Alcune considerazioni ambiente

sabato 1 aprile 2023 10:32

# Variabili d'ambiente - alcuni dettagli

**Quando un'applicazione è attiva, possiamo cambiarle le variabili ambientali?**

| | |
|---|---|
| PWD | direttorio di lavoro corrente |
| HOME | directory principale d'utente |
| PATH | specifica di ricerca di eseguibili |
| DISPLAY | specifica di host dove reindirizzare l'output grafico |
| LOGNAME | login dell'utente |

**Esse sono all'interno dell'address space, in particolare nella stack area in basso, in una zona non utilizzata realmente per le operazioni di stack, e quella è READ/WRITE, quindi se arriviamo in quella zona con dei pointer possiamo fare quello che vogliamo.**

**Abbiamo una serie di funzioni dello standard di sistema che ci permettono di interagire/ modificare queste variabili ambientali.**

## GETENV()

char *getenv(char *name)

| | |
|---|---|
| **Descrizione** | preleva il valore di una variabile d'ambiente |
| **Parametri** | *name indica il nome della variabile d'ambiente |
| **Restituzione** | NULL oppure la stringa che definisce il valore della variabile |

**Getenv ci restituisce, dato il nome di una variabile d'ambiente, il puntatore alla variabile effettiva.**

**Data una variabile d'ambiente che si chiama Process Working Directory (PWD), o INPUT_METHOD, mi dai per favore il puntatore a questa stringa nell'ambiente?**

**"La funzione getenv() ottiene il valore corrente della variabile d'ambiente, name. L'applicazione non deve modificare la stringa puntata dalla funzione getenv().**

**La funzione getenv() restituisce il valore della variabile di ambiente come stringa con terminazione NUL. Se il nome della variabile non è nell'ambiente corrente, viene restituito NULL"**

**-Manuale ufficiale - June 20, 2007**

## PUTENV()

int putenv(char *string)

| | |
|---|---|
| **Descrizione** | setta il valore di una variabile d'ambiente |
| **Parametri** | *string identifica il nome della variabile d'ambiente + valore da assegnare (nella forma "nome=valore") |
| **Restituzione** | 0 in caso di successo – valore diverso da zero in caso di fallimento |

**"La funzione putenv() accetta un argomento nella forma ``nome=valore'' ed è equivalente a: setenv(nome, valore, 1);**
**La stringa puntata da string diventa parte dell'ambiente. Un programma non dovrebbe alterare o liberare la stringa e non dovrebbe usare stack o altre variabili di stringa transitorie come argomenti per putenv(). La funzione setenv() è fortemente preferita a putenv().**

**Le funzioni setenv(), putenv() e unsetenv() restituiscono il valore 0 in caso di esito positivo; in caso contrario, viene restituito il valore -1 e viene impostata la variabile globale errno per indicare l'errore."**

**-Manuale ufficiale - June 20, 2007**

1) se la variabile d'ambiente non esiste viene anche creata

2) la congiunzione di valori avviene attraverso il carattere ':'

ES. PATH=/user/local/bin:/bin/:/home/quaglia/bin

# SETENV()

int setenv(char *name, char *value, int overwrite)

**Descrizione** crea una variabile d'ambiente e setta il suo valore

**Parametri**
1) *name: nome della variabile d'ambiente
2) *value: valore da assegnare
3) overwrite: flag di sovrascrittura in caso la variabile esista

**Restituzione** 0 in caso di successo, -1 in caso di fallimento

**"La funzione setenv() inserisce o reimposta il nome della variabile di ambiente nell'elenco di ambienti corrente. Se il nome della variabile non esiste nell'elenco, viene inserito con il valore dato. Se la variabile esiste, viene verificato l'overwrite dell'argomento; se overwrite è zero, la variabile non viene resettata, altrimenti viene resettata al valore dato."**

**-Manuale ufficiale - June 20, 2007**

**Ti do il nome di una variabile d'ambiente, ti dico quale deve essere il nuovo valore che quella variabile deve avere: questo è il secondo parametro.**
**Se non esiste la crei, se esiste farai un lavoro che dipende dal terzo parametro overwrite.**

# UNSETENV()

int unsetenv(char *name)

**Descrizione** elimina una variabile d'ambiente

**Parametri** *name identifica il nome della variabile d'ambiente

**Restituzione** 0 in caso di successo, -1 in caso di fallimento