

Manual Page Search Parameters

Q ldid

man

apropos

1 - General Commands

All Architectures

ldid

NAME

ldid — Link Identity Editor

SYNOPSIS

```
ldid [-Acputype:subtype] [-a] [-C[adhoc | enforcement | expires | hard | host | kill | library-validation | restrict | runtime]] [-D] [-d] [-Eenum:file] [-e] [-H[sha1 | sha256]] [-h] [-Iname] [-Kkey.pl2] [-Upassword]] [-M] [-P[num]] [-Qrequirements] [-q] [-r] [-Sfile.xml] [-s] [-u] [-arch archtype] file ...
```

DESCRIPTION

ldid adds SHA1 and SHA256 hashes to a Mach-O file so that they can be run on a system that has validation but not signature verification.

-Acpu_{type}:sub_{type}

When used with **-a**, **-D**, **-e**, **-h**, **-q**, or **-u**, only act on the slice specified by *cpu_{type}* and *sub_{type}*. *cpu_{type}* and *sub_{type}* should both be integers.

-a

Print the CPU types and subtypes in hexadecimal.

-arch arch_{type}

The same as **-A**, except the name of the architecture is used. The list of currently known *arch_{types}* can be found in [arch\(3\)](#). This is a Procursus extension.

-C[ad_{hoc} | enforcement | expires | hard | host | kill | library-validation | restrict | runtime]

Specify the option flags to embed in the code signature. See [codesign\(1\)](#) for details about these options.

-D

Reset the cryptid.

-d

Print the cryptid in the binaries if it exists.

-Eenum:file

Embed the hashes of *file* in the special codesign slot at *num*.

-e

Print the entitlements in each slice, or the slice specified by **-A** or **-arch** to *stdout*.

-H[sha1 | sha256]

Disable the hash not specified. This is useful to replicate the default behavior of [codesign\(1\)](#), which only provides an sha256 signature.

-h

Print information about the signature, such as hash types, flags, CDHash, and CodeDirectory version to *stdout*.

-Iname

Set the identifier used in the binaries signature to *name*. If not specified, the basename of the binary is used.

-Kkey.pl2

Sign using the identity in *key.pl2*. This will give the binary a valid signature so that it can be run on a system with signature validation. If *key.pl2* has a password you will be prompted for it, or you can specify from the command line with **-U**.

-M

When used with **-S**, merge the new and existing entitlements instead of replacing the existing entitlements, this is useful for adding a few specific entitlements to a handful of binaries.

-P[num]

Mark the Mach-O as a platform binary. If *num* is specified, the platform field in the CodeDirectory will be set to that number. The default is 13, as per Apple binaries. Specifying the platform to set to using **-P** is a Procursus extension.

-Qrequirements.xml

Embed the requirements found in *requirements*.

-q

Print embedded requirements of the binaries.

-r

Remove the signature from the Mach-O.

-S[file.xml]

Pseudo-sign the Mach-O binaries. If *file.xml* is specified then the entitlements found in *file.xml* will be embedded in the Mach-O.

-s

Resign the Mach-O binaries while keeping the existing entitlements.

-Upass_{word}

Use *password* as the password for the p12 certificate instead of prompting. This is a Procursus extension.

-u

If the binary was linked against UIKit, then print the UIKit version that the Mach-O binaries were linked against.

EXAMPLES

The command:

```
ldid -S file
```

will fakesign *file* with no entitlements.

The command:

```
ldid -Cadhoc -K/path/to/key.pl2 -Sent.xml file
```

will sign *file* using the key in */path/to/key.pl2* with the entitlements found in *ent.xml*, and mark it as an adhoc signature.

The command:

```
ldid -Sent.xml -M file
```

will add the entitlements in *ent.xml* to the entitlements already in *file*.

The command:

```
ldid -e file > ent.xml
```

will save the entitlements found in each slice of *file* to *ent.xml*.

SEE ALSO

[codesign\(1\)](#)

HISTORY

The **ldid** utility was written by Jay "Saurik" Freeman. iPhoneOS 1.2.0 and 2.0 support was added on April 6, 2008. **-S** was added on June 13, 2008. SHA256 support was added on August 25, 2016, fixing iOS 11 support. iOS 14 support was added on July 31, 2020 by Kabir Oberai. iOS 15 support was added on June 11, 2021.