libreria legittima con una libreria dannosa. Ciò può comportare l'esecuzione di comandi (e payload) dannosi.

La vulnerabilità si esplicita in due forme distinte: l'aggressore controlla l'indirizzo della libreria all'interno del programma, oppure controlla l'ambiente e quindi la libreira puntata dal programma.

Come difendersi

Oltre al consueto principio dei minimi privilegi e il controllo dell'input, qui occorre verificare sempre l'attendibilità delle librerie importate.
L'applicazione non deve caricare librerie non necessarie o delle quali può fare a meno.
Invece dei path relativi, l'applicazione deve utilizzare path assoluti per individuare il percorso delle librerie da caricare.

Esempio:
Nel seguente codice la libreria viene caricata a partire da un indirizzo scritto nel registry. Chiunque acceda al registry può sostituirlo con l'indirizzo di una copia manipolata della libreria medesima.

```
RegQueryValueEx(hkey, "APP_HOME_DIR", 0, 0, (BYTE*)appHomeDir, &size);
char* libreria=(char*)malloc(strlen(appHomeDir)+strlen(INITLIB));
if (libreria) {
  strcpy(libreria, appHomeDir);
  strcat(libreria,INITCMD);
  LoadLibrary(libreria);
}
```

Se si utilizza un percorso assoluto, la libreria viene prelevata da un percorso più difficilmente manipolabile.
Utilizzare la System.load() in luogo della System.loadLibrary(), perché più sicura.

Per ulteriori informazioni si veda: http://cwe.mitre.org/data/definitions/114.html,
CWE-114: Process Control.

7.1.8 Ulteriori indicazioni per lo sviluppo sicuro

La raccolta di Best Practices che segue, è conforme ai dettami degli standard CERT C / C++ Programming Language Secure Coding.

7.1.8.1 Dichiarazioni

- È consigliato dimensionare gli array non utilizzando costanti numeriche ma piuttosto costanti simboliche definite
  Esempio:
  Forma non corretta:
  ```
  int mesi[13];
  ```
  Forma corretta:
  ```
  int mesi[TOT_MESI + 1];
  ```

- Dichiarare le costanti utilizzando la keyword "const"
  Esempio:
  Forma non corretta:
  ```
  int mesi = 12;
  ```
  Forma corretta:
  ```
  const unsigned int mesi = 12;
  ```
- Dichiarare le variabili che possono avere valori positivi utilizzando la keyword "unsigned"
- Il tipo "char" deve essere unsigned
- Non utilizzare float e double quando non è necessario (calcoli scientifici)
- Le classi che hanno funzioni virtuali devono sempre avere distruttori virtuali