NAME

gmautil – General Tcl Utility Routines

SYNOPSIS

(If package installed globally)
package require gmautil

(Otherwise)
source gmautil.tcl

::gmautil::is_git *path*

::gmautil::verify *data signature*

::gmautil::version_compare *v1 v2*

::gmautil::upgrade *destination tmp url file old new strip launch msg proxy curl*

DESCRIPTION

This module provides a set of miscellaneous utility functions that could be useful to more than one part of the overall product suite.

::gmautil::is_git *path*

Returns true if the directory *path* resides inside a Git working directory tree.

::gmautil::verify *data signature*

Given a binary data string *data*, return true if the binary cryptographic signature in *signature* is valid for that data. This uses our built-in product public key.

::gmautil::version_compare *v1 v2*

Compares two version number strings to see which order they should go in, chronologically. Each string must consist of one or more integers (as ASCII digit sequences) separated by dots. The first such number is the most significant (major) version, the next is the next-most significant sub-version within the major one, and so forth. Thus, 1.2 comes before 1.10, 2.2.4 comes before 2.3, etc. Returns 0 if *v1* and *v2* are equal or equivalent, a number <0 if *v1*<*v2* (i.e., *v1* comes before *v2*), or a number >0 if *v1*>*v2*.

::gmautil::upgrade *destination tmp url file old new strip launch msg proxy curl*

This function facilitates automatic upgrades within an application. The application is downloaded using the CURL program whose pathname is given in *curl* (using a proxy server as specified by *proxy* if that value is non-empty). The file itself is obtained from the server directory indicated by the *url* value, with the base filename *file* and suffix **.tar.gz** and **.tar.gz.sig** (the latter being a cryptographic signature verifying the authenticity of the compressed tar file). These files are downloaded to the temporary directory *tmp*.

Once the files are downloaded, the signature is checked, and then the files are extracted into the destination diretory *destination*, although a leading *strip* (possibly prefixed with **./**) is removed from the names as they exist in the tar file.

The tar file is expected to contain a manifest file in its top-level directory, called **__checksums__**. Each line of this file consists of a SHA256 checksum in hex, whitespace, a type character (space for text files, ***** for binary files, **?** for portable-newline files, or ^ for BITS files), then the path of the file. Only binary files are supported at this time. Every file extracted from the archive must have an entry in this manifest file, and the installed file's checksum must match the manifest entry.

Once that is all accomplished, the downloaded files are removed and, if *launch* is non-empty, the program will attempt to start the program *launch* relative to *destination* and then the calling program is terminated.

old and *new* are the current and to-be-installed version numbers. Warnings will be issued before downgrading. The routine will refuse to continue if the versions are equal to each other.

DIAGNOSTICS

An exception is thrown if a serious error is encountered.

SEE ALSO

openssl(1).

AUTHOR

Steve Willoughby / steve@madscience.zone.

HISTORY

Initial version created 17 July 2020.

COPYRGHT

Part of the GMA software suite, copyright © 1992–2021 by Steven L. Willoughby (Software Alchemy), Aloha, Oregon, USA. All Rights Reserved. Distributed under BSD-3-Clause License.