Rocky Enterprise Linux 9.2 Manual Pages on command 'update-crypto-policies.8'

$ man update-crypto-policies.8

UPDATE-CRYPTO-POLI(8) UPDATE-CRYPTO-POLI(8)

NAME

update-crypto-policies - manage the policies available to the various cryptographic back-ends.

SYNOPSIS

update-crypto-policies [COMMAND]

DESCRIPTION

update-crypto-policies(8) is used to set the policy applicable for the various cryptographic back-ends, such as SSL/TLS libraries. The policy aims to control the back-end default algorithm selections unless the application user configures them otherwise.

The available policies are described in the crypto-policies(7) manual page.

The desired system policy is selected in /etc/crypto-policies/config and this tool will generate the individual policy requirements for all back-ends that support such configuration. After this tool is called and all the affected applications are restarted, the administrator is assured that any application that utilizes the supported back-ends will

follow the specified policy.

Note that the above assurance does apply to the extent that applications are configured to follow the default policy (the details vary on the back-end, see below for more information).

The generated back-end policies will be placed in /etc/crypto-policies/back-ends. Currently the supported back-ends (and directive scopes they respect) are:

? GnuTLS library (GnuTLS, SSL, TLS)

? OpenSSL library (OpenSSL, SSL, TLS)

? NSS library (NSS, SSL, TLS)

? OpenJDK (java-tls, SSL, TLS)

? Libkrb5 (krb5, kerberos)

? BIND (BIND, DNSSec)

? OpenSSH (OpenSSH, SSH)

? Libreswan (libreswan, IKE, IPSec)

? libssh (libssh, SSH)

Applications and languages which rely on any of these back-ends will follow the system policies as well. Examples are apache httpd, nginx, php, and others.

In general after changing the system crypto policies with the update-crypto-policies --set command it is recommended to restart the system for the effect to fully take place as the policy configuration files are loaded on application start-up. Otherwise applications started before the command was run need to be restarted to load the updated configuration.

COMMANDS

The following commands are available in update-crypto-policies tool.

? --set: Sets the current policy and overwrites the config file.

? --show: Shows the currently applied crypto policy.

? --is-applied: Returns success if the currently configured policy in the config file was applied by running the update-crypto-policies.

? --check: Returns success if freshly generated configuration files match the current policy. The check will return failure if there is

a manual modification of the back-end configuration files or a newer version of crypto-policies package is installed without running update-crypto-policies. This should not happen with system updates because update-crypto-policies is run implicitly there.

OPTIONS

The following options are available in update-crypto-policies tool.

? --no-reload: By default this tool causes some running applications to reload the configured policy. This option skips the reloading.

APPLICATION SUPPORT

Applications in the operating system that provide a default configuration file that includes a cryptographic policy string will be modified gradually to support these policies.

When an application provides a configuration file, the changes needed to utilize the system-wide policy are the following.

? Applications using OpenSSL: If an application allows the configuration of ciphersuite string, the special cipher string "PROFILE=SYSTEM" should replace any other cipher string. Applications which use the default library settings automatically adhere to the policy. Applications following the policy inherit the settings for cipher suite preference. By default the OpenSSL library reads a configuration file when it is initialized. If the application does not override loading of the configuration file, the policy also sets the minimum TLS protocol version and default cipher suite preference via this file. If the application is long-running such as the httpd server it has to be restarted to reload the configuration file after policy is changed. Otherwise the changed policy cannot take effect.

? Applications using GnuTLS: Applications using GnuTLS will load the crypto policies by default. To prevent applications from adhering to the policy the GNUTLS_SYSTEM_PRIORITY_FILE environment variable must be set on an empty file (e.g., /dev/null). The policy covers the settings for cipher suite preference, TLS and DTLS protocol versions, allowed elliptic curves, and limits for cryptographic

keys.

? Applications using NSS: Applications using NSS will load the crypto policies by default. They inherit the settings for cipher suite preference, TLS and DTLS protocol versions, allowed elliptic curves, and limits for cryptographic keys. To prevent applications from adhering to the policy the NSS_IGNORE_SYSTEM_POLICY environment variable must be set to 1 prior to executing that application.

? Applications using Java: No special treatment is required. Applications using Java will load the crypto policies by default. These applications will then inherit the settings for allowed cipher suites, allowed TLS and DTLS protocol versions, allowed elliptic curves, and limits for cryptographic keys. To prevent openjdk applications from adhering to the policy the <java.home>/jre/lib/security/java.security file should be edited to contain security.useSystemPropertiesFile=false or the system property java.security.disableSystemPropertiesFile be set to true. Note that the system property java.security.properties is loaded with a lower preference than the crypto policies, so you can?t use this property to override crypto policies without also preventing openjdk applications from adhering to the policy.

? Applications using libkrb5: No special treatment is required. Applications will follow the crypto policies by default. These applications inherit the settings for the permitted encryption types for tickets as well as the cryptographic key limits for the PKINIT protocol. A system-wide opt-out is available by deleting the /etc/krb5.conf.d/crypto-policies link.

? BIND: This application inherits the set of disabled algorithms. To opt-out from the policy, remove the policy include directive in the named.conf file.

? OpenSSH: Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange algorithms. To opt-out from the policy for client,

override the global ssh_config with a user-specific configuration in ~/.ssh/config. See ssh_config(5) for more information. To override some configuration option in server, use a drop-in directory /etc/ssh/sshd_config.d/ to create a file lexicographically preceding 05-redhat.conf which is currently including crypto policies configuration file.

? Libreswan: Both servers and clients inherit the ESP and IKE preferences, if they are not overridden in the connection configuration file. Note that due to limitations of libreswan, crypto policies is restricted to supporting IKEv2. To opt-out from the policy, comment the line including /etc/crypto-policies/back-ends/libreswan.config from /etc/ipsec.conf.

? Applications using libssh: Both client and server applications using libssh will load the crypto policies by default. They inherit the ciphers, key exchange, message authentication, and signature algorithms preferences.

POLICY CONFIGURATION

One of the supported policies should be set in /etc/crypto-policies/config and this script should be run afterwards. In case of a parsing error no policies will be updated.

CUSTOM POLICIES

The custom policies can take two forms. First form is a full custom policy file which is supported by the update-crypto-policies tool in the same way as the policies shipped along the tool in the package. The second form can be called a subpolicy or policy modifier. This form modifies aspects of any base policy file by removing or adding algorithms or protocols. The subpolicies can be appended on the update-crypto-policies --set command line to the base policy separated by the : character. There can be multiple subpolicies appended. The resulting configuration is the same as if the policy and subpolicies were concatenated together.

Let?s suppose we have a subpolicy ECDHE-ONLY that enforces ECDHE usage

for key exchange, and a subpolicy SHA1 that enables support for several usages of SHA-1 hash function. You can set the DEFAULT policy with ECDHE key exchange and SHA-1 hash function enabled by running the following command:

update-crypto-policies --set DEFAULT:ECDHE-ONLY:SHA1

This command generates and applies configuration that will be a modification of the DEFAULT policy with changes specified in the ECDHE-ONLY and SHA1 subpolicies.

FILES

/etc/crypto-policies/config

The file contains the system policy to be applied when update-crypto-policies is run without any arguments. It should contain a string of one of the policies listed in the crypto-policies(7) page (e.g., DEFAULT) or any custom policy name with subpolicies separated by the : character. The file is overwritten when update-crypto-policies --set is executed.

/etc/crypto-policies/back-ends

Contains the generated policies in separated files, and in a format readable by the supported back ends.

/etc/crypto-policies/local.d

Contains additional files to be appended to the generated policy files. The files present must adhere to $app-XXX.config file naming, where XXX is any arbitrary identifier. For example, to append a line to GnuTLS' generated policy, create a gnutls-extra-line.config file in local.d. This will be appended to the generated gnutls.config during update-crypto-policies. Please note that because the mechanism just appends a line to the back-end configuration the effect varies among the back-ends. For some of the back-ends the override fully replaces the original policy and for other back-ends the override might not be effective at all.

/etc/crypto-policies/state/current

The file contains the current system policy name with eventual subpolicies as of the last execution of the update-crypto-policies

command.

/etc/crypto-policies/state/CURRENT.pol

The file contains the current system policy definition with all the modifications from eventual subpolicies applied and is written when the update-crypto-policies command is executed.

SEE ALSO

crypto-policies(7), fips-mode-setup(8)

AUTHOR

Written by Nikos Mavrogiannopoulos.

update-crypto-policies 12/15/2022 UPDATE-CRYPTO-POLI(8)