

NAME

openssl-rehash, c_rehash - Create symbolic links to files named by the hash values

SYNOPSIS

openssl rehash [-h] [-help] [-old] [-compat] [-n] [-v] [-provider *name*] [-provider-path *path*] [-propquery *propq*] [*directory*] ...

c_rehash [-h] [-help] [-old] [-n] [-v] [-provider *name*] [-provider-path *path*] [-propquery *propq*] [*directory*] ...

DESCRIPTION

This command is generally equivalent to the external script **c_rehash**, except for minor differences noted below.

openssl rehash scans directories and calculates a hash value of each *.pem*, *.crt*, *.cer*, or *.crl* file in the specified directory list and creates symbolic links for each file, where the name of the link is the hash value. (If the platform does not support symbolic links, a copy is made.) This command is useful as many programs that use OpenSSL require directories to be set up like this in order to find certificates.

If any directories are named on the command line, then those are processed in turn. If not, then the **SSL_CERT_DIR** environment variable is consulted; this should be a colon-separated list of directories, like the Unix **PATH** variable. If that is not set then the default directory (installation-specific but often */etc/ssl/certs*) is processed.

In order for a directory to be processed, the user must have write permissions on that directory, otherwise an error will be generated.

The links created are of the form *HHHHHHHH.D*, where each *H* is a hexadecimal character and *D* is a single decimal digit. When a directory is processed, all links in it that have a name in that syntax are first removed, even if they are being used for some other purpose. To skip the removal step, use the **-n** flag. Hashes for CRL's look similar except the letter **r** appears after the period, like this:
HHHHHHHH.rD.

Multiple objects may have the same hash; they will be indicated by incrementing the *D* value. Duplicates are found by comparing the full SHA-1 fingerprint. A warning will be displayed if a duplicate is found.

A warning will also be displayed if there are files that cannot be parsed as either a certificate or a CRL or if more than one such object appears in the file.

Script Configuration

The **c_rehash** script uses the **openssl** program to compute the hashes and fingerprints. If not found in the user's **PATH**, then set the **OPENSSL** environment variable to the full pathname. Any program can be used, it will be invoked as follows for either a certificate or CRL:

```
$OPENSSL x509 -hash -fingerprint -noout -in FILENAME
```

```
$OPENSSL crl -hash -fingerprint -noout -in FILENAME
```

where *FILENAME* is the filename. It must output the hash of the file on the first line, and the fingerprint on the second, optionally prefixed with some text and an equals sign.

OPTIONS

-help -h

Display a brief usage message.

-old Use old-style hashing (MD5, as opposed to SHA-1) for generating links to be used for releases before 1.0.0. Note that current versions will not use the old style.

-n Do not remove existing links. This is needed when keeping new and old-style links in the same directory.

-compat

Generate links for both old-style (MD5) and new-style (SHA1) hashing. This allows releases before 1.0.0 to use these links along-side newer releases.

-v Print messages about old links removed and new links created. By default, this command only lists each directory as it is processed.

-provider *name*

-provider-path *path*

-propquery *propq*

See "Provider Options" in **openssl(1)**, **provider(7)**, and **property(7)**.

ENVIRONMENT

OPENSSL

The path to an executable to use to generate hashes and fingerprints (see above).

SSL_CERT_DIR

Colon separated list of directories to operate on. Ignored if directories are listed on the command line.

SEE ALSO

openssl(1), **openssl-crl(1)**, **openssl-x509(1)**

COPYRIGHT

Copyright 2015-2020 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the Apache License 2.0 (the "License"). You may not use this file except in compliance with the License. You can obtain a copy in the file LICENSE in the source distribution or at <https://www.openssl.org/source/license.html>.