



**NAME**

`munge_enum_is_valid`, `munge_enum_int_to_str`, `munge_enum_str_to_int` – MUNGE enumeration functions

**SYNOPSIS**

```
#include <munge.h>
```

```
int munge_enum_is_valid (munge_enum_t type, int val);
```

```
const char * munge_enum_int_to_str (munge_enum_t type, int val);
```

```
int munge_enum_str_to_int (munge_enum_t type, const char *str);
```

```
cc ... -lmunge
```

**DESCRIPTION**

The `munge_enum_is_valid()` function checks if the given value *val* is a valid MUNGE enumeration of the specified type *type* in the software configuration as currently compiled. Some enumerations correspond to options that can only be enabled at compile-time.

The `munge_enum_int_to_str()` function converts the MUNGE enumeration *val* of the specified type *type* into a text string.

The `munge_enum_str_to_int()` function converts the NUL-terminated case-insensitive string *str* into the corresponding MUNGE enumeration of the specified type *type*.

**RETURN VALUE**

The `munge_enum_is_valid()` function returns non-zero if the given value *val* is a valid enumeration.

The `munge_enum_int_to_str()` function returns a NUL-terminated constant text string, or NULL on error; this string should not be freed or modified by the caller.

The `munge_enum_str_to_int()` function returns a MUNGE enumeration on success (i.e.,  $\geq 0$ ), or  $-1$  on error.

**ENUM TYPES**

The following enumeration types can be specified.

**MUNGE\_ENUM\_CIPHER**

Specify enumerations for the available cipher types.

**MUNGE\_ENUM\_MAC**

Specify enumerations for the available MAC types.

**MUNGE\_ENUM\_ZIP**

Specify enumerations for the available compression types.

**ERRORS**

Refer to `munge(3)` for a complete list of errors.

**EXAMPLE**

The following example program illustrates how a list of available cipher types can be queried.

```
#include <stdio.h>           /* for printf() */
#include <stdlib.h>          /* for exit() */
#include <munge.h>

int
main (int argc, char *argv[])
{
    int    i;
```



```
const char *p;
mungenum_t t = MUNGE_ENUM_CIPHER;

for (i = 0; (p = munge_enum_int_to_str (t, i)) != NULL; i++) {
    if (mungenum_is_valid (t, i)) {
        printf ("%2d = %s\n", i, p);
    }
}
exit (0);
}
```

## AUTHOR

Chris Dunlap <cdunlap AT llnl DOT gov>

## COPYRIGHT

Copyright (C) 2007-2011 Lawrence Livermore National Security, LLC.

Copyright (C) 2002-2007 The Regents of the University of California.

MUNGE is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

Additionally for the MUNGE library (libmunge), you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

## SEE ALSO

**munge(1)**, **remunge(1)**, **unmunge(1)**, **munge(3)**, **munge\_ctx(3)**, **munge(7)**, **munged(8)**.

**<http://munge.googlecode.com/>**

