user

package standard library

Version: go1.20.1 Latest | Published: Feb 14, 2023 | License: BSD-3-Clause | Imports: 8 | Imported by: 22,869

Details Valid go.mod file ? Redistributable license ? Tagged version ?

Stable version ?

Learn more

Repository cs.opensource.google/go/go

Links Report a Vulnerability

Documentation

Documentation

Rendered for linux/amd64

Overview

Package user allows user account lookups by name or id.

For most Unix systems, this package has two internal implementations of resolving user and group ids to names, and listing supplementary group IDs. One is written in pure Go and parses /etc/passwd and /etc/group. The other is cgo-based and relies on the standard C library (libc) routines such as getpwuid_r, getgrnam_r, and getgrouplist.

When cgo is available, and the required routines are implemented in libc for a particular platform, cgo-based (libc-backed) code is used. This can be overridden by using osusergo build tag, which enforces the pure Go implementation.

Index

Constants

This section is empty.

Variables

This section is empty.

Functions

This section is empty.

Types

type Group

added in go1.7

```
type Group struct {
	Gid  string // group ID
	Name string // group name
}
```

Group represents a grouping of users.

On POSIX systems Gid contains a decimal number representing the group ID.

func LookupGroup

added in go1.7

```
func LookupGroup(name string) (*Group, error)
```

LookupGroup looks up a group by name. If the group cannot be found, the returned error is of type UnknownGroupError.

func LookupGroupId

added in go1.7

```
func LookupGroupId(gid string) (*Group, error)
```

LookupGroupId looks up a group by groupid. If the group cannot be found, the returned error is of type UnknownGroupIdError.

type UnknownGroupError

added in go1.7

```
type UnknownGroupError string
```

UnknownGroupError is returned by LookupGroup when a group cannot be found.

func (UnknownGroupError) Error

added in go1.7

```
func (e UnknownGroupError) Error() string
```

type UnknownGroupIdError

added in go1.7

```
type UnknownGroupIdError string
```

UnknownGroupIdError is returned by LookupGroupId when a group cannot be found.

func (UnknownGroupIdError) Error

added in go1.7

```
func (e UnknownGroupIdError) Error() string
```

type UnknownUserError

```
type UnknownUserError string
```

UnknownUserError is returned by Lookup when a user cannot be found.

func (UnknownUserError) Error

```
func (e UnknownUserError) Error() string
```

type UnknownUserIdError

```
type UnknownUserIdError int
```

UnknownUserIdError is returned by LookupId when a user cannot be found.

func (UnknownUserIdError) Error

```
func (e UnknownUserIdError) Error() string
```

type User

```
type User struct {
	// Uid is the user ID.
	// On POSIX systems, this is a decimal number representing the uid.
	// On Windows, this is a security identifier (SID) in a string format.
	// On Plan 9, this is the contents of /dev/user.
	Uid string
	// Gid is the primary group ID.
	// On POSIX systems, this is a decimal number representing the gid.
	// On Windows, this is a SID in a string format.
	// On Plan 9, this is the contents of /dev/user.
```

```
    Gid string
    // Username is the login name.
    Username string
    // Name is the user's real or display name.
    // It might be blank.
    // On POSIX systems, this is the first (or only) entry in the GECOS field
    // list.
    // On Windows, this is the user's display name.
    // On Plan 9, this is the contents of /dev/user.
    Name string
    // HomeDir is the path to the user's home directory (if they have one).
    HomeDir string
}
```

User represents a user account.

func Current

```
func Current() (*User, error)
```

Current returns the current user.

The first call will cache the current user information. Subsequent calls will return the cached value and will not reflect changes to the current user.

func Lookup

```
func Lookup(username string) (*User, error)
```

Lookup looks up a user by username. If the user cannot be found, the returned error is of type UnknownUserError.

func LookupId

```
func LookupId(uid string) (*User, error)
```

LookupId looks up a user by userid. If the user cannot be found, the returned error is of type UnknownUserIdError.

func (*User) GroupIds

added in go1.7

```
func (u *User) GroupIds() ([]string, error)
```

GroupIds returns the list of group IDs that the user is a member of.

Source Files

View all

cgo_listgroups_unix.go cgo_lookup_cgo.go cgo_lookup_unix.go getgrouplist_unix.go lookup.go user.go

Google