- ritorno a carrello (\x0a);
- new Line (\x0c);
- NULL byte (\x00);
- altri.

Esempio:
Esempio di script vulnerabile a Shell Execution Command:

Contromisure

Scrivere il codice in modo che non venga eseguita nessuna shell dei comandi.
È deprecata l'invocazione diretta dei comandi di sistema, soprattutto se utilizza l'input utente. Per accedere alle funzioni del sistema operativo, è obbligatorio utilizzare le API messe a disposizione dalle librerie dei vari linguaggi di programmazione.
Se dovessero permanere nel sorgente delle shell dipendenti dall'input dell'utente, occorre allora validare l'input, filtrando parole e caratteri potenzialmente dannosi. Meglio ancora se si verifica preventivamente l'input dell'utente confrontandolo con una white list di valori ammessi.

6.1.2 File Inclusion

Le problematiche di File Inclusion sono solitamente riscontrabili nelle applicazioni web. Si sono diffuse negli ultimi anni con il boom dei linguaggi e delle tecnologie di scripting (ASP, PHP, Python, Perl, etc..) e si manifestano quando i parametri passati ad uno script vulnerabile non vengono opportunamente verificati prima di essere utilizzati per includere dei file in determinati punti di un portale.
Le problematiche di File Inclusion si distinguono solitamente in due categorie:

- **Local File Inclusion**: si manifestano quando un aggressore passa, come parametri di uno script vulnerabile, dei file residenti localmente nel sistema. Il loro contenuto viene così visualizzato a video nell'esatto punto del portale in cui si verifica l'inclusione. Un aggressore può in questo modo ottenere gli hash delle password di sistema o accedere ad informazioni riservate collocate all'esterno della document root del Web Server. Le problematiche di Local File Inclusion possono anche essere sfruttate per eseguire comandi remoti se l'aggressore ha la possibilità di collocare localmente un file contenente codice malevolo, che può essere puntato dallo script vulnerabile. Il file può essere trasmesso utilizzando i classici servizi di rete (ftp, ssh, cifs, etc..) o usufruendo di una qualsiasi procedura di upload richiamabile da Web
- **Remote File Inclusion**: è la più pericolosa perché permette a un aggressore di passare, come parametri di uno script vulnerabile, un file che risiede in un altro web server (ad esempio da egli stesso controllato). L'aggressore può collocare all'interno di questo file del codice di scripting (ad esempio codice PHP malevolo) per eseguire comandi remoti sul sistema.

Esempio:
Un URL costruito come segue:

```
http://vulnerable_host/preview.php?file=example.html
```

Può essere modificato come segue, per visualizzare, ad esempio, un file locale dal contenuto sensibile:

```
http://vulnerable_host/preview.php?file=../../../../etc/passwd
```

Contromisure

Occorre evitare di utilizzare file esterni il cui contenuto sia di difficile verifica. Nel caso in cui non se ne possa fare a meno, occorre predisporre una white list di file ammessi. Solo tali file saranno selezionabili da parte dell'utente, per esempio tramite un indice numerico. Tale approccio è molto facile da mettere in