

Il Cross Site Scripting può essere reflected o stored. Nel primo caso, uno script inoculato è valido solo all'interno della sessione corrente, ma i suoi effetti possono essere molto dannosi per il sito vittima dell'attacco.

Ancora più grave è il Cross Site Scripting di tipo stored. In questo caso lo script inoculato viene memorizzato come parte integrante della pagina all'interno di un database e ripristinato ogni qual volta la pagina in questione viene caricata. Quest'attacco è stato sfruttato ampiamente nel recente passato, soprattutto laddove veniva consentito agli utenti di inserire recensioni, commenti e altri contributi.

Volendo schematizzare, possiamo categorizzare gli attacchi XSS nelle seguenti tipologie:

- Reflected XSS, in cui la stringa dannosa proviene dalla richiesta dell'utente.
- Stored XSS, in cui la stringa dannosa proviene dal database del sito Web.

Esiste anche un Cross Site Scripting dovuto a una lacuna nella codifica UTF-7, che permettere di mascherare i caratteri "<" e ">", facendoli sfuggire al controllo. Questa minaccia non è più possibile nei moderni browser, ad eccezione di Microsoft Internet Explorer 11.

### **Come difendersi**

- Validare tutti gli input, indipendentemente dalla loro provenienza. Per la validazione, si consiglia l'approccio white list (sono accettati solo i dati che adottano una struttura specificata nella white list, scartando quelli che non la rispettano). Occorre controllare il tipo del dato, la sua dimensione, l'intervallo di validità (range), il formato ed eventuali valori attesi (white list).
- Codificare completamente tutti i dati dinamici prima di incorporarli (encoding). La codifica dovrebbe essere sensibile al contesto, in base al tipo di dato che si vuole neutralizzare: se ci si aspetta che possa esserci codice HTML abusivo, occorre codificare gli eventuali tag HTML, se ci si potrebbe trovare di fronte a uno script, allora bisogna codificare gli elementi sintattici di Javascript, ecc.
- Si consiglia di utilizzare la libreria di codifica ESAPI.
- Nell'intestazione di risposta Content-Type HTTP, è necessario definire esplicitamente la codifica dei caratteri (charset) per l'intera pagina.
- Impostare l'attributo HTTPOnly per proteggere il cookie della sessione da indebite letture da parte di script malevoli.

Per ulteriori informazioni si veda: <http://cwe.mitre.org/data/definitions/79.html>.

CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting').

### **Esempio:**

Nel seguente codice, un dato in input (parametro per la ricerca viene stampato a video, senza averlo prima sottoposto ad alcun controllo):

```
Sub cmdSearch _Click(Source As Object, _ e As EventArgs)

    // Do Search....

    lblResult.Text="You Searched for: " & txtInput.Text

    // [loop per mostrare I risultati della query]

End Sub
```

## **7.8.2 Code Injection**

### **Come riconoscerla**