hdiutil

Manipulate disk images (attach, verify, burn, etc).

```
Syntax
      hdiutil verb [options]

DESCRIPTION

     hdiutil uses the DiskImages framework to manipulate disk images.  Common
     verbs include attach, detach, verify, create, convert, compact, and burn.

     The rest of the verbs are currently: help, info, checksum, chpass,
     erasekeys, unflatten, flatten, imageinfo, isencrypted, mountvol, unmount,
     plugins, udifrez, udifderez, internet-enable, resize, segment,
     makehybrid, and pmap.

BACKGROUND
```

Disk images are containers that emulate disks. Like disks, they can be partitioned and formatted. Many uses of disk images blur the distinction between the disk image container and its content, but this distinction is critical to understanding disk images. The terms "attach" and "detach" are used to distinguish the way disk images are connected to and disconnected from the system.

For example, when you double-click a d ge is "attached" to the system just like an external drive. Then, the kern lf any file structures are understood, the associated volumes wil Finder.

Always consider whether a "disk image e-oriented) content of the image. For example, verify verifies that t created. On the other hand, create -srcfolder creates a disk image c in it, and then copies the specified files

```
COMMON OPTIONS

     The following option desc:

     -verbose be verbose: prod
              This option can
              tion failed.  At
              will be detailed

     -quiet   close stdout and
              indicate success
              -quiet.

     -debug   be very verbose.
              progress informa
              enables -verbose

     Many hdiutil verbs unders

     -plist          provide result output in plist format.  Other programs
                     invoking hdiutil are expected to use -plist rather than
                     try to parse the human-readable output.  The usual output
                     is consistent but generally unstructured.

     -puppetstrings  provide progress output that is easy for another program
                     to parse.  PERCENTAGE outputs can include the value -1
                     which means hdiutil is performing an operation that will
                     take an indeterminate amount of time to complete.  Any
                     program trying to interpret hdiutil's progress should use
                     -puppetstrings.

     -srcimagekey key=value
                     specify a key/value pair for the disk image recognition
                     system.  (-imagekey is normally a synonym)

     -tgtimagekey key=value
                     specify a key/value pair for any image created.
                     (-imagekey is only a synonym if there is no input image).

     -encryption [AES-128|AES-256]
                     specify a particular type of encryption or, if not speci-
                     fied, the default encryption algorithm.  The default
                     algorithm is the AES cipher with a 128-bit key.

     -stdinpass      read a null-terminated passphrase from standard input.
```

If the standard input is a tty, the passphrase will be read with readpassphrase(3). -stdinpass replaces -passphrase though the latter is still supported for compatibility. Beware that the password will contain any newlines before the NULL. See EXAMPLES.

-agentpass
: force the default behavior of prompting for a passphrase. Useful with -pubkey to create an image protected by both a passphrase and a public key.

-recover *keychain_file*
: specify a keychain containing the secret corresponding to the certificate specified with -certificate when the image was created.

-certificate *cert_file*
: specify a secondary access certificate for an encrypted image. cert_file must be DER-encoded certificate data, which can be created by Keychain Access or openssl(1).

-pubkey *PK1*,*PK2*,...,*PKn*
: specify a list of public keys, identified by their hexadecimal hashes, to be used to protect the encrypted image being created.

-cacert *cert*
: specify a certificate authority certificate. cert can be either a PEM file or a directory of certificates processed by curl(1).

-insecurehttp
: ignore SS… signed se… unavailab… server na…

-shadow [*shadowfile*]
: Use a sha… primary i… the origi… attached … image, bl… in the ba… device wi… specified … shadow fi… taking im… -insecure…

Verbs that create images … any filenames if the exte… engine also examines the … changes its behavior acco… created without specifyin… .sparsebundle extension t…

VERBS

Each verb is listed with … the verbs can be passed i…

help
: Display minimal usage information for each verb. hdiutil verb -help will provide basic usage information for that verb.

attach *image* [*options*]
: Attach a disk image as a device. attach will return information about an already-attached image as if it had attached it. mount is a poorly-named synonym for attach. See BACKGROUND.

  Beware that an image freshly created and attached is treated as a new removable device. See hdid(8) and the EXAMPLES section below for more details about how owners are ignored on filesystems on such devices.

  The output of attach has been stable since macOS 10.0 (though it was called hdid(8) then) and is intended to be program-readable. It consists of the /dev node, a tab, a content hint (if applicable), another tab, and a mount point (if any filesystems were mounted). Because content hints are derived from the partition data, GUID Partition Table types can leak through. Common GUIDs such as "48465300-0000-11AA-AA11-0030654" are mapped to their human-readable counterparts (here "Apple_HFS").

  Common options: -encryption, -stdinpass, -recover, -imagekey, -shadow, -puppetstrings, and -plist.

```
Options:
-readonly           Force the resulting device to be read-only
-readwrite          Attempt to override the DiskImages frame-
                    work's decision to attach a particular
                    image read-only.  For example, -readwrite
                    can be used to modify the HFS filesystem on
                    a HFS/ISO hybrid CD image.
-nokernel           Attach with a helper process.  This is
                    again the default as of macOS 10.5.
-kernel             attempt to attach this image without a
                    helper process; fail if unsupported.  Only
                    UDRW, UDRO, UDZO, and UDSP images are sup-
                    ported in-kernel.  Encryption and HTTP-
                    backed images are also supported.
-notremovable       Prevent this image from being detached.
                    Only root can use this option.

-mount required|optional|suppressed
                    Indicate whether filesystems in the image
                    should be mounted or not.  The default is
                    required (attach will fail if no filesys-
                    tems mount).
-nomount            Identical to -mount suppressed.
-mountroot path     mount volumes on subdirectories of path
                    instead of under /Volumes.  path must
                    exist.  Full mount point paths must be less
-mountrandom pa
-mountpoint pat
-nobrowse
-owners on|off
-drivekey key=
-section subsp                                          count>
The following
com.apple.fram                                          ered in both
the positive a
-[no]verify                                             h
                    its timestamp changes.
                    To maintain backwards compatibility,
                    hdid(8) does not attempt to verify images
                    before attaching them.
                    Preferences keys: skip-verify, skip-verify-
                    remote, skip-verify-locked, skip-previously-
                    verified
-[no]ignorebadchecksums
                    specify whether bad checksums should be ignored.
                    The default is to cancel when a bad
                    checksum is detected.
                    Preferences key: ignore-bad-checksums
-[no]autoopen       do [not] auto-open volumes (in the Finder) after attaching
                    an image.  By default, dou ble-clicking a read-only
                    disk image causes the resulting volume to be opened in the
                    Finder.  hdiutil defaults to -noautoopen.
-[no]autoopenro     do [not] auto-open read-only volumes.
                    Preferences key: auto-open-ro-root
-[no]autoopenrw     do [not] auto-open read/write volumes.
                    Preferences key: auto-open-rw-root
-[no]autofsck       do [not] force automatic file system check-
                    ing before mounting a disk image.  By
                    default, only quarantined images (e.g. down-
                    loaded from the Internet) that have not pre-
                    viously passed fsck are checked.
                    Preferences key: auto-fsck
```

```
detach dev_name [-force]
          detach a disk image and terminate any associated process.
          dev_name is a partial /dev node path (e.g. "disk1").  As of OS
          X 10.4, dev_name can also be a mountpoint.  If Disk Arbitra-
          tion is running, detach will use it to unmount any filesystems
          and detach the image.  If not, detach will attempt to unmount
          any filesystems and detach the image directly (using the
          `eject' ioctl).  If Disk Arbitration is not running, it can be
          necessary to unmount the filesystems with umount(8) before
          detaching the image.  eject is a synonym for detach.

          Options:
          -force   ignore open files on mounted volumes, etc.

verify image [options]
          compute the checksum of a "read-only" or "compressed" image
          and verify it against the value stored in the image.
          Read/write images don't contain checksums and thus can't be
          verified.  verify accepts the common options -encryption,
          -stdinpass, -srcimagekey, -puppetstrings, and -plist.

create size_spec image
          create a new image of the given size or from the provided
          data.  If image already exists, -ov must be specified or
create will fail.  To make a cross-platform CD or DVD, use
makehybrid instead.  See also EXAMPLES below.

The size speci
Filesystem and
default GPTSPU
for the filesy
Size specifier
-size ??b|??k|
          Spe
          mkf
          exa
          of
          ful
-sectors secto
          Spe
          tor
-megabytes siz
          Spe
          (10

-srcfolder sou
          cop
          ima
          men
          ing
          sin
          spa
          plu
          fil
          of
          rid
          -si
          based on the source content, allowing for more or
          less free space in the resulting filesystem.
          -srcfolder can be specified more than once, in
          which case the image volume will be populated at
          the top level with a copy of each specified
          filesystem object.  -srcdir is a synonym.
-srcdevice device
          specifies that the blocks of device should be used
          to create a new image.  The image size will match
          the size of device.  resize can be used to adjust
          the size of resizable filesystems and writable
          images.  Both -srcdevice and -srcfolder can run
          into errors if there are bad blocks on a disk.  One
          way around this problem is to write over the files
          in question in the hopes that the drive will remap
          the bad blocks.  Data will be lost, but the image
          creation operation will subsequently succeed.
          Filesystem options (like -fs, -volname, -stretch,
          or -size) are invalid and ignored when using
          -srcdevice.

Common options: -encryption, -stdinpass, -certificate,
-pubkey, -imagekey, -tgtimagekey, -puppetstrings, and -plist.

-imagekey di-sparse-puma-compatible=TRUE and -imagekey
di-shadow-puma-compatible=TRUE will create, respectively, sparse and
shadow images that can be attached on macOS 10.1.
```

```
-imagekey encrypted-encoding-version can select between version 1 and version 2
of the encrypted encoding.  The framework preferences have a corresponding
key to change the default for all images.
Version 2 is not compatible with macOS 10.2 but is more robust for
SPARSE (UDSP) images.  Version 1 is the default for non-sparse images.
As of macOS 10.4.7, sparse encrypted images always use version 2 and as
of macOS 10.5, all encrypted images default to version 2.

General options:
-align alignment
          specifies a size to which the final data partition
          will be aligned.  The default is 4K.

-type UDIF|SPARSE|SPARSEBUNDLE
          -type is particular to create and is used to specify
          the format of empty read/write images.  It is inde-
          pendent of -format which is used to specify the
          final read-only image format when populating an
          image with pre-existing content.

          UDIF is the default type.  If specified, a UDRW of
          the specified size will be created.  SPARSE creates
          a UDSP: a read/write single-file image which expands
          as is is filled with data.  SPARSEBUNDLE creates a
          UDSB: a read/write image backed by a directory bun-
          dle.

          By d
          Intr
          whic
          spar
          numb
          time
          rang

          The
          the
          exab
          amou
          spar
          tem
          can
          HFS+
          tual
          TENT

-fs filesystem
          wher
          HFSX
          tem
          imag
          appr
          ment
          fied
          and
-volname volna
          The
          The
          HFS+'s default volume name is `untitled'.  -volname
          is invalid and ignored when using -srcdevice.
-uid uid  the root of the newly-created volume will be owned
          by the given numeric user id.  99 maps to the magic
          `unknown' user (see hdid(8)).
-gid gid  the root of the newly-created volume will be owned
          by the given numeric group id.  99 maps to
          `unknown'.
-mode mode the root of the newly-created volume will have mode
          (in octal) mode.  The default mode is determined by
          the filesystem's newfs unless -srcfolder is speci-
          fied, in which case the default mode is derived from
          the specified filesystem object.
-[no]autostretch
          do [not] suppress automatically making backwards-
          compatible stretchable volumes when the volume size
          crosses the auto-stretch-size threshold (default:
          256 MB).  See also asr(8).
-stretch max_stretch
          -stretch initializes HFS+ filesystem data such that
          it can later be stretched on older systems (which
          could only stretch within predefined limits) using
          hdiutil resize or by asr(8).  max_stretch is speci-
          fied like -size.  -stretch is invalid and ignored
          when using -srcdevice.
-fsargs newfs_args
```

```
                additional arguments to pass to whatever newfs pro-
                gram is implied by -fs.  newfs_hfs(8) has a number
                of options that can reduce the amount of space
                needed by the filesystem's data structures.  Sup-
                pressing the journal with -fs HFS+ and passing argu-
                ments such as -c c=64,a=16,e=16 to -fsargs will min-
                imize gaps at the front of the filesystem, allowing
                resize to squeeze more space from the filesystem.
                For truly optimal filesystems, use makehybrid.

        -layout layout
                Specify the partition layout of the image.  layout
                can be anything supported by MediaKit.framework.
                NONE creates an image with no partition map.  When
                such an image is attached, a single /dev entry will
                be created (e.g. /dev/disk1).

                'SPUD' causes a DDM and an Apple Partition Scheme
                partition map with a single entry to be written.
                'GPTSPUD' creates a similar image but with a GUID
                Partition Scheme map instead.  When attached, multi-
                ple /dev entries will be created, with either slice
                1 (GPT) or slice 2 (APM) as the data partition.
                (e.g. /dev/disk1, /dev/disk1s1, /dev/disk1s2).

                Unless overridden by -fs, the default layout is
                'GPTSPUD' (PPC systems used 'SPUD' prior to macOS
                10.6
                crea

        -library bundl
                spec
                Medi

        -partitionType
                Chan
                disk
                impl

        -ov     over
                over

        -attach atta                                          ified
                via                                           equired behavior.

Image from sou
-format format




Options specif
-segmentSize s                                                bigger



Options specif
-[no]crossdev                                                 em.

        -[no]scrub      Do [not] skip temporary files when imaging a volume.
                        Scrubbing is the default when the source is the root
                        of a mounted volume.
                        Scrubbed items include trashes, temporary directories, swap files, etc.

        -[no]anyowners  Do not fail if the user invoking hdiutil can't
                        ensure correct file ownership for the files in
                        the image.

        -skipunreadable Skip files that can't be read by the copying user and don't authenticate.

        -[no]atomic     Do [not] copy files to a temporary location and then rename them to their
                        destination.  Atomic copies are the default.
                        Non-atomic copying may be slightly faster.

        -copyuid user   Perform the copy as the given user.  Requires root privilege.
                        If user can't read or create files with the needed owners,
                        -anyowners or  -skipunreadable must be used to prevent the
                        operation from failing.

By default, create -srcfolder attempts to maintain the permissions present in the source
directory.  It prompts for authentication if it detects an unreadable file, a file owned by
someone other than the user creating the image, or a SGID file in a group that the copying
user is not in.
```

```
convert image -format format -o outfile
          convert image to type format and write the result to outfile.

          As with create, the correct filename extension will be added only if it isn't part
          of the provided name.  Format is one of:

               UDRW - UDIF read/write image
               UDRO - UDIF read-only image
               UDCO - UDIF ADC-compressed image
               UDZO - UDIF zlib-compressed image
               ULFO - UDIF lzfse-compressed image (OS X 10.11+ only)
               ULMO - UDIF lzma-compressed image (macOS 10.15+ only)
               UDBZ - UDIF bzip2-compressed image (deprecated)
               UDTO - DVD/CD-R master for export
               UDSP - SPARSE (grows with content)
               UDSB - SPARSEBUNDLE (grows with content; bundle-backed)
               UFBI - UDIF entire image with MD5 checksum

         In addition to the compression offered by some formats, the UDIF read-only format skips
         unused space in HFS, APFS, ExFAT, and MS-DOS (FAT, FAT32) filesystems.
         For UDZO, -imagekey zlib-level=value allows the zlib compression level to be specified
         a la gzip(1).  The default compression level is 1 (fastest).

          Common options: -encryption, -stdinpass, -certificate,
          -srcimagekey, -tgtimagekey, -shadow and related,
          -puppetstrings, and -plist.

          Other options:
          -align alignme
                        T

          -pmap         a
                        W
                        o
                        d

          -segmentSize [
                        N
                        S                                          ile is being written.
                        T                                           is
                        2
                        a
                        s
                        ?

          -tasks task_co
                        W                                           the number of
                        t
                        T                                          ent system.

burn image

          Burn image to
          In all cases,                                            ive has been found.
          Common options                                          strings, and -stdinpass.

          Other options:
          -device

          -testburn
          -anydevice    explicitly allow burning to devices not qual-
                        ified by Apple (kept for backwards compati-
                        bility as burn will burn to any device by
                        default as of macOS 10.4).

          -[no]eject    do [not] eject disc after burning. The
                        default is to eject the disc.
          -[no]verifyburn  do [not] verify disc contents after burn.
                        The default is to verify.

          -[no]addpmap  do [not] add partition map if necessary.
                        Some filesystem types will not be recognized
                        when stored on optical media unless they are
                        enclosed in a partition map.  This option
                        will add a partition map to any bare filesys-
                        tem which needs a partition map in order to
                        be recognized when burned to optical media.
                        The default is to add the partition map if
                        needed.

          -[no]skipfinalfree do [not] skip final free partition.  If
                        there is a partition map on the image speci-
                        fying an Apple_Free partition as the last
                        partition, that Apple_Free partition will not
                        be burned.  The burned partition map will
                        still reference the empty space.  The default
```

```
                  is to skip burning a final free partition.

-[no]optimizeimage do [not] optimize filesystem for burning.
                  Optimization can reduce the size of an HFS or
                  HFS+ volume to the size of the data contained
                  on the volume.  This option will change what
                  is burned such that the disc will have a dif-
                  ferent checksum than the image it came from.
                  The default is to burn all blocks of the disk
                  image (minus any trailing Apple_Free).

-[no]forceclose   do [not] force the disc to be closed after
                  burning.  Further burns to the disc will be
                  impossible.  The default is not to close the
                  disc.

-nounderrun       Disable the default buffer underrun protec-
                  tion.

-[no]synthesize   [Don't] Synthesize a hybrid filesystem for
                  the disc.  The default is to create a new
                  (HFS/ISO) filesystem when the source image's
                  blocks could not be legally burned to a disc.

-speed x_factor   1, 2, 4, 6, ...  `max'
                  The desired "x-factor".  e.g. 8 means the
                  drive will be instructed burn at "8x speed"

-sizequery

-erase

-fullerase

-list

makehybrid -o image sourc
           Generate a pot
           image using th
           system.  This
           drutil(1) can
```



```
source can either be a directory or a disk image.  The generated image can
later be burned using burn, or converted to another read-only format with convert.
By  default, the filesystem will be readable on most modern computing platforms.
The generated filesystem is not intended for conversion to read/write, but can
safely have its files copied to a read/write filesystem using ditto(8).

hdiutil supports generating El Torito-style bootable ISO9660 filesystems, which are
commonly used for booting x86-based hardware. The specification includes several
emulation modes. By default, an El Torito boot image emulates either a 1.2MB, 1.44MB,
or 2.88MB floppy drive, depending on the size of the image.  Also  available are
"No Emulation" and "Hard Disk Emulation" modes, which allow the boot image to either
be loaded directly into memory, or be virtualized as a partitioned hard disk,
respectively. The El Torito options should not be used for data CDs.

Filesystem options:
-hfs    Generate an HFS+ filesystem.
        This filesystem can be present on an image simultaneously with an ISO9660
        or Joliet or UDF filesystem.  On Operating Systems that understand HFS+ as well
        as ISO9660 and UDF, like Mac OS 9 or OS X, HFS+ is usually the preferred
        filesystem for hybrid images.
-iso    Generate an ISO9660 Level 2 filesystem with Rock Ridge extensions.
        This filesystem can be present on an image simultaneously with an HFS+ or Joliet or
        UDF filesystem.  ISO9660 is the standard cross-platform interchange format for
        CDs and some DVDs, and is understood by virtually all Operating Systems.
        If an ISO9660 or Joliet filesystem is present on a disk image or CD, but not
        HFS+, OS X will use the ISO9660 (or Joliet) filesystem.
```

```
-joliet Generate Joliet extensions to ISO9660.
       This view of the filesystem can be present on an image simultaneously with HFS+,
       and requires the presence of an ISO9660 filesystem.
       Joliet supports Unicode filenames, but is only supported on some Operating Systems.
       If both an ISO9660 and Joliet filesystem are present on a disk image
       or CD, but not HFS+, OS X will prefer the Joliet filesystem.
-udf   Generate a UDF filesystem. This filesystem can be present on an image
       simultaneously with HFS+, ISO9660, and Joliet. UDF is the standard interchange
       format for DVDs, although Operating System support varies based on OS version
       and UDF version.

By default, if no filesystem is specified, the image will be created with all four
filesystems as a hybrid image.  When multiple filesystems are selected, the data area
of the image is shared between all filesystems, and only directory information and
volume meta-data are unique to each filesystem.  This means that creating a
cross-platform ISO9660/HFS+ hybrid has a minimal overhead when compared to a
single filesystem image.

Other options (most take a single argument):
-hfs-blessed-directory Path to directory which should be "blessed" for OS X booting
                       on the generated filesystem.
                       This assumes the directory has been otherwise prepared,
                       for example with bless -bootinfo to create a valid BootX file.
                       (HFS+ only).
-hfs-openfolder        Path to a directory that will be opened by the Finder
                       automatically.  See also  the -openfolder option in bless(8)
                       (HFS+ only)
-hfs-startupfi                                                    fied size,

-abstract-file                                                     the root of the generated
                                                                  ct file

-bibliography-                                                     the root of the generated
                                                                  graphy file

-copyright-fil                                                     the root of the generated
                                                                  ght file

-application
-preparer
-publisher
-system-id

-keep-mac-spec                                                    e) in non-HFS+ filesystems

-eltorito-boot                                                    e directory. By default,
                                                                  t be one of 1200KB, 1440KB, or
                                                                  r -no-emul-boot or
                                                                  tion" or "Hard Disk Emulation"

-hard-disk-boo                                                    n MBR partition map and a

-no-emul-boot                                                     ware will load the number of
                                                                   it, without emulating any

-no-boot                                                          stem firmware can still
                                                                  s option is not recommended
                       (ISO9660/Joliet).
-boot-load-seg         For a No Emulation boot image, load the data at the specified segment address.
                       This options is not recommended, so that the system firmware can use its
                       default address (ISO9660/Joliet)
-boot-load-size        For a No Emulation boot image, load the specified number of 512-byte emulated
                       sectors into memory and execute it. By default, 4 sectors (2KB) will be loaded
                       (ISO9660/Joliet).
-eltorito-platform     Use the specified numeric platform ID in the El Torito Boot Catalog Valida-
                       tion Entry or Section Header. Defaults to 0 to identify x86 hardware
                       (ISO/Joliet).
-eltorito-specification For complex layouts involving multiple boot images, a plist-formatted string
                       can be provided, using either OpenStep-style syntax or XML syntax, represent-
                       ing an array of dictionaries. Any of the El Torito options can be set in the
                       sub-dictionaries and will apply to that boot image only. If
                       -eltorito-specification is provided in addition to the normal El Torito com-
                       mand-line options, the specification will be used to populate secondary non-
                       default boot entries.
-udf-version           Version of UDF filesystem to generate.
                       This can be either "1.02" or "1.50". If not specified, it defaults to "1.50"
                       (UDF).

-default-volume-name   Default volume name for all filesystems, unless overridden.
                       If not specified, defaults to the last path component of source.
-hfs-volume-name       Volume name for just the HFS+ filesystem if it should be different
                       (HFS+ only).
-iso-volume-name       Volume name for just the ISO9660 filesystem if it should be different
```

```
                        (ISO9660 only).
-joliet-volume-name     Volume name for just the Joliet filesystem if it should be different
                        (Joliet only).
-udf-volume-name        Volume name for just the UDF filesystem if it should be different
                        (UDF only).

-hide-all               A glob expression of files and directories that should not
                        be exposed in the generated filesystems.  The string might
                        need to be quoted to avoid shell expansion, and will be
                        passed to glob(3) for evaluation.  Although this option cannot
                        be used multiple times, an arbitrarily complex glob expression
                        can be used.
-hide-hfs               A glob expression of files and directories that should not
                        be exposed via the HFS+ filesystem, although the data can
                        still be present for use by other filesystems (HFS+ only).
-hide-iso               A glob expression of files and directories that should not
                        be exposed via the ISO filesystem, although the data can
                        still be present for use by other filesystems (ISO9660 only).  Per above,
                        the Joliet hierarchy will supersede the ISO hierarchy when the hybrid is
                        mounted as an ISO 9660 filesystem on macOS.  Therefore, if Joliet is
                        being generated (the default) -hide-joliet will also be needed to
                        hide the file from mount_cd9660(8).
-hide-joliet            A glob expression of files and directories that should not be exposed via the
                        Joliet filesystem, although the data can still be present for use by other
                        filesystems (Joliet only).  Because OS X's ISO 9660 filesystem uses the Joliet
                        catalog if it is available, -hide-joliet effectively supersedes
                        -hide-iso when the resulting filesystem is mounted as ISO on macOS.
-hide-udf                                                                  should not
                                                                           data can
                                                                           DF only).
-only-udf                                                                   exposed in UDF.
-only-iso                                                                   exposed in ISO.
-only-joliet                                                                exposed in Joliet.

-print-size


-plistin                                                                   standard plist
                                                                           f the hybrid image
                                                                           a key in the
                                                                           ue should be
                                                                            for number arguments,
                                                                           gument should use a
                                                                           f "output".

If a disk imag                                                             d paths will be
evaluated rela                                                             an be used in
this case.  If                                                             o files or directories
either via an                                                              ing directory.

The volume nam                                                             o be mapped onto
the legal char                                                             obey naming
restrictions.                                                              en string would be remapped.

The -abstract-                                                              directly in the source
directory, not                                                             lity with ISO9660 Level 1.
```



```
compact image options
          scans the bands of a sparse (SPARSE or SPARSEBUNDLE) disk image containing an APFS or HFS+
          filesystem, removing those parts of the image which are no longer being
          used by the filesystem.  Depending on the location of files in the hosted filesystem,
          compact may or may not shrink the image.  For SPARSEBUNDLE images, completely unused
          band files are simply removed.

          Options:
          -batteryallowed  Allow compacting on battery power.
                           SPARSE images could be damaged if power is lost during a compact operation.
                           The default is not allowed.

          -sleepallowed    Allow machine to idle sleep while compacting, which cancels the compact operation.
                           The default is not allowed, which prevents idle sleep until
                           compact completes.
                           User-initiated sleep, such as a lid close, will always cancel compact.

          Common options: -encryption, -stdinpass, -srcimagekey, -shadow and related, -puppetstrings, and -plist.

info      display information about DiskImages.framework, the disk image driver, and any images that are
          currently attached.  hdiutil info accepts -plist.

checksum image -type type
          Calculate the specified checksum on the image data, regardless of image type.

          Common options: -shadow and related, -encryption, -stdinpass,
```

```
            -srcimagekey, -puppetstrings, and -plist.

            type is one of:
                  UDIF-CRC32 - CRC-32 image checksum
                  UDIF-MD5 - MD5 image checksum
                  DC42 - Disk Copy 4.2
                  CRC28 - CRC-32 (NDIF)
                  CRC32 - CRC-32
                  MD5 - MD5
                  SHA - SHA
                  SHA1 - SHA-1
                  SHA256 - SHA-256
                  SHA384 - SHA-384
                  SHA512 - SHA-512

      chpass image
            change the passphrase for an encrypted image.  The default is
            to change the password interactively.

            Common options: -recover and -srcimagekey.  The options
            -oldstdinpass and -newstdinpass allow, in the order specified,
            the null-terminated old and new passwords to be read from the
            standard input in the same manner as with -stdinpass.

      erasekeys image
            securely overwrite keys used to access an encrypted image,
            quickly rendering the image completely inaccessible.  Once
            erasekeys has
            sible way to r

            Common options

      unflatten image
            unflatten a UD
            image file (no
            tation of the
            tion will fail

            Common options

      flatten image
            Flatten a read
            single-fork fi
            XML (for the k
            (for macOS 10.

            Common options
            Since images a
            required if th

            Other options:
            -noxml      do
                        Th
            -norsrcfork do
                        no

      fsid image
            Print informat
            As usual, image
            cal disk.  See the NOTE ON DEV ENTRY ACCESS section.  More
            detailed information is presented for HFS file systems.

            Common options: -encryption, -stdinpass, -srcimagekey, and
            -shadow and related.

      mountvol dev_name
            mount the filesystem in dev_name using Disk Arbitration (simi-
            lar to diskutil(8)'s mount). XML output is available from
            -plist.  Note that mountvol (rather than mount, though it
            often works in macOS 10.5 and later) is the correct way to
            remount a volume after it has been unmounted by unmount.

            Prior to macOS 10.5, mount/attach would treat a /dev entry as a
            disk image to be attached (creating another /dev entry).  That
            behavior was undesirable.

      unmount volume [-force]
            unmount a mounted volume without detaching any associated
            image.  Volume is a /dev entry or mountpoint.  NOTE: unmount
            does NOT detach any disk image associated with the volume.
            Images are attached and detached; volumes are mounted and
            unmounted.  mountvol will remount a volume that has been
            unmounted by unmount.

            Options:
```

```
            -force   unmount filesystem regardless of open files on that
                     filesystem.  Similar to umount -f.

    imageinfo image
            Print out information about a disk image.

            Common options: -encryption, -stdinpass, -srcimagekey, -shadow
            and related, and -plist.

            Options are any of:
            -format   just print out the image format
            -checksum just print out the image checksum

    isencrypted image
            print a line indicating whether image is encrypted.  If it is,
            additional details are printed.

            Common options: -plist.

    plugins print information about DiskImages framework plugins.  The
            user, system, local, and network domains are searched for
            plugins (i.e.  ~/Library/Plug-ins/DiskImages,
            /System/Library/Plug-ins/DiskImages,
            /Library/Plug-ins/DiskImages,
            /Network/Library/Plug-ins/DiskImages).
            Common options: -plist.

    internet-enable [-yes] | -
            Enable or disa
            the default.
            will "unpack"
            be copied into
            image will be

            Common options
            -plist.

    resize size_spec image
            Resize a disk
            containing a t
            resize the ima
            within it by a
            the end of the
            tems other tha

            resize can shr
            be converted t
            hdiutil resize
            data.  diskuti
            help hdiutil r
            can also be us

            resize is limi
            UDSP vs. UDSB)
            and the filesy
            inside of GPT
            space, the lim
            created with a
            Before macOS 10.4, resize was limited by how the filesystem was
            created (see hdiutil create -stretch).

            hdiutil burn does not burn Apple_Free partitions at the end of
            the devices, so an image with a resized filesystem can be
            burned to create a CD-R/DVD-R master that contains only the
            actual data in the hosted filesystem (assuming minimal data
            fragmentation).

            Common options: -encryption, -stdinpass, -srcimagekey, -shadow
            and related, and -plist.

            Size specifiers:
            -size ??b|??k|??m|??g|??t??p|??e
            -sectors sector_count | min
                           Specify the number of 512-byte sectors to
                           which the partition should be resized.  If
                           this falls outside the mininum valid value or
                           space remaining on the underlying file sys-
                           tem, an error will be returned and the parti-
                           tion will not be resized.  min automatically
                           determines the smallest possible size.

            Other options:
            -imageonly     only resize the image file, not the parti-
                           tion(s) and filesystems inside of it.
```

```
-partitiononly   only resize a partition / filesystem in the
                 image, not the image.  -partitiononly will
                 fail if the new size won't fit inside the
                 image.  On APM, shrinking a partition results
                 in an explicit Apple_Free entry taking up the
                 remaining space in the image.
-partitionNumber partitionNumber
                 specifies which partition to resize (UDIF
                 only -- see HISTORY below).  partitionNumber
                 is 0-based, but, per hdiutil pmap, partition
                 0 is the partition map itself.

-growonly        only allow the image to grow
-shrinkonly      only allow the image to shrink
-nofinalgap      allow resize to entirely eliminate the trail-
                 ing free partition in an APM map.  Restoring
                 such images to very old hardware can inter-
                 fere with booting.

-limits          Displays the minimum, current, and maximum
                 sizes (in 512-byte sectors) for the image.
                 In addition to any hosted filesystem con-
                 straints, UDRW images are constrained by
                 available disk space in the filesystem host-
                 ing the image.  -limits does not modify the
                 image.

segment
          segment -o fir
          segment -o fir
          segment a NDIF
          around limitat
          filesystems, n
          not the segmen
          passed to segm

          Common options
          -tgtimagekey,

          Options:
          -segmentCount
                       S
                       -
          -segmentSize s
                       S
                       s
                       s
                       n
                       l
                       O
                       h
                       n

          -firstSegmentS
                       S
                       s
                       r
          -restricted  M
                       r
          -ov          overwrite any existing files.

pmap [options] image
          display the partition map of an image or device.  By default,
          this report includes starting offsets and significant amounts
          of free space.  image is either a plain or special file (for
          example, a /dev/disk entry).  See NOTE ON DEV ENTRY ACCESS.

          Common options: -encryption, -stdinpass, -srcimagekey, and
          -shadow and related.

          -simple       generate MediaKit's minimal report: basic parti-
                        tion types, names, and sizes in human-readable
                        units.
          -standard     generate MediaKit's standard report, which adds
                        partition offsets and uses 512-byte sectors.
          -complete     generate MediaKit's comprehensive report, with
                        end offsets, significant free space, etc.

          -endoffsets   indicate last block of each partition.
          -nofreespace  suppress all free space reporting.  Not valid
                        with -shims.
          -shims        report free space < 32 sectors.
          -uuids        show per-instance UUIDs for each partition.  APM
                        does not store instance UUIDs so these will be
                        randomly generated for APM maps.
```

```
     udifrez [options] image
               embed resources (e.g. a software license agreement) in a disk
               image.

               You must specify one of the following options:
               -xml file
                    Copy resources from the XML in file.
               -rsrcfork file
                    Copy resources from file's resource fork.
               -replaceall
                    Delete all pre-existing resources in image.

     udifderez [options] image
               extract resources from image.

               Options:
               -xml    emit XML output (default)
               -rez    emit Rez format output

               Common options: -encryption, -stdinpass, and -srcimagekey.

EXAMPLES

     Verifying:
           hdiutil verify myimage.img
                 verifies an image against its internal checksum

     Segmenting:
           hdiutil segment -se
                 creates aseg.

     Converting:
           hdiutil convert mas
                 converts maste
           hdiutil convert /de
                 converts the
                 file.  authop
                 is not availa

     Burning:
           hdiutil burn myImag
                 burns the ima
           hdiutil burn myRawI
                 burns the ima
                 disc.  Volume

     Creating a 50 MB encrypte
           hdiutil create -enc

     Creating a 50 MB encrypte
           hdiutil create -enc
               -pubkey F534A3B

     Creating a 50 MB encrypte
           hdiutil create -enc
               -pubkey F534A3B

     Note that these two -pubke
     sponding to this public key is currently in the user's keychain or smart
     card.  For additional information on smart card authorization setup see
     sc_autch(8).

     Creating an encrypted single-partition image without user interaction:
           printf pp|hdiutil create -encryption -stdinpass -size 9m sp.dmg

     Creating a "1 GB" SPARSE image (a 1 GB filesystem in a growable file):
           hdiutil create -type SPARSE -size 1g -fs HFS+ growableTo1g

     Creating a "1 GB" SPARSEBUNDLE (a 1 GB filesystem in a growable bundle):
           hdiutil create -type SPARSEBUNDLE -size 1g -fs HFS+ growableTo1g

     Creating a new mounted volume backed by an image:
           hdiutil create -volname Dick -size 1.3m -fs HFS+ -attach Moby.dmg

     Using a shadow file to attach a read-only image read-write to modify it,
     then convert it back to a read-only image. This method eliminates the
     time/space required to convert a image to read-write before modifying it.

           hdiutil attach -owners on Moby.dmg -shadow
           /dev/disk2   Apple_partition_scheme
           /dev/disk2s1 Apple_partition_map
           /dev/disk2s2 Apple_HFS                /Volumes/Moby

           ditto /Applications/Preview.app /Volumes/Moby
```

```
      hdiutil detach /dev/disk2
      hdiutil convert -format UDZO Moby.dmg -shadow

Using makehybrid to create cross-platform data with files overlapping
between filesystem views.  With these files:
      albumlist.txt song2.wma     song4.m4a     song6.mp3     song8.mp3
      song1.wma     song3.m4a     song5.mp3     song7.mp3

      hdiutil makehybrid -o MusicBackup.iso Music -hfs -iso -joliet \
          -hide-hfs 'Music/*.wma' -hide-joliet 'Music/{*.m4a,*.mp3}' \
          -hide-iso 'Music/*.{wma,m4a}'

will create an image with three filesystems pointing to the same blocks.
The HFS+ filesystem, typically only visible on Macintosh systems, will
not include the .wma files, but will show the .m4a and .mp3 files. The
Joliet filesystem will not show the .m4a and .mp3 files, but will show
the .wma files. The ISO9660 filesystem, typically the default filesystem
for optical media on many platforms, will only show the .mp3 files. All
three filesystems will include the "albumlist.txt" files.

Image from directory (new-style):
      hdiutil create -srcfolder mydir mydir.dmg

Image from directory (10.1-style; of historical interest):
      du -s myFolder            # du(1) will count resource forks
      10542
      hdiutil create -sectors 10642 folder     # add ~1% for filesytem
      hdid -nomount folde
      ...
      /dev/disk1s2
      newfs_hfs -v myFold
      hdiutil detach disk
      hdid folder.dmg
      ...
      /dev/disk1s2
      sudo mount -u -t hf
      # optionally enable

      ditto -rsrcFork myF
      hdiutil detach disk
      hdiutil convert -fo

Manually changing ownersh
      hdiutil attach myim
      ...
      /dev/disk1s2
      diskutil unmount di
      mkdir /Volumes/myVo
      sudo mount -r -t hf
      # -o owners is the

Forcing a known image to
      hdiutil attach -ima

ENVIRONMENT

     The following environment

     com_apple_hdid_verbose
               enable -verbose behavior for attach.

     com_apple_hdid_debug
               enable -debug behavior for attach.

     com_apple_hdid_nokernel
               similar to -nokernel but works even with, for example, create
               -attach.

     com_apple_hdid_kernel
               attempt to attach in-kernel first (like attach -kernel). In OS
               X 10.4.x, in-kernel was the default behavior for UDRW and
               SPARSE images.  On macOS 10.5, these and other kernel-compati-
               ble images, including RAM-based images described in hdid(8),
               will attach with a user process unless attach -kernel is used
               or the corresponding variable is set.  If an image is not
               "kernel-compatible" and -kernel is specified, the attach will
               fail.  (WARNING: ram:// images currently use wired memory when
               attached in-kernel).

     com_apple_diskimages_insecureHTTP
               disable SSL peer verification the same way -insecurehttp does.
               Useful for clients of DiskImages such as asr(8) which don't
               support a similar command line option.

ERRORS
```

DiskImages uses many frameworks and can encounter many error codes. In general, it tries to turn these error numbers into localized strings for the user. For background, intro(2) is a good explanation of our primary error domain: the BSD errno values. For debugging, -verbose should generally provide enough information to figure out what has gone wrong. The following is a list of interesting errors that hdiutil can encounter:

[ENXIO] Device not configured. This error is returned explicitly by DiskImages when its kernel driver or framework helper cannot be contacted. It also often shows up when a device has been removed while I/O is still active. One common case of the helper not being found is when Foundation's Distributed Objects RPC mechanism cannot be configured. D.O. doesn't work under dead Mach bootstrap contexts such as can exist in a reattached screen(1) session. Root users can take advantage of StartupItemContext(8) (in /usr/libexec) to access the startup item Mach bootstrap context.

[EINVAL] Invalid argument. This error is used in many contexts and is often a clue that hdiutil's arguments are subtly non-sensical (e.g. an invalid layout name passed to create -layout).

[EFBIG] File too large. DiskImages uses this error explicitly when attempting to access a disk image over HTTP that is too
This e
old-st

[EAUTH] Authen
libcur
Securi
enter
-cacer
peers.

[EBUSY] Resour
cannot
volume

[EAGAIN] Resour
DiskIm
to pre
machin
return
be obt

EACCES vs. EPERM EACCES
"opera
tion t
rect e
sion d
access

USING PERSISTENT SPARSE IMAGES

As of macOS 10.5, a more reliable, efficient, and scalable sparse format, UDSB (SPARSEBUNDLE), is recommended for persistent sparse images as long as a backing bundle (directory) is acceptable. macOS 10.5 also introduced F_FULLFSYNC over AFP (on client and server), allowing proper journal flushes for HFS+J-bearing images. Critical data should never be stored in sparse disk images on file servers that don't support F_FULLFSYNC.

SPARSE (UDSP) images and shadow files were designed for intermediate use when creating other images (e.g. UDZO) when final image sizes are unknown. As of macOS 10.3.2, partially-updated SPARSE images are properly handled and are thus safe for persistent storage. SPARSE images are not recommended for persistent storage on versions of macOS earlier than 10.3.2 and should be avoided in favor of SPARSEBUNDLE images or UDRW
images and resize.

If more space is needed than is referenced by the hosted filesystem, hdiutil resize or diskutil(8) resize can help to grow or shrink the filesystem in an image. compact reclaims unused space in sparse images. Though they request that hosted HFS+ filesystems use a special "front first" allocation policy, beware that sparse images can enhance the
effects of any fragmentation in the hosted filesystem.

To prevent errors when a filesystem inside of a sparse image has more free space than the volume holding the sparse image, HFS volumes inside sparse images will report an amount of free space slightly less than the amount of free space on the volume on which image resides. The image filesystem currently only behaves this way as a result of a direct attach action and will not behave this way if, for example, the filesystem is unmounted and remounted.

/dev Entry Access

Since any /dev entry can be treated as a raw disk image, it is worth noting which devices can be accessed when and how. /dev/rdisk nodes are character-special devices, but are "raw" in the BSD sense and force block-aligned I/O. They are closer to the physical disk than the buffer cache. /dev/disk nodes, on the other hand, are buffered block-special
devices and are used primarily by the kernel's filesystem code.

It is not possible to read from a /dev/disk node while a filesystem is mounted from it, but anyone with read access to the appropriate /dev/rdisk node can use hdiutil verbs such as fsid or pmap with it. The DiskImages framework will attempt to use authopen(1) to open any device which it can't open (due to EACCES) for reading with open(2). This might cause apparent hangs while trying to access /dev entries while logged in remotely (an authorization panel is waiting on console).

Generally, the /dev/disk node is preferred for imaging devices (e.g. convert or create -srcdevice operations), while /dev/rdisk is usable for the quick pmap or fsid. In particular, converting the blocks of a mounted journaled filesystem to a read-only image will prevent the volume in the image from mounting (the journal will be permanently dirty).

Compatibility

macOS 10.0 supported the disk images of Disk Copy 6 on Mac OS 9. macOS 10.1 added sparse, encrypted, and zlib-compressed images. These images will not be recognized on macOS 10.0 (or will attach read/write, possibly allowing for their destruction). As the sparse, shadow, and encrypted formats have evolved, switches have been added to facilitate the creation of images that are compatible with older OS versions (at the expense of the performance and reliability improvements offered by the format enhancements). In particular, sparse images should not be expected to attach on versions of macOS older than that which created them.

With macOS 10.2, the most common image formats went "in-kernel" (i.e. the DiskImages kernel extension served them without a helper process), image meta-data began being sto Copy.app "compressed" format became UDZO (breaking compat ormat which provides smaller images (especially when combined with m

In macOS 10.4.7, the resource for l metadata length limitations imposed by resource fork structures. As a r nized by either macOS 10.1 or macOS 10.0. flatten can be used t

macOS 10.5 introduced sparse b S versions. macOS 10.6 removed support for attaching SPARSEBU macOS 10.7 removed double-clic ng flatten and unflatten, or convert.

History

Disk images were first invented to acturing replication. These images of floppies are typically referred to as estored them to floppy disks. Disk Copy 4.2 images were block-for-b disk, with no notion of compressio on.

NDIF (New Disk Image Format) in hats and to support images larger than a floppy disk. With NDIF and s under Mac OS 9. Apple Data Compression (ADC) -- which care used to compress images that we

UDIF (Universal Disk Image Form sent entire block devices and all the data therein: DDM, partition map, n can then be replicated from an image.
To ensure single-fork files (NDIF w s the native image format for macOS.

Raw disk images from other Operating Systems (e.g. .iso files) will be recognized as disk images and can be attached and mounted if macOS recognizes the filesystems. They can also be burned with hdiutil burn.

What's New

In macOS 10.12 Apple will provide an updated `hdutil` command able to work with the new file system.

macOS 10.7 added the ability to quickly render encrypted images inaccessible using the new erasekeys verb, which saves time versus securely overwriting the entire image.

In macOS 10.6, pmap was rewritten to use MediaKit's latest reporting routines so that it can properly support GPT partition maps. Also -debug now implies -verbose for all verbs.

macOS 10.5 changed the behavior of attach when run on an existing image or /dev node: if the image was attached but no volume was mounted, the volume would be mounted. Prior systems would return the /dev without mounting the volume. This change effectively removes the ability to create a second /dev node from an existing one.

Examples

Mount a Disk Image:

```
$ hdiutil attach /path/to/diskimage.dmg
```

Unmount a Disk Image:

```
$ hdiutil detach /dev/disk2s1
```

Create a Disk Image from a folders contents:

```
$ hdiutil create -volname "Volume Name" -srcfolder /path/to/folder -ov diskimage.dmg
```

Create an encrypted Disk Image from a folders contents:

```
$ hdiutil create -encryption -stdinpass -volname "Volume Name" -srcfolder /path/to/folder -ov encrypted.dmg
```

The required password can be piped into the hdiutil command:

```
echo -n SEcurePa$$w0rd | hdiutil...
```

Burn a Disk Image file (.iso, .img or .dmg) to a DVD:

```
$ hdiutil burn /path/to/image_file
```

"The beginning of wisdom is to call things by their right names" ~ Chinese Proverb

Related macOS commands

asr - Apple Software Restore.
dd - Convert and copy a file, clone disks
diskutil - Disk utilities - Format, Verify, R
ditto - Copy files and folders.
authopen(1), hdid(8), ioreg(8), drutil(1), /DiskImageMounter.app.