 수 있습니다. DNI는 VLAN 태그를 지원하고 MAC 주소를 사용자 지정합니다. DNI의 트래픽은 보안 그룹에 의해 보호되지 않습니다.

Snowcone 디바이스는 EC2 호환 인스턴스당 8개의 DNI를 지원하며 디바이스당 최대 8개의 DNI를 지원합니다.

주제

- [필수 조건](#)
- [가상 네트워크 인터페이스\(VNI\) 설정](#)
- [다이렉트 네트워크 인터페이스\(DNI\) 설정](#)

필수 조건

VNI 또는 DNI를 구성하려면 먼저 다음 사전 조건을 충족해야 합니다.

1. 디바이스에 전원이 연결되어 있고 RJ45 포트와 같은 물리적 네트워크 인터페이스 중 하나가 IP 주소와 연결되어 있는지 확인합니다.
2. Snow Family 디바이스에서 사용 중인 물리적 네트워크 인터페이스에 연결된 IP 주소를 얻습니다.
3. Snowball Edge 클라이언트를 구성합니다. 자세한 내용은 [Snowcone 클라이언트용 프로필 구성](#)을 참조하십시오.
4. 디바이스의 잠금을 해제합니다. 디바이스 잠금을 AWS OpsHub for Snow Family 해제할 때는 사용하는 것이 좋습니다. 자세한 지침은 [디바이스 잠금 해제](#)를 참조하십시오.

CLI 명령을 사용하려면 다음 명령을 실행하고 대화 상자에 나타나는 정보를 제공하십시오.

```
snowballEdge configure
```

Snowball Edge Manifest Path: manifest.bin

Unlock Code: *unlock code*

Default Endpoint: https://*device ip*

5. 다음 명령을 실행합니다.

```
snowballEdge unlock-device
```

디바이스 디스플레이 업데이트는 잠금이 해제되었음을 나타냅니다.

6. 디바이스에서 EC2 호환 인스턴스를 시작합니다. VNI를 이 인스턴스와 연결합니다.
7. snowballEdge describe-device 명령을 실행하여 물리적 네트워크 인터페이스 ID의 목록을 가져옵니다.
8. 사용하려는 물리적 네트워크 인터페이스의 ID를 알아내 이를 기록해 둡니다.

가상 네트워크 인터페이스(VNI) 설정

물리적 네트워크 인터페이스의 ID를 식별한 후 가상 네트워크 인터페이스(VNI)를 설정할 수 있습니다. 다음 절차에 따라 VNI를 설정합니다. VNI를 생성하기 전에 필수 작업을 수행해야 합니다.

VNI를 생성하고 IP 주소를 연결합니다.

1. snowballEdge create-virtual-network-interface 명령을 실행합니다. 다음 예시에서는 두 가지 서로 다른 IP 주소 배정 방법, 즉 DHCP 또는 STATIC을 사용하여 이 명령을 실행하는 방법을 보여줍니다. DHCP 메서드는 Dynamic Host Configuration Protocol(DHCP)를 사용합니다.

```
snowballEdge create-virtual-network-interface \
--physical-network-interface-id s.ni-abcd1234 \
--ip-address-assignment DHCP
```

//OR//

```
snowballEdge create-virtual-network-interface \
--physical-network-interface-id s.ni-abcd1234 \
--ip-address-assignment STATIC \
```

```
--static-ip-address-configuration IpAddress=192.0.2.0,Netmask=255.255.255.0
```

이 명령은 IP 주소가 포함된 JSON 구조를 반환합니다. 프로세스 후반부에 `ec2 associate-address` AWS CLI 명령에 사용할 해당 IP 주소를 기록해 두십시오.

이 IP 주소가 필요할 때마다 `snowballEdge describe-virtual-network-interfaces` Snowball Edge 클라이언트 명령 또는 명령을 사용하여 IP 주소를 가져올 수 있습니다. `aws ec2 describe-addresses` AWS CLI

2. 새로 생성된 IP 주소를 인스턴스와 연결하려면 다음 명령을 사용하여 빨간색 텍스트를 사용자의 값으로 바꿉니다.

```
aws ec2 associate-address --public-ip 192.0.2.0 --instance-id s.i-01234567890123456
--endpoint http://Snow Family device physical IP address:8008
```

다이렉트 네트워크 인터페이스(DNI) 설정

Note

다이렉트 네트워크 인터페이스 기능은 2021년 1월 12일 또는 그 이후에 사용할 수 있으며 Snow Family 디바이스가 제공되는 모든 AWS 리전 곳에서 사용할 수 있습니다.

필수 조건

직접 네트워크 인터페이스(DNI)를 설정하기 전에 사전 조건 섹션의 태스크를 수행해야 합니다.

1. DNI를 설정하기 전에 필수 태스크를 수행합니다. 지침은 [필수 조건](#) 섹션을 참조하세요.
2. 또한 디바이스에서 인스턴스를 시작하고, VNI를 생성한 다음, 인스턴스와 연결해야 합니다. 지침은 [가상 네트워크 인터페이스\(VNI\) 설정](#) 섹션을 참조하십시오.

Note

in-the-field 소프트웨어 업데이트를 수행하여 기존 장치에 다이렉트 네트워킹을 추가한 경우 이 기능을 완전히 활성화하려면 장치를 두 번 다시 시작해야 합니다.

DNI를 생성하고 IP 주소를 연결합니다.

1. 다음 명령을 실행하여 직접 네트워크 인터페이스를 생성하고 Amazon EC2 호환 인스턴스에 연결합니다. 다음 단계에서는 디바이스의 MAC 주소가 필요합니다.

```
create-direct-network-interface [--endpoint endpoint] [--instance-id instanceId]
  [--mac macAddress]
  [--physical-network-interface-id physicalNetworkInterfaceId]
  [--unlock-code unlockCode] [--vlan vlanId]
```

OPTIONS

--endpoint <endpoint> 이 요청을 보낼 엔드포인트입니다. 디바이스의 엔드포인트는 https 스키마 뒤에 IP 주소를 사용하는 URL이 됩니다. 예를 들어 디바이스의 IP 주소가 123.0.1.2인 경우 디바이스의 엔드포인트는 https://123.0.1.2가 됩니다.

--instance-id <instanceId> 인터페이스를 연결할 EC2 호환 인스턴스 ID입니다(선택 사항).

--mac <macAddress> 네트워크 인터페이스의 MAC 주소를 설정합니다(선택 사항).

--physical-network-interface-id <physicalNetworkInterfaceId> 새 가상 네트워크 인터페이스를 생성할 물리적 네트워크 인터페이스의 ID입니다. describe-device 명령을 사용하여 Snowball Edge에서 사용할 수 있는 물리적 네트워크 인터페이스를 결정할 수 있습니다.

--vlan <vlanId> 인터페이스에 할당된 VLAN을 설정합니다(선택 사항). 지정된 경우 인터페이스에서 전송되는 모든 트래픽에는 지정된 VLAN ID의 태그가 지정됩니다. 수신 트래픽은 지정된 VLAN ID로 필터링되며, 인스턴스로 전달되기 전에 모든 VLAN 태그가 제거됩니다.

2. 1단계에서 DNI를 인스턴스와 연결하지 않은 경우 [다이렉트 네트워크 인터페이스 업데이트](#) 명령을 실행하여 인스턴스를 연결할 수 있습니다.
3. DNI를 생성하여 EC2 호환 인스턴스에 연결한 후에는 Amazon EC2 호환 인스턴스 내에서 구성을 두 번 변경해야 합니다.
 - 첫 번째는 EC2 호환 인스턴스와 연결된 VNI를 위한 패킷이 eth0을 통해 전송되도록 변경하는 것입니다.
 - 두 번째는 부팅 시 DHCP 또는 고정 IP를 사용하도록 직접 네트워크 인터페이스를 구성하는 것입니다.

다음은 이러한 구성을 변경하는 Amazon Linux 2 및 CentOS Linux용 셸 스크립트의 예시입니다.

Amazon Linux 2

```
# Mac address of the direct network interface.
# You got this when you created the direct network interface.
DNI_MAC=[MAC ADDRESS FROM CREATED DNI]

# Configure routing so that packets meant for the VNI always are sent through
eth0.
PRIVATE_IP=$(curl -s http://169.254.169.254/latest/meta-data/local-ipv4)
PRIVATE_GATEWAY=$(ip route show to match 0/0 dev eth0 | awk '{print $3}')
ROUTE_TABLE=10001
echo "from $PRIVATE_IP table $ROUTE_TABLE" > /etc/sysconfig/network-scripts/
rule-eth0
echo "default via $PRIVATE_GATEWAY dev eth0 table $ROUTE_TABLE" > /etc/
sysconfig/network-scripts/route-eth0
echo "169.254.169.254 dev eth0" >> /etc/sysconfig/network-scripts/route-eth0

# Query the persistent DNI name, assigned by udev via ec2net helper.
# changable in /etc/udev/rules.d/70-persistent-net.rules
DNI=$(ip --oneline link | grep -i $DNI_MAC | awk -F ': ' '{ print $2 }')

# Configure DNI to use DHCP on boot.
cat << EOF > /etc/sysconfig/network-scripts/ifcfg-$DNI
DEVICE="$DNI"
NAME="$DNI"
HWADDR=$DNI_MAC
ONBOOT=yes
NOZEROCONF=yes
BOOTPROTO=dhcp
TYPE=Ethernet
MAINROUTETABLE=no
EOF

# Make all changes live.
systemctl restart network
```

CentOS Linux

```
# Mac address of the direct network interface. You got this when you created the
direct network interface.
DNI_MAC=[MAC ADDRESS FROM CREATED DNI]
# The name to use for the direct network interface. You can pick any name that
isn't already in use.
DNI=eth1

# Configure routing so that packets meant for the VNIC always are sent through
eth0
PRIVATE_IP=$(curl -s http://169.254.169.254/latest/meta-data/local-ipv4)
PRIVATE_GATEWAY=$(ip route show to match 0/0 dev eth0 | awk '{print $3}')
ROUTE_TABLE=10001
echo from $PRIVATE_IP table $ROUTE_TABLE > /etc/sysconfig/network-scripts/rule-
eth0
echo default via $PRIVATE_GATEWAY dev eth0 table $ROUTE_TABLE > /etc/sysconfig/
network-scripts/route-eth0

# Configure your direct network interface to use DHCP on boot.
cat << EOF > /etc/sysconfig/network-scripts/ifcfg-$DNI
DEVICE="$DNI"
NAME="$DNI"
HWADDR="$DNI_MAC"
ONBOOT=yes
NOZEROCONF=yes
BOOTPROTO=dhcp
TYPE=Ethernet
EOF

# Rename DNI device if needed.
CURRENT_DEVICE_NAME=$(LANG=C ip -o link | awk -F ':' -v IGNORECASE=1 '!/link\/
ieee802\.\.11/ && /'"$DNI_MAC"'/ { print $2 }')
ip link set $CURRENT_DEVICE_NAME name $DNI

# Make all changes live.
systemctl restart network
```

추가 명령

AWS Snowcone 디바이스에 사용할 수 있는 모든 명령은 [Snowball Edge 클라이언트 사용을 참조](#) 하십시오.

SSH를 사용하여 Snowcone의 컴퓨팅 인스턴스에 연결

SSH를 사용하여 Snowcone 디바이스의 컴퓨팅 인스턴스에 연결하려면 작업을 생성하기 전에 먼저 SSH 키를 AMI에 제공해야 합니다. 이 절차에 대한 자세한 내용은 [디바이스에서 시작된 컴퓨팅 인스턴스에 연결되도록 AMI를 구성하여 SSH 사용](#) 섹션을 참조하십시오. 해당 절차를 따르지 않으면 SSH를 사용하여 인스턴스에 연결할 수 없습니다.

SSH로 인스턴스에 연결하려면

1. 디바이스에 전원이 인가되어 있고 네트워크에 연결되어 있으며 잠금 해제가 되어 있는지 확인합니다.
2. 컴퓨팅 인스턴스에 대해 네트워크 설정이 구성되어 있는지 확인합니다. 자세한 내용은 [컴퓨팅 인스턴스에 대한 네트워크 구성](#) 섹션을 참조하십시오.
3. 이러한 특정 인스턴스에 대해 사용한 PEM 또는 PPK 키 페어를 찾기 위해 메모를 확인합니다. 이 파일을 컴퓨터에 복사합니다. PEM 파일의 경로를 적어 둡니다.
4. 다음 예시 명령과 같이 SSH를 통해 인스턴스에 연결합니다. IP 주소는 [컴퓨팅 인스턴스에 대한 네트워크 구성](#)에서 설정한 가상 네트워크 인터페이스(VNIC)의 IP 주소입니다.

```
ssh -i path/to/PEM/key/file instance-user-name@192.0.2.0
```

자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [SSH를 사용하여 Linux 인스턴스에 연결](#)을 참조하십시오.

컴퓨팅 인스턴스에 대한 Snowcone 클라이언트 명령

Snowball Edge 클라이언트는 로컬 서버에서 실행되는 독립형 터미널 애플리케이션입니다. 이 애플리케이션은 Snowcone 디바이스에서 일부 관리 작업을 할 수 있게 해 줍니다. 서비스를 시작 및 중지하는 방법을 비롯하여 Snowball Edge 클라이언트를 사용하는 방법에 대한 자세한 내용은 [AWS Snowball Edge 클라이언트 사용](#) 섹션을 참조하십시오.

아래에서 사용 예시를 포함하여 컴퓨팅 인스턴스와 관련된 Snowball Edge 클라이언트 명령에 대한 정보를 찾을 수 있습니다. AWS Snowcone 디바이스에서 사용할 수 있는 Amazon EC2 호환 명령 목록은 [Snowcone에서 지원되는 아마존 EC2 호환 AWS CLI 명령](#)을 참조하십시오.

Note

클러스터와 관련된 명령은 지원되지 않으며 오류를 반환합니다.

Amazon EC2 호환 인스턴스를 자동 시작하는 시작 구성 생성

잠금이 해제된 후 AWS Snowcone 디바이스에서 Amazon EC2 호환 컴퓨팅 인스턴스를 자동으로 시작하려면 시작 구성을 생성하면 됩니다. 이를 위해 다음 예시와 같이 `snowballEdge create-autostart-configuration` 명령을 사용하십시오.

사용량

```
snowballEdge create-autostart-configuration --physical-connector-type
[SFP_PLUS or RJ45]
--ip-address-assignment [DHCP or STATIC]
[--static-ip-address-configuration
IpAddress=[IP address],NetMask=[Netmask]]
--launch-template-id
[--launch-template-version]
```

EC2 호환 인스턴스를 자동 시작하는 시작 구성 업데이트

Snowcone의 기존 시작 구성을 업데이트하려면 `snowballEdge update-autostart-configuration` 명령을 사용합니다. 다음과 같은 사용법을 찾아볼 수 있습니다. 시작 구성을 활성화 또는 비활성화하려면 `--enabled` 파라미터를 지정합니다.

사용량

```
snowballEdge update-autostart-configuration --autostart-configuration-
arn
--physical-connector-type [SFP_PLUS or
RJ45]]
--ip-address-assignment [DHCP or STATIC]]
[--static-ip-address-configuration
IpAddress=[IP address],NetMask=[Netmask]]
--launch-template-id
[--launch-template-version]
[--enabled]
```


EC2 호환 인스턴스를 자동 시작하는 시작 구성 삭제

더 이상 사용하지 않는 시작 구성을 삭제하려면 `snowballEdge delete-autostart-configuration` 명령을 사용합니다. 다음과 같은 사용법을 찾아볼 수 있습니다.

사용량

```
snowballEdge delete-autostart-configuration --autostart-configuration-arn
```

EC2 호환 인스턴스를 자동 시작하는 시작 구성 나열

Snowcone에서 생성한 시작 구성을 나열하려면 `describe-autostart-configurations` 명령을 사용합니다. 다음과 같은 사용법을 찾아볼 수 있습니다.

사용량

```
snowballEdge describe-autostart-configurations
```

가상 네트워크 인터페이스 생성

Snowcone에서 컴퓨팅 인스턴스를 실행하거나 Snowcone에서 NFS 인터페이스를 시작하려면 먼저 가상 네트워크 인터페이스 (VNIC) 를 생성합니다. 각 Snowcone에는 이 디바이스의 물리적 네트워크 인터페이스 컨트롤러인 네트워크 인터페이스(NIC)가 세 개 있습니다. 디바이스 후면에 있는 RJ45 포트가 바로 이 인터페이스입니다.

각 VNIC는 물리적 인터페이스에 기반을 두고 있으므로 각 NIC에 VNIC를 얼마든지 연결할 수 있습니다. 가상 네트워크 인터페이스를 생성하려면 `snowballEdge create-virtual-network-interface` 명령을 사용하십시오.

Note

`--static-ip-address-configuration` 파라미터는 `--ip-address-assignment` 파라미터에 대해 `STATIC` 옵션을 사용하는 경우에만 유효합니다.

사용법

Snowball Edge 클라이언트를 구성하거나 Snowball Edge 클라이언트를 구성하지 않는 두 가지 방법으로 이 명령을 사용할 수 있습니다. 다음 사용 예시는 Snowball Edge 클라이언트가 구성된 방법을 보여줍니다.

```
snowballEdge create-virtual-network-interface --ip-address-assignment [DHCP or STATIC]
--physical-network-interface-id [physical network interface id] --static-ip-address-
configuration IPAddress=[IP address],NetMask=[Netmask]
```

다음 사용 예시는 Snowball Edge 클라이언트가 구성되지 않은 방법을 보여줍니다.

```
snowballEdge create-virtual-network-interface --endpoint https://[ip address]
--manifest-file /path/to/manifest --unlock-code [unlock code] --ip-address-
assignment [DHCP or STATIC] --physical-network-interface-id [physical network interface
id] --static-ip-address-configuration IPAddress=[IP address],NetMask=[Netmask]
```

Example 예시: VNIC 생성(DHCP 사용)

```
snowballEdge create-virtual-network-interface --ip-address-assignment dhcp --physical-
network-interface-id s.ni-8EXAMPLEaEXAMPLEd
{
  "VirtualNetworkInterface" : {
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-8EXAMPLE8EXAMPLEf",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.0",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "EX:AM:PL:E1:23:45"
  }
}
```

가상 네트워크 인터페이스 설명

디바이스에서 앞서 생성한 VNIC를 설명하려면 `snowballEdge describe-virtual-network-interfaces` 명령을 사용합니다. 다음과 같은 사용법을 찾아볼 수 있습니다.

사용법

Snowball Edge 클라이언트를 구성하거나 Snowball Edge 클라이언트를 구성하지 않는 두 가지 방법으로 이 명령을 사용할 수 있습니다. 다음 사용 예시는 Snowball Edge 클라이언트가 구성된 방법을 보여줍니다.

```
snowballEdge describe-virtual-network-interfaces
```

다음 사용 예시는 Snowball Edge 클라이언트가 구성되지 않은 방법을 보여줍니다.

```
snowballEdge describe-virtual-network-interfaces --endpoint https://[ip address] --manifest-file /path/to/manifest --unlock-code [unlock code]
```

Example 예시: VNIC 설명

```
snowballEdge describe-virtual-network-interfaces
[
  {
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-8EXAMPLE8EXAMPLE8",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.0",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "EX:AM:PL:E1:23:45"
  },{
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device:::interface/
s.ni-1EXAMPLE1EXAMPLE1",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.2",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "12:34:5E:XA:MP:LE"
  }
]
```

가상 네트워크 인터페이스 업데이트

VNIC(가상 네트워크 인터페이스)를 생성한 후에 `snowballEdge update-virtual-network-interface` 명령을 사용하여 해당 구성을 업데이트할 수 있습니다. 특정 VNIC에 ARN(Amazon 리소스 이름)을 제공한 후 업데이트하려는 모든 요소에 대해 이 값만 제공하면 됩니다.

사용량

Snowball Edge 클라이언트를 구성하거나 Snowball Edge 클라이언트를 구성하지 않는 두 가지 방법으로 이 명령을 사용할 수 있습니다. 다음 사용 예시는 Snowball Edge 클라이언트가 구성된 방법을 보여줍니다.

```
snowballEdge update-virtual-network-interface --virtual-network-interface-arn [virtual network-interface-arn] --ip-address-assignment [DHCP or STATIC] --physical-network-interface-id [physical network interface id] --static-ip-address-configuration IpAddress=[IP address],NetMask=[Netmask]
```

다음 사용 예시는 Snowball Edge 클라이언트가 구성되지 않은 방법을 보여줍니다.

```
snowballEdge update-virtual-network-interface --endpoint https://[ip address] --manifest-file /path/to/manifest --unlock-code [unlock code] --virtual-network-interface-arn [virtual network-interface-arn] --ip-address-assignment [DHCP or STATIC] --physical-network-interface-id [physical network interface id] --static-ip-address-configuration IpAddress=[IP address],NetMask=[Netmask]
```

Example 예시: VNIC 업데이트(DHCP 사용)

```
snowballEdge update-virtual-network-interface --virtual-network-interface-arn arn:aws:snowball-device:::interface/s.ni-8EXAMPLEbEXAMPLEd --ip-address-assignment dhcp
```

가상 네트워크 인터페이스 삭제

가상 네트워크 인터페이스를 삭제하려면 `snowballEdge delete-virtual-network-interface` 명령을 사용하면 됩니다.

사용량

Snowball Edge 클라이언트를 구성하거나 Snowball Edge 클라이언트를 구성하지 않는 두 가지 방법으로 이 명령을 사용할 수 있습니다. 다음 사용 예시는 Snowball Edge 클라이언트가 구성된 방법을 보여줍니다.

```
snowballEdge delete-virtual-network-interface --virtual-network-interface-arn [virtual network-interface-arn]
```

다음 사용 예시는 Snowball Edge 클라이언트가 구성되지 않은 방법을 보여줍니다.

```
snowballEdge delete-virtual-network-interface --endpoint https://[ip address] --
manifest-file /path/to/manifest --unlock-code [unlock code] --virtual-network-
interface-arn [virtual network-interface-arn]
```

Example 예시: VNIC 삭제

```
snowballEdge delete-virtual-network-interface --virtual-network-interface-arn
arn:aws:snowball-device:::interface/s.ni-8EXAMPLEbEXAMPLEd
```

Amazon EC2 호환 인스턴스에서 Snow용 인스턴스 메타데이터 서비스 사용

Snow용 IMDS는 Snow의 Amazon EC2 호환 인스턴스에 대한 인스턴스 메타데이터 서비스(IMDS)를 제공합니다. 인스턴스 메타데이터는 인스턴스에 대한 정보 카테고리입니다. 여기에는 호스트 이름, 이벤트, 보안 그룹 등의 범주가 포함됩니다. Snow용 IMDS를 사용하면 인스턴스 메타데이터로 Amazon EC2 호환 인스턴스를 시작할 때 지정한 사용자 데이터에 액세스할 수 있습니다. 예를 들어, Snow용 IMDS를 사용하여 인스턴스 구성을 위한 파라미터를 지정하거나 단순 스크립트에 이러한 파라미터를 포함시킬 수 있습니다. 일반 AMI를 빌드하고 사용자 데이터를 사용하여 시작 시간에 제공되는 구성 파일을 수정할 수 있습니다.

인스턴스 메타데이터와 사용자 데이터 및 Snow EC2 호환 인스턴스에 대해 알아보려면 이 설명서의 [지원되는 인스턴스 메타데이터 및 사용자 데이터](#)를 참조하십시오.

Important

사용자는 인스턴스 자체 내에서 인스턴스 메타데이터 및 사용자 데이터에만 액세스할 수 있지만, 데이터는 인증 또는 암호화 방법으로 보호되지 않습니다. 인스턴스에 직접 액세스할 수 있는 모든 사람과 인스턴스에서 실행 중인 모든 소프트웨어는 메타데이터를 볼 수 있습니다. 따라서 암호 또는 수명이 긴 암호화 키와 같은 민감한 데이터를 사용자 데이터로 저장해서는 안 됩니다.

Note

이 섹션의 예에서는 인스턴스 메타데이터 서비스의 IPv4 주소(169.254.169.254)를 사용합니다. 링크-로컬 IPv6 주소를 사용한 인스턴스 메타데이터 검색은 지원되지 않습니다.

주제

- [IMDS 버전](#)
- [IMDSv1 및 IMDSv2를 사용하여 인스턴스 메타데이터를 검색하는 예](#)

IMDS 버전

IMDS 버전 2 또는 IMDS 버전 1을 사용하여 실행 중인 인스턴스에서 인스턴스 메타데이터에 액세스할 수 있습니다.

- 인스턴스 메타데이터 서비스 버전 2(IMDSv2) – 세션 지향 방법
- 인스턴스 메타데이터 서비스 버전 1(IMDSv1) – 요청/응답 방법

Snow 소프트웨어 버전에 따라 IMDSv1 또는 IMDSv2를 사용하거나 둘 다 사용할 수 있습니다. 이는 EC2 호환 인스턴스에서 실행되는 AMI 유형에 따라서도 달라집니다. Ubuntu 20.04를 실행하는 AMI와 같은 일부 AMI에는 IMDSv2가 필요합니다. 인스턴스 메타데이터 서비스는 PUT 또는 GET 헤더의 존재 여부에 따라 IMDSv1 요청과 IMDSv2 요청을 구별합니다. IMDSv2는 이 두 헤더를 모두 사용합니다. IMDSv1은 GET 헤더만 사용합니다.

AWS에서는 IMDSv2에 더 높은 보안이 포함되어 있으므로 IMDSv1보다는 IMDSv2를 사용하는 것을 권장합니다. 자세한 내용은 [EC2 인스턴스 메타데이터 서비스의 향상된 기능을 통해 개방형 방화벽, 역방향 프록시 및 SSRF 취약성에 대한 심층적인 방어 기능 추가](#)를 참조하십시오.

IMDSv2

IMDSv2는 세션 지향 요청을 사용합니다. 세션 지향 요청으로 세션 기간을 정의하는 세션 토큰을 생성합니다. 세션 기간은 최소 1초, 최대 6시간입니다. 지정된 기간 중에는 후속 요청에 동일한 세션 토큰을 사용할 수 있습니다. 지정된 기간이 만료된 후에는 향후 요청에 사용할 새로운 세션 토큰을 생성해야 합니다.

다음 예에서는 Linux 셸 스크립트와 IMDSv2를 사용하여 최상위 인스턴스 메타데이터 항목을 가져옵니다. 이 예에서는

1. PUT 요청을 사용하여 6시간(21,600초) 동안 지속되는 세션 토큰을 생성합니다.
2. TOKEN이라는 변수에 세션 토큰 헤더를 저장합니다.
3. 토큰을 사용하여 최상위 메타데이터 항목을 요청합니다.

별도의 두 명령을 실행하거나 둘을 결합할 수 있습니다.

별도의 명령

먼저 다음 명령을 사용하여 토큰을 생성합니다.

Note

X-aws-ec2-metadata-token-ttl-seconds는 필수 헤더입니다. 이 헤더가 포함되지 않은 경우 400 - 파라미터 누락 또는 잘못된 파라미터 오류 코드를 받게 됩니다.

```
[ec2-user ~]$ TOKEN=curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"
```

그런 다음 해당 토큰을 사용하여 다음 명령으로 최상위 메타데이터 항목을 생성합니다.

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/
```

결합된 명령

토큰을 저장하고 명령을 결합할 수 있습니다. 다음 예는 위의 두 명령을 결합하고 TOKEN이라는 변수에 세션 토큰 헤더를 저장합니다.

Note

토큰을 만드는 데 오류가 발생하면 유효한 토큰 대신에 오류 메시지가 변수에 저장되고 명령이 작동하지 않습니다.

Example 결합된 명령

```
[ec2-user ~]$ TOKEN=curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/
```

토큰을 생성한 후에는 만료될 때까지 토큰을 재사용할 수 있습니다. 다음 예제 명령에서는 인스턴스를 시작하는 데 사용한 AMI의 ID를 가져오고 이전 예에서 생성한 \$TOKEN에 이를 저장합니다.

Example 토큰 재사용의 경우

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/ami-id
```

IMDSv2를 사용하여 인스턴스 메타데이터를 요청하는 경우 요청에는 다음 규칙이 포함되어야 합니다.

1. PUT 요청을 사용하여 인스턴스 메타데이터 서비스의 세션을 초기화합니다. PUT 요청은 후속 GET 요청에 포함되어야 하는 토큰을 인스턴스 메타데이터 서비스에 반환합니다. 토큰은 IMDSv2를 사용하여 메타데이터에 액세스하는 데 필요합니다.
2. 인스턴스 메타데이터 서비스에 대한 모든 GET 요청에 토큰을 포함시킵니다.
 - a. 토큰은 인스턴스에 특정한 키입니다. 토큰은 다른 인스턴스에서 유효하지 않으며 해당 토큰이 생성된 인스턴스 외부에서 사용하려고 시도하면 거부됩니다.
 - b. PUT 요청에는 토큰의 TTL(Time to Live)를 최대 6시간(21,600초)까지 초 단위로 지정하는 헤더가 포함되어야 합니다. 토큰은 논리 세션을 나타냅니다. TTL은 토큰이 유효한 시간 길이를 지정하며 따라서 세션 기간을 지정합니다.
 - c. 토큰이 만료된 후 인스턴스 메타데이터에 계속 액세스하려면 다른 PUT 요청을 사용하여 새 세션을 생성해야 합니다.
 - d. 토큰을 재사용하거나 모든 요청에 새 토큰을 생성하도록 선택할 수 있습니다. 요청 수가 적은 경우 인스턴스 메타데이터 서비스에 액세스해야 할 때마다 토큰을 생성하고 즉시 사용하는 것이 더 간편할 수 있습니다. 하지만 효율성을 향상하려면 인스턴스 메타데이터를 요청해야 할 때마다 PUT 요청을 작성하는 대신 토큰에 더 긴 기간을 지정하고 토큰을 재사용할 수 있습니다. 동시 토큰 수에는 실질적인 제한이 없으며 각각은 자체 세션을 나타냅니다.

IMDSv2 인스턴스 메타데이터 요청에서는 HTTP GET 및 HEAD 메서드가 허용됩니다. PUT 요청은 X-Forwarded-For 헤더가 포함된 경우 거부됩니다.

기본적으로 PUT 요청에 대한 응답에는 IP 프로토콜 수준에서 1의 응답 홉 제한(TTL(Time to Live))이 있습니다. Snow용 IMDS는 PUT 응답에 대한 홉 제한을 수정할 수 없습니다.

IMDSv1

IMDSv1은 요청-응답 모델을 사용합니다. 인스턴스 메타데이터를 요청하려면 인스턴스 메타데이터 서비스에 GET 요청을 전송합니다.

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
```

인스턴스 메타데이터 검색

실행 중인 인스턴스에서 인스턴스 메타데이터를 사용할 수 있기 때문에 Amazon EC2 콘솔 또는 AWS CLI를 사용할 필요가 없습니다. 이는 인스턴스에서 실행할 스크립트를 작성할 때 유용합니다. 예를 들어, 사용자는 인스턴스 메타데이터에서 인스턴스의 로컬 IP 주소에 액세스하여 외부 애플리케이션과의 연결을 관리할 수 있습니다. 인스턴스 메타데이터는 몇 가지 범주로 분류될 수 있습니다. 각 인스턴스 메타데이터 범주에 대한 설명은 이 설명서의 [지원되는 인스턴스 메타데이터 및 사용자 데이터를 참조](#)를 참조하십시오.

실행 중인 모든 인스턴스 메타데이터 범주를 살펴보려면 다음 IPv4 URI를 사용하십시오.

```
http://169.254.169.254/latest/meta-data/
```

IP 주소는 링크-로컬 주소이며 인스턴스에서만 유효합니다. 자세한 내용은 Wikipedia의 [Link-local address](#)를 참조하십시오.

응답 및 오류 메시지

모든 인스턴스 메타데이터는 텍스트(HTTP 콘텐츠 유형 text/plain)로 반환됩니다.

특정 메타데이터 리소스를 요청하면 적절한 값이 반환되거나 리소스를 이용할 수 없는 경우 404 - 찾을 수 없음 HTTP 오류 코드가 반환됩니다.

일반 메타데이터 리소스(/로 끝나는 URI)를 요청한 경우 이용 가능한 리소스 목록이 반환되거나 해당 리소스가 없는 경우 404 - 찾을 수 없음 HTTP 오류 코드가 반환됩니다. 목록 항목은 개별 라인에 표시되고 줄바꿈(ASCII 문자 코드 10)으로 끝납니다.

IMDSv1을 사용하여 수행한 요청의 경우 다음 HTTP 오류 코드가 반환될 수 있습니다.

- 400 - 파라미터 누락 또는 유효하지 않음 - PUT 요청이 유효하지 않습니다.

- 401 - 권한이 없음 - GET 요청이 유효하지 않은 토큰을 사용합니다. 권장되는 작업은 새 토큰을 생성하는 것입니다.
- 403 - 금지됨 - 요청이 허용되지 않거나 인스턴스 메타데이터 서비스가 꺼져 있습니다.

IMDSv1 및 IMDSv2를 사용하여 인스턴스 메타데이터를 검색하는 예

다음 예에서는 Linux 인스턴스에서 사용할 수 있는 명령을 제공합니다.

Example 인스턴스 메타데이터의 사용 가능한 버전 가져오기

이 예를 통해 이용 가능한 인스턴스 메타데이터 버전을 가져올 수 있습니다. 각 버전은 새 인스턴스 메타데이터 카테고리가 릴리스될 때 인스턴스 메타데이터 빌드를 참조합니다. 이전 버전의 구조 및 정보를 사용하는 스크립트인 경우 이전 버전을 사용할 수 있습니다.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://192.0.2.0/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
Dload  Upload  Total    Spent    Left  Speed
 100    56    100    56      0     0    3733    0  --:--:--
--:--:-- --:--:-- 3733
* Trying 192.0.2.0...
* TCP_NODELAY set
* Connected to 192.0.2.0 (192.0.2.0) port 80 (#0)
> GET / HTTP/1.1
> Host: 192.0.2.0
> User-Agent: curl/7.61.1
> Accept: */*
> X-aws-ec2-metadata-token:
MDAXcxNFLbAwJIYx8KzgNckcHTdxT4Tt69TzpKExlXKTULHIQnjEtXvD
>
* HTTP 1.0, assume close after body
< HTTP/1.0 200 OK
< Date: Mon, 12 Sep 2022 21:58:03 GMT
< Content-Length: 274
< Content-Type: text/plain
< Server: EC2ws
<
1.0
```

```
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
2016-06-30
2016-09-02
2018-03-28
2018-08-17
2018-09-24
2019-10-01
2020-10-27
2021-01-03
2021-03-23
* Closing connection 0
```

IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
```

```
2014-02-25
2014-11-05
2015-10-20
2016-04-19
2016-06-30
2016-09-02
2018-03-28
2018-08-17
2018-09-24
2019-10-01
2020-10-27
2021-01-03
2021-03-23
latest
```

Example 최고 수준 메타데이터 항목 가져오기

이 예는 최고 수준 메타데이터 항목을 가져옵니다. 최상위 메타데이터 항목에 대한 자세한 내용은 이 설명서의 [지원되는 인스턴스 메타데이터 및 사용자](#) 데이터를 참조하십시오.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://192.0.2.0/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v
http://192.0.2.0/latest/meta-data/
ami-id
hostname
instance-id
instance-type
local-hostname
local-ipv4
mac
network/
reservation-id
security-groups
```

IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/  
ami-id  
hostname  
instance-id  
instance-type  
local-hostname  
local-ipv4  
mac  
network/  
reservation-id  
security-groups
```

Example 최상위 메타데이터의 값 가져오기

다음 예에서는 이전 예에서 얻은 몇 가지 최상위 메타데이터 항목의 값을 가져옵니다. IMDSv2 요청에서는 토큰이 만료되지 않았다고 가정하고 이전 예제 명령에서 생성한 저장된 토큰을 사용합니다.

ami-id IMDSv2

```
curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/latest/meta-data/  
ami-id ami-0abcdef1234567890
```

ami-id IMDSv1

```
curl http://192.0.2.0/latest/meta-data/ami-id ami-0abcdef1234567890
```

reservation-id IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/  
latest/meta-data/reservation-id r-0efghijk987654321
```

reservation-id IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/reservation-id \
r-0efghijk987654321
```

local-hostname IMDSv2

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://192.0.2.0/
latest/meta-data/local-hostname ip-00-000-00-00
```

local-hostname IMDSv1

```
[ec2-user ~]$ curl http://192.0.2.0/latest/meta-data/local-hostname ip-00-000-00-00
```

Amazon EC2 호환 엔드포인트 사용

다음에서 Amazon Elastic Compute Cloud 호환(Amazon EC2) 엔드포인트의 개요를 찾을 수 있습니다. 엔드포인트를 통해 Amazon Machine Image(AMI)를 관리하고 Amazon EC2 호환 API 작업을 사용하여 인스턴스를 프로그래밍 방식으로 컴퓨팅할 수 있습니다.

주제

- [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#)
- [Snowcone에서 지원되지 않는 Amazon EC2 기능](#)
- [Snowcone에서 지원되는 아마존 EC2 호환 AWS CLI 명령](#)
- [지원되는 Amazon EC2 API 작업](#)

Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI

를 사용하여 AWS Snowcone 디바이스에 명령을 실행할 때 엔드포인트가 Amazon EC2 호환 엔드포인트가 되도록 지정할 수 있습니다. AWS CLI 다음과 같이 HTTPS 엔드포인트 또는 보안되지 않은 HTTP 엔드포인트를 사용할 수 있습니다.

HTTPS 보안 엔드포인트

```
aws ec2 describe-instances --endpoint https://192.0.2.0:8243 --ca-bundle path/to/certificate
```

HTTP 보안되지 않는 엔드포인트

```
aws ec2 describe-instances --endpoint http://192.0.2.0:8008
```

HTTPS 엔드포인트 8243을 사용하는 경우 전송 중 데이터는 암호화됩니다. 이 암호화는 Snowcone이 잠금 해제될 때마다 생성하는 인증서로 보장됩니다. 인증서가 생기면 이를 로컬 `ca-bundle.pem` 파일에 저장할 수 있습니다. 그런 다음, 다음 설명과 같이 인증서 경로를 포함하도록 AWS CLI 프로필을 구성할 수 있습니다.

인증서를 Amazon EC2 호환 엔드포인트에 연결하려면

1. Snowcone을 전원과 네트워크에 연결한 후 켭니다.
2. 디바이스의 잠금 해제가 완료되면 로컬 네트워크상의 IP 주소를 적어 둡니다.
3. 네트워크의 터미널에서 Snowcone을 ping할 수 있는지 확인합니다.
4. 터미널에서 `snowballEdge get-certificate` 명령을 실행합니다. 이 명령에 대한 자세한 내용은 [데이터를 전송하기 위해 인증서 가져오기](#)를 참조하십시오.
5. `snowballEdge get-certificate` 명령의 출력을 파일(예: `ca-bundle.pem`)에 저장합니다.
6. 터미널에서 다음 명령을 실행합니다.

```
aws configure set profile.snowcone.ca_bundle /path/to/ca-bundle.pem
```

절차를 완료한 후에는 이러한 로컬 자격 증명, 인증서 및 지정된 엔드포인트로 CLI 명령을 실행할 수 있습니다.

Snowcone에서 지원되지 않는 Amazon EC2 기능

Amazon EC2 호환 엔드포인트를 사용하면 Snowcone에서 Amazon EC2 호환 API 작업으로 AMI 및 컴퓨팅 인스턴스를 프로그래밍 방식으로 관리할 수 있습니다. 하지만 모든 기능 및 API 작업이 Snowcone 디바이스에서 사용할 수 있도록 지원되는 것은 아닙니다.

이 설명서에 지원되는 것으로 명시적으로 나열되지 않은 기능 또는 작업은 지원되지 않습니다. 예를 들어, 다음 Amazon EC2 작업은 Snowcone에서 사용할 수 있도록 지원되지 않습니다.

- [create-nat-gateway](#)

- [create-key-pair](#)

Snowcone에서 지원되는 아마존 EC2 호환 AWS CLI 명령

Amazon EC2 호환 엔드포인트를 통해 Snow Family 디바이스에서 컴퓨팅 인스턴스를 관리할 수 있습니다. 이 유형의 엔드포인트는 SDK의 많은 Amazon EC2 CLI 명령 및 작업을 지원합니다. AWS CLI [호출 대상 지정을 AWS CLI포함하여 설치 및 설정에 대한 자세한 내용은 사용 AWS 리전 설명서를 참조하십시오.AWS Command Line Interface](#)

Snowcone에서 지원되는 Amazon EC2 호환 AWS CLI 명령 목록

다음은 Snowcone 디바이스에서 지원되는 Amazon EC2의 AWS CLI 명령 및 옵션 하위 집합에 대한 설명을 확인할 수 있습니다. 아래에 나열되지 않은 명령이나 옵션은 지원되지 않습니다. 명령과 함께 지원되지 않는 옵션을 선언할 수 있습니다. 하지만 이러한 옵션은 무시됩니다.

- [associate-address](#) – 가상 IP 주소를 디바이스의 물리적 네트워크 인터페이스 3개 중 하나에 사용되는 인스턴스와 연결합니다.
 - --instance-id – 단일 sbe 인스턴스의 ID입니다.
 - --public-ip – 인스턴스에 액세스하기 위해 사용할 가상 IP 주소입니다.
- [attach-volume](#) — Amazon EBS 볼륨을 디바이스에서 중지되거나 실행 중인 인스턴스에 연결하고 지정된 AWS Snowcone 디바이스 이름을 가진 인스턴스에 노출합니다.
 - --device value - 디바이스의 이름입니다.
 - --instance-id - 대상 Amazon EC2 인스턴스의 ID입니다.
 - --volume-id value - EBS 볼륨의 ID입니다.
- [authorize-security-group-egress](#)— Snowcone 디바이스와 함께 사용할 하나 이상의 이그레스 규칙을 보안 그룹에 추가합니다. 특히 이 작업을 통해 인스턴스는 하나 이상의 대상 IPv4 CIDR 주소 범위로 트래픽을 보낼 수 있습니다. 자세한 설명은 [Snow 디바이스의 보안 그룹](#) 섹션을 참조하세요.
 - --group-id value - 보안 그룹의 ID입니다.
 - [--ip-permissions value] - 하나 이상의 IP 권한 집합입니다.
- [authorize-security-group-ingress](#)— 보안 그룹에 하나 이상의 인그레스 규칙을 추가합니다. authorize-security-group-ingress 호출 시 group-name 또는 group-id에 대한 값을 지정해야 합니다.
 - [--group-name value] - 보안 그룹의 이름입니다.
 - [--group-id value] - 보안 그룹의 ID입니다.
 - [--ip-permissions value] - 하나 이상의 IP 권한 집합입니다.

- [--protocol value] IP 프로토콜. 가능한 값은 tcp, udp 및 icmp입니다. "모든 프로토콜" 값이 지정되지 않는 한(-1) --port 인수가 필요합니다.
- [--port value] - TCP 또는 UDP의 경우, 허용할 포트 범위입니다. 이 값은 단일 정수 또는 범위(최소~최대)일 수 있습니다.

ICMP의 경우, 단일 정수 또는 범위(type~code). 여기서 type은 ICMP 유형 번호를 나타내고 code는 ICMP 코드 번호를 나타냅니다. -1의 값은 모든 ICMP 유형에 대한 모든 ICMP 코드를 나타냅니다. type에만 해당되는 -1의 값은 지정된 ICMP 유형에 대한 모든 ICMP 코드를 나타냅니다.

- [--cidr value] - CIDR IP 범위입니다.
- [create-launch-template](#) — 시작 템플릿을 생성합니다. 시작 템플릿에는 인스턴스를 시작하기 위한 파라미터가 포함되어 있습니다. RunInstances를 사용하여 인스턴스를 시작하면 요청에 시작 파라미터를 제공하는 대신 시작 템플릿을 지정할 수 있습니다. AWS Snowcone 디바이스당 최대 100개의 템플릿을 생성할 수 있습니다.
- launch-template-name string-- 시작 템플릿의 이름.
- -- launch-template-data structure — 시작 템플릿에 대한 정보. 다음과 같은 속성이 지원됩니다.
 - ImageId
 - InstanceType
 - SecurityGroupIds
 - TagSpecifications
 - UserData

JSON 구문:

```
{
  "ImageId": "string",
  "InstanceType": "sbe-c.large",
  "SecurityGroupIds": [
    "string",
    "...",
  ],
  "TagSpecifications": [
    {
      "ResourceType": "instance",
      "Tags": [
        {
          "Key": "Name",
```

```

        "Value": "Test"
      },
      {
        "Key": "Stack",
        "Value": "Gamma"
      }
    ]
  }
}

```

- `[--version-description string]` - 시작 템플릿의 첫 번째 버전에 대한 설명입니다.
- `--endpoint snowballEndpoint` - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [create-launch-template-version](#)— 시작 템플릿의 새 버전을 만듭니다. 새 버전을 기반으로 하는 시작 템플릿의 기존 버전을 지정할 수 있습니다. 시작 템플릿 버전은 생성한 순서대로 번호가 지정됩니다. 시작 템플릿 버전의 숫자는 지정, 변경 또는 대체할 수 없습니다. 각 시작 템플릿의 버전을 최대 100 개까지 생성할 수 있습니다.

요청에서 시작 템플릿 ID 또는 시작 템플릿 이름을 지정합니다.

- `-- launch-template-id string` — 시작 템플릿의 ID.
- `-- launch-template-name string` — 시작 템플릿의 이름.
- `-- launch-template-data structure` — 시작 템플릿에 대한 정보. 다음과 같은 속성이 지원됩니다.
 - ImageId
 - InstanceType
 - SecurityGroupIds
 - TagSpecifications
 - UserData

JSON 구문:

```

'{'
  "ImageId": "string",
  "InstanceType": "sbe-c.large",
  "SecurityGroupIds": ["string", ...],

```

```
"TagSpecifications":[{"ResourceType":"instance","Tags":
[{"Key":"Name","Value":"Test"},
{"Key":"Stack","Value":"Gamma"}]},
"UserData":"this is my user data"
}]'
```

- `--source-version string` - 새 버전을 기반으로 하는 시작 템플릿의 버전 번호입니다. 새 버전은 `launch-template-data`에 지정된 파라미터를 제외하고 소스 버전과 동일한 시작 파라미터를 상속합니다.
- `--version-description string` - 시작 템플릿의 첫 번째 버전에 대한 설명입니다.
- `--endpoint snowballEndpoint` - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [create-tags](#) - 지정된 리소스에 대해 하나 이상의 태그를 추가하거나 덮어씁니다. 각 리소스는 최대 50개의 태그를 보유할 수 있습니다. 각 태그는 키와 값(선택 사항)으로 구성됩니다. 태그 키는 리소스에 대해 고유해야 합니다. 다음 리소스가 지원됩니다.
 - AMI
 - Instance
 - 시작 템플릿
 - 보안 그룹
- [create-security-group](#)— Snowcone 디바이스에 보안 그룹을 생성합니다. 최대 50개의 보안 그룹을 생성할 수 있습니다. 보안 그룹을 생성할 때 친근한 이름을 직접 선택하여 지정합니다.
 - `--group-name value` - 보안 그룹의 이름입니다.
 - `--description value` - 보안 그룹의 설명입니다. 이는 참조용일 뿐입니다. 이 값의 최대 길이는 255 자입니다.
- [볼륨 생성](#) - 디바이스의 인스턴스에 연결할 수 있는 Amazon EBS 볼륨을 생성합니다. AWS Snowcone
 - `--sizevalue` - 볼륨 크기 (단위) 로 GiBs, 1GiB에서 1TB (GiBs1000) 사이일 수 있습니다.
 - `--snapshot-id value` - 볼륨이 생성되는 스냅샷의 ID입니다.
 - `--volume-type value` - 볼륨 유형입니다. 값을 지정하지 않을 경우 기본값은 `sbp1`입니다. 가능한 값은 다음을 포함합니다.
 - 마그네틱 볼륨인 `sbg1`
 - SSD 볼륨인 `sbp1`
 - `--tag-specification value` - 생성 중 볼륨에 적용되는 태그 목록입니다.

- [delete-launch-template](#)— 시작 템플릿을 삭제합니다. 시작 템플릿을 삭제하면 모든 버전이 삭제됩니다.

요청에서 시작 템플릿 ID 또는 시작 템플릿 이름을 지정합니다.

- -- launch-template-id string — 시작 템플릿의 ID.
- -- launch-template-name string — 시작 템플릿의 이름.
- --endpoint snowballEndpoint - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [delete-launch-template-version](#)— 시작 템플릿의 버전을 하나 이상 삭제합니다. 시작 템플릿의 기본 버전은 삭제할 수 없으며, 먼저 다른 버전을 기본 버전으로 할당해야 합니다. 기본 버전이 시작 템플릿의 유일한 버전인 경우 delete-launch-template 명령을 사용하여 전체 시작 템플릿을 삭제합니다.

요청에서 시작 템플릿 ID 또는 시작 템플릿 이름을 지정합니다.

- -- launch-template-id string — 시작 템플릿의 ID.
- -- launch-template-name string — 시작 템플릿의 이름.
- --versions(list) "string" "string" - 삭제할 하나 이상의 시작 템플릿 버전에 대한 버전 번호입니다.
- --endpoint snowballEndpoint - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다.
- [delete-security-group](#)— 보안 그룹을 삭제합니다.

인스턴스와 연결된 보안 그룹이나 다른 보안 그룹에서 참조하는 보안 그룹의 삭제를 시도하는 경우 DependencyViolation으로 작업이 실패합니다.

- --group-name value – 보안 그룹의 이름입니다.
- --description value – 보안 그룹의 설명입니다. 이는 참조용일 뿐입니다. 이 값의 최대 길이는 255 자입니다.
- [delete-tags](#) - 지정된 리소스(AMI, 컴퓨팅 인스턴스, 시작 템플릿 또는 보안 그룹)에서 지정된 태그 세트를 삭제합니다.
- [delete-volume](#) - 지정된 Amazon EBS 볼륨을 삭제합니다. 이때 볼륨은 available 상태(인스턴스에 연결되지 않음)여야 합니다.
- --volume-id value - 볼륨의 ID입니다.

- [describe-address](#) - 디바이스에서 동일한 수의 sbe 인스턴스와 관련된 하나 이상의 가상 IP 주소를 설명합니다.
 - --public-ips - 인스턴스와 연결된 한 개 이상의 가상 IP 주소입니다.
- [describe-images](#) - 사용 가능한 하나 이상의 이미지(AMI)를 설명합니다. 사용 가능한 이미지는 작업 생성 중에 Snowcone 디바이스에 추가됩니다.
 - --image-id - AMI의 Snowcone AMI ID입니다.
- [describe-instance-attribute](#)— 지정된 인스턴스의 지정된 속성을 설명합니다. 한 번에 하나의 속성만 지정할 수 있습니다. 다음과 같은 속성이 지원됩니다.
 - instanceInitiatedShutdownBehavior
 - instanceType
 - userData
- [describe-instances](#) - 하나 이상의 인스턴스를 설명합니다. 응답은 인스턴스에 할당된 보안 그룹을 반환합니다.
 - --instance-ids - 디바이스에서 중지된 하나 이상의 sbe 인스턴스 ID입니다.
 - --page-size - 직접 호출을 받아들일 각 페이지의 크기입니다. 이 값은 명령의 출력을 통해 반환되는 항목의 수에는 영향을 미치지 않습니다. 페이지 크기를 더 작게 설정하면 디바이스에 대한 추가 호출이 이루어져 각 호출에서 더 적은 항목을 가져옵니다. 따라서 호출이 시간 초과되는 것을 방지하는 데 도움이 됩니다.
 - --max-items - 명령의 출력을 통해 반환될 항목의 총 수입니다. 사용 가능한 총 항목 수가 지정된 값을 초과하는 경우 명령의 출력에 NextToken이 제공됩니다. 페이지 매김을 재개하려면 후속 명령의 starting-token 인수에 NextToken 값을 제공합니다.
 - --starting-token - 페이지 매김을 시작할 위치를 지정하기 위한 토큰입니다. 이 토큰은 이전에 잘린 응답에서 도출된 NextToken 값입니다.
- [describe-launch-templates](#)— 하나 이상의 시작 템플릿에 대해 설명합니다. describe-launch-templates 명령은 페이지 매김 작업입니다. 다중 호출을 생성하여 결과의 전체 데이터 세트를 검색할 수 있습니다.

요청에서 시작 템플릿 ID 또는 시작 템플릿 이름을 지정합니다.

- -- launch-template-ids (목록) "string" "string" — 시작 템플릿의 ID 목록입니다.
- -- launch-template-names (목록) "string" "string" — 시작 템플릿의 이름 목록입니다.
- --page-size - 직접 호출을 받아들일 각 페이지의 크기입니다. 이 값은 명령의 출력을 통해 반환되는 항목의 수에는 영향을 미치지 않습니다. 페이지 크기를 더 작게 설정하면 디바이스에 대한 추가 호출이 이루어져 각 호출에서 더 적은 항목을 가져옵니다. 따라서 호출이 시간 초과되는 것을 방지하는 데 도움이 됩니다.

- `--max-items` - 명령의 출력을 통해 반환될 항목의 총 수입니다. 사용 가능한 총 항목 수가 지정된 값을 초과하는 경우 명령의 출력에 `NextToken`이 제공됩니다. 페이지 매김을 재개하려면 후속 명령의 `starting-token` 인수에 `NextToken` 값을 제공합니다.
- `--starting-token` - 페이지 매김을 시작할 위치를 지정하기 위한 토큰입니다. 이 토큰은 이전에 잘린 응답에서 도출된 `NextToken` 값입니다.
- `--endpoint snowballEndpoint` - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [describe-launch-template-versions](#)— 지정된 시작 템플릿의 하나 이상의 버전을 설명합니다. 모든 버전, 개별 버전 또는 특정 버전 범위를 설명할 수 있습니다. `describe-launch-template-versions` 명령은 페이지 매김 작업입니다. 다중 호출을 생성하여 결과의 전체 데이터 세트를 검색할 수 있습니다.

요청에서 시작 템플릿 ID 또는 시작 템플릿 이름을 지정합니다.

- `-- launch-template-id string` — 시작 템플릿의 ID.
- `-- launch-template-name string` — 시작 템플릿의 이름.
- `[--versions (list) "string" "string"]` - 삭제할 하나 이상의 시작 템플릿 버전에 대한 버전 번호입니다.
- `[--min-version string]` - 시작 템플릿 버전을 설명하는 최소 버전 번호입니다.
- `[--max-version string]` - 시작 템플릿 버전을 설명하는 최대 버전 번호입니다.
- `--page-size` - 직접 호출을 받아들일 각 페이지의 크기입니다. 이 값은 명령의 출력을 통해 반환되는 항목의 수에는 영향을 미치지 않습니다. 페이지 크기를 더 작게 설정하면 디바이스에 대한 추가 호출이 이루어져 각 호출에서 더 적은 항목을 가져옵니다. 따라서 호출이 시간 초과되는 것을 방지하는 데 도움이 됩니다.
- `--max-items` - 명령의 출력을 통해 반환될 항목의 총 수입니다. 사용 가능한 총 항목 수가 지정된 값을 초과하는 경우 명령의 출력에 `NextToken`이 제공됩니다. 페이지 매김을 재개하려면 후속 명령의 `starting-token` 인수에 `NextToken` 값을 제공합니다.
- `--starting-token` - 페이지 매김을 시작할 위치를 지정하기 위한 토큰입니다. 이 토큰은 이전에 잘린 응답에서 도출된 `NextToken` 값입니다.
- `--endpoint snowballEndpoint` - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [describe-security-groups](#)— 하나 이상의 보안 그룹을 설명합니다.

`describe-security-groups` 명령은 페이지 매김 작업입니다. 다중 API 직접 호출을 발행하여 결과의 전체 데이터 세트를 검색할 수 있습니다.

- `[--group-name value]` - 보안 그룹의 이름입니다.
- `[--group-id value]` - 보안 그룹의 ID입니다.
- `[--page-size value]` — AWS 서비스 콜에 포함할 각 페이지의 크기입니다. 이 크기는 명령의 출력에서 반환된 항목의 수에 영향을 주지 않습니다. 페이지 크기를 작게 설정하면 AWS 서비스에 대한 직접 호출이 많아지고 각 직접 호출의 항목에서 검색 항목이 줄어듭니다. 이 방법을 사용하면 AWS 서비스 호출 시간이 초과되는 것을 방지할 수 있습니다. 사용 예시는 AWS Command Line Interface 사용 설명서의 [페이지 매김](#) 섹션을 참조하세요.
- `[--max-items value]` - 명령의 출력을 통해 반환될 항목의 총 수입니다. 사용 가능한 총 항목 수가 지정된 값을 초과하는 경우 명령의 출력에 NextToken이 제공됩니다. 페이지 매김을 재개하려면 후속 명령의 `starting-token` 인수에 NextToken 값을 제공합니다. NextToken 외부에서 직접 AWS CLI 응답 요소를 사용하면 안 됩니다. 사용 예시는 AWS Command Line Interface 사용 설명서의 [페이지 매김](#) 섹션을 참조하세요.
- `[--starting-token value]` - 페이지 매김이 시작되는 곳을 지정하는 토큰입니다. 이 토큰은 이전에 잘린 응답에서 도출된 NextToken 값입니다. 사용 예시는 AWS Command Line Interface 사용 설명서의 [페이지 매김](#) 섹션을 참조하세요.
- [describe-tags](#) - 지정된 리소스(image, instance 또는 보안 그룹)에 대한 하나 이상의 태그를 설명합니다. 이 명령에서는 다음 필터가 지원됩니다.
 - `launch-template`
 - `resource-id`
 - `resource-type` - image 또는 instance입니다.
 - 키
 - 값
- [describe-volumes](#) - 지정된 EBS 볼륨을 설명합니다.
 - `[--max-items value]` - 명령의 출력을 통해 반환될 항목의 총 수입니다. 사용 가능한 총 항목 수가 지정된 값을 초과하는 경우 명령의 출력에 NextToken이 제공됩니다. 페이지 매김을 재개하려면 후속 명령의 `starting-token` 인수에 NextToken 값을 제공합니다.
 - `[--starting-token value]` - 페이지 매김이 시작되는 곳을 지정하는 토큰입니다. 이 토큰은 이전에 잘린 응답에서 도출된 NextToken 값입니다.
 - `[--volume-ids value]` - 하나 이상의 볼륨 ID입니다.
- [detach-volume](#) - 중지되거나 실행 중인 인스턴스에서 Amazon EBS 볼륨을 분리합니다.

- `[--device value]` - 디바이스의 이름입니다.
- `[--instance-id]` - 대상 Amazon EC2 인스턴스의 ID입니다.
- `--volume-id value` - 볼륨의 ID입니다.
- [disassociate-address](#) - 가상 IP 주소를 연결된 인스턴스에서 분리합니다.
 - `--public-ip` - 인스턴스에서 분리하려고 하는 가상 IP 주소입니다.
- [get-launch-template-data](#)— 지정된 인스턴스의 구성 데이터를 검색합니다. 이 데이터를 사용하여 시작 템플릿을 생성할 수 있습니다.
 - `--instance-id` - 단일 sbe 인스턴스의 ID입니다.
 - `--endpoint snowballEndpoint` - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [modify-launch-template](#)— 시작 템플릿을 수정합니다. 기본 버전으로 설정할 시작 템플릿의 버전을 지정할 수 있습니다. 시작 템플릿 버전을 지정하지 않고 인스턴스를 시작하면 시작 템플릿의 기본 버전이 적용됩니다.

요청에서 시작 템플릿 ID 또는 시작 템플릿 이름을 지정합니다.

- `-- launch-template-id string` — 시작 템플릿의 ID.
- `-- launch-template-name string` — 시작 템플릿의 이름.
- `--default-version string` - 기본 버전으로 설정할 시작 템플릿의 버전 번호입니다.
- `--endpoint snowballEndpoint` - Amazon EC2 API 작업을 사용하여 프로그래밍 방식으로 컴퓨팅 인스턴스를 관리할 수 있는 값입니다. 자세한 설명은 [Amazon EC2 호환 엔드포인트를 엔드포인트로 지정 AWS CLI](#) 섹션을 참조하세요.
- [modify-instance-attribute](#)— 지정된 인스턴스의 속성을 수정합니다. 다음과 같은 속성이 지원됩니다.
 - `instanceInitiatedShutdownBehavior`
 - `userData`
- [revoke-security-group-egress](#)— 보안 그룹에서 하나 이상의 송신 규칙을 제거합니다.
 - `[--group-id value]` - 보안 그룹의 ID입니다.
 - `[--ip-permissions value]` - 하나 이상의 IP 권한 집합입니다.
- [revoke-security-group-ingress](#)— 보안 그룹에 대한 하나 이상의 인그레스 규칙을 취소합니다. `revoke-security-group-ingress` 호출 시 `group-name` 또는 `group-id`에 대한 값을 지정해야 합니다.
 - `[--group-name value]` - 보안 그룹의 이름입니다.

- [--group-id value] - 보안 그룹의 ID입니다.
- [--ip-permissions value] - 하나 이상의 IP 권한 집합입니다.
- [--protocol value] IP 프로토콜. 가능한 값은 tcp, udp 및 icmp입니다. "모든 프로토콜" 값이 지정되지 않는 한(-1) --port 인수가 필요합니다.
- [--port value] - TCP 또는 UDP의 경우, 허용할 포트 범위입니다. 단일 정수 또는 범위(최소~최대 값)입니다.

ICMP의 경우, 단일 정수 또는 범위(type~code). 여기서 type은 ICMP 유형 번호를 나타내고 code는 ICMP 코드 번호를 나타냅니다. -1의 값은 모든 ICMP 유형에 대한 모든 ICMP 코드를 나타냅니다. type에만 해당되는 -1의 값은 지정된 ICMP 유형에 대한 모든 ICMP 코드를 나타냅니다.

- [--cidr value] - CIDR IP 범위입니다.
- [run-instances](#) - AMI에 대해 Snowcone AMI ID를 사용하여 많은 컴퓨팅 인스턴스를 시작합니다.

Note

컴퓨팅 인스턴스의 크기 및 유형에 따라 Snowcone 디바이스에서 컴퓨팅 인스턴스를 시작하는 데 최대 한 시간 반이 소요될 수 있습니다.

- [-- block-device-mappings (list)] — 블록 디바이스 매핑 항목. DeleteOnTermination, VolumeSize, VolumeType 파라미터를 지원합니다. 부트 볼륨은 sbg1 유형이어야 합니다.

이 명령어의 JSON 구문은 다음과 같습니다.

```
{
  "DeviceName": "/dev/sdh",
  "Ebs":
  {
    "DeleteOnTermination": true|false,
    "VolumeSize": 100,
    "VolumeType": "sbp1"|"sbg1"
  }
}
```

- --count - 시작할 인스턴스의 개수입니다. 단일 숫자가 제공된 경우 최소 개수를 시작하는 것으로 가정합니다(1로 기본 설정). 범위가 min:max 형식으로 제공될 경우 첫 번째 수는 시작할 인스턴스의 최소 수로 해석하고 두 번째 수는 시작할 인스턴스의 최대 수로 해석합니다.

- `--image-id - describe-images`를 직접 호출하여 얻을 수 있는 AMI의 Snowcone AMI ID입니다. AMI는 인스턴스를 시작하는 데 필요합니다.
- `-- InstanceInitiatedShutdownBehavior` — 기본적으로 `shutdown` 또는 `poweroff`와 같은 명령을 사용하여 인스턴스 종료를 시작하면 인스턴스가 중지됩니다. 인스턴스가 중지되지 않고 종료되도록 이 동작을 변경할 수 있습니다. `stop` 및 `terminate` 파라미터가 지원됩니다. 기본값은 `stop`입니다. 자세한 정보는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스가 시작하는 종료 동작 변경](#) 섹션을 참조하세요.
- `--instance-type - sbe` 인스턴스 유형입니다.
- `--launch-template structure` - 인스턴스를 시작하는 데 사용할 시작 템플릿입니다. `run-instances` 명령에서 파라미터를 지정하면 시작 템플릿에 있는 동일한 파라미터가 재정의됩니다. 시작 템플릿의 이름이나 ID는 지정할 수 있지만, 둘 다 지정할 수는 없습니다.

```
{
  "LaunchTemplateId": "string",
  "LaunchTemplateName": "string",
  "Version": "string"
}
```

- `security-group-ids` -- 하나 이상의 보안 그룹 ID. 를 사용하여 보안 그룹을 생성할 수 [CreateSecurityGroup](#) 있습니다. 값이 제공되지 않는 경우 기본 보안 그룹의 ID가 생성된 인스턴스에 할당됩니다.
- `--tag-specifications` - 시작 시 리소스에 적용되는 태그입니다. 시작 시에만 인스턴스에 태그를 지정할 수 있습니다. 지정된 태그는 시작 시 생성된 모든 인스턴스에 적용됩니다. 생성된 후에 리소스에 태그를 지정하려면 `create-tags`를 사용합니다.
- `--user-data` - 인스턴스에 사용할 수 있는 사용자 데이터입니다. 를 사용하는 AWS CLI 경우 base64 인코딩이 자동으로 수행되며 파일에서 텍스트를 로드할 수 있습니다. 그렇지 않으면 base64 인코딩 형식의 텍스트를 입력해야 합니다.
- [start-instances](#) - 이전에 중지했던 sbe 인스턴스를 시작합니다. 인스턴스에 연결된 모든 자원은 시작 및 중지를 통해 지속되지만 인스턴스가 해지되면 삭제됩니다.
 - `--instance-ids` - 디바이스에서 중지된 하나 이상의 sbe 인스턴스 ID입니다.
- [stop-instances](#) - 실행 중인 sbe 인스턴스를 중지합니다. 인스턴스에 연결된 모든 자원은 시작 및 중지를 통해 지속되지만 인스턴스가 해지되면 삭제됩니다.
 - `--instance-ids` - 디바이스에서 중지된 하나 이상의 sbe 인스턴스 ID입니다.

- [terminate-instances](#) - 하나 이상의 인스턴스를 종료합니다. 이 작업은 멱등성을 갖기 때문에 인스턴스를 2회 이상 종료해도 모든 호출이 성공적으로 처리됩니다. 인스턴스에 연결된 모든 리소스는 시작 및 중지 사태가 발생해도 지속되지만 인스턴스가 종료되면 데이터가 삭제됩니다.

Note

기본적으로 shutdown 또는 poweroff 등의 명령을 사용하여 인스턴스에서 종료를 시작하면 인스턴스가 중지됩니다. 그러나 이러한 명령이 인스턴스를 종료하도록 InstanceInitiatedShutdownBehavior 속성을 사용하여 이 동작을 변경할 수 있습니다. 자세한 정보는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스가 시작하는 종료 동작 변경](#) 섹션을 참조하세요.

- --instance-ids - 디바이스에서 종료된 하나 이상의 sbe 인스턴스 ID입니다. 이러한 인스턴스에 대해 저장된 연결된 데이터는 모두 손실됩니다.

지원되는 Amazon EC2 API 작업

다음에서 Amazon EC2 API 참조의 설명 링크와 함께 Snowcone 디바이스에서 사용할 수 있는 Amazon EC2 API 작업을 찾을 수 있습니다. Amazon EC2 API 호출에는 서명 버전 4(SigV4) 서명이 필요합니다. AWS CLI 또는 AWS SDK를 사용하여 이러한 API 호출을 수행하는 경우 SigV4 서명이 자동으로 처리됩니다. 그렇지 않은 경우 자체 SigV4 서명 솔루션을 구현해야 합니다.

- [AssociateAddress](#)— 엘라스틱 IP 주소를 인스턴스 또는 네트워크 인터페이스와 연결합니다.
- [AttachVolume](#)— 다음과 같은 요청 파라미터가 지원됩니다.
 - Device
 - InstanceId
 - VolumeId
- [AuthorizeSecurityGroupEgress](#)— Snowcone 디바이스와 함께 사용할 하나 이상의 이그레스 규칙을 보안 그룹에 추가합니다. 특히 이 작업을 통해 인스턴스는 하나 이상의 대상 IPv4 CIDR 주소 범위로 트래픽을 보낼 수 있습니다.
- [AuthorizeSecurityGroupIngress](#)— 보안 그룹에 하나 이상의 인그레스 규칙을 추가합니다. AuthorizeSecurityGroupIngress 호출할 때는 GroupName 또는 GroupId 에 대한 값을 지정해야 합니다.
- [CreateVolume](#)— 다음과 같은 요청 파라미터가 지원됩니다.

- SnapshotId
- Size
- VolumeType
- TagSpecification.N
- [CreateLaunchTemplate](#)— 다음과 같은 요청 매개변수가 지원됩니다.
 - ImageId
 - InstanceType
 - SecurityGroupIds
 - TagSpecifications
 - UserData
- [CreateLaunchTemplateVersion](#)
- [CreateTags](#)— 다음과 같은 요청 매개변수가 지원됩니다.
 - AMI
 - Instance
 - Launch template
 - Security group
- [CreateSecurityGroup](#)— Snowcone에 보안 그룹을 생성합니다. 최대 50개의 보안 그룹을 생성할 수 있습니다. 보안 그룹을 생성할 때 친근한 이름을 직접 선택하여 지정합니다.
- [DeleteLaunchTemplate](#)
- [DeleteLaunchTemplateVersions](#)
- [DeleteSecurityGroup](#)— 보안 그룹을 삭제합니다. 인스턴스와 연결된 보안 그룹이나 다른 보안 그룹에서 참조하는 보안 그룹의 삭제를 시도하는 경우 DependencyViolation으로 작업이 실패합니다.
- [DeleteTags](#)— 지정된 리소스 세트에서 지정된 태그 세트를 삭제합니다.
- [DeleteVolume](#)— 다음과 같은 요청 파라미터가 지원됩니다.
 - VolumeId
- [DescribeAddresses](#)
- [DescribeImages](#)
- [DescribeInstanceAttribute](#)— 지원되는 속성은 다음과 같습니다.
 - instanceType
 - userData

- [DescribeLaunchTemplates](#)
- [DescribeLaunchTemplateVersions](#)
- [DescribeInstances](#)
- [DescribeSecurityGroups](#)— 하나 이상의 보안 그룹을 설명합니다. DescribeSecurityGroups 페이지 매김 작업입니다. 다중 API 직접 호출을 발행하여 결과의 전체 데이터 세트를 검색할 수 있습니다.
- [DescribeTags](#)— 이 명령을 사용하면 다음과 같은 필터가 지원됩니다.
 - resource-id
 - resource-type - AMI 또는 컴퓨팅 인스턴스만 해당합니다.
 - key
 - value
- [DescribeVolume](#)— 다음과 같은 요청 파라미터가 지원됩니다.
 - MaxResults
 - NextToken
 - VolumeId.N
- [DetachVolume](#)— 다음과 같은 요청 매개변수가 지원됩니다.
 - Device
 - InstanceId
 - VolumeId
- [DisassociateAddress](#)
- [GetLaunchTemplateData](#)
- [ModifyLaunchTemplate](#)
- [ModifyInstanceAttribute](#)— userData 속성만 지원됩니다.
- [RevokeSecurityGroupEgress](#)— 보안 그룹에서 하나 이상의 송신 규칙을 제거합니다.
- [RevokeSecurityGroupIngress](#)— 보안 그룹에 대한 하나 이상의 인그레스 규칙을 취소합니다. RevokeSecurityGroupIngress 호출할 때는 또는 에 대한 값을 지정해야 합니다. group-name group-id
- [RunInstances](#) -

Note

컴퓨팅 인스턴스의 크기 및 유형에 따라 Snowcone에서 컴퓨팅 인스턴스를 시작하는 데 최대한 시간 반이 소요될 수 있습니다.

- [StartInstances](#)
- [StopInstances](#)— 중지된 인스턴스와 연결된 리소스는 유지됩니다. 이러한 리소스를 확보하기 위해 인스턴스를 종료할 수 있습니다. 하지만 연결된 모든 데이터가 삭제됩니다.
- [TerminateInstances](#)

시작 템플릿을 사용하여 Amazon EC2 호환 인스턴스 자동 시작

시작 템플릿과 Snowball Edge 클라이언트 시작 구성 명령을 사용하여 AWS Snowcone 디바이스에서 Amazon EC2 호환 인스턴스를 자동으로 시작할 수 있습니다. 인스턴스가 종료되면 자동 시작이 시작되지만 인스턴스를 삭제하거나 인스턴스의 자동 시작 구성을 업데이트하면 자동 시작 기능이 새 인스턴스를 시작합니다.

시작 템플릿에는 Snowcone에 Amazon EC2 호환 인스턴스를 생성하는 데 필요한 구성 정보가 포함되어 있습니다. 시작 템플릿을 사용하여 시작 파라미터를 저장할 수 있으므로 Snowcone에서 EC2 호환 인스턴스를 시작할 때마다 시작 파라미터를 지정할 필요가 없습니다.

Snowcone에서 자동 시작 구성을 사용하면 Amazon EC2 호환 인스턴스를 시작할 파라미터를 구성할 수 있습니다. Snowcone이 구성되면 재부팅하고 잠금을 해제할 때 자동 시작 구성을 사용하여 지정한 파라미터로 인스턴스를 시작합니다. 자동 시작 구성을 사용하여 시작한 인스턴스가 중지되면 디바이스를 잠금 해제할 때 인스턴스가 실행을 시작합니다.


Note

자동 시작 구성을 처음 구성한 후에는 디바이스를 다시 시작하십시오. 이후의 모든 인스턴스 시작 (계획된 재부팅 또는 예상치 못한 재부팅 이후) 은 디바이스 잠금이 해제된 후 자동으로 실행됩니다. AWS Snowcone

시작 템플릿은 해당 인스턴스를 시작할 때 Amazon EC2 호환 인스턴스에 대한 Amazon Machine Image(AMI) ID, 인스턴스 유형, 사용자 데이터, 보안 그룹 및 태그를 지정할 수 있습니다.

Snowcone에서 EC2 호환 인스턴스를 자동으로 시작하려면 다음 단계를 수행합니다.


1. AWS Snowcone 디바이스를 주문할 때 컴퓨팅 인스턴스가 포함된 Snow Family 디바이스를 주문하는 작업을 생성하십시오. 자세한 설명은 [컴퓨팅 인스턴스가 있는 작업 생성](#) 섹션을 참조하세요.
2. Snowcone을 받은 후 잠금을 해제합니다.
3. EC2 API 명령 `aws ec2 create-launch-template`을 사용하여 시작 템플릿을 생성합니다. 자세한 설명은 [Snowcone에서 지원되는 Amazon EC2 호환 AWS CLI 명령 목록](#) 섹션을 참조하세요.

 Note

Amazon EC2 엔드포인트는 디바이스 엔드포인트입니다.

4. Snowball Edge 클라이언트 명령 `snowballEdge create-autostart-configuration`을 사용하여 EC2 호환 시작 템플릿을 네트워크 구성에 바인딩합니다. 자세한 설명은 [Amazon EC2 호환 인스턴스를 자동 시작하는 시작 구성 생성](#) 섹션을 참조하세요.
5. AWS Snowcone 기기를 재부팅한 다음 잠금을 해제하세요. EC2 호환 인스턴스는 시작 템플릿과 Snowcone 클라이언트 명령 `create-autostart-configuration`에 지정된 속성을 사용하여 자동으로 시작됩니다.

실행 중인 인스턴스 상태를 확인하려면 EC2 API 명령 `describe-autostart-configurations`를 사용하십시오.

 Note

템플릿 실행을 AWS Snowball 지원하는 콘솔 또는 작업 관리 API는 없습니다. EC2 및 Snowball Edge 클라이언트 CLI 명령을 사용하여 AWS Snowcone 디바이스에서 EC2 호환 인스턴스를 자동으로 시작합니다.

Amazon EC2 호환 인스턴스에 블록 스토리지 사용

Snowcone의 블록 스토리지는 애플리케이션 요구 사항에 따라 블록 스토리지를 추가하거나 제거할 수 있도록 합니다. Amazon EC2 호환 인스턴스에 연결된 볼륨은 스토리지 볼륨으로 표시되며, 인스턴스 수명과 관계없이 지속됩니다. 익숙한 Amazon EBS API를 사용해 블록 스토리지를 관리할 수 있습니다.

Amazon EC2 엔드포인트를 사용하면 일부 EBS 명령어가 지원됩니다. 지원되는 명령어에는 `attach-volume`, `create-volume`, `delete-volume`, `detach-volume`, `describe-volumes`가 포함됩니

다. 이런 명령어에 대한 자세한 내용은 [Snowcone에서 지원되는 Amazon EC2 호환 AWS CLI 명령 목록](#) 섹션을 참조하십시오.

⚠ Important

볼륨을 분리하기 전에 운영 체제 안에서 디바이스의 파일 시스템을 모두 탑재 해제합니다. 해제하지 않으면 잠재적으로 데이터가 손실될 수 있습니다.

다음으로 Amazon EBS 볼륨 할당량 및 AWS Snowcone 디바이스의 Amazon EBS 볼륨과 클라우드의 Amazon EBS 볼륨 차이를 설명합니다.

- Amazon EBS 볼륨은 볼륨을 호스팅하는 AWS Snowcone 디바이스에서 실행되는 EC2 호환 인스턴스에서만 사용할 수 있습니다.
- 볼륨 유형은 용량 최적화 HDD(sbg1) 또는 성능 최적화 SSD(sbp1)로 제한됩니다. 기본 볼륨 유형은 sbg1입니다.
- Amazon EC2 루트 볼륨은 항상 IDE 드라이버를 사용합니다. 추가 Amazon EBS 볼륨은 가능한 경우 Virtio 드라이버를 우선 사용합니다. Virtio 드라이버를 이용할 수 없는 경우 SBE는 IDE 드라이버를 기본으로 사용합니다. 성능이 우수하기 때문에 Virtio 드라이버의 사용을 추천합니다.
- Amazon EBS 볼륨을 생성할 때 encrypted 파라미터는 지원되지 않습니다. 그러나 디바이스의 모든 데이터는 기본적으로 암호화됩니다.
- 볼륨 크기는 1GB에서 10TB까지 가능합니다.
- 하나의 EC2 호환 인스턴스에 최대 10개의 Amazon EBS 볼륨을 연결할 수 있습니다.
- AWS Snowcone 디바이스에 추가할 수 있는 Amazon EBS 볼륨 수에 대한 공식적인 제한은 없습니다. 하지만 Amazon EBS의 총 볼륨 용량은 AWS Snowcone 디바이스의 사용 가능한 공간에 따라 제한됩니다.

Snow 디바이스의 보안 그룹

보안 그룹은 하나 이상의 인스턴스에 대한 트래픽을 제어하는 가상 방화벽 역할을 합니다. 인스턴스를 시작할 때 하나 이상의 보안 그룹을 인스턴스와 연결합니다. 연결된 인스턴스와 트래픽을 주고받을 수 있게 하는 규칙을 각 보안 그룹에 추가할 수 있습니다. 자세한 정보는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 인스턴스용 Amazon EC2 보안 그룹](#)을 참조하세요.

Snowcone 디바이스의 보안 그룹은 AWS 클라우드의 보안 그룹과 비슷합니다. Virtual Private Cloud(VPC)는 Snowcone 디바이스에서 지원되지 않습니다.

다음에서 Snowcone 보안 그룹과 EC2-VPC 보안 그룹의 기타 차이를 확인할 수 있습니다.

- 각 Snowcone의 보안 그룹 수는 50개로 제한됩니다.
- 기본 보안 그룹은 인바운드와 아웃바운드 트래픽을 모두 허용합니다.
- 로컬 인스턴스 간의 트래픽은 프라이빗 인스턴스 IP 주소 또는 퍼블릭 IP 주소를 사용할 수 있습니다. 예를 들어 SSH를 사용하여 인스턴스 A에서 인스턴스 B로 연결하려고 합니다. 이 경우 보안 그룹이 이 트래픽을 허용하면 대상 IP 주소는 인스턴스 B의 퍼블릭 IP 또는 프라이빗 IP 주소일 수 있습니다.
- AWS CLI 작업 및 API 호출에 대해 나열된 파라미터만 지원됩니다. 일반적으로 이는 EC2-VPC 인스턴스에서 지원되는 파라미터의 하위 집합입니다.

지원되는 AWS CLI 작업에 대한 자세한 내용은 [을 참조하십시오](#) [Snowcone에서 지원되는 Amazon EC2 호환 AWS CLI 명령 목록](#). 지원되는 API 작업에 대한 자세한 내용은 [지원되는 Amazon EC2 API 작업](#) 섹션을 참조하십시오.

지원되는 인스턴스 메타데이터 및 사용자 데이터

인스턴스 메타데이터는 실행 중인 인스턴스를 구성 또는 관리하는 데 사용될 수 있는 인스턴스 관련 데이터입니다. Snowcone에서는 컴퓨팅 인스턴스에 대해 인스턴스 메타데이터 범주의 하위 집합을 지원합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 메타데이터 및 사용자 데이터](#)를 참조하십시오.

다음과 같은 범주가 지원됩니다. 다른 카테고리를 사용하여 404 오류 메시지를 반환합니다.

Snowcone에서 지원되는 인스턴스 메타데이터 범주

데이터	설명
ami-id	인스턴스를 시작하기 위해 사용된 AMI ID.
hostname	인스턴스의 프라이빗 IPv4 DNS 호스트 이름.
instance-id	이 인스턴스의 ID.
instance-type	인스턴스 유형.
local-hostname	인스턴스의 프라이빗 IPv4 DNS 호스트 이름.
local-ipv4	인스턴스의 프라이빗 IPv4 주소.

데이터	설명
mac	인스턴스의 미디어 액세스 제어(MAC) 주소.
network/interfaces/macs/ <i>mac</i> /local-hostname	인터페이스의 로컬 호스트이름.
network/interfaces/macs/ <i>mac</i> /local-ipv4s	프라이빗 IPv4 주소는 인터페이스와 연결됩니다.
network/interfaces/macs/ <i>mac</i> /mac	인스턴스의 MAC 주소.
network/interfaces/macs/ <i>mac</i> /public-ipv4s	탄력적 IP 주소는 인터페이스와 연결됩니다.
public-ipv4	퍼블릭 IPv4 주소.
public-keys/0/openssh-key	퍼블릭 키. 시작 시에 인스턴스가 제공된 경우에만 사용할 수 있습니다.
reservation-id	예약 ID:
userData	시작 시 인스턴스에 명령을 전송하기 위한 Shell 스크립트입니다.

Snowcone에서 지원되는 인스턴스 동적 데이터 범주

데이터	설명
instance-identity/document	인스턴스 속성이 포함된 JSON입니다. instanceId , imageId, privateIp 및 instanceType 에만 값이 있고 반환된 기타 속성은 null입니다. 자세한 내용은 Linux용 Amazon EC2 사용 설명서의 인스턴스 자격 증명 문서 섹션을 참조하세요.

Snowcone 컴퓨팅 인스턴스의 사용자 데이터 변경

Snowcone 디바이스에서 컴퓨팅 인스턴스용 셸 스크립트에 사용할 수 있도록 사용자 데이터가 지원됩니다. Shell 스크립트를 사용하면 시작 시 인스턴스에 명령을 전송할 수 있습니다. modify-instance-attribute AWS CLI 명령 또는 ModifyInstanceAttribute API 작업을 사용하여 사용자 데이터를 변경할 수 있습니다.

사용자 데이터 변경하기

1. stop-instances AWS CLI 명령으로 컴퓨팅 인스턴스를 중지합니다.
2. modify-instance-attribute AWS CLI 명령을 사용하여 userData 속성을 수정합니다.
3. start-instances AWS CLI 명령을 사용하여 컴퓨팅 인스턴스를 다시 시작합니다.

컴퓨팅 인스턴스에서는 Shell 스크립트만 지원됩니다. Snowcone에서 실행 중인 컴퓨팅 인스턴스에서는 cloud-init 패키지 명령이 지원되지 않습니다. 명령 사용에 대한 자세한 내용은 [AWS CLI 명령 참조](#)를 참조하십시오.

Snowcone 디바이스에서 컴퓨팅 인스턴스 관련 문제 해결

다음에서 컴퓨팅 인스턴스와 Snowcone 작업에 대한 문제해결 조언을 확인할 수 있습니다.

주제

- [가상 네트워크 인터페이스의 IP 주소가 0.0.0.0인 경우](#)
- [라지 컴퓨팅 인스턴스를 시작할 때 Snowcone이 중단되는 경우](#)
- [내 인스턴스에 루트 볼륨이 한 개 있는 경우](#)
- [보호되지 않는 프라이빗 키 파일 오류](#)

가상 네트워크 인터페이스의 IP 주소가 0.0.0.0인 경우

이 문제는 VNIC(가상 네트워크 인터페이스)에 연결된 물리적 NIC(네트워크 인터페이스)의 IP 주소도 0.0.0.0인 경우 발생할 수 있습니다. NIC에 IP 주소가 구성되지 않은 경우(예: 디바이스 전원을 방금 켜는 경우)에도 이 문제가 생길 수 있습니다. 또한 잘못된 RJ45 인터페이스를 사용 중인 경우에도 이 문제가 발생할 수 있습니다. Snowcone에는 두 개의 RJ45 인터페이스가 있습니다. 잘못된 물리적 인터페이스를 지정하고 있을 수 있습니다.

취할 조치

이러한 상황이 발생하면 다음 작업을 수행할 수 있습니다.

- IP 주소가 있는 NIC와 연결된 새 VNIC를 생성합니다. 자세한 설명은 [컴퓨팅 인스턴스에 대한 네트워크 구성](#) 섹션을 참조하세요.
- 기존 VNIC를 업데이트합니다. 자세한 설명은 [가상 네트워크 인터페이스 업데이트](#) 섹션을 참조하세요.

라지 컴퓨팅 인스턴스를 시작할 때 Snowcone이 중단되는 경우

Snowcone이 인스턴스 시작을 중지한 것으로 보일 수 있습니다. 이는 일반적인 사례가 아닙니다. 그러나 최대 규모의 컴퓨팅 인스턴스를 시작하는 데 한 시간 이상 걸릴 수 있습니다. Snowcone의 HTTP 또는 HTTPS Amazon EC2 엔드포인트에 대해 AWS CLI 명령을 `aws ec2 describe-instances` 실행하여 인스턴스의 상태를 확인할 수 있습니다.

내 인스턴스에 루트 볼륨이 한 개 있는 경우

인스턴스에는 기본적으로 한 개의 루트 볼륨이 있습니다. 모든 sbe 인스턴스에는 루트 볼륨이 한 개 있습니다.

볼륨 추가에 대한 자세한 내용은 <https://docs.aws.amazon.com/snowball/latest/snowcone-guide/snowcone-snowcone-ebs.html>을 참조하십시오.

보호되지 않는 프라이빗 키 파일 오류

이 오류는 컴퓨팅 인스턴스의 .pem 파일에 읽기/쓰기 권한이 없는 경우 발생할 수 있습니다.

취할 조치

다음 절차를 통해 파일에 대한 권한을 변경하여 이 문제를 해결할 수 있습니다.

1. 터미널을 열고 .pem 파일을 저장한 위치로 이동합니다.
2. 다음 명령을 입력합니다.

```
chmod 400 filename.pem
```

AWS DataSync를 사용하여 파일 전송

인터넷 또는 AWS Direct Connect를 통해 AWS 스토리지 서비스에서 송수신되는 대량의 데이터 복사를 간소화, 자동화 및 가속화는 온라인 데이터 전송 서비스인 AWS DataSync로 작업하십시오.

DataSync 에이전트는 Snowcone 디바이스에 사전 설치되어 있습니다. 디바이스와 Amazon S3 버킷, Amazon EFS 및 Windows File Server용 Amazon FSx 간에 데이터를 전송할 수 있습니다. AWS

DataSync에서는 파일 및 객체 이동, 데이터 전송 예약, 전송 진행 상황 모니터링, 암호화, 데이터 전송 확인, 고객 문제 알림 등을 자동으로 처리합니다.

DataSync 에이전트를 시작하기 전에 Snowcone 디바이스에서 네트워크 파일 시스템(NFS)을 활성화 하십시오. [NFS 인터페이스 자동 구성](#) 및 [NFS 시작 및 액세스 제한](#) 섹션을 참조하십시오.

DataSync 에이전트는 Snowcone 작업 준비 중에 Snowcone 디바이스에 AMI로 사전 설치됩니다. 온라인으로 데이터를 AWS로 전송하려면 Snowcone 디바이스를 외부 네트워크에 연결하고 AWS OpsHub 또는 CLI를 사용하여 DataSync 에이전트 AMI를 시작합니다. AWS 관리 콘솔에서 DataSync 에이전트를 활성화하거나 CLI를 사용하여 Snowcone NFS 스토어와 Amazon S3, Amazon EFS 또는 Amazon FSx for Windows File Server 간에 온라인 데이터 전송 작업을 설정합니다.

Snowcone에서 실행되는 AWS DataSync를 다음과 같은 용도로 사용할 수 있습니다.

- 엣지 컴퓨팅 애플리케이션은 데이터를 수집하고 데이터를 처리하여 즉각적인 인사이트를 확보한 다음 온라인으로 데이터를 AWS로 전송합니다.
- 센서 또는 머신에서 지속적으로 생성되는 데이터를 온라인으로 공장의 AWS 또는 다른 엣지 위치로 전송합니다.
- AWS 스토리지 서비스의 미디어, 과학 또는 기타 콘텐츠를 파트너와 고객에게 온라인으로 배포합니다.
- 미디어, 과학 또는 기타 콘텐츠를 엣지 로케이션에서 AWS로 온라인으로 전송하여 콘텐츠를 집계하십시오.
- DataSync 에이전트의 VNI에 고정 IP 주소를 사용해야 합니다. DHCP를 사용하면 문제가 발생할 수 있습니다. 디바이스가 재부팅되면 다른 IP 주소를 사용하고 있기 때문에 AWS 클라우드에서 DataSync 에이전트와 DataSync 활성화를 모두 재구성해야 하기 때문입니다.
- Snowcone에서 Datasync 에이전트를 사용하려면 Snowcone의 NFS 구성에서 “AllowedHosts” 항목에 DataSync 에이전트 IP 주소가 포함되어 있는지 확인해야 합니다. DataSync 에이전트를 활성화하려면 Snowcone에서 NFS 서비스를 중지하고 다시 시작해야 할 수 있습니다.
- NFS 서비스를 중지할 때마다 DataSync 에이전트도 중지해야 합니다. 참고: 나열된 DataSync 에이전트를 허용한 NFS 서비스를 중지하면 DataSync 에이전트의 작동이 중지됩니다.
- Snowcone 디바이스의 작업당 최대 파일 수는 [AWS DataSync 작업 할당량](#)을 확인하십시오. 파일 수 200,000개를 초과하는 경우 DataSync 태스크는 메모리 할당 오류를 보고하고 실행을 중단합니다.

일회성 엣지 컴퓨팅 또는 데이터 전송 워크플로의 경우 또는 광역 네트워크 (WAN) 링크가 없거나 부적절한 WAN 대역폭이 없는 엣지 로케이션의 Snowcone 워크플로의 경우 Snowcone 디바이스를 AWS로 반송하여 데이터 전송을 완료해야 합니다.

NFS 인터페이스 관리

네트워크 파일 시스템 (NFS) 인터페이스를 사용하여 마치 장치가 운영 체제의 로컬 스토리지인 것처럼 Snow Family 장치에 파일을 업로드할 수 있습니다. 이렇게 하면 파일 복사, 드래그 앤 드롭 또는 기타 그래픽 사용자 인터페이스 기능과 같은 운영 체제 기능을 사용할 수 있으므로 보다 사용자 친화적인 방식으로 데이터를 전송할 수 있습니다. 디바이스의 각 S3 버킷은 NFS 인터페이스 엔드포인트로 사용할 수 있으며 데이터를 복사하기 위해 마운트할 수 있습니다. NFS 인터페이스는 가져오기 작업에 사용할 수 있습니다.

시작 시 NFS 인터페이스는 1GB의 메모리와 1개의 CPU를 사용합니다. 이로 인해 Snow Family 디바이스에서 실행되는 다른 서비스의 수 또는 실행할 수 있는 EC2 호환 인스턴스의 수가 제한될 수 있습니다.

NFS 인터페이스를 통해 전송된 데이터는 전송 중에 암호화되지 않습니다. NFS 인터페이스를 구성할 때 CIDR 블록을 제공할 수 있으며 Snow Family 디바이스는 해당 블록에 주소가 있는 클라이언트 컴퓨터에서 NFS 인터페이스에 대한 액세스를 제한합니다.

Amazon S3로 반환되면 디바이스에 있는 파일이 Amazon S3로 전송됩니다. 자세한 내용은 [Snowcone 작동 방식을](#) 참조하십시오.

컴퓨터 운영 체제에서 NFS를 사용하는 방법에 대한 자세한 내용은 운영 체제 설명서를 참조하십시오.

NFS 인터페이스를 사용할 때는 다음 세부 정보를 염두에 두십시오.

- 파일 이름은 Snow Family 디바이스의 로컬 S3 버킷에 있는 객체 키입니다. 키 이름은 최대 1,024바이트 길이와 UTF-8 인코딩을 사용하는 유니코드 문자의 시퀀스입니다. 가능한 경우 NFSv4.1을 사용하고 파일 이름을 유니코드 UTF-8 형식으로 인코딩하여 데이터를 성공적으로 가져오는 것이 좋습니다. UTF-8 인코딩으로 인코딩되지 않은 파일 이름은 사용하는 NFS 인코딩에 따라 S3에 업로드되지 않거나 다른 파일 이름으로 S3에 업로드될 수 있습니다.
- 파일 경로의 최대 길이는 1,024자 미만이어야 합니다. Snow Family 디바이스는 1,024자를 초과하는 파일 경로를 지원하지 않습니다. 이 파일 경로 길이를 초과하면 파일 가져오기 오류가 발생합니다.
- 자세한 내용은 Amazon 심플 스토리지 서비스 사용 설명서의 [객체 키](#)를 참조하십시오.
- NFS 기반 전송의 경우 Snow Family 디바이스에서 Amazon S3로 객체를 가져올 때 표준 POSIX 스타일 메타데이터가 객체에 추가됩니다. 또한 NFS 옵션을 사용한 Snow Family 디바이스 가져오기를 위해 현재 Amazon S3로 내부 가져오기 메커니즘의 AWS DataSync 일부로 사용하고 있는 메타데이터 "x-amz-meta-user-agent aws-datasync"가 표시됩니다.

Note

NFS 서비스가 시작되기 전까지는 Snowcone 디바이스의 사용 가능한 스토리지 공간이 정확하지 않습니다.

또한 GUI 도구를 사용하여 AWS OpsHub NFS 인터페이스를 구성하고 관리할 수 있습니다. 자세한 내용은 오프라인 파일 전송을 위한 NFS를 [사용한 NFS](#) 참조하십시오.

Snow Family 디바이스의 NFS 구성

NFS 인터페이스는 기본적으로 Snow Family 디바이스에서 실행되지 않으므로 디바이스로 데이터를 전송할 수 있으려면 인터페이스를 시작해야 합니다. Snow Family 디바이스에서 실행되는 가상 네트워크 인터페이스 (VNI) 의 IP 주소를 제공하고 필요한 경우 파일 공유에 대한 액세스를 제한하여 NFS 인터페이스를 구성할 수 있습니다. NFS 인터페이스를 구성하기 전에 Snow Family 디바이스에 가상 네트워크 인터페이스 (VNI) 를 설정하십시오. 자세한 내용은 [컴퓨팅 인스턴스를 위한 네트워크 구성](#)을 참조하십시오.

Snow 패밀리 디바이스를 NFS 인터페이스용으로 구성하십시오.

- describe-service 명령을 사용하여 NFS 인터페이스가 활성 상태인지 확인합니다.

```
snowballEdge describe-service --service-id nfs
```

이 명령은 NFS 서비스의 상태 또는 를 반환합니다. ACTIVE INACTIVE

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "ACTIVE"
  }
}
```

State 이름 값이 인 경우 NFS 인터페이스 서비스가 활성화되며 ACTIVE Snow Family 디바이스 NFS 볼륨을 마운트할 수 있습니다. 자세한 설명은

NFS 인터페이스가 시작된 후 엔드포인트를 클라이언트 컴퓨터에 로컬 스토리지로 마운트합니다.

다음은 윈도우, 리눅스, macOS 운영 체제의 기본 마운트 명령입니다.

- Windows:

```
mount -o nolock rsize=128 wsize=128 mtype=hard nfs-interface-ip-address:/
buckets/BucketName *
```

- Linux:

```
mount -t nfs nfs-interface-ip-address:/buckets/BucketName mount_point
```

- macOS:

```
mount -t nfs -o vers=3,rsize=131072,wsize=131072,nolocks,hard,retrans=2 nfs-
interface-ip-address:/buckets/$bucketname mount_point
```

섹션을 참조하세요. 값이 INACTIVE 이면 서비스를 시작해야 합니다.

Snow Family 디바이스에서 NFS 서비스 시작

필요한 경우 가상 네트워크 인터페이스 (VNI) 를 시작한 다음 Snow Family 디바이스에서 NFS 서비스를 시작합니다. 필요한 경우 NFS 서비스를 시작할 때 허용된 네트워크 주소 블록을 제공하십시오. 주소를 제공하지 않으면 NFS 엔드포인트에 대한 액세스가 제한되지 않습니다.

1. `describe-virtual-network-interface` 명령을 사용하여 Snow Family 디바이스에서 사용할 수 있는 VNI를 확인하세요.

```
snowballEdge describe-virtual-network-interfaces
```


Snow Family 디바이스에서 하나 이상의 VNI가 활성화된 경우 명령은 다음을 반환합니다.

```
snowballEdge describe-virtual-network-interfaces
[
  {
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device::interface/
s.ni-8EXAMPLE8EXAMPLE8",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.0",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "EX:AM:PL:E1:23:45"
  },{
    "VirtualNetworkInterfaceArn" : "arn:aws:snowball-device::interface/
s.ni-1EXAMPLE1EXAMPLE1",
    "PhysicalNetworkInterfaceId" : "s.ni-8EXAMPLEaEXAMPLEd",
    "IpAddressAssignment" : "DHCP",
    "IpAddress" : "192.0.2.2",
    "Netmask" : "255.255.255.0",
    "DefaultGateway" : "192.0.2.1",
    "MacAddress" : "12:34:5E:XA:MP:LE"
  }
]
```

NFS 인터페이스와 함께 사용할 VNI `VirtualNetworkInterfaceArn` 이름 값을 기록해 둡니다.

2. 사용할 수 있는 VNI가 없는 경우 `create-virtual-network-interface` 명령을 사용하여 NFS 인터페이스용 VNI를 생성합니다. 자세한 내용은 [가상 네트워크 인터페이스 \(VNI\) 설정을 참조](#)하십시오.
3. `start-service` 명령을 사용하여 NFS 서비스를 시작하고 VNI에 연결합니다. NFS 인터페이스에 대한 액세스를 제한하려면 명령에 `service-configuration` 및 `AllowedHosts` 매개 변수를 포함하십시오.

```
snowballEdge start-service --virtual-network-interface-arns arn-of-vni --service-id
nfs --service-configuration AllowedHosts=CIDR-address-range
```

4. `describe-service` 명령을 사용하여 서비스 상태를 확인합니다. State 이름 값이 일 때 실행됩니다. ACTIVE.

```
snowballEdge describe-service --service-id nfs
```

이 명령은 서비스 상태뿐만 아니라 NFS 엔드포인트의 IP 주소 및 포트 번호, 엔드포인트에 액세스할 수 있는 CIDR 범위를 반환합니다.

```
{
  "ServiceId" : "nfs",
  "Status" : {
    "State" : "ACTIVE"
  },
  "Endpoints" : [ {
    "Protocol" : "nfs",
    "Port" : 2049,
    "Host" : "192.0.2.0"
  } ],
  "ServiceConfiguration" : {
    "AllowedHosts" : [ "10.24.34.0/23", "198.51.100.0/24" ]
  }
}
```

클라이언트 컴퓨터에 NFS 엔드포인트를 마운트합니다.

NFS 인터페이스가 시작된 후 엔드포인트를 클라이언트 컴퓨터에 로컬 스토리지로 마운트합니다.

다음은 윈도우, 리눅스, macOS 운영 체제의 기본 마운트 명령입니다.

- Windows:

```
mount -o nolock rsize=128 wsize=128 mtype=hard nfs-interface-ip-address:/
buckets/BucketName *
```

- Linux:

```
mount -t nfs nfs-interface-ip-address:/buckets/BucketName mount_point
```

- macOS:

```
mount -t nfs -o vers=3,rsiz=131072,wsiz=131072,nolocks,hard,retrans=2 nfs-  
interface-ip-address:/buckets/$bucketname mount_point
```

NFS 인터페이스 중지

NFS 인터페이스를 통한 파일 전송을 완료하고 Snow Family 디바이스의 전원을 끄기 전에 `stop-service` 명령을 사용하여 NFS 서비스를 중지하십시오.

```
snowballEdge stop-service --service-id nfs
```

AWS IoT Greengrass를 사용하여 Amazon EC2 호환 인스턴스에서 사전 설치된 소프트웨어 실행

AWS IoT Greengrass는 디바이스에서 IoT 애플리케이션을 구축, 배포 및 관리하는 데 도움이 되는 오픈 소스 사물 인터넷(IoT) 엣지 런타임 및 클라우드 서비스입니다. AWS IoT Greengrass를 사용하여 디바이스가 생성한 데이터에 대해 로컬로 작업하고, 기계 학습 모델을 기반으로 예측을 실행하고, 디바이스 데이터를 필터링 및 집계할 수 있도록 하는 소프트웨어를 구축할 수 있습니다. AWS IoT Greengrass에 대한 자세한 내용은 AWS IoT Greengrass Version 2 개발자 설명서의 [AWS IoT Greengrass란 무엇입니까?](#)를 참조하십시오.

Snow Family 디바이스에서 AWS IoT Greengrass를 사용하면 데이터가 생성되는 위치와 더 가까운 곳에서 디바이스가 데이터를 수집 및 분석하고, 로컬 이벤트에 자율적으로 대응하고, 로컬 네트워크의 다른 디바이스와 안전하게 통신할 수 있습니다.

Amazon EC2 호환 인스턴스 설정

Note

Snow Family 디바이스에 AWS IoT Greengrass Version 2를 설치하려면 디바이스가 인터넷에 연결되어 있어야 합니다. 설치 후에는 Snow Family 디바이스가 AWS IoT Greengrass와 함께 작동하는 데 인터넷이 필요하지 않습니다.

AWS IoT Greengrass V2에 대해 EC2 호환 인스턴스를 설정하려면

1. 퍼블릭 IP 주소와 SSH 키를 사용하여 AWS IoT Greengrass 검증된 AMI를 시작합니다.
 - a. AWS CLI 사용: [run-instances](#)
 - b. AWS OpsHub 사용: [Amazon EC2 인스턴스 시작하기](#)를 참조하십시오.

Note

인스턴스와 연결된 퍼블릭 IP 주소 및 SSH 키 이름을 기록해 둡니다.

2. SSH를 이용해 EC2 호환 인스턴스에 연결합니다. 이를 위해 디바이스에 연결된 컴퓨터에서 다음 명령을 실행합니다. `ssh-key`를 EC2 호환 인스턴스를 시작하는 데 사용한 키로 대체합니다. `public-ip-address`를 EC2 호환 인스턴스의 퍼블릭 IP 주소로 대체합니다.

```
ssh -i ssh-key ec2-user@ public-ip-address
```

Important

컴퓨터에서 이전 버전의 Microsoft Windows를 사용하는 경우 SSH 명령이 없거나 SSH가 있지만 EC2 호환 인스턴스에 연결하지 못할 수 있습니다. EC2 호환 인스턴스에 연결하려면 무료 오픈 소스 SSH 클라이언트인 PuTTY를 설치하고 구성하면 됩니다. SSH 키를 .pem 형식에서 PuTTY 형식으로 변환하고 EC2 인스턴스에 연결해야 합니다. .pem에서 PuTTY 형식으로 변환하는 방법은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [PuTTYgen을 사용하여 프라이빗 키 변환](#)을 참조하십시오.

AWS IoT Greengrass 설치

다음으로 EC2 호환 인스턴스를 로컬 개발에 사용할 수 있는 AWS IoT Greengrass 코어 디바이스로 설정합니다.

AWS IoT Greengrass을(를) 설치하려면

1. 다음 명령을 사용하여 AWS IoT Greengrass에 대한 필수 소프트웨어를 설치합니다. 이 명령은 AWS Command Line Interface(AWS CLI) v2, Python 3, Java 8을 설치합니다.

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
&& unzip awscliv2.zip && sudo ./aws/install && sudo yum -y install python3
java-1.8.0-openjdk
```

2. 루트 사용자에게 AWS IoT Greengrass 소프트웨어를 실행할 수 있는 권한을 부여하고 sudoers 구성 파일에서 루트 권한을 root ALL=(ALL) ALL에서 root ALL=(ALL:ALL) ALL로 수정합니다.

```
sudo sed -in 's/root\tALL=(ALL)/root\tALL=(ALL:ALL)/' /etc/sudoers
```

3. 다음 명령을 사용하여 AWS IoT Greengrass 코어 소프트웨어를 다운로드합니다.

```
curl -s https://d2s8p88vqu9w66.cloudfront.net/releases/greengrass-nucleus-
latest.zip > greengrass-nucleus-latest.zip && unzip greengrass-nucleus-latest.zip -
d GreengrassCore && rm greengrass-nucleus-latest.zip
```

4. 다음 명령을 사용하여 AWS IoT Greengrass 코어 소프트웨어를 설치할 수 있는 자격 증명을 제공합니다. 예시 값을 사용자의 자격 증명으로 바꿉니다.

```
export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

Note

이는 Snow Family 디바이스가 아니라 해당 AWS 리전의 IAM 사용자가 제공한 자격 증명입니다.

5. 다음 명령을 실행하여 AWS IoT Greengrass 코어 소프트웨어를 시작합니다. 이 명령은 코어 소프트웨어가 작동하는 데 필요한 AWS 리소스를 생성하고 코어 소프트웨어를 AMI 부팅 시 실행되는 시스템 서비스로 설정합니다.

명령에서 다음 파라미터를 교체합니다.

- `region`: 리소스를 찾거나 생성할 AWS 리전입니다.
- `MyGreengrassCore`: AWS IoT Greengrass 코어 디바이스에 대한 AWS IoT 사물의 이름입니다.
- `MyGreengrassCoreGroup`: AWS IoT Greengrass 코어 디바이스에 대한 AWS IoT 사물의 이름입니다.

```
sudo -E java -Droot="/greengrass/v2" -Dlog.store=FILE \  
-jar ./GreengrassInstaller/lib/Greengrass.jar \  
--aws-region region \  
--thing-name MyGreengrassCore \  
--thing-group-name MyGreengrassCoreGroup \  
--thing-policy-name GreengrassV2IoTThingPolicy \  
--tes-role-name GreengrassV2TokenExchangeRole \  
--tes-role-alias-name GreengrassCoreTokenExchangeRoleAlias \  
--component-default-user ggc_user:ggc_group \  
--provision true \  
--setup-system-service true \  
--deploy-dev-tools true
```

Note

이 명령은 Amazon Linux 2 AMI를 실행하는 Amazon EC2 호환 인스턴스를 위한 명령입니다. Windows AMI의 경우 [AWS IoT Greengrass 코어 소프트웨어 설치](#)를 참조하십시오.

작업을 마치면 로컬에서 사용할 수 있도록 Snow Family 디바이스에서 AWS IoT Greengrass 코어를 실행하게 됩니다.

AWS Snowcone 디바이스에서 AWS 서비스를 사용하는 데 필요한 포트

AWS 서비스가 AWS Snowcone 디바이스에서 제대로 작동하려면 서비스에 필요한 네트워크 포트를 허용해야 합니다.

다음은 각 AWS 서비스에 필요한 네트워크 포트 목록입니다.

포트	프로토콜	Comment
22	SSH	디바이스 상태 점검 및 EC2 SSH
2049	NFS	NFS 엔드포인트
8008	HTTP	EC2 HTTP 엔드포인트
8243	HTTPS	EC2 HTTPS 엔드포인트
9091	HTTP	디바이스 관리를 위한 엔드포인트

Snowcone 디바이스 반납

Snowcone 디바이스에 데이터를 모두 전송했으면 AWS에 반송할 준비를 합니다. 계속하기 전에 디바이스에 대한 모든 데이터 전송이 중단되었는지 확인합니다.

디바이스와의 모든 통신이 종료된 후 전원 버튼을 눌러서 전원을 끕니다. 디바이스를 종료하는 데 20초 정도 걸립니다.

Snowcone 디바이스 연결 해제

Snowcone 케이블을 분리합니다. 반송 라벨이 디바이스 상단의 전자 잉크 디스플레이에 표시되면 반송할 준비가 된 것입니다. 리전의 배송 업체를 확인하려면 [AWS Snowcone에 대한 배송 고려 사항](#) 섹션을 참조하십시오.

작업 유형별 고려사항

Important

데이터를 가져올 경우 프로세스를 종료할 때 AWS로 가져오기가 성공하고 데이터 전송 결과를 확인할 수 있을 때까지 전송된 데이터의 로컬 복사본을 삭제해서는 안 됩니다.

Note

Amazon S3로 가져오기 위해 Snow 디바이스를 반납하면 AWS에서 디바이스가 변조되지 않았는지, 디바이스가 정상인지 확인한 후 데이터 수집을 시작합니다. 디바이스의 데이터가 대상 S3 버킷으로 수집되는 것을 원하지 않는 경우 Snow 작업을 취소를 요청할 수 있습니다. 작업을 취소하면 데이터 전송을 건너뛰고 설정된 프로세스에 따라 디바이스를 안전하게 삭제합니다. 엄격한 관리 연속성 및 운영 절차로 인해 당사는 귀하의 데이터가 들어 있는 디바이스를 당사 시설에 보관할 수 없습니다.

디바이스 배송 방법에 대한 자세한 내용은 [AWS Snowcone에 대한 배송 고려 사항](#) 섹션을 참조하십시오.

디바이스에서 데이터 보호

AWS Snowcone 디바이스의 데이터를 보호하는 데 도움이 되는 다음 권장 사항을 고려하세요.

주제

- [AWS Snowcone 보호](#)
- [NFC 태그 검증](#)

AWS Snowcone 보호

다음은 Snowcone을 사용할 때 고려할 몇 가지 보안 관련 사항과 디바이스가 처리를 위해 AWS에 도착하면 AWS가 취하는 기타 보안 관련 예방 조치에 대한 몇 가지 상위 수준 정보를 기술한 것입니다.

다음 보안 접근 방식을 따르는 것이 좋습니다.

- 디바이스가 처음 도착하면 손상되거나 명백히 변경된 부분이 있는지 검사합니다. 디바이스에서 의심이 가는 부분이 있으면 내부 네트워크에 연결하지 마세요. 그 대신에 [AWS Support](#)에 연락하면 새 디바이스를 받을 수 있습니다.
- 작업 보안 인증 정보가 누출되지 않도록 보호해야 합니다. 작업의 매니페스트 및 잠금 해제 코드에 대한 액세스 권한이 있는 모든 개인은 해당 작업에 대해 전송되는 디바이스의 콘텐츠에 액세스할 수 있습니다.
- 디바이스를 로딩 독에 두지 마세요. 로딩 독에 방치하면 여러 요소에 노출될 수 있습니다. 각 Snowcone 디바이스는 견고하지만 아무리 튼튼한 하드웨어도 악천후로 인해 손상될 수 있습니다. 디바이스가 도난, 분실 또는 파손된 경우 최대한 빨리 보고하세요. 문제를 빨리 보고할수록 다른 디바이스를 더 빨리 배송받아 작업을 완료할 수 있습니다.

Note

Snowcone 디바이스는 AWS의 자산입니다. 디바이스를 함부로 변경하는 것은 AWS 이용 정책을 위반하는 행위입니다. 자세한 내용은 <http://aws.amazon.com/aup/>를 참조하세요.

다음 보안 단계를 수행합니다.

- 파일 인터페이스를 통해 데이터를 전송할 때 객체 메타데이터가 유지됩니다.

- 디바이스가 AWS에 도착하면 변조 흔적이 있는지 검사하고 TPM(Trusted Platform Module)에서 탐지한 변경 사항이 없는지 확인합니다. Snowcone은 변조 방지 인클로저, 256비트 암호화, 데이터 보안 및 완전한 관리 체인을 제공하도록 설계된 업계 표준 TPM을 포함하여 데이터를 보호하도록 설계된 다중 보안 계층을 사용합니다.
- 데이터 전송 작업이 처리 및 확인되면 AWS는 미디어 삭제를 위한 미국 국립 표준 기술 연구소 (NIST) 지침에 따라 Snowcone 디바이스의 소프트웨어를 삭제합니다.

NFC 태그 검증

AWS Snowcone 디바이스에는 NFC 태그가 내장되어 있습니다. Android에서 사용 가능한 Snowcone Verification 앱을 사용하여 이러한 태그를 스캔할 수 있습니다. 이러한 NFC 태그를 스캔하고 검증함으로써 디바이스 사용 전에 디바이스가 변조되지 않았는지 확인할 수 있습니다.

NFC 태그 검증 과정에는 Snowball Edge 클라이언트를 사용해 디바이스별 QR 코드를 생성하여 스캔 중인 태그가 올바른 디바이스에 부합하는지 확인하는 작업이 포함됩니다. 자세한 내용은 [NFC 검증에 대한 QR 코드 가져오기](#) 섹션을 참조하세요.

다음 절차는 Snowcone 디바이스에서 NFC 태그를 검증하는 방법을 설명합니다. 시작하기 전에, 다음과 같은 시작하기 실습의 첫 단계를 수행했는지 확인해야 합니다.

1. 첫 번째 작업을 생성합니다. 자세한 내용은 [Snow Family 디바이스 주문을 위한 작업 생성](#)을 참조하십시오.
2. 디바이스를 수령합니다.
3. 로컬 네트워크에 연결합니다.
4. 보안 인증 정보 및 도구를 가져옵니다. 자세한 설명은 [자격 증명 받기](#) 섹션을 참조하세요.
5. Snowball Edge 클라이언트를 다운로드 및 설치합니다. 자세한 설명은 [AWS Snowball Edge 클라이언트 사용](#) 섹션을 참조하세요.

AWS Snowcone 디바이스의 NFC 태그 검증

1. `snowballEdge get-app-qr-code` Snowball Edge 클라이언트 명령을 실행합니다. 이 명령의 사용에 대한 자세한 내용은 [NFC 검증에 대한 QR 코드 가져오기](#) 섹션을 참조하세요.

QR 코드는 원하는 위치에 .png 파일로 저장됩니다.

2. 저장한 .png 파일로 이동하여 앱으로 QR 코드를 스캔할 수 있도록 엽니다.

3. 휴대폰으로 NFC 태그를 스캔하려면 Snowcone Verification 앱을 다운로드하여 설치합니다.
Android 휴대폰을 사용하는 경우 Google Play 스토어에서 앱을 다운로드합니다.
4. 앱을 시작하고 화면의 지침을 따릅니다.

이제 디바이스에 대한 NFC 태그를 성공적으로 스캔했고 검증했습니다.

스캔 중 문제가 발생하는 경우, 다음을 시도합니다.

- 앱을 다른 휴대폰에 다운로드하여 다시 시도합니다.
- 디바이스가 다른 NFC 태그의 간섭을 벗어나도록 방의 격리된 공간으로 이동하여 다시 시도합니다.
- 문제가 지속되면 [AWS Support](#)로 문의하세요.

AWS Snowcone 작업 상태 이해

AWS Snowcone 작업을 생성하면 작업 상태가 전환되고 상태가 AWS Snow 패밀리 관리 콘솔에 표시됩니다.

작업 상태 보기

1. [AWS Snow 패밀리 관리 콘솔](#)에 로그인합니다.
2. 작업 대시보드에서 작업을 선택합니다.
3. 콘솔에서 작업 이름을 클릭합니다.
4. 작업 상태 창은 상단 근처에 있으며 작업 상태를 반영합니다.

Note

구성한 액세스 권한 관련 문제로 인해 Snow 디바이스에서 데이터 센터로 데이터를 가져올 수 없는 경우에는 사용자에게 이를 알리려고 시도하며, 사용자는 알림을 받은 날로부터 30일 이내에 문제를 해결할 수 있습니다. 문제가 해결되지 않으면 AWS Snow Family 작업을 취소하고 디바이스에서 데이터를 삭제할 수 있습니다.

AWS Snowcone 디바이스 작업 상태

작업 상태	의미
작업 생성됨	작업이 방금 생성되었습니다. 작업이 내보내기 작업인 경우 이 상태인 동안에만 작업 또는 작업 파트를 취소할 수 있습니다.
어플라이언스 준비 중	작업에 사용할 디바이스를 AWS가 준비하고 있습니다.
배송 준비 중	디바이스를 배송하기 위해 AWS에서 준비하고 있습니다.
배송 중	작업 생성 중에 제공한 주소로 디바이스가 배송되었습니다.

작업 상태	의미
사용자에게 배달됨	작업 생성 중에 제공한 주소로 디바이스가 도착했습니다.
AWS로 배송 중	디바이스를 AWS로 반환했습니다.
분류 시설에 있음	이 작업의 디바이스는 내부 분류 시설에 있습니다. Amazon S3로 가져오기 작업에 대한 모든 추가 처리는 일반적으로 향후 2일 내에 시작됩니다.
AWS에 있음	배송물이 AWS에 도착했습니다. 데이터를 가져올 경우 가져오기는 일반적으로 도착 날짜 내에 시작됩니다.
가져오는 중	AWS가 데이터를 Amazon S3로 가져오고 있습니다.
완료됨	작업 또는 작업 파트를 성공적으로 완료했습니다.
취소됨	작업을 취소했습니다.

Snow Family 디바이스 알림

Snow가 Amazon SNS를 사용하는 방법

Snow 서비스는 Amazon Simple Notification Service(SNS)에서 제공하는 강력한 알림을 활용하도록 설계되었습니다. Snow 디바이스를 주문하기 위한 작업을 생성하는 동안 작업 상태 변경에 대한 알림을 받을 이메일 주소를 제공할 수 있습니다. 기존 SNS 주제를 선택하거나 새로운 주제를 생성할 수 있습니다. SNS 주제가 암호화된 경우 주제에 대해 고객 관리형 KMS 암호화를 활성화하고 고객 관리형 KMS 키 정책을 설정해야 합니다. [알림 기본 설정 선택](#) 섹션을 참조하십시오.

작업을 생성한 후 Amazon SNS 알림을 받기 위해 지정한 모든 이메일 주소는 AWS 알림으로부터 주제 구독 확인을 요청하는 이메일 메시지를 받게 됩니다. 이메일 계정 사용자는 가입 확인을 선택하여 가입 여부를 확인해야 합니다. Amazon SNS 알림 이메일은 각 작업 상태에 맞게 조정되며 [AWS Snow 패밀 리 관리 콘솔](#)에 대한 링크를 포함합니다.

또한 Amazon SNS 콘솔에서 상태 변경 알림을 위한 문자 메시지를 보내도록 Amazon SNS를 구성할 수 있습니다. 자세한 내용은 Amazon Simple Notification Service 개발자 가이드의 [모바일 문자 메시지 \(SMS\)](#)를 참조하십시오.

Snow 작업 상태 변경에 대한 SNS 주제 암호화

Snow 작업 상태 변경 알림의 SNS 주제에 대한 고객 관리형 KMS 암호화를 활성화하십시오. Snow import IAM 역할은 Decrypt 및 GenerateDataKey 작업을 수행할 AWS 관리형 KMS 키에 액세스할 수 없기 때문에 AWS 관리형 암호화로 암호화된 SNS 주제는 Snow 작업 상태 변경을 수신할 수 없습니다. 또한 AWS 관리형 KMS 키 정책은 편집할 수 없습니다.

Amazon SNS 관리 콘솔을 사용하여 SNS 주제에 대해 서버 측 암호화(SSE)를 활성화하려면

1. AWS Management Console에 로그인하고 <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 주제를 선택합니다.
3. 주제 페이지에서 작업 상태 변경 알림에 사용할 주제를 선택한 다음 편집을 선택합니다.
4. 암호화 섹션을 확장하고 다음을 수행합니다.
 - a. 암호화 활성을 선택합니다.
 - b. AWS KMS 키를 지정합니다. 다음을 참조하십시오.
 - c. 각 KMS 유형에 대해 설명, 계정, KMS ARN이 표시됩니다.

5. AWS 계정에서 사용자 지정 키를 사용하려면 AWS KMS 키 필드를 선택한 다음 목록에서 사용자 지정 KMS를 선택합니다. 사용자 지정 KMS 생성에 대한 지침은 AWS Key Management Service 개발자 설명서의 [키 생성](#)을 참조하십시오.

AWS 계정 또는 다른 AWS 계정에서 사용자 지정 KMS ARN을 사용하려면 AWS KMS 키 필드에 KMS 키 ARN을 입력합니다.

6. Save changes(변경 사항 저장)를 선택합니다. 주제에 대해 서버 측 암호화가 활성화되고 주제 페이지가 표시됩니다.

고객 관리형 KMS 키 정책 설정

Snow 작업 상태 변경에 대한 알림을 받을 SNS 주제에 대한 암호화를 활성화한 후, SNS 주제 암호화에 대한 KMS 정책을 업데이트하고 Snow 서비스 보안 주체 "importexport.amazonaws.com"에 "kms:Decrypt" 및 "kms:GenerateDataKey*" 조치를 허용합니다.

KMS 키 정책에서 가져오기 및 내보내기 서비스 역할을 허용하려면

1. AWS Management Console에 로그인하고 <https://console.aws.amazon.com/kms>에서 AWS Key Management Service(AWS KMS) 콘솔을 엽니다.
2. AWS 리전을 변경하려면 페이지의 오른쪽 상단 모서리에 있는 리전 선택기를 사용합니다.
3. 콘솔 오른쪽 상단에서 Snow 디바이스를 주문한 리전과 동일한 리전으로 콘솔의 AWS 리전을 변경합니다.
4. 탐색 창에서 고객 관리형 키(Customer managed keys)를 선택합니다.
5. KMS 키 목록에서 업데이트하려는 KMS 키의 별칭 또는 키 ID를 선택합니다.
6. 키 정책 설명에서 키 정책 탭을 선택하여 KMS 키에 대한 액세스 권한이 부여된 보안 주체를 볼 수 있고, 이러한 보안 주체가 수행할 수 있는 작업을 볼 수 있습니다.
7. Snow 서비스 보안 주체 "importexport.amazonaws.com"의 경우 "kms:Decrypt" 및 "kms:GenerateDataKey*" 조치에 대해 다음 정책 설명을 추가합니다.

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "service.amazonaws.com"
  },
  "Action": [
```

```

"kms:Decrypt",
"kms:GenerateDataKey"
],
"Resource": "*",
"Condition": {
  "ArnLike": {
    "aws:SourceArn": "arn:aws:service:region:customer-account-id:resource-type/
customer-resource-id"
  },
  "StringEquals": {
    "kms:EncryptionContext:aws:sns:topicArn": "arn:aws:sns:your_region:customer-
account-id:your_sns_topic_name"
  }
}
}
}

```

8. 변경 사항 저장을 선택하여 변경 내용을 적용하고 정책 편집기를 종료합니다.

SNS 알림 예시

Amazon SNS 알림은 작업 상태가 변경될 때 다음과 같은 이메일 메시지를 생성합니다. 이러한 메시지는 Email-JSON SNS 주제 프로토콜의 예입니다.

작업 상태	SNS 알림 JSON
작업 생성됨	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) has been created. More info - https://console.aws.amazon. com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", </pre>

작업 상태	SNS 알림 JSON
	<pre> "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtdRJ1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 2223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
어플라이언스 준비 중	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is being prepared. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
내보내는 중	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is being Exported. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkg ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
배송 중	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is in transit to you. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
사용자에게 배달됨	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) was delivered to you. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
AWS로 배송 중	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is in transit to AWS. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
분류 시설에 있음	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is at AWS sorting facility. More info - https:// console.aws.amazon.com/impor texport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1shOBWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi11lhIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd507lX1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD0lzmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtdRJIyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
AWS에 있음	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is at AWS. More info - https://console.aws.amazon.com/ importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi11lhIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
가져오는 중	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) is being imported. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkg ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON
완료됨	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) complete.\nThanks for using AWS Snow Family.\nCan you take a quick survey on your experienc e? Survey here: http://bit.ly/1pLQ JMY. More info - https://console.aw s.amazon.com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1shOBWsJHkrW2xkR58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi11lhIkq ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtdRJIyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

작업 상태	SNS 알림 JSON

작업 상태	SNS 알림 JSON
취소됨	<pre> { "Type" : "Notification", "MessageId" : "dc1e94d9-56c5-5e9 6-808d-cc7f68faa162", "TopicArn" : "arn:aws:sns:us-ea st-2:111122223333:ExampleTopic1", "Message" : "Your job Job-name (JID8bca334a-6c2f-4cd0-97e2 -3f5a4dc9bd6d) was canceled. More info - https://console.aws.amazon. com/importexport", "Timestamp" : "2023-02-23T00:27: 58.831Z", "SignatureVersion" : "1", "Signature" : "FMG5t1ZhJNHLHUXvZ gtZz1k24FzVa7oX0T4P03neeXw8 ZEXZx6z35j2F0TuNYShn2h0bKNC/ zLTnMyIxEzmi2X1sh0BWsJHkrW2xkr58ABZ F+4uWHEE73yDVR4SyYAIkP9jstZzDRm +bcVs8+T0yaLiEGLrIIIL4esi111hIkG ErCuy5btPcWXBdio2fpCRD5x9oR 6gmE/rd5071X1c1uvnv4r1Lkk4pqP2/ iUfxFZva1xLSRvgyfm6D9hNk1VyPfy+7 Ta1MD01zmJu0rExtnSIbZew3foxgx8GT +1bZkLd0ZdtRj1IyPRP44eyq78sU0Eo/ LsDr0Iak4ZDpg8dXg==", "SigningCertURL" : "https:// sns.us-east-1.amazonaws.com/ SimpleNotificationService-010a507c1 833636cd94bdb98bd93083a.pem", "UnsubscribeURL" : "https:// sns.us-east-2.amazonaws.com/? Action=Unsubscribe&SubscriptionArn =arn:aws:sns:us-east-2:1111 22223333:ExampleTopic1:e103 9402-24e7-40a3-a0d4-797da162b297" } </pre>

AWS Snowcone 주문 프로세스에 대한 이해

AWS Snowcone를 주문할 때는 두 가지 옵션이 있습니다. AWS Snow 패밀리 관리 콘솔을 통해 주문하거나 작업 관리 API(JMAPI)를 사용할 수 있습니다.

배송 프로세스에 대한 이해

이 섹션에서는 AWS Snowcone 디바이스의 배송 처리 방식에 대한 정보와 지원되는 AWS 리전 목록을 확인할 수 있습니다. 지원되는 리전과 엔드포인트에 대한 자세한 내용은 AWS 일반 참조의 [AWS Snow Family 엔드포인트 및 할당량](#)을 참조하십시오. 작업에 대해 선택하는 배송료는 해당 작업에 사용되는 Snowcone 디바이스의 전송과 수신 모두에 적용됩니다. 배송 요금에 대한 자세한 내용은 [AWS Snowcone 요금](#)을 참조하십시오.

Note

Snowcone 디바이스는 디바이스가 주문된 AWS 리전 내에서 데이터를 가져오거나 내보내는 데에만 사용할 수 있습니다.

Snowcone 디바이스 반납

E Ink 디스플레이의 선불 배송 정보에는 디바이스를 반송할 정확한 주소가 포함되어 있습니다. Snowcone 디바이스 반송 방법에 대한 자세한 내용은 [배송 업체](#) 섹션을 참조하십시오. 디바이스는 AWS 분류 시설로 배송되었다가 다시 AWS 데이터 센터로 전달됩니다. 리전의 배송 업체를 통해 패키지를 추적할 수 있습니다. AWS Snow 패밀리 관리 콘솔을 사용하여 작업에 대한 상태 변경을 추적할 수 있습니다.

Important

AWS에서 달리 지시하지 않는 한 Snowcone 디바이스에 별도의 배송 라벨을 부착해서는 안 됩니다. 항상 디바이스의 E Ink 디스플레이에 표시된 배송 라벨을 사용하십시오.

AWS Management Console 사용

AWS Snow 패밀리 관리 콘솔을 사용하여 Snowcone 디바이스를 주문할 수 있습니다.

콘솔에서 Snowcone 주문

AWS Snowball을 사용하여 Snowcone을 주문하는 방법에 대한 단계별 지침은 [시작하기](#) 섹션을 참조하십시오.

작업 관리 API 사용

작업 관리 API(JMAPI)를 사용하면 AWS Snow 패밀리 관리 콘솔에서 사용할 수 있는 것과 동일한 기능에 프로그래밍 방식으로 액세스할 수 있습니다. 이를 통해 작업 기능을 자동화할 수 있습니다. JMAPI를 사용하면 작업 상태를 확인하고, 작업을 생성하고, 매니페스트 파일을 다운로드하고, 코드를 잠금 해제하고, 작업 완료 보고서를 볼 수 있습니다. 직접 호출은 API를 통해 이루어지므로 직접 호출을 사용자 지정 애플리케이션 또는 웹 프론트 엔드에 통합할 수 있습니다.

주제

- [JMAPI의 일반적인 용도](#)
- [JMAPI 필수 문자열](#)
- [JMAPI 엔드포인트](#)
- [JMAPI CLI 명령](#)
- [예시](#)

JMAPI의 일반적인 용도

- Snowcone 디바이스 주문 자동화
- 매니페스트 파일 다운로드
- 잠금 해제 파일 다운로드
- 현재 Snowcone 작업 목록 작성
- Snowcone 작업 완료 보고서 다운로드

JMAPI 필수 문자열

작업 관리 API를 통해 주문할 때는 예시와 함께 표시된 다음 필수 파라미터를 사용합니다.

```
--job-type
--resources
```

```
--address-id
--region
--role-arn
--kms-key-arn
--shipping-option
--device-type
--description
```

JMAPI 엔드포인트

API 엔드포인트

각 엔드포인트를 직접 호출하기 위한 형식은 `snowballEdge.region.amazonaws.com`입니다. 다음은 엔드포인트의 분류를 이해하는 데 도움이 되는 몇 가지 예시입니다.

Example

리전	엔드포인트
미국 동부(버지니아 북부)	snowball.us-east-1.amazonaws.com
미국 서부(오레곤)	snowball.us-west-2.amazonaws.com

JMAPI CLI 명령

작업 관리 API

다음은 작업 관리 API에 대해 수행할 수 있는 CLI 직접 호출입니다.

명령	예시
작업 나열	aws snowball list-jobs
작업 설명	aws snowball describe-job --job-id <i>[JOB ID]</i>
주소 설명	aws snowball describe-address --address-id

명령	예시
주소 생성	<code>aws snowball create-address --cli-input-json file://create-address.json</code>
작업 생성	<code>aws snowball create-job --cli-input-json file://create-job.json</code>
작업 취소	<code>aws snowball cancel-job --job-id <i>[JOB ID]</i></code>

예시

다음은 작업 관리 API를 사용하는 명령의 예시입니다.

KMS JSON 예시

다음 JSON 예시는 AWS KMS 정책 파일을 사용하기 위한 올바른 형식의 JSON 파일입니다.

```
{
  "KeyMetadata": {
    "Origin": "AWS_KMS",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Description": "",
    "KeyManager": "CUSTOMER",
    "Enabled": true,
    "KeyUsage": "ENCRYPT_DECRYPT",
    "KeyState": "Enabled",
    "CreationDate": 1502910355.475,
    "Arn": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333"
  }
}
```

주소 생성 예시

다음 예시는 주소 생성을 위한 명령 형식을 지정하는 방법과 주소 생성 성공 시 응답에 대한 내용을 보여줍니다.


```
aws snowball create-address --address "Name=Bob,Company=AWS,Street1=1234 Fake
St.,City=All,StateOrProvince=Any,Country=US,PostalCode=12345,PhoneNumber=1234567890"
```

출력 예시

```
{
  "AddressId": "ADID3be640c8-1111-1111-1111-917f201ffa42"
}
```

작업 생성 예시

다음 명령은 create-job 명령을 실행하기 위한 예시 명령을 보여줍니다.

```
aws snowball create-job --job-type IMPORT --resources file://path/to/resources.json
--address-id ADID3be640c8-1111-1111-1111-917f201ffa42 --region us-east-1 --role-
arn arn:aws:iam::123456789123:role/example_role --kms-key-arn arn:aws:kms:us-
west-2:000000000000:key/Example --snowball-capacity-preference T14 --device-
configuration file://path/to/configuration.json --shipping-option SECOND_DAY --
snowball-type SNC1_SSD
```

위 create-job 명령을 실행하면 T14 용량 기본 설정이 적용되고 SECOND_DAY 배송 옵션이 있는 SNC1_SSD 유형 Snowcone 디바이스를 사용하여 us-east-1 리전에 가져오기 작업이 생성됩니다.

Amazon EC2 작업에 사용할 AMI 내보내기

이 섹션에서는 Amazon Machine Image(AMI)를 내보내 AWS Snowcone 디바이스의 Amazon EC2 호환 컴퓨팅 인스턴스와 함께 사용하는 방법에 대한 개요를 제공합니다.

주제

- [디바이스에서 시작된 컴퓨팅 인스턴스에 연결되도록 AMI를 구성하여 SSH 사용](#)
- [콘솔을 사용하여 작업 생성](#)
- [AWS CLI에서 작업 생성](#)

디바이스에서 시작된 컴퓨팅 인스턴스에 연결되도록 AMI를 구성하여 SSH 사용

Secure Shell(SSH)을 사용하여 Snowcone 디바이스의 컴퓨팅 인스턴스에 연결하려면 작업을 생성하기 전에 다음 절차를 수행하여야 합니다. 이 절차는 작업을 생성하기 전 AMI에 SSH 키를 추가합니다.

또한, 이 절차를 사용하여 작업의 AMI로 사용하려는 인스턴스에 애플리케이션을 설정하는 것이 좋습니다.

⚠ Important

이 절차를 따르지 않으면 Snowcone 디바이스의 수령 시 인스턴스를 SSH와 연결할 수 없습니다.

SSH 키를 AMI에 넣기

- 호환되는 AMI 이미지를 사용하여 새 인스턴스를 AWS 클라우드에서 시작합니다(참조 <https://docs.aws.amazon.com/snowball/latest/developer-guide/using-ami.html>).

인스턴스의 시작 시 인스턴스에 할당하는 스토리지 크기가 Snowcone 디바이스에서 나중에 사용할 용도로 적합한지 확인합니다. Amazon EC2 콘솔에서는 4단계: 스토리지 추가에서 이 작업을 수행합니다. Snowcone에서 컴퓨팅 인스턴스 스토리지 볼륨에 지원되는 크기의 목록은 '[ec2-snowcone-limits](#)'를 참조하십시오.

- Snowcone에서 실행하고 싶은 애플리케이션을 설치 및 구성한 후 예상대로 작동하는지 테스트합니다.
- 이 인스턴스를 생성하기 위해 SSH 키 페어에 사용한 PEM/PPK 파일의 사본을 만듭니다. 이 파일을 Snowcone과 통신하기 위해 사용하려는 서버에 저장합니다. 이 파일은 디바이스에서 시작된 인스턴스에 SSH를 사용하여 연결할 때 필요하므로 이 파일의 경로를 적어둡니다.
- 인스턴스를 AMI로 저장합니다. 자세한 내용은 [Amazon EBS 지원 Linux AMI 생성](#)을 참조하십시오.
- SSH를 사용하여 연결하려는 각 인스턴스에 대해 이 절차를 반복합니다. 서로 다른 SSH 키 페어의 사본이 있는지 확인하고 관련된 AMI에 유의합니다.

콘솔을 사용하여 작업 생성

다음 단계는 Snow Family 디바이스를 주문하기 위한 작업을 생성하는 것입니다. 작업의 유형은 클러스터를 포함해 어떤 유형이어도 됩니다. 를 사용하여 [AWS Snow 패밀리 관리 콘솔Snow Family 장치 주문을 위한 작업 생성하기](#) 참조에 제공된 지침을 따르십시오. 작업 생성 마법사의 3단계: 작업 세부 사항 제공 페이지에 따라 후속 단계를 추가합니다.

- EC2로 컴퓨팅 활성화를 선택합니다.
- AMI 추가를 선택합니다.

3. 이때 열리는 대화 상자에서 AMI를 선택한 다음, 저장을 선택합니다.
4. 디바이스 유형에 따라 작업에 AMI를 총 20개까지 추가할 수 있습니다.
5. 일반적인 작업 생성 절차를 계속 진행합니다.

AWS CLI에서 작업 생성

AWS Command Line Interface(AWS CLI)를 사용하여 작업을 생성할 수도 있습니다. 이렇게 하려면 터미널을 열고 다음 명령을 실행하여 빨간색으로 표시된 텍스트를 실제 값으로 교체합니다.

```
aws snowball create-job --job-type IMPORT --resources '{"S3Resources": [{"BucketArn": "arn:aws:s3:::bucket-name"}], "Ec2AmiResources": [{"AmiId": "ami-12345678"}]}' --description Example --address-id ADIEXAMPLE60-1234-1234-5678-41fEXAMPLE57 --kms-key-arn arn:aws:kms:us-west-2:012345678901:key/eEXAMPLE-1234-1234-5678-5b4EXAMPLE8e --role-arn arn:aws:iam::123456789012:role/snowcone-import-snowcone-role --shipping-option SECOND_DAY --snowball-type SNC1_HDD --snowball-capacity-preference T8 --device-configuration '{"SnowconeDeviceConfiguration":{"WirelessConnection":{"IsWifiEnabled": false}}}'
```

디바이스가 도착하면 잠금을 해제한 후 Snowball Edge 클라이언트를 사용하여 로컬 자격 증명을 얻습니다. 자세한 내용은 [자격 증명 받기](#)(를) 참조하세요.

AWS Snowcone에 대한 배송 고려 사항

아래에서 AWS Snowcone 디바이스에 대한 배송 처리 방법에 대한 정보를 알아보고 지원되는 각 AWS 리전 리전을 보여주는 목록을 확인할 수 있습니다. 작업에 대해 선택하는 배송료는 해당 또는 그 작업에 사용되는 AWS Snowcone 디바이스의 전송과 수신 모두에 적용됩니다. 배송 요금에 대한 자세한 내용은 [AWS Snowcone 요금](#)을 참조하십시오.

주제

- [AWS Snowcone 디바이스 배송 준비](#)
- [리전 기반 배송 제한](#)
- [AWS Snowcone 디바이스 배송](#)

Snow Family 디바이스를 주문하기 위한 작업을 생성할 때 배송 주소와 배송 속도를 지정합니다. 이 배송 기간은 작업을 생성한 날로부터 AWS Snowcone 디바이스를 수신하는 데 걸리는 시간을 나타내지 않습니다. AWS에서 배송 주소로 디바이스가 배송되는 데 걸리는 시간만 나타냅니다. 작업 유형(일반적으로 내보내기 작업이 가져오기 작업보다 더 오래 걸림)을 포함한 다양한 요소에 따라 달라지는 처리 시간은 여기에 포함되지 않습니다. 또한 배송 업체에서는 일반적으로 발신 AWS Snowcone 디바이스를 하루에 한 번만 픽업합니다. 따라서 배송 전 처리에 하루 이상 걸릴 수 있습니다.

Note

Snow Family 디바이스는 디바이스를 주문한 동일한 AWS 리전로만 반품할 수 있습니다. 일부 AWS 리전에서는 Snow Family 디바이스를 주문한 국가가 아닌 다른 국가로 보내는 것을 지원합니다. 자세한 정보는 [리전 기반 배송 제한](#) 섹션을 참조하십시오.

AWS Snowcone 디바이스 배송 준비

아래에서는 Snowcone을 준비하여 AWS에 반송하는 방법을 설명합니다.

AWS Snowcone 디바이스의 배송을 준비하려면 다음을 수행합니다.

1. AWS Snowcone 디바이스에 대해 이 작업의 모든 데이터 전송(업로드 또는 다운로드)을 완료해야 합니다. [디바이스의 잠금을 해제합니다.](#)
2. 표시등 근처, 네트워크 포트 맞은편에 있는 디바이스 전면의 전원 버튼을 누릅니다. 디바이스의 전원이 꺼지는 데 20초 정도 걸립니다.

전원을 끄고 Snowcone 디바이스의 플러그를 뽑았는데 약 1분이 지나도 E Ink 화면에 배송 정보가 나타나지 않는 경우 [Snow Family 디바이스 반품 문제 해결](#)을 참조하십시오.

리전 기반 배송 제한

Snow Family 디바이스를 주문하기 위한 작업을 생성하려면 먼저 데이터가 AWS 리전 있는 콘솔에서 콘솔에 로그인해야 합니다. Snow Family 디바이스는 아시아 태평양(인도)에서 아시아 태평양(호주)으로 배송되는 경우와 같이 해외 국가 간에는 배송되지 않습니다.

유럽 연합(EU) 회원국에서는 국가 간 배송에 대한 예외가 있습니다. 유럽 AWS 리전에서의 데이터 전송의 경우, 당사는 다음에 나열된 EU 회원국으로만 디바이스를 배송합니다.

- 오스트리아, 벨기에, 불가리아, 크로아티아, 사이프러스, 체코 공화국, 덴마크, 에스토니아, 핀란드, 프랑스, 독일, 그리스, 헝가리, 이탈리아, 아일랜드, 라트비아, 리투아니아, 룩셈부르크, 몰타, 네덜란드, 폴란드, 포르투갈, 루마니아, 슬로바키아, 슬로베니아, 스페인, 스웨덴.

동일한 국가 내 국내 배송은 허용됩니다. 예시:

- 영국 리전에서의 데이터 전송의 경우, 당사는 디바이스를 영국 국내로 배송합니다.
- 아시아 태평양(뭄바이)에서 데이터를 전송할 경우 인도 내에서만 디바이스를 배송합니다.

Note

AWS에서는 Snow Family 디바이스를 우체국 박스로 배송하지 않습니다.

AWS Snowcone 디바이스 배송

E Ink 디스플레이의 선불 배송 라벨에는 AWS Snowcone 디바이스를 반송할 정확한 주소가 포함되어 있습니다. AWS Snowcone 디바이스 반송 방법에 대한 자세한 내용은 [the section called “배송 업체”](#) 섹션을 참조하십시오. AWS Snowcone 디바이스는 AWS 분류 시설로 배송되었다가 다시 AWS 데이터 센터로 전달됩니다. 리전의 배송 업체를 통해 패키지를 추적할 수 있습니다. AWS Snow 패밀리 관리 콘솔을 사용하여 작업에 대한 상태 변경을 추적할 수 있습니다.

전원을 끄고 Snowcone 디바이스의 플러그를 뽑았는데 약 1분이 지나도 E Ink 화면에 배송 정보가 나타나지 않는 경우 [Snow Family 디바이스 반품 문제 해결](#)을 참조하십시오.

배송 업체

Snow Family 디바이스를 주문하기 위한 작업을 생성할 때 AWS Snowcone 디바이스를 배송할 주소를 입력합니다. 고객 리전을 지원하는 배송 업체에서 AWS에서 고객에게 보내는 AWS Snowcone 디바이스 배송과 고객이 AWS로 보내는 반송을 처리합니다. AWS Snowcone 디바이스를 배송할 때 추적 번호가 제공됩니다. [AWS Snow 패밀리 관리 콘솔](#)의 작업 대시보드 또는 작업 관리 API에 대한 API 호출을 사용하여 각 작업의 추적 번호와 추적 웹사이트에 대한 링크를 찾을 수 있습니다.

다음은 리전별로 AWS Snowcone 디바이스에 대해 지원되는 배송 업체 목록입니다.

- 인도의 경우 Blue Dart가 배송 업체입니다.
- 한국, 일본, 호주, 인도네시아, 이스라엘, 싱가포르의 경우 Kuehne + Nagel이 배송 업체입니다.
- 중국의 경우 S.F. Express가 배송 업체입니다.
- 다른 모든 리전의 경우 [UPS](#)가 배송 업체입니다.

캐나다, EU, 남아프리카, 미국 내 AWS Snowcone 픽업

캐나다, EU, 남아프리카, 미국에서는 UPS의 AWS Snowcone 디바이스 픽업과 관련하여 다음 정보에 유의하십시오.

- UPS에 직접 픽업을 예약하여 UPS에서 AWS Snowcone 디바이스를 직접 픽업하도록 예약하거나, AWS에 반송하도록 UPS 패키지 인도 시설에 가져다 줍니다
- E Ink 디스플레이의 선불 UPS 배송 라벨에는 AWS Snowcone 디바이스를 반송할 정확한 주소가 포함되어 있습니다.
- AWS Snowcone 디바이스는 AWS 분류 시설로 배송되었다가 다시 AWS 데이터 센터로 전달됩니다. UPS는 발송물에 대한 추적 번호를 자동으로 제공합니다.

Snow Family 디바이스에 대한 UPS 서비스는 특정 국가 내에서만 국내 서비스를 제공합니다.

브라질에서 AWS Snowcone 픽업

브라질에서는 UPS의 Snowcone 픽업과 관련하여 다음 정보에 유의하십시오.

- Snowcone을 반환할 준비가 되면 0800-770-9035으로 연락하여 UPS 픽업을 예약하십시오.
- Snowcone은 26개 주와 연방구가 포함된 브라질 국내에서 사용할 수 있습니다.
- Cadastro Nacional de Pessoa Juridica(CNPJ) 세금 ID가 있는 경우에는 작업을 생성하기 전에 이 ID를 알고 있어야 합니다.

- Snowcone 디바이스를 반송하려면 관련 문서를 발행해야 합니다. ICMS 등록에 따라 거주하고 있는 주에서 어떤 문서가 필요한지 세무 부서에 확인하시기 바랍니다.
- 상파울루 내 - 일반적으로 비 ICMS 신고서와 전자 세금 계산서(NF-e)가 필요합니다.
- 상파울루 이외 리전 - 일반적으로 다음이 필요합니다.
 - ICMS를 제외한 세금 신고서
 - nota fiscal avulsa
 - 전자 세금 계산서(NF-e)

Note

ICMS를 제외한 세금 신고서의 경우에는 보관용 1부와 발송용 3부 등 총 4부를 복사하는 것이 좋습니다.

이스라엘 내 AWS Snowcone 픽업

이스라엘에서는 snowball-shipping@amazon.com 이메일로 AWS에 연락하여 픽업을 예약하십시오. 제목에 Snowcone 픽업 요청을 입력하고 다음 정보를 포함하십시오.

- 작업 ID - 반품하는 Snow 디바이스와 관련된 작업 ID입니다. 작업 ID는 [AWS Snow 패밀리 관리 콘솔](#)에서 찾을 수 있습니다.
- 픽업 주소 - 디바이스를 픽업할 주소입니다.
- 픽업 날짜 - 디바이스를 픽업하려는 가장 빠른 날짜입니다.
- 연락처 세부 정보 - 운송 서비스인 Kuehne + Nagel이 필요에 따라 픽업 관련 정보를 제공하는 데 사용할 수 있는 이름, 이메일 주소, 현지 전화번호입니다.

곧 디바이스를 픽업하도록 제공한 주소에 대한 정보가 포함된 AWS Support의 후속 이메일을 수신할 것입니다. 디바이스를 배송할 준비를 하고 보통 현지 시간으로 12:00 이전에 픽업할 준비를 하십시오. [AWS Snowcone 디바이스 배송 준비](#) 섹션을 참조하십시오.

영국 내 AWS Snowcone 픽업

영국의 경우 UPS가 AWS Snowcone 디바이스를 픽업할 수 있도록 다음 정보를 옆두에 두십시오.

- UPS에 직접 픽업을 예약하여 UPS에서 AWS Snowcone 디바이스를 직접 픽업하도록 예약하거나, AWS에 반송하도록 UPS 패키지 인도 시설에 가져다 줍니다.

- E Ink 디스플레이의 선불 UPS 배송 라벨에는 AWS Snowcone 디바이스를 반송할 정확한 주소가 포함되어 있습니다.
- AWS Snowcone 디바이스는 AWS 분류 시설로 배송되었다가 다시 AWS 데이터 센터로 전달됩니다. UPS에서는 작업에 대한 추적 번호를 자동으로 보고합니다.

⚠ Important

AWS에서 사용자 개인에게 달리 지시하지 않는 한 AWS Snowcone 디바이스에 별도의 배송 라벨을 부착하면 안 됩니다. 항상 디바이스의 E Ink 디스플레이에 표시된 배송 라벨을 사용하십시오.

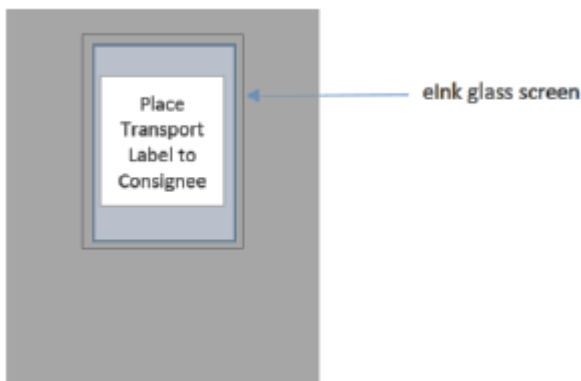
Snow 제품군에 대한 UPS 서비스는 특정 국가 내에서만 제공됩니다.

ℹ Note

2021년 1월부터 영국은 더 이상 EU에 속하지 않습니다. 영국과 다른 EU 국가 간의 주문은 국제 주문이며, 일반 배송 절차가 아닌 경우 특별 국제 절차를 통해서만 승인됩니다. 승인을 받아 EU 국가에서 LHR로 또는 영국에서 EU 국가로 디바이스를 반품하는 고객은 먼저 <snowball-shipping@amazon.com>에 반품을 요청해야 합니다. 그러면 UPS와 픽업/드롭 오프를 협의하기 전에 상업 송장을 제공할 수 있습니다.

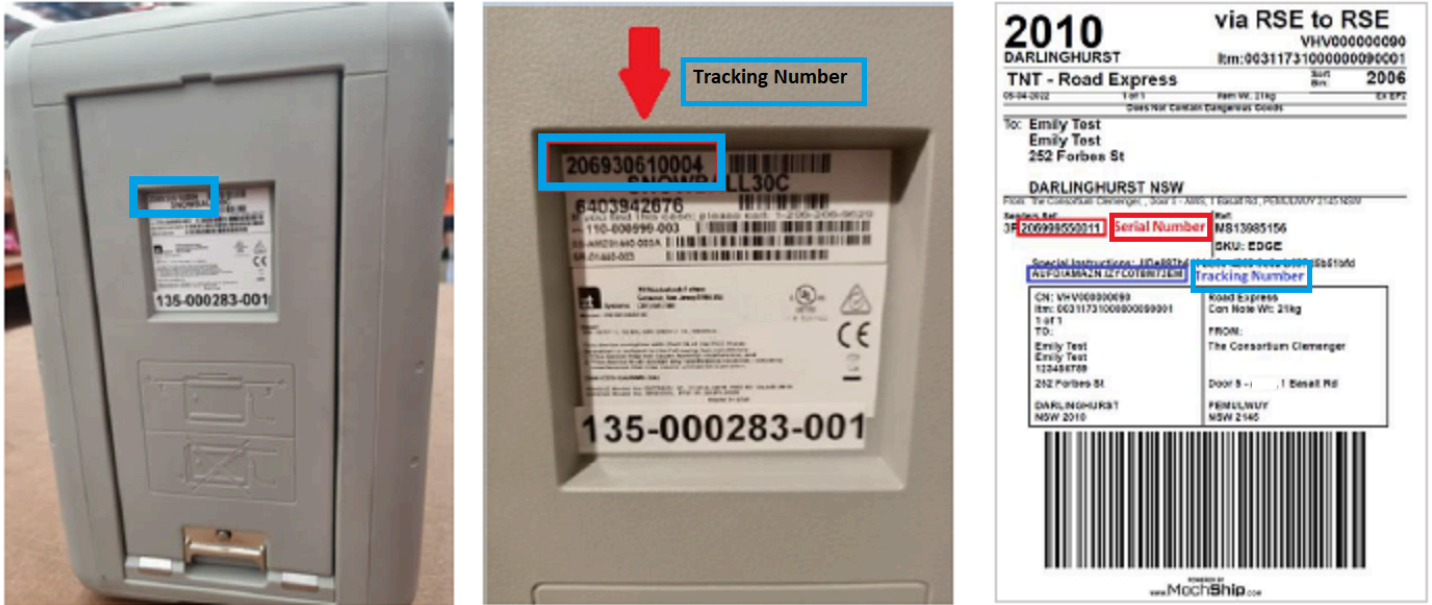
오스트레일리아에서 AWS Snowcone 픽업

오스트레일리아에서는 AWS Snowcone 디바이스를 AWS로 반송하는 경우 Snow 디바이스의 eInk 라벨 위에 반품 운송 라벨(이 지침이 들어 있는 파우치에 있음)을 부착하십시오.



Snow 디바이스와 함께 반품 라벨을 받지 못한 경우, 디바이스 일련 번호 또는 참조 번호를 포함하여 knau.snowball_return@kuehne-nagel.com으로 이메일을 보내주십시오.

Example where to locate tracking number and serial number



Snow 디바이스를 반품하려면 다음 정보를 포함하여 knau.snowball_return@kuehne-nagel.com으로 이메일을 보내주십시오.

- 이름.
- 추적 번호(파란색으로 표시)
- 디바이스 일련 번호(빨간색으로 표시)
- 전체 컬렉션 주소
- 픽업 시 담당자에게 연락하십시오
- 픽업 시 연락처 전화번호
- 수거 날짜
- 수집 기간(업무 시간 내 최소 3시간)

Note

수거 날짜 및 시간은 영업일 업무 시간 내여야 합니다.
Kuehne + Nagel 팀이 픽업 요청 접수 확인을 위해 회신할 것입니다.

인도의 AWS Snowcone 픽업

인도에서는 Blue Dart가 Snowcone 디바이스를 픽업하러 옵니다. Snowcone 디바이스를 반환할 준비가 되면 디바이스를 끄고 반환 배송을 위해 준비합니다. 픽업 일정을 잡으려면 제목에 Snowcone 픽업 요청을 기재하여 snowball-pickup@amazon.com으로 이메일을 보내십시오. 이메일 본문에 다음 정보를 포함합니다.

- 작업 ID - AWS에 반송하려는 Snowcone과 연결된 작업 ID입니다.
- AWS 계정 ID - 작업을 생성한 AWS 계정에 대한 ID입니다.
- 가장 빠른 픽업 시간(현지 시간) - Snowcone 픽업을 원하는 가장 빠른 시간입니다.
- 가장 늦은 픽업 시간(현지 시간) - Snowcone 픽업을 원하는 가장 늦은 시간입니다.
- 특별 지침(선택 사항) - 픽업을 조정하기 위한 연락처 세부 정보를 포함하여 Snowcone 픽업에 대한 특별 지침입니다.

Snowcone 팀이 Blue Dart의 픽업을 주선하고 확인 이메일을 보냅니다. Blue Dart가 종이 배송 라벨을 제공하고 Snowcone 디바이스를 픽업합니다.

Important

인도에서 Snowcone을 사용하는 경우 해당 주에 관련 세금 서류를 모두 제출하십시오.

한국 내 AWS Snowcone 픽업

한국에서 Kuehne + Nagel이 픽업을 처리합니다. 디바이스를 반송할 준비가 되면 제목 줄에 Snowball 픽업 요청이라고 입력한 후 snowball-shipping@amazon.com에 이메일을 보내 픽업을 예약할 수 있습니다. 이메일 본문에 다음 정보를 포함합니다.

- 작업 ID - AWS에 반송하려는 Snowcone과 연결된 작업 ID입니다.
- 픽업 주소 - 디바이스를 픽업할 주소입니다.
- 픽업 날짜 - 디바이스를 픽업하려는 가장 빠른 날짜입니다.
- 연락처 세부 정보 - 필요한 경우 Kuehne + Nagel이 사용자와 연락하는 데 사용할 수 있는 이름, 이메일 주소 및 현지 전화 번호입니다.

곧 디바이스를 픽업하도록 제공한 주소에 대한 정보가 포함된 AWS Support의 후속 이메일을 수신할 것입니다. 디바이스의 배송을 준비([AWS Snowcone 디바이스 배송 준비](#) 참조)하고 현지 시간으로 보통 1300시~1500시 사이에 픽업할 수 있도록 준비하십시오.

홍콩에서 AWS Snowcone 픽업

홍콩에서는 S.F. Express에서 픽업을 처리합니다. 기기를 반납할 준비가 되면 제목에 Snowcone 픽업 요청을 포함하여 snowball-shipping-ap-east-1@amazon.com 으로 이메일을 보내주시면 픽업 일정을 잡을 수 있습니다. 이메일 본문에 다음 정보를 포함합니다.

- 작업 ID
- AWS 계정 ID
- 담당자 이름
- 연락처 전화 번호
- 연락처 이메일 주소
- 디바이스 픽업을 원하는 날짜
- 원하는 가장 빠른 픽업 시간
- 원하는 가장 늦은 픽업 시간
- 픽업 주소

Note

S.F. Express에서 픽업 날짜를 조정 한 후에는 일정을 변경할 수 없습니다.

디바이스는 S.F. Express를 통해 AWS에 배송됩니다. 반송물의 S.F. Express 추적 번호를 사용하여 배송이 언제 완료되었는지 알 수 있습니다.

인도네시아, 일본, 싱가포르에서의 AWS Snowcone 픽업

인도네시아, 일본, 싱가포르에서는 디바이스를 반품할 준비가 되면 반품용 E Ink 라벨에 표시된 QR 코드를 휴대폰으로 스캔하십시오. 그러면 이메일 템플릿으로 바로 이동할 수 있습니다. 픽업 날짜, 시간 및 연락처 세부 정보를 입력하십시오.

RETURN

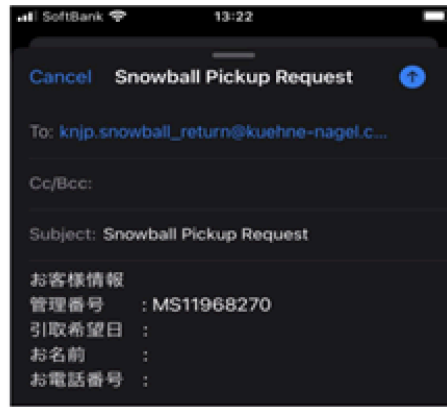
AWS Jobs ID QF6LNZGKTZPF
 シリアル番号 2R 207138750022
 管理番号 MS14003547



返送のご案内

“以下のQRコードをスキャンし情報を入力の上、
 メールにてご連絡をお願い致します。”

送信先アドレス knjp.snowball_return@kuehne-nagel.com
 件名 Snow Ball Pickup Request
 お客様情報
 管理番号 : MS14003547
 引取希望日 : 要記入
 お名前 : 要記入
 お電話番号 : 要記入



배송 기간

국가마다 이용 가능한 배송 기간이 다릅니다. 이러한 배송 기간은 AWS Snowcone 디바이스를 배송 중인 국가에 따라 다릅니다. 배송 기간은 다음과 같습니다.

- 호주, 인도네시아, 일본, 싱가포르, 한국 - 이러한 국가 내에서 배송하는 경우 1~3일의 표준 배송 속도를 이용할 수 있습니다.
- 브라질 - 브라질 내에서 배송하는 경우 영업일 기준 2일 이내 영업 시간에 배달하는 UPS Domestic Express Saver 배송을 이용할 수 있습니다. 배송 속도는 국경에서 지연되는 시간의 영향을 받을 수 있습니다.
- EU(유럽 연합) - EU 내 국가로 배송할 경우 특급 배송을 이용할 수 있습니다. 일반적으로 Snowcone 디바이스 특급 배송은 약 1일 이내에 전달됩니다. 또한 EU 내 대부분의 국가에서는 표준 배송을 이용하며 일반적으로 편도 기준으로 1주일이 채 걸리지 않습니다.
- 인도 - 인도 내에서 배송 시 AWS Snowcone 디바이스는 AWS가 모든 관련 세금 문서를 수신한 지 7일 이내(근무일 기준)에 발송됩니다.
- 이스라엘 - 이스라엘로 배송하는 경우 당일 배송이 가능합니다.
- 미국(US) 및 캐나다 - 미국 또는 캐나다 내에서 배송하는 경우 1일 배송 및 2일 배송을 이용할 수 있습니다.
- 영국(UK) - 영국 내에서 배송하는 경우 특급 배송을 이용할 수 있습니다. 일반적으로 Snowcone 디바이스 특급 배송은 약 1일 이내에 전달됩니다. 또한 표준 배송을 이용하면 일반적으로 편도 기준으로 1주일이 채 걸리지 않습니다.

Snowcone 디바이스의 소프트웨어 업데이트

AWS 보유하고 있는 Snow Family 장치에 새 소프트웨어를 사용할 수 있게 되면 알려 드립니다. 알림은 이메일을 통해 AWS Health Dashboard, 그리고 CloudWatch 이벤트로 제공됩니다. 이메일 알림은 Amazon Web Services, Inc.에서 Snow Family 디바이스를 주문할 때 사용한 AWS 계정에 연결된 이메일 주소로 전송됩니다. 알림을 받으면 이 주제의 지침에 따라 가능한 한 빨리 업데이트를 다운로드하여 설치하여 디바이스 사용이 중단되지 않도록 하십시오. 에 대한 AWS Health Dashboard 자세한 내용은 [AWS Health 사용 설명서를](#) 참조하십시오. CloudWatch 이벤트에 대한 자세한 내용은 [Amazon CloudWatch Events 사용 설명서를](#) 참조하십시오.

온프레미스 환경의 Snowcone 장치에서 소프트웨어 업데이트를 AWS 다운로드하고 설치할 수 있습니다. 이러한 업데이트는 백그라운드에서 발생합니다. 최신 소프트웨어가 디바이스로 AWS 안전하게 다운로드되는 동안에도 평소처럼 디바이스를 계속 사용할 수 있습니다. 하지만 다운로드한 업데이트를 적용하려면 디바이스에서 실행 중인 서비스를 중지하고 업데이트가 완료된 후 다시 시작해야 합니다.

Snowball Edge/Snowcone 디바이스 (어플라이언스) AWS 에 대해 제공하는 소프트웨어 업데이트는 서비스 약관의 섹션 9에 따른 어플라이언스 소프트웨어입니다.

소프트웨어 업데이트는 AWS를 대신하여 해당 어플라이언스에 소프트웨어 업데이트를 설치할 목적으로만 제공됩니다. 귀하는 (i) 해당 어플라이언스에 소프트웨어 업데이트를 설치하는 데 필요한 것 이외의 소프트웨어 업데이트 사본을 만들거나 (ii) 소프트웨어 업데이트에 적용된 암호화를 포함하되 이에 국한되지 않는 소프트웨어 업데이트의 기능 또는 조치를 우회 또는 비활성화하거나 이를 시도하지 않으며, 제3자가 이를 수행하거나 수행을 시도하는 것을 허용하거나 승인하지 않습니다. 해당 어플라이언스에 소프트웨어 업데이트를 설치한 후에는 어플라이언스에 소프트웨어 업데이트를 설치하는 데 사용된 모든 미디어에서 소프트웨어 업데이트를 삭제하는 데 동의합니다.

Warning

업데이트를 설치하기 전에 디바이스의 모든 활동을 일시 중지하는 것이 좋습니다. 디바이스를 업데이트하고 다시 시작하면 인스턴스 실행이 중단되고 로컬 Amazon S3 버킷에 대한 모든 쓰기가 중단됩니다.

주제

- [필수 조건](#)
- [업데이트 다운로드](#)
- [업데이트 설치](#)

- [SSL 인증서 업데이트](#)
- [Snow Family 디바이스의 Amazon Linux 2 AMI 업데이트](#)

필수 조건

디바이스를 업데이트하기 전에 다음 사전 조건이 충족되어야 합니다.

- 작업을 생성했고, 온프레미스에 디바이스가 있으며, 이를 잠금 해제했습니다. 자세한 내용은 [시작하기](#) 섹션을 참조하십시오.
- Snowcone 디바이스 업데이트는 Snowball Edge 클라이언트를 통해 이루어집니다. 업데이트하려는 디바이스에 네트워크로 연결된 로컬 환경의 컴퓨터에 최신 버전의 Snowball Edge 클라이언트를 다운로드하여 설치해야 합니다. 자세한 내용은 [AWS Snowball Edge 클라이언트 사용](#)을 참조하십시오.
- (선택 사항) Snowball Edge 클라이언트에 대한 프로필을 구성하는 것이 좋습니다. 자세한 내용은 [Snowball Edge 클라이언트의 프로필 구성](#) 섹션을 참조하십시오.

이러한 작업을 완료했다면 Snowcone 디바이스에 대한 업데이트를 다운로드 및 설치할 수 있습니다.

업데이트 다운로드

Snow Family 장치용 업데이트를 다운로드할 수 있는 두 가지 기본 방법은 다음과 같습니다.

- 특정 Snowball Edge 클라이언트 명령을 사용하여 언제든지 수동 업데이트를 트리거할 수 있습니다.
- 프로그래밍 방식으로 디바이스를 자동으로 업데이트할 시간을 결정할 수 있습니다.

다음 절차는 수동 업데이트 다운로드 프로세스를 간략하게 설명합니다. Snowcone 디바이스 자동 업데이트에 대한 자세한 내용은 [Snowcone 업데이트](#)의 `configure-auto-update-strategy`를 참조하십시오.

Note

디바이스가 인터넷에 액세스할 수 없는 경우 API를 사용하여 업데이트 파일을 다운로드할 수 있습니다. [GetSoftwareUpdates](#) 그런 다음 `uri` 파라미터를 `download-updates` 사용하여 호출할 때 다음 예와 같이 로컬 파일 위치를 가리키십시오.

```
snowballEdge download-updates --uri file:///tmp/local-update
```

Windows 운영 체제의 경우 다음과 같이 uri 매개 변수 값의 형식을 지정합니다.

```
snowballEdge download-updates --uri file://C:/path/to/local-update
```

Snowcone 소프트웨어 업데이트를 확인하고 다운로드하려면

1. 터미널 창을 열고 Snowcone 디바이스가 describe-device 명령으로 잠금 해제되었는지 확인합니다. 디바이스가 잠긴 경우 unlock-device 명령을 사용하여 잠금 해제합니다. 자세한 내용은 [Snow Family 장치 잠금 해제를 참조하십시오](#).
2. 디바이스가 잠금 해제되면 snowballEdge check-for-updates 명령을 사용합니다. 이 명령은 Snowball Edge 소프트웨어의 최신 사용 가능한 버전을 반환하고, 디바이스의 현재 설치된 버전을 반환합니다.
3. 디바이스 소프트웨어가 최신이 아닌 경우 snowballEdge download-updates 명령을 실행합니다.

Note

기기가 인터넷에 연결되어 있지 않은 경우 먼저 [GetSoftwareUpdates](#) API를 사용하여 업데이트 파일을 다운로드하세요. 그런 다음 다음 예와 같이 다운로드한 파일의 로컬 경로가 포함된 uri 파라미터를 사용하여 snowballEdge download-updates 명령을 실행합니다.

```
snowballEdge download-updates --uri file:///tmp/local-update
```

Windows 운영 체제의 경우 다음과 같이 uri 매개 변수 값의 형식을 지정합니다.

```
snowballEdge download-updates --uri file://C:/path/to/local-update
```

4. snowballEdge describe-device-software 명령을 사용하여 다운로드 상태를 확인할 수 있습니다. 업데이트를 다운로드하는 동안 이 명령을 사용하면 상태가 표시됩니다.

Example **describe-device-software** 명령 출력

```
Install State: Downloading
```

업데이트 설치

업데이트를 다운로드한 후 이를 설치하고 디바이스를 재시작해야 업데이트가 적용됩니다. 다음 절차는 수동으로 업데이트를 설치하는 방법을 안내합니다.

Note

소프트웨어 업데이트를 설치하기 전에 디바이스의 모든 활동을 일시 중단합니다. 업데이트를 설치하면 인스턴스 실행이 중지되고 디바이스의 Amazon S3 버킷에 대한 모든 쓰기가 중단됩니다. 이로 인해 데이터가 손실될 수 있습니다.

독립형 Snow Family 디바이스에 이미 다운로드된 소프트웨어 업데이트를 설치하려면

1. 터미널 창을 열고 Snowcone 디바이스가 `describe-device` 명령으로 잠금 해제되었는지 확인합니다. 디바이스가 잠긴 경우 `unlock-device` 명령을 사용하여 잠금 해제합니다. 자세한 내용은 [Snow Family 디바이스 잠금 해제를](#) 참조하십시오.
2. `list-services` 명령을 실행하여 장치에서 사용할 수 있는 서비스를 확인하세요. 이 명령은 기기에서 사용할 수 있는 각 서비스의 서비스 ID를 반환합니다.

```
snowballEdge list-services
```

Example **list-services** 명령 출력의

```
{
  "ServiceIds" : [ "greengrass", "fileinterface", "s3", "ec2", "s3-snow" ]
}
```

3. `list-services` 명령으로 식별된 각 서비스 ID에 대해 `describe-service` 명령을 실행하여 상태를 확인합니다. 이 정보를 사용하여 중지할 서비스를 식별할 수 있습니다.


```
snowballEdge describe-service --service-id service-id
```

Example `describe-service` 명령 출력의

```
{
  "ServiceId" : "s3",
  "Status" : {
    "State" : "ACTIVE"
  },
  "Storage" : {
    "TotalSpaceBytes" : 99608745492480,
    "FreeSpaceBytes" : 99608744468480
  },
  "Endpoints" : [ {
    "Protocol" : "http",
    "Port" : 8080,
    "Host" : "192.0.2.0"
  }, {
    "Protocol" : "https",
    "Port" : 8443,
    "Host" : "192.0.2.0",
    "CertificateAssociation" : {
      "CertificateArn" : "arn:aws:snowball-
device::certificate/6d955EXAMPLEdb71798146EXAMPLE3f0"
    }
  } ]
}
```

이 출력은 s3 서비스가 활성 상태이며 `stop-service` 명령을 사용하여 중지해야 함을 보여줍니다.

- 명령을 사용하면 `stop-service` 명령 출력에 State 이름 값이 ACTIVE 있는 각 서비스를 중지할 수 있습니다. `list-services` 둘 이상의 서비스가 실행 중인 경우 계속하기 전에 각 서비스를 중지하십시오.

Note

Amazon S3 어댑터, Amazon EC2 및 IAM 서비스는 중지할 수 없습니다. AWS STS Snow Family 디바이스에서 Amazon S3 호환 스토리지가 실행 중인 경우 업데이트를 설치하기

전에 중지합니다. Snow Family 디바이스의 Amazon S3 호환 스토리지는 s3-snow 다음과 같습니다serviceId.

```
snowballEdge stop-service --service-id service-id --device-ip-addresses snow-device-1-ip-address snow-device-2-ip-address snow-device-3-ip-address --manifest-file path/to/manifest/file.bin --unlock-code 29-character-unlock-code --endpoint https://snow-device-ip-address
```

Example **stop-service** 명령 출력의

Stopping the AWS service on your Snowball Edge. You can determine the status of the AWS service using the describe-service command.

5. snowballEdge install-updates 명령을 실행합니다.
6. snowballEdge describe-device-software 명령을 사용하여 설치 상태를 확인할 수 있습니다. 업데이트를 설치하는 동안 이 명령을 사용하면 상태가 표시됩니다.

출력 예시

```
Install State: Installing //Possible values[NA, Installing, Requires Reboot]
```

Snowcone 디바이스에 대한 소프트웨어 업데이트가 설치되었습니다. 업데이트를 설치해도 디바이스에 대한 업데이트는 자동으로 적용되지 않습니다. 업데이트 설치를 완료하려면 디바이스를 재시작해야 합니다.

Warning

디바이스의 모든 활동을 정지하고 않고 Snow Family 디바이스를 재시작하면 데이터 손실이 발생할 수 있습니다.

7. 디바이스의 모든 서비스가 중지되면 디바이스를 재부팅하고 디바이스를 잠금 해제한 다음 다시 부팅합니다. 이렇게 하면 다운로드한 소프트웨어 업데이트 설치가 완료됩니다. 장치 재부팅에 대

한 자세한 내용은 Snow Family 장치 [재부팅](#)을 참조하십시오. 장치 잠금 해제에 대한 자세한 내용은 Snow Family 장치 [잠금 해제](#)를 참조하십시오.

8. 두 번째 재부팅 후 디바이스의 전원이 켜지면 디바이스의 잠금을 해제하십시오.
9. `check-for-updates` 명령을 실행합니다. 이 명령은 Snowcone 소프트웨어의 최신 사용 가능한 버전을 반환하고, 디바이스의 현재 설치된 버전도 반환합니다.

이제 Snow Family 장치 또는 장치 클러스터를 성공적으로 업데이트하고 최신 Snow Family 소프트웨어로의 업데이트를 확인했습니다.

SSL 인증서 업데이트

Snow Family 디바이스를 360일 이상 보관할 계획이라면 디바이스 사용이 중단되지 않도록 디바이스의 SSL (Secure Sockets Layer) 인증서를 업데이트해야 합니다. 인증서가 만료되면 디바이스를 사용할 수 없으므로 디바이스를 AWS에 반환해야 합니다.

AWS 보유한 Snow Family 디바이스의 SSL 인증서가 만료되기 30일 전에 알려드립니다. 알림은 이메일을 통해 AWS Health Dashboard, 그리고 CloudWatch 이벤트로 제공됩니다. 이메일 알림은 Amazon Web Services, Inc.에서 Snow Family 디바이스를 주문할 때 사용한 AWS 계정에 연결된 이메일 주소로 전송됩니다. 알림을 받으면 이 주제의 지침에 따라 최대한 빨리 업데이트를 요청하여 디바이스 사용이 중단되지 않도록 하십시오. 에 대한 AWS Health Dashboard 자세한 내용은 [AWS Health 사용 설명서](#)를 참조하십시오. CloudWatch 이벤트에 대한 자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#)를 참조하십시오.

이 항목에서는 인증서 만료 시기를 결정하는 방법과 디바이스를 업데이트하는 방법에 대해 설명합니다.

1. `snowballEdge describe-device-software` 명령을 사용하여 인증서 만료 시기를 결정하십시오. 명령 출력의 값에는 인증서가 만료되는 날짜 및 시간이 `CertificateExpiry` 값에 포함됩니다.

Example `describe-device-software` 출력

```
Installed version: 101
Installing version: 102
Install State: Downloading
CertificateExpiry : Thur Jan 01 00:00:00 UTC 1970
```

2. AWS Support 연락하여 SSL 인증서 업데이트를 요청하십시오.
3. AWS Support 업데이트 파일을 제공합니다. 업데이트 파일을 [다운로드](#)하고 [설치](#)합니다.
4. [코드와 매니페스트 파일을 사용하십시오.](#)

Snow Family 디바이스의 Amazon Linux 2 AMI 업데이트

보안을 위한 모범 사례는 Amazon Linux 2 AMI를 Snow Family up-to-date 디바이스에 보관하는 것입니다. 에서 [Amazon Linux 2 AMI \(HVM\), SSD 볼륨 유형 \(64비트 x86\)](#) 을 정기적으로 확인하여 업데이트를 확인하십시오. AWS Marketplace AMI 업데이트의 필요성이 확인되면 최신 Amazon Linux 2 이미지를 Snow 디바이스로 가져옵니다. [Amazon EC2 호환 AMI로 디바이스로 이미지 가져오기](#)를 참조하십시오.

AWS CLI의 `ssm get-parameters` 명령을 사용하여 최신 Amazon Linux 2 이미지 ID를 가져올 수도 있습니다.

```
aws ssm get-parameters --names /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2 --query 'Parameters[0].[Value]' --region your-region
```

이 명령은 AMI의 최신 이미지 ID를 반환합니다. 예:

```
ami-0ccb473bada910e74
```

AWS Snowcone 디바이스의 모범 사례

AWS Snowcone 디바이스의 이점과 만족을 최대한으로 얻으려면 다음 모범 사례를 따르는 것이 좋습니다.

주제

- [보안](#)
- [네트워크](#)
- [리소스 관리](#)
- [EC2 호환 인스턴스 관리](#)

보안

- AWS Snowcone 디바이스에 대해 의심이 가는 부분이 있으면 내부 네트워크에 연결하지 마십시오. 그 대신에 [AWS Support](#)에 연락하면 새 AWS Snowcone 디바이스를 받을 수 있습니다.
- 잠금 해제 코드의 복사본을 해당 작업에 대한 매니페스트와 동일한 워크스테이션 내 위치에 저장하지 않는 것이 좋습니다. 이러한 항목을 따로 저장하면 권한 없는 당사자가 AWS Snowcone 디바이스에 대한 액세스 권한을 획득하는 것을 방지할 수 있습니다. 예를 들어 매니페스트 복사본을 로컬 서버에 저장하고 디바이스를 잠금 해제하려는 사용자에게 코드를 이메일로 전송할 수 있습니다. 이 접근 방식은 AWS Snowcone 디바이스에 대한 액세스를 서버에 저장된 파일과 해당 사용자의 이메일 주소에 액세스할 수 있는 개인으로 제한합니다.
- Snowball Edge 클라이언트 명령 `snowballEdge list-access-keys`와 `snowballEdge get-secret-access-key`를 차례로 실행할 때 표시되는 자격 증명은 액세스 키 및 비밀 키의 페어입니다. 이러한 키는 작업과 디바이스의 로컬 리소스에만 연결됩니다. 사용자 AWS 계정 또는 다른 AWS 계정 중 어떤 것에도 매핑되지 않습니다. 이러한 키를 사용하여 AWS 클라우드의 서비스 및 리소스에 액세스하려고 하면 해당 키는 작업과 관련된 로컬 리소스에 대해서만 작동하기 때문에 실패합니다.
- NFS 공유에 대한 액세스를 제한할 수 있습니다. 자세한 내용은 [NFS 실행 시 NFS 공유에 대한 액세스 제한](#) 섹션을 참조하십시오.
- Snowcone 디바이스를 끄거나 전원을 껐다가 다시 켜면 디바이스가 잠긴 상태가 됩니다.

네트워크

- AWS Snowcone 디바이스의 로컬 버킷에서 데이터를 읽고 쓸 때는 한 번에 한 가지 방법만 사용하는 것이 좋습니다. 둘 다 사용

동일한 S3 버킷에서 NFS 탑재와 DataSync를 동시에 사용하면 읽기/쓰기 충돌이 발생할 수 있습니다.

- 데이터 손상을 방지하려면 데이터를 전송하는 동안 AWS Snowcone 디바이스의 연결을 해제하거나 네트워크 설정을 변경하면 안 됩니다.
- 파일은 디바이스에 기록되는 동안 정적 상태에 있어야 합니다. 파일이 쓰여지는 동안 수정된 파일은 읽기/쓰기 충돌을 일으킬 수 있습니다.
- AWS Snowcone 디바이스의 성능 개선에 대한 자세한 내용은 [Snowcone 성능](#) 섹션을 참조하십시오.

리소스 관리

- 온프레미스 데이터 전송을 수행할 수 있는 5일 무료 기간은 AWS Snowcone 디바이스가 데이터 센터에 도착한 다음날부터 시작합니다.

EC2 호환 인스턴스 관리

AWS Snowcone 디바이스에서 작성한 Amazon EC2 호환 인스턴스를 실수로 삭제하지 않도록 운영 체제에서 인스턴스를 종료하면 안 됩니다. 예를 들어, shutdown 또는 reboot 명령을 사용하지 마십시오. 운영 체제에서 인스턴스를 종료하면 [terminate-instances](#) 명령을 호출하는 것과 효과가 동일합니다.

대신 [stop-instances](#) 명령을 사용하여 보존하려는 Amazon EC2 호환 인스턴스를 일시 중단합니다.

Snowcone 성능

다음 표는 네트워크의 전송 속도가 AWS Snowcone를 데이터로 채우는 데 걸리는 시간에 어떤 영향을 주는지를 간략하게 보여줍니다.

속도(MB/s)	8TB 전송 시간	4TB 전송 시간
100	21.17시간	10.59시간
60	36.57시간	18.29시간
30	68.57시간	34.29시간
10	210.29시간	105.15시간

AWS Snowcone 할당량

아래에서 Snowcone 디바이스 사용 할당량 관련 정보를 확인할 수 있습니다.

Snowcone 디스크 스토리지

내부적으로 Snowcone 디바이스에는 8TB 또는 14TB의 디스크 스토리지가 포함되어 있습니다. 이 스토리지는 로컬 Amazon Elastic Block Store(Amazon EBS) 볼륨 프레젠테이션을 통해 내부 NFS(Network File System) 서비스 또는 로컬 Amazon EC2 호환 인스턴스와 함께 사용할 수 있습니다. 이 스토리지는 NFS 또는 Amazon EBS 중 하나에 사용할 수 있지만 둘 다에 사용할 수는 없습니다. 사용 사례에 따라 스토리지를 할당할 수 있습니다. Snowcone 주문 시 스토리지가 사전 할당된다는 점에 유의하세요.

컴퓨팅 작업 스토리지

작업 유형이 로컬 컴퓨팅인 경우 총 8TB 또는 14TB의 로컬 Amazon EBS 볼륨을 생성하여 Amazon EC2 호환 인스턴스에 연결할 수 있습니다. Amazon EBS 볼륨을 사용하면 루트 볼륨만 사용할 때보다 더 많은 로컬 용량에 로컬 Amazon EC2 인스턴스가 액세스할 수 있습니다. 로컬 스토리지만 가능하므로 디바이스를 AWS로 반환하면 Amazon EBS 볼륨에 쓴 데이터가 손실됩니다. Amazon S3로 가져올 수 없기 때문입니다.

주제

- [컴퓨팅 리소스 할당량](#)
- [Snowcone 디바이스 배송에 대한 제한](#)
- [반환된 Snowcone 디바이스를 가져오기 위한 처리 제한](#)
- [사용 가능 AWS 리전](#)

컴퓨팅 리소스 할당량

Snowcone 디바이스에 사용할 수 있는 컴퓨팅 리소스 스토리지 할당량

다음 표에는 Snowcone 디바이스에 사용할 수 있는 컴퓨팅 리소스와 스토리지 할당량이 요약되어 있습니다.

인스턴스 유형	vCPU 코어	메모리(GiB)	할당량
snc1.micro	1	1	2
snc1.small	1	2	2
snc1.medium	2	4	1

Snowcone 디바이스에 대한 Amazon Machine Image(AMI) 및 Amazon EC2 용량 할당량

다음 표에는 Snowcone 디바이스의 AMI 및 Amazon EC2 호환 인스턴스 용량 할당량이 요약되어 있습니다.

Amazon EC2	크기
Amazon EC2 호환 AMI	125GB(고객 작업에 사용되는 모든 AMI를 합친 크기)
Amazon EC2 호환 인스턴스	150GB(인스턴스의 백업 스토리지 볼륨에 사용할 수 있는 스페이스스)

Snowcone 디바이스 배송에 대한 제한

Snowcone 디바이스 배송에 대한 제한 사항은 다음과 같습니다.

- AWS는 Snowcone 디바이스를 사서함으로 배송하지 않습니다.
- 작업을 생성할 때 지정한 해외 주소로 Snowcone 디바이스를 이동하는 것은 AWS 서비스 약관 위반이므로 허용되지 않습니다.

배송에 대한 자세한 내용은 [AWS Snowcone에 대한 배송 고려 사항](#) 섹션을 참조하세요.

반환된 Snowcone 디바이스를 가져오기 위한 처리 제한

AWS 클라우드로 데이터를 가져오려면 Snowcone 디바이스가 다음 요구 사항을 충족해야 합니다.

- Snowcone 디바이스가 손상되어서는 안 됩니다. 전면과 후면의 도어를 여는 것 외에는 어떤 이유로든 Snowcone 디바이스를 열지 마세요.
- 디바이스가 물리적으로 손상되어서는 안 됩니다. 손상을 방지하려면 Snowcone 디바이스의 도어 두개를 닫고 걸쇠에서 딸깍거리는 소리가 날 때까지 누릅니다.
- AWS에서 배송 라벨을 제공하지 않는 한, Snowcone 디바이스의 E Ink 디스플레이가 표시되어야 하며, Snowcone 디바이스로 데이터 전송을 완료했을 때 자동으로 생성된 반환 라벨이 표시되어야 합니다.

Note

이러한 요구사항을 충족하지 않은 채 반송되는 모든 Snowcone 디바이스는 별도의 작업 없이 삭제됩니다.

사용 가능 AWS 리전

지원되는 AWS 리전 및 엔드포인트에 대한 자세한 내용은 AWS 일반 참조의 [AWSSnow Family 엔드포인트 및 할당량](#)을 참조하세요.

Snowcone 문제 해결

이 섹션에서는 발생한 일부 문제를 해결하기 위한 AWS Snowcone 디바이스/서비스 관련 지침과 인사이트를 제공합니다.

주제

- [컴퓨팅 문제 해결](#)
- [네트워크 문제 해결](#)
- [Datasync의 Amazon EC2 호환 인스턴스 문제 해결](#)
- [데이터 전송 문제 해결](#)
- [Snow Family 디바이스 반품 문제 해결](#)

컴퓨팅 문제 해결

컴퓨팅 인스턴스 문제 해결은 [여기](#)에 설명되어 있습니다.

네트워크 문제 해결

IP 주소는 0.0.0.0입니다.

네트워크와 전원 케이블을 연결했지만 디바이스 IP 주소에 0.0.0.0이 표시됩니다.

취할 조치

Snowcone과 네트워크 디바이스 간의 케이블 연결이 제대로 확인되었는지 확인하십시오.

디바이스가 여전히 IP 주소를 표시하지 못하는 경우, 라우터에서 IP 구성 실패로 이어질 수 있는 오작동 또는 구성과 관련된 문제가 있는지 확인하십시오.

네트워크에 DHCP가 활성화되지 않은 경우 Snowcone 디스플레이의 STATIC 옵션을 사용하여 디바이스에 고정 IP를 설정하고 적절한 고정 IP 주소로 디바이스를 프로그래밍해야 합니다.

디바이스를 잠금 해제할 수 없습니다.

OpsHub 또는 Snowball Edge 클라이언트를 사용하여 디바이스를 잠금 해제할 수 없습니다.

취할 조치

클라이언트가 AWS Snowcone 디바이스와 동일한 서브넷 및 네트워크에 있는지 확인하십시오.

Snowcone에 여러 IP 주소 할당(예: WiFi 및 이더넷)이 있는 경우 여러 클라이언트가 Snowcone에 응답하려고 시도하는 경우 동일한 IP 주소를 사용하여 연결해야 합니다. 한 번에 하나의 인터페이스만 사용할 수 있습니다.

위 단계를 수행해도 문제가 해결되지 않으면 AWS Support와 연계하여 디바이스의 로그를 제공하십시오. Snowball Edge 클라이언트와 SnowBalleEdge ge-support-logs 명령을 사용하십시오.

Datasync의 Amazon EC2 호환 인스턴스 문제 해결

오류: 인스턴스 시작 실패

Snowcone은 이 요청에 대한 인스턴스를 시작할 수 있는 용량이 충분하지 않습니다.

취할 조치

에이전트로 AWS Snowcone 디바이스에서 DataSync를 실행하기 위한 요구 사항은 2개의 CPU 코어와 4GiB의 메모리를 제공하는 기본 인스턴스 snc1.medium을 사용합니다. Snowcone 디바이스에 새 Amazon EC2 호환 인스턴스 시작 요청을 프로비저닝할 수 있는 충분한 리소스가 있는지 확인하십시오.

데이터 전송 문제 해결

서버에 의한 액세스 거부

오류: mount.nfs - 192.168.1.214를 탑재하는 동안 서버에서 액세스가 거부되었습니다.

취할 조치

빠른 설치를 사용하여 NFS를 구성한 경우 기본적으로 시스템만 이 디바이스로 파일을 전송할 수 있습니다. 다른 호스트가 Snowcone에 데이터를 업로드하도록 허용하려면 NFS 서비스를 비활성화하고 NFS 구성에서 허용하려는 IP 주소 목록을 사용하여 서비스를 다시 활성화하십시오.

데이터 전송 중 연결 시간 초과

Amazon S3 인터페이스(AWS CLI를 통한)를 사용하여 데이터를 AWS Snowcone로 전송하려고 시도하는 중에 연결 시간이 초과되었습니다.

이러한 동작이 예상되는 이유

AWS Snowcone 디바이스는 오프라인 전송 및 AWS DataSync 온라인 전송의 경우 NFS 탑재를 통한 전송만 지원합니다. Amazon S3 인터페이스는 현재 지원되지 않습니다. 이를 통해 AWS CLI를 사용하여 연결할 때 시간 초과가 발생하는 이유를 알 수 있습니다.

Showmount ENOENT 생성

오류: 확인되지 않은 오류 - Showmount ENOENT를 생성합니다. Error NO ENTry(또는 Error NO ENtity)의 약어로, 실제로는 파일/디렉터리 이외의 용도로도 사용할 수 있습니다.

취할 조치

이 오류는 NFS 클라이언트가 워크스테이션에서 실행되지 않는 경우에 발생할 수 있습니다. AWS OpsHub에서 NFS를 구성하기 전에 NFS 서비스를 시작하지 않은 경우 이 오류가 표시될 수 있습니다.

Windows 클라이언트에서는 다음 단계를 수행하여 NFS 서비스가 실행 중인지 확인할 수 있습니다.

1. 클라이언트 머신에서 시작을 열고 제어판을 선택한 다음 프로그램을 선택합니다.
2. Windows 기능 사용/사용 안 함을 선택합니다.
3. Services for NFS(NFS용 서비스)에서 Client for NFS(NFS용 클라이언트)를 선택하고 확인을 선택합니다. 참조: [Windows 클라이언트에 NFS 탑재](#)

이 오류는 실행 중인 AWS OpsHub 워크스테이션과 연결을 차단할 수 있는 디바이스 사이에 방화벽/바이러스 백신이 있는 경우에도 발생할 수 있습니다.

워크스테이션에서 NFS 포트 번호 2049의 telnet to the AWS Snowcone device IP address 명령을 실행하여 연결이 성공적으로 설정되었는지 확인합니다.

Snow Family 디바이스 반품 문제 해결

반품을 준비하기 위해 Snow Family 디바이스의 전원을 끈 후 반품용 배송 정보가 E Ink 디스플레이에 표시되지 않는 경우가 있습니다.

취할 조치

1. AWS Snow 패밀리 관리 콘솔에 로그인합니다.
2. 디바이스의 작업을 확인하십시오.

3. 세부 정보 섹션의 반품 배송 레이블 섹션에서 반품 레이블을 보고 인쇄하려면 이 링크를 따르십시오.
오를 선택합니다.
4. 레이블을 인쇄하여 디바이스에 부착합니다.
5. [Snowcone 디바이스 반납](#)에 따라 디바이스를 반납합니다.

작업 관리 API 참조

- [작업 관리 API 참조](#)
 - [작업](#)
 - [데이터 형식](#)
 - [공통 파라미터](#)
 - [일반적인 오류](#)

AWS Snowcone 사용 설명서에 대한 문서 기록

다음 표는 이 AWS Snowcone 릴리스 관련 설명서를 소개합니다.

- API 버전: 최신
- 설명서 최종 업데이트: 2023년 8월 25일

변경 사항	설명	날짜
디바이스 주문 시 사용자 지정 AMI를 포함하세요.	이제 AWS Snow Family 작업을 주문하는 동안 사용자 지정 Amazon 머신 이미지를 미리 로드할 수 있습니다. 자세한 내용은 Snowcone에서의 Amazon EC2 사용을 참조하십시오 .	2023년 11월 15일
신규 AWS 리전 지원	이제 이스라엘(텔아비브) AWS 리전에서 AWS Snowcone SSD 디바이스를 사용할 수 있습니다. 이 리전의 픽업 정보가 추가되었습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요. 배송에 대한 자세한 내용은 AWS Snowcone의 배송 고려 사항 을 참조하세요.	2023년 8월 25일
신규 AWS 리전 지원	이제 유럽(파리) AWS 리전에서 Snowcone 디바이스를 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요. 배송에 대한 자세한 내용은 AWS	2022년 6월 29일

[Snowcone의 배송 고려 사항을 참조하세요.](#)

[신규 Snowcone 문제 해결 팁](#)

Snowball Edge와 유사하게 AWS Snowcone 문제 해결 팁 섹션이 추가되었습니다. 추가된 새 섹션은 [Snowcone 문제 해결](#)을 참조하세요.

2022년 4월 15일

[신규 AWS 리전 지원](#)

이제 아시아 태평양(뭄바이) 및 브라질 리전에서 AWS Snowcone을 사용할 수 있습니다. EU, 캐나다, 싱가포르, 브라질의 픽업 정보가 추가되었습니다. 자세한 내용은 AWS 일반 참조의 [AWS Snow Family 엔드포인트 및 할당량](#)을 참조하세요.

2022년 2월 23일

[신규 AWS 리전 지원](#)

이제 유럽(런던) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 [AWS Snow Family 엔드포인트 및 할당량](#)을 참조하세요.

2022년 1월 5일

[NTP\(Network Time Protocol\) 서버 구성 지원](#)

이제 Snowcone 디바이스가 외부 NTP(Network Time Protocol) 서버 구성을 지원합니다.

2021년 11월 16일

신규 AWS 리전 지원	이제 미국 서부(캘리포니아 북부), 미국 동부(오하이오), 아시아 태평양(싱가포르), 아시아 태평양(도쿄), 아시아 태평양(시드니) 리전에서 AWS Snowcone SSD를 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요.	2021년 11월 3일
신규 AWS 리전 지원	이제 미국 서부(캘리포니아 북부), 미국 동부(오하이오), 남아메리카(상파울루) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요.	2021년 9월 29일
신규 AWS 리전 지원	이제 아시아 태평양(싱가포르) 및 아시아 태평양(도쿄) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요.	2021년 8월 26일
Snowcone 디바이스를 통한 Amazon S3에서 오프라인 데이터 내보내기 지원	이제 Amazon S3 데이터를 Snowcone 디바이스로 전송하여 내보내도록 AWS에 요청할 수 있습니다. 아울러 디바이스는 해당 위치로 직접 배송됩니다. 자세한 내용은 가져오기 및 내보내기 작업 작동 방식 을 참조하세요.	2021년 8월 4일

[AWS Snow Device Management 소개](#)

Snow Device Management를 사용하면 AWS Snowcone 디바이스 및 로컬 AWS 서비스를 원격으로 관리할 수 있습니다. 모든 Snowcone 디바이스는 Snow Device Management를 지원하며, Snowcone을 사용할 수 있는 대부분의 AWS 리전에서 새 디바이스에 사전 설치된 후 제공됩니다. 자세한 내용은 [AWS Snow Device Management를 사용하여 디바이스 관리](#)를 참조하세요.

2021년 8월 4일

[신규 AWS 리전 지원](#)

이제 캐나다(중부) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 [AWS Snow Family 엔드포인트 및 할당량](#)을 참조하세요.

2021년 4월 28일

[신규 AWS 리전 지원](#)

이제 아시아 태평양(시드니) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 [AWS Snow Family 엔드포인트 및 할당량](#)을 참조하세요.

2021년 3월 24일

[직접 네트워크 인터페이스 지원](#)

이제 AWS Snowcone에 멀티캐스트 스트림, 라우팅, 로드 밸런싱과 같은 사용 사례를 지원하는 고급 네트워크 기능인 직접 네트워크 인터페이스(DNI) 구성이 추가되었습니다. 자세한 내용은 [컴퓨팅 인스턴스를 위한 네트워크 구성](#)을 참조하세요.

2021년 1월 12일

신규 AWS 리전 지원	이제 유럽(프랑크푸르트) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요.	2020년 11월 18일
신규 AWS 리전 지원	이제 유럽(아일랜드) 리전에서 AWS Snowcone을 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조의 AWS Snow Family 엔드포인트 및 할당량 을 참조하세요.	2020년 9월 16일
AWS Snowcone 소개	AWS Snowcone은 엣지 컴퓨팅 및 데이터 전송을 위한 견고하고 안전한 휴대용 디바이스입니다. 오프라인에서 디바이스를 AWS로 배송하거나 온라인에서 AWS Snowcone를 사용하여 AWS DataSync을 통해 데이터를 수집, 처리하고 AWS로 이동할 수 있습니다. 자세한 내용은 AWS Snowcone이란 무엇인가요? 를 참조하세요.	2020년 6월 17일

AWS 용어집

최신 AWS 용어는 AWS 용어집 참조서의 [AWS 용어집](#)을 참조하세요.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.