



ONTAP 명령줄 인터페이스를 사용합니다

ONTAP 9

NetApp
February 12, 2026

목차

ONTAP 명령줄 인터페이스를 사용합니다	1
ONTAP 명령줄 인터페이스에 대해 알아봅니다	1
CLI 명령의 다른 ONTAP 셸에 대해 알아봅니다	1
클러스터 셸에서 노드 명령 및 옵션 액세스	2
사용 가능한 노드 대체 명령을 표시합니다	2
ONTAP CLI 명령 디렉토리를 탐색하는 방법	3
ONTAP CLI에서 값을 지정하는 규칙을 이해합니다	4
ONTAP 명령 기록을 보고 기록에서 명령을 다시 실행합니다	4
CLI 명령 편집을 위한 ONTAP 바로 가기 키	5
ONTAP CLI 명령에 대한 권한 수준을 이해합니다	7
ONTAP CLI에서 권한 수준을 설정한다	7
ONTAP CLI에 대한 디스플레이 기본 설정을 지정합니다	8
ONTAP CLI에서 쿼리 연산자를 사용합니다	9
ONTAP CLI에서 modify 및 delete 명령에 확장 쿼리를 사용합니다	10
fields 매개 변수를 사용하여 ONTAP show 명령의 출력을 제한합니다	11
명령 입력에서 ONTAP CLI 위치 매개 변수를 사용합니다	12
위치 매개 변수란 무엇입니까	12
위치 매개 변수를 식별합니다	13
위치 매개 변수 사용 예	13
ONTAP CLI man 페이지에 액세스하는 방법	14

ONTAP 명령줄 인터페이스를 사용합니다

ONTAP 명령줄 인터페이스에 대해 알아봅니다

ONTAP CLI(Command-Line Interface)는 관리 인터페이스의 명령 기반 뷰를 제공합니다. 스토리지 시스템 프롬프트에서 명령을 입력하면 명령 결과가 텍스트로 표시됩니다.

CLI 명령 프롬프트는 `cluster_name::>`로 표시됩니다.

권한 수준(즉, 'set' 명령의 '-Privilege' 매개 변수)을 'advanced'로 설정하면 다음과 같은 별표(*)가 표시됩니다.

'cluster_name

*>

에 대한 자세한 내용은 `set "ONTAP 명령 참조입니다"`을 참조하십시오.

CLI 명령의 다른 ONTAP 셸에 대해 알아봅니다

클러스터에는 CLI 명령을 위한 `clustershell`, `nodeshell` 및 `_systemshell`의 세 가지 셸이 있습니다. 셸은 서로 다른 용도로 사용되며 각각 다른 명령 집합을 가집니다.

- 클러스터 셸은 클러스터에 로그인할 때 자동으로 시작되는 네이티브 셸입니다.

클러스터 구성 및 관리에 필요한 모든 명령을 제공합니다. `clustershell` CLI 도움말(`clustershell` 프롬프트에서 트리거됨)에는 ? 사용 가능한 `clustershell` 명령이 표시됩니다. `man`clustershell`의 명령은 지정된 `clustershell` 명령에 대한 `man` 페이지를 (``man <command_name>` 표시합니다.)에 대한 자세한 내용은 ``man "ONTAP 명령 참조입니다"`을 참조하십시오.

- 노드 수준에서만 적용되는 명령을 위한 특수 셸입니다.

노드 셸은 명령을 통해 액세스할 수 `system node run` 있습니다.에 대한 자세한 내용은 `system node run "ONTAP 명령 참조입니다"`을 참조하십시오.

노드 셸 CLI 도움말(또는 `help` 노드 셸 프롬프트에서 트리거됨 ?)에는 사용 가능한 노드 셸 명령이 표시됩니다. ``man`노드 셸의 명령은 지정된 노드 셸 명령에 대한 man 페이지를 표시합니다.`

일반적으로 사용되는 노드 셸의 명령 및 옵션은 클러스터 셸에 터널링되거나 별칭되어 클러스터 셸에서도 실행될 수 있습니다.

- 시스템 셸은 진단 및 문제 해결용으로만 사용되는 하위 셸입니다.

시스템 셸 및 관련 `diag` 계정은 저수준 진단 목적으로 사용됩니다. 이러한 액세스를 위해서는 진단 권한 수준이 필요하며, 문제 해결 작업을 수행하기 위한 기술 지원용으로만 예약되어 있습니다.

이 절차에서 설명하는 명령에 대한 자세한 내용은 `를 "ONTAP 명령 참조입니다"`참조하십시오.

클러스터 셸에서 노드 명령 및 옵션 액세스

노드 셸을 통해 명령 및 옵션을 액세스할 수 있습니다.

```
'system node run - node_nodename_'
```

일반적으로 사용되는 노드 셸의 명령 및 옵션은 클러스터 셸에 터널링되거나 별칭되어 클러스터 셸에서도 실행될 수 있습니다.

클러스터 셸에서 지원되는 Nodeshell 옵션은 다음을 사용하여 액세스할 수 있습니다. `vserver options clustershell` 명령. 이러한 옵션을 보려면 다음을 사용하여 클러스터 셸 CLI를 쿼리할 수 있습니다. `vserver options -vserver nodename_or_clustername -option-name ?`

클러스터 셸에 노드 셸이나 레거시 명령 또는 옵션을 입력하고 명령 또는 옵션에 동일한 클러스터 셸 명령이 있으면 ONTAP에서 사용할 클러스터 셸 명령을 알려줍니다.

클러스터 셸에서 지원되지 않는 노드 셸이나 레거시 명령 또는 옵션을 입력하면 ONTAP에서 명령이나 옵션에 대한 ""지원되지 않음" 상태를 알려줍니다.

사용 가능한 노드 대체 명령을 표시합니다

CLI 도움말을 사용하여 노드 셸에서 사용 가능한 노드 셸의 명령 목록을 얻을 수 있습니다.

단계

1. 노드 셸에 액세스하려면 클러스터 셸의 시스템 프롬프트에 다음 명령을 입력합니다.

```
* 시스템 노드 run-node {nodename|local} *
```

로컬 은 클러스터에 액세스하는 데 사용한 노드입니다.



'system node run' 명령에는 'run'이라는 별칭 명령이 있습니다.

2. 노드 셸에 다음 명령을 입력하여 사용 가능한 노드 셸 명령 목록을 표시합니다.

```
**[CommandName] 도움말 * "
```

'CommandName'은 표시할 수 있는 명령의 이름입니다. 'CommandName'을 포함하지 않으면 CLI는 사용 가능한 모든 notes지옥명령을 표시합니다.

'exit'를 입력하거나 Ctrl-d를 입력하여 클러스터 셸 CLI로 돌아갑니다.

에 대한 자세한 내용은 `exit` ["ONTAP 명령 참조입니다"](#)을 참조하십시오.

사용 가능한 노드 셸의 명령 표시 예

다음 예에서는 node2라는 노드의 노드 중 노드 셸에 액세스하여 노드 셸의 명령 "환경"에 대한 정보를 표시합니다.

```

cluster1::> system node run -node node2
Type 'exit' or 'Ctrl-D' to return to the CLI

node2> environment help
Usage: environment status |
      [status] [shelf [<adapter>[.<shelf-number>]]] |
      [status] [shelf_log] |
      [status] [shelf_stats] |
      [status] [shelf_power_status] |
      [status] [chassis [all | list-sensors | Temperature | PSU 1 |
PSU 2 | Voltage | SYS FAN | NVRAM6-temperature-3 | NVRAM6-battery-3]]

```

ONTAP CLI 명령 디렉토리를 탐색하는 방법

CLI의 명령은 명령 디렉토리별로 계층 구조로 구성됩니다. 전체 명령 경로를 입력하거나 디렉토리 구조를 탐색하여 계층에서 명령을 실행할 수 있습니다.

CLI를 사용하는 경우 프롬프트에 디렉토리 이름을 입력하고 Enter 키를 눌러 명령 디렉토리에 액세스할 수 있습니다. 그러면 디렉토리 이름이 프롬프트 텍스트에 포함되어 해당 명령 디렉터리와 상호 작용하고 있음을 나타냅니다. 명령 계층 구조로 더 자세히 이동하려면 명령 하위 디렉터리의 이름을 입력한 다음 Enter 키를 누릅니다. 그러면 하위 디렉터리 이름이 프롬프트 텍스트에 포함되고 컨텍스트는 해당 하위 디렉터리로 이동합니다.

전체 명령을 입력하여 여러 명령 디렉토리를 탐색할 수 있습니다. 예를 들어, 프롬프트에 'storage disk show' 명령을 입력하여 디스크 드라이브에 대한 정보를 표시할 수 있습니다. 다음 예제와 같이 한 번에 하나의 명령 디렉토리를 탐색하여 명령을 실행할 수도 있습니다.

```

cluster1::> storage
cluster1::storage> disk
cluster1::storage disk> show

```

에 대한 자세한 내용은 `storage disk show` "[ONTAP 명령 참조입니다](#)"을 참조하십시오.

현재 디렉터리에 고유한 명령을 만드는 명령의 최소 문자 수만 입력하여 명령을 축약할 수 있습니다. 예를 들어, 이전 예제의 명령을 축약하려면 'st d sh'를 입력할 수 있습니다. Tab 키를 사용하여 약식 명령을 확장하고 기본 매개 변수 값을 비롯한 명령의 매개 변수를 표시할 수도 있습니다.

'top' 명령을 사용하여 명령어 계층구조의 최상위 수준으로 이동하고 'up' 명령이나...' 명령을 사용하여 명령어 계층구조에서 한 단계 위로 이동할 수 있습니다.



CLI에서 별표(*)가 앞에 오는 명령 및 명령 옵션은 고급 권한 수준 이상에서만 실행할 수 있습니다.

관련 정보

- "[위쪽](#)"
- "[위로](#)"

ONTAP CLI에서 값을 지정하는 규칙을 이해합니다

대부분의 명령에는 하나 이상의 필수 매개 변수 또는 선택적 매개 변수가 포함됩니다. 많은 매개 변수는 값을 지정해야 합니다. CLI에서 값을 지정할 때 적용되는 몇 가지 규칙이 있습니다.

- 값은 숫자, 부울 지정자 또는 텍스트 문자열이거나 사전 정의된 값의 열거 목록 중 한 값을 선택하여 사용할 수 있습니다.

일부 매개 변수에는 둘 이상의 값을 심표로 구분하여 지정할 수 있습니다. 심표로 구분된 값 목록은 따옴표(" ")에 있을 필요가 없습니다. 텍스트, 공백 또는 쿼리 문자(보다 작음 또는 보다 큰 기호로 시작하는 텍스트나 쿼리가 아닌 경우)를 지정할 때는 항상 엔티티를 따옴표로 묶어야 합니다.

- CLI에서 물음표(?)는 특정 명령의 도움말 정보를 표시하는 명령으로 해석됩니다.
- 명령 이름, 매개 변수 및 특정 값과 같이 CLI에서 입력하는 일부 텍스트는 대/소문자를 구분하지 않습니다.

예를 들어, 'vserver cifs' 명령에 대한 매개 변수 값을 입력하면 대문자 표시가 무시됩니다. 하지만 노드 이름, SVM(스토리지 가상 머신), 애그리게이트, 볼륨, 논리 인터페이스 등과 같은 대부분의 매개 변수 값은 대/소문자를 구분합니다.

- 문자열이나 목록을 사용하는 매개 변수 값을 지우려면 빈 따옴표("") 또는 대시("-")를 지정합니다.
- (# 파운드 기호라고도 하는 해시 기호는 명령줄 입력에 대한 주석을 나타냅니다. 사용할 경우 명령줄에서 마지막 매개 변수 뒤에 표시되어야 합니다.

CLI에서는 과 줄 사이의 텍스트를 # 무시합니다.

다음 예에서는 텍스트 코멘트로 SVM을 작성합니다. 그런 다음 SVM이 수정되어 설명을 삭제합니다.

```
cluster1::> vserver create -vserver vs0 -subtype default -rootvolume
root_vs0
-aggregate aggr1 -rootvolume-security-style unix -language C.UTF-8 -is
-repository false -ipspace ipspaceA -comment "My SVM"
cluster1::> vserver modify -vserver vs0 -comment ""
```

다음 예제에서 부호를 사용하는 명령줄 주석은 # 명령이 수행하는 작업을 나타냅니다.

```
cluster1::> security login create -vserver vs0 -user-or-group-name new-
admin
-application ssh -authmethod password #This command creates a new user
account
```

에 대한 자세한 내용은 security login create ["ONTAP 명령 참조입니다"](#)을 참조하십시오.

ONTAP 명령 기록을 보고 기록에서 명령을 다시 실행합니다

각 CLI 세션은 해당 세션에서 실행된 모든 명령의 기록을 유지합니다. 현재 사용 중인 세션의

명령 기록을 볼 수 있습니다. 명령을 다시 실행할 수도 있습니다.

명령어 이력을 보기 위해 'history' 명령어를 사용할 수 있다.

명령을 재발급하려면 다음 인수 중 하나와 함께 "redo" 명령을 사용할 수 있습니다.

- 이전 명령의 일부와 일치하는 문자열입니다

예를 들어, 실행한 유일한 'volume' 명령이 'volume show'인 경우 'redo volume' 명령을 사용하여 명령을 다시 실행할 수 있습니다.

- history 명령으로 나열한 이전 명령의 숫자 ID

예를 들어, 'redo 4' 명령을 사용하여 기록 목록에서 네 번째 명령을 다시 실행할 수 있습니다.

- 기록 목록의 끝에서 음수 오프셋

예를 들어, redo-2 명령을 사용하여 이전에 두 명령을 실행한 명령을 다시 실행할 수 있습니다.

예를 들어, 명령 기록 끝에서 세 번째 명령을 다시 실행하려면 다음 명령을 입력합니다.

```
cluster1::> redo -3
```

관련 정보

- ["기록"](#)
- ["다시 실행"](#)
- ["볼륨"](#)

CLI 명령 편집을 위한 ONTAP 바로 가기 키

현재 명령 프롬프트에 있는 명령은 활성 명령입니다. 바로 가기 키를 사용하면 활성 명령을 빠르게 편집할 수 있습니다. 이러한 바로 가기 키는 UNIX tcsh 셸 및 Emacs 편집기의 바로 가기와 유사합니다.

다음 표에는 CLI 명령을 편집하기 위한 바로 가기 키가 나와 있습니다. `Ctrl`-`Ctrl` 키를 누른 상태에서 뒤에 지정된 문자를 입력함을 나타냅니다. `Esc`-`Esc` 키를 눌렀다가 놓은 다음 그 뒤에 지정된 문자를 입력함을 나타냅니다.

원하는 작업	다음 바로 가기 키 중 하나를 사용합니다.
커서를 한 문자씩 뒤로 이동합니다	<ul style="list-style-type: none">• <code>Ctrl+B</code>를 누릅니다• 뒤로 화살표
커서를 한 문자씩 앞으로 이동합니다	<ul style="list-style-type: none">• <code>Ctrl + F</code>• 앞으로 화살표

원하는 작업	다음 바로 가기 키 중 하나를 사용합니다.
커서를 한 단어씩 뒤로 이동합니다	Esc-B를 누릅니다
커서를 한 단어씩 앞으로 이동합니다	Esc - F
커서를 줄의 시작 부분으로 이동합니다	Ctrl-A를 누릅니다
커서를 줄의 끝으로 이동합니다	Ctrl-E
줄의 시작 부분에서 커서까지의 명령줄 내용을 제거하고 잘라낸 버퍼에 저장합니다. 잘라내기 버퍼는 일부 프로그램에서 <code>_clipboard_</code> 라고 하는 것과 유사한 임시 메모리처럼 작동합니다.	Ctrl+U
커서에서 줄의 끝까지 명령줄의 내용을 제거하고 절단 버퍼에 저장합니다	Ctrl+K를 누릅니다
커서에서 다음 단어의 끝까지 명령줄 내용을 제거하고 절단 버퍼에 저장합니다	Esc-D 를 누릅니다
커서 앞에 있는 단어를 제거하고 잘라낸 버퍼에 저장합니다	Ctrl+W
절단 버퍼의 내용을 बैं킹하고 커서의 명령줄에 밀어 넣습니다	Ctrl+Y
커서 앞의 문자를 삭제합니다	<ul style="list-style-type: none"> • Ctrl + H • 백스페이스
커서가 있는 문자를 삭제합니다	Ctrl+D를 누릅니다
선을 지웁니다	Ctrl+C
화면을 지웁니다	Ctrl+L
명령줄의 현재 내용을 기록 목록의 이전 항목으로 바꿉니다. 키보드 단축키를 반복할 때마다 기록 커서가 이전 항목으로 이동합니다.	<ul style="list-style-type: none"> • Ctrl+P • Esc-P • 위쪽 화살표
명령줄의 현재 내용을 기록 목록의 다음 항목으로 바꿉니다. 키보드 단축키를 반복할 때마다 기록 커서가 다음 항목으로 이동합니다.	<ul style="list-style-type: none"> • Ctrl+N • Esc-N • 아래쪽 화살표

원하는 작업	다음 바로 가기 키 중 하나를 사용합니다.
부분적으로 입력한 명령을 확장하거나 현재 편집 위치에서 유효한 입력을 나열합니다	<ul style="list-style-type: none"> • 탭을 클릭합니다 • Ctrl+I
상황에 맞는 도움말을 표시합니다	?
물음표 (?) 문자에 대한 특수 매핑을 해제합니다. 예를 들어 명령 인수에 물음표를 입력하려면 Esc 키를 누른 다음 ? 문자를 누릅니다.	Esc-?
TTY 출력을 시작합니다	Ctrl-Q를 누릅니다
TTY 출력을 중지합니다	Ctrl+S

ONTAP CLI 명령에 대한 권한 수준을 이해합니다

ONTAP 명령 및 매개 변수는 *admin*, *advanced* 및 *_diagnostic_* 의 세 가지 권한 레벨로 정의됩니다. 권한 수준은 작업을 수행하는 데 필요한 기술 수준을 반영합니다.

- * 관리자 *

대부분의 명령과 매개 변수는 이 레벨에서 사용할 수 있습니다. 일반 또는 일상적인 작업에 사용됩니다.

- 고급 *

이 레벨의 명령 및 매개 변수는 자주 사용되지 않으며, 고급 지식이 필요하며 부적절하게 사용될 경우 문제를 일으킬 수 있습니다.

고급 명령 또는 매개 변수는 지원 담당자의 조언과 함께 사용해야 합니다.

- * 진단 *

진단 명령 및 매개 변수가 잠재적으로 중단될 수 있습니다. 문제 진단 및 해결을 위해 지원 담당자만이 사용합니다.

ONTAP CLI에서 권한 수준을 설정한다

CLI에서 '설정' 명령어를 이용하여 권한 수준을 설정할 수 있다. 권한 수준 설정 변경 사항은 현재 진행 중인 세션에만 적용됩니다. 세션 간에 지속적이지 않습니다.

단계

1. CLI에서 권한 수준을 설정하려면 '-Privilege' 파라미터와 함께 '설정' 명령어를 사용한다.

권한 수준 설정의 예

다음 예제에서는 권한 수준을 Advanced 로 설정한 다음 admin 으로 설정합니다.

```
cluster1::> set -privilege advanced
Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by NetApp personnel.
Do you wish to continue? (y or n): y
cluster1::*> set -privilege admin
```

에 대한 자세한 내용은 [set "ONTAP 명령 참조입니다"](#)을 참조하십시오.

ONTAP CLI에 대한 디스플레이 기본 설정을 지정합니다

'설정' 명령과 '행' 명령을 사용하여 CLI 세션에 대한 디스플레이 기본 설정을 설정할 수 있습니다. 설정한 기본 설정은 현재 진행 중인 세션에만 적용됩니다. 세션 간에 지속적이지 않습니다.

이 작업에 대해

다음과 같은 CLI 디스플레이 기본 설정을 설정할 수 있습니다.

- 명령 세션의 권한 수준입니다
- 잠재적으로 운영 중단 명령에 대해 확인을 수행할지 여부를 나타냅니다
- '표시' 명령이 모든 필드를 표시하는지 여부
- 필드 구분 기호로 사용할 문자 또는 문자입니다
- 데이터 크기를 보고할 때의 기본 단위입니다
- 인터페이스가 출력을 일시 중지하기 전에 화면이 현재 CLI 세션에 표시되는 행의 수입니다

원하는 행 수를 지정하지 않으면 터미널의 실제 높이에 따라 자동으로 조정됩니다. 실제 높이가 정의되지 않은 경우 기본 행 수는 24입니다.

- 기본 스토리지 가상 시스템(SVM) 또는 노드
- 오류가 발생할 경우 계속 명령을 중지할지 여부를 나타냅니다

단계

1. CLI display preference를 설정하려면 '설정' 명령어를 사용한다.

현재 CLI 세션에서 화면에 표시되는 행의 수를 설정하려면 "rows" 명령을 사용할 수도 있습니다.

및 rows 에 대한 자세한 set 내용은 ["ONTAP 명령 참조입니다"](#)참조하십시오.

CLI에서 디스플레이 기본 설정을 설정하는 예

다음 예제에서는 쉼표를 필드 구분 기호로 설정하고 기본 데이터 크기 단위로 GB를 설정하고 행 수를 50으로 설정합니다.

```
cluster1::> set -showseparator "," -units GB
cluster1::> rows 50
```

관련 정보

- "표시"
- "설정"
- "행"

ONTAP CLI에서 쿼리 연산자를 사용합니다

관리 인터페이스는 쿼리 및 UNIX 스타일 패턴과 와일드카드를 지원하여 명령 매개 변수 인수의 여러 값을 일치시킬 수 있습니다.

다음 표에서는 지원되는 쿼리 연산자를 설명합니다.

운영자	설명
*	모든 항목과 일치하는 와일드카드입니다. 예를 들어, 'volume show-volume\ * tmp *' 명령은 이름에 'tmp' 문자열이 포함된 모든 볼륨의 목록을 표시합니다.
!	연산자가 아닙니다. 일치하지 않는 값을 나타냅니다. 예를 들어, '!vs0'은 값 "vs0"과 일치하지 않음을 나타냅니다.
있습니다. 비교할 두 개의 값을 구분합니다. 예를 들어, '* vs0	vs2*'는 vs0 또는 vs2와 일치합니다.
b *	\ * c * 와 같이 여러 개 또는 구문을 지정할 수 있습니다. 이 항목은 a 항목, b로 시작하는 모든 항목, c를 포함하는 모든 항목과 일치합니다.
...	범위 연산자. 예를 들어, ' * 5..10 *'은 '5'에서 '10'까지의 모든 값을 포함합니다.
를 누릅니다	보다 작음 연산자입니다. 예를 들어, '*<20 *'은 "20"보다 작은 모든 값과 일치합니다.
를 누릅니다	보다 큰 연산자. 예를 들어, ' * > 5 * '는 '5'보다 큰 값을 찾습니다.

운영자	설명
=	보다 작거나 같은 연산자. 예를 들어, " $* \leq 5*$ "는 "5"보다 작거나 같은 값을 찾습니다.
>=	보다 크거나 같은 연산자입니다. 예를 들어, " $>= 5$ "는 '5'보다 크거나 같은 값을 찾습니다.
{query}	확장된 쿼리. 확장 쿼리는 다른 매개 변수 앞에 명령 이름 뒤의 첫 번째 인수로 지정해야 합니다. 예를 들어, 'volume modify{-volume\ * tmp *} -state offline' 명령은 이름에 'tmp' 문자열이 포함된 모든 볼륨을 오프라인으로 설정합니다.

쿼리 문자를 리터럴로 구문 분석하려면 문자를 큰따옴표로 묶어야 합니다(예: "<10", "0..100", "*abc*", 또는 "a|b")를 참조하십시오.

특수 문자가 해석되지 않도록 원시 파일 이름을 큰따옴표로 묶어야 합니다. 이는 클러스터 셸에서 사용하는 특수 문자에도 적용됩니다.

한 명령줄에 여러 쿼리 연산자를 사용할 수 있습니다. 예를 들어 명령은 `volume show -size >1GB -percent-used <50 -vserver !vs1` 크기가 1GB보다 크고, 50% 미만이고, 이름이 ""VS1""인 SVM(스토리지 가상 머신)에 없는 모든 볼륨을 표시합니다.

관련 정보

["CLI 명령 편집을 위한 바로 가기 키"](#)

ONTAP CLI에서 **modify** 및 **delete** 명령에 확장 쿼리를 사용합니다

확장 쿼리를 사용하여 지정된 값이 있는 개체를 일치시키고 작업을 수행할 수 있습니다.

확장 쿼리는 중괄호({})로 둘러싸서 지정합니다. 확장 쿼리는 다른 매개 변수 앞에 명령 이름 뒤의 첫 번째 인수로 지정해야 합니다. 예를 들어 이름이 "tmp"인 모든 볼륨을 오프라인으로 설정하려면 다음 예제에서 명령을 실행합니다.

```
cluster1::> volume modify {-volume *tmp*} -state offline
```

확장 쿼리는 일반적으로 수정 및 삭제 명령에만 유용합니다. 이들은 참조나 표현에 아무런 의미가 없습니다.

쿼리와 수정 작업의 조합은 유용한 도구입니다. 그러나 잘못 구현하면 혼동과 오류가 발생할 수 있습니다. 예를 들어, (advanced 권한) 'system node image modify' 명령을 사용하여 노드의 기본 소프트웨어 이미지를 설정하면 다른 소프트웨어 이미지가 자동으로 기본값이 아닌 기본값으로 설정됩니다. 다음 예제의 명령은 null 작업입니다.

```
cluster1::*> system node image modify {-isdefault true} -isdefault false
```

이 명령은 현재 기본 이미지를 기본 이미지가 아닌 이미지로 설정한 다음 새 기본 이미지(이전의 기본 이미지가 아닌 이미지)를 기본 이미지가 아닌 이미지로 설정하여 원래 기본 설정을 유지합니다. 다음 예제와 같이 명령을 사용하여 작업을 올바르게 수행할 수 있습니다.

```
cluster1::*> system node image modify {-iscurrent false} -isdefault true
```

fields 매개 변수를 사용하여 ONTAP show 명령의 출력을 제한합니다

매개 변수를 명령과 함께 show 사용하여 세부 정보를 표시하면 -instance 출력이 오래 걸리고 필요한 것보다 더 많은 정보가 포함될 수 있습니다. -fields`명령의 매개 변수를 `show 사용하면 지정한 정보만 표시할 수 있습니다.

예를 들어 를 실행하면 volume show -instance 여러 개의 정보 화면이 표시될 수 있습니다. 을 사용하여 항상 표시되는 기본 필드 외에 지정된 필드만 포함하도록 출력을 사용자 지정할 수 있습니다 volume show -fields *fieldname[,fieldname...]*. 를 사용하면 명령에 대한 유효한 필드를 표시할 show 수 -fields ? 있습니다.

다음 예제는 매개 변수와 -fields 매개 변수 간의 출력 차이를 보여줍니다 -instance.

```

cluster1::> volume show -instance

                                Vserver Name: cluster1-1
                                Volume Name: vol0
                                Aggregate Name: aggr0
                                Volume Size: 348.3GB
                                Volume Data Set ID: -
                                Volume Master Data Set ID: -
                                Volume State: online
                                Volume Type: RW
                                Volume Style: flex
                                ...
                                Space Guarantee Style: volume
                                Space Guarantee in Effect: true
                                ...
Press <space> to page down, <return> for next line, or 'q' to quit...
...
cluster1::>

cluster1::> volume show -fields space-guarantee,space-guarantee-enabled

vserver  volume  space-guarantee  space-guarantee-enabled
-----  -
cluster1-1  vol0    volume          true
cluster1-2  vol0    volume          true
vs1        root_vol
          volume          true
vs2        new_vol
          volume          true
vs2        root_vol
          volume          true
...
cluster1::>

```

명령 입력에서 ONTAP CLI 위치 매개 변수를 사용합니다

ONTAP CLI의 위치 매개 변수 기능을 활용하여 명령 입력의 효율성을 높일 수 있습니다. 명령을 쿼리하여 명령의 위치 매개 변수를 식별할 수 있습니다.

위치 매개 변수란 무엇입니까

- 위치 매개 변수는 매개 변수 값을 지정하기 전에 매개 변수 이름을 지정할 필요가 없는 매개 변수입니다.
- "**command_name?**" 출력에 표시된 것과 같이 명령 입력에서 상대 시퀀스를 동일한 명령에서 다른 위치 매개 변수와 관찰하는 한 위치 매개 변수를 위치 매개 변수와 함께 사용할 수 있습니다.

- 위치 매개 변수는 명령에 대한 필수 매개 변수이거나 선택적 매개 변수일 수 있습니다.
- 매개 변수는 한 명령에 대한 위치일 수 있지만 다른 명령에는 위치 지정되지 않을 수 있습니다.



특히 위치 매개 변수가 명령에서 선택 항목이거나 선택 매개 변수가 앞에 나열되는 경우에는 스크립트의 위치 매개 변수 기능을 사용하지 않는 것이 좋습니다.

위치 매개 변수를 식별합니다

'**command_name?**' 명령 출력에서 위치 매개 변수를 식별할 수 있습니다. 위치 매개 변수에는 다음 형식 중 하나로 매개 변수 이름을 둘러싸는 대괄호가 있습니다.

- `[-parameter_name]parameter_value`는 위치 매개 변수를 나타냅니다.
- `[[parameter_name]parameter_value]`는 위치 매개 변수를 선택적으로 표시합니다.

예를 들어, '**command_name?**' 출력에 다음과 같이 표시될 경우 매개 변수는 표시되는 명령의 위치입니다.

- `[-lif]<lif-name>`
- `[[lif]<lif-name>]`

그러나 다음과 같이 표시될 경우 매개 변수는 표시되는 명령의 위치 이외의 매개 변수입니다.

- `-lif<lif-name>`
- `[lif<lif-name>]`

위치 매개 변수 사용 예

다음 예제에서 '***volume create? ***' 출력에서는 '-volume', '-aggregate', '-size' 명령의 위치를 나타내는 세 개의 매개 변수가 표시됩니다.

```
cluster1::> volume create ?
  -vserver <vserver name>           Vserver Name
  [-volume] <volume name>          Volume Name
  [-aggregate] <aggregate name>     Aggregate Name
  [[-size] {<integer>[KB|MB|GB|TB|PB]}] Volume Size
  [ -state {online|restricted|offline|force-online|force-offline|mixed} ]
                                         Volume State (default: online)
  [ -type {RW|DP|DC} ]                Volume Type (default: RW)
  [ -policy <text> ]                  Export Policy
  [ -user <user name> ]              User ID
  ...
  [ -space-guarantee|-s {none|volume} ] Space Guarantee Style (default:
volume)
  [ -percent-snapshot-space <percent> ] Space Reserved for Snapshot
Copies
  ...
```

다음 예에서는 positional parameter 기능을 이용하지 않고 'volume create' 명령을 지정합니다.

```
'cluster1::> volume create-vserver svm1-volume vol1-aggregate * aggr1-size 1g * - percent-snapshot-space 0'
```

다음 예제에서는 위치 매개 변수 기능을 사용하여 명령 입력의 효율성을 높입니다. 'volume create' 명령에서 위치 매개 변수가 위치 매개 변수와 비위치 매개 변수가 서로 연결되어 있으며 위치 매개 변수 값은 매개 변수 이름 없이 지정됩니다. 위치 매개 변수는 '* volume create?*' 출력에 표시된 것과 동일한 순서로 지정됩니다. 즉, -volume의 값은 -size의 앞에 차례로 지정된 -aggregate의 값 앞에 지정됩니다.

```
'cluster1::> 볼륨 create * vol2 * aggr1 * * 1g * -vserver svm1-percent-snapshot-space 0'
```

```
'cluster1::> volume create -vserver svm1 * vol3 * -snapshot-policy default * aggr1 * -nvfail off * 1g * -space-보증 없음'
```

ONTAP CLI man 페이지에 액세스하는 방법

ONTAP 수동(man) 페이지에서는 ONTAP CLI 명령을 사용하는 방법을 설명합니다. 이러한 페이지는 명령줄에서 사용할 수 있으며 release-specific_command_references_에도 게시됩니다.

ONTAP 명령줄에서 명령을 사용하여 man <command_name> 지정된 명령의 수동 페이지를 표시합니다. 명령 이름을 지정하지 않으면 수동 페이지 인덱스가 표시됩니다. 명령을 사용하여 명령 자체에 대한 정보를 볼 man 수 man man 있습니다. 를 입력하여 man 페이지를 종료할 수 q 있습니다.

해당 릴리즈에서 사용할 수 있는 관리자 레벨 및 고급 레벨 ONTAP 명령에 대해 자세히 ["ONTAP 명령 참조입니다"](#) 알아보십시오.

저작권 정보

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

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이선스와 고지사항이 적용됩니다.

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

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

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

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

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

상표 정보

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