**Surviving the Web: A Journey into Web Session Security**

*A web session is a sequence of network HTTP request and response belonging to a particular user. Hence, it is very important for an application to be secure in order to maintain data integrity and confidentiality of a user.*

Below is the overview of the research, the authors have made regarding critical web session attacks and the solutions proposed to mitigate and prevent them.

**Type of Attackers classified based on Threat Model**: Web and Network Attackers.

**Web Session Security Attacks**:

* Web Attacks - Content Injection, Cross-Site Request Forgery (CSRF), Login CSRF, Cookie Forcing, Session Fixation.
* Network Attacks – Eavesdropping sensitive information over HTTP or a mixture of HTTP/ HTTPS(S) network.

**Mitigation Techniques:** Http Only Cookies, Session Shield and Zan, Request Filtering Approaches

**Preventive Techniques:**

* Client-side Filtering, Server-side Filtering, BEEP (Browser Enforced Embedded Policies), Blueprint, Noncespaces, Document Structure Integrity, Critical Evaluation.
* CSRF Preventive techniques- Allowed Referrer List, Tokenization, NoForge, Origin Checking.
* Cookie Forcing and Session Fixation Preventive Techniques – Serene, Origin Cookies, Authentication Cookies Renewal.
* Network Attacks - HTTPS with secure cookies, HProxy, HTTP Strict Transport Security, HTTPS Everywhere.
* Defense Against Multiple Attacks - Origin-Bound Certificates, Browser based Information Flow Control, Browser based Information Flow Control, Security Policies for JavaScript, Ajax Intrusion Detection System, Escudo, CookiExt, SessInt, Same Origin Mutual Approval, App-Isolation.

**Guidelines formulated by the authors to be followed while developing a secured website:**

* Transparency – A combination of usability and compatibility is the most important factor to be considered for large scale deployment of any web defensive solution.
* Security by Design – Backward compatibility should be considered while developing new websites and also developers should be provided with tools and methodologies that allow them to take security into account from the first phases of development.
* Ease of Adoption – The security solutions implemented should be easily understood and adopted by all the web developers.
* Formal Specification and Verification – This step should be considered from the initial stages of development so that all possible attacks violating the security properties can be detected which could be missed during the design phase. Underlying assumptions can also be listed to get to the best solution by comparing them and helps understand the security problem in a better by focusing on each of the security property instead of the mechanism involved.
* Defensive Solutions should be Declarative - Web developers should be given access to an appropriate policy specification language, but the enforcement of the policy should not be their concern.

**Conclusion:** The authors have made a thorough research about existing web security solutions (against several web session attacks that violate security properties at different application layers) and their impact on compatibility, usability and ease of deployment. They have also synthesized a list of guidelines which would be useful for developing a secure web application in a systematic and an innovative way.

**ACM Reference Information:**

Surviving the Web: A Journey into Web Session Security by [Stefano Calzavara](https://dl.acm.org/author_page.cfm?id=81479661055&coll=DL&dl=ACM&trk=0), [Riccardo Focardi](https://dl.acm.org/author_page.cfm?id=81100634484&coll=DL&dl=ACM&trk=0), [Marco Squarcina](https://dl.acm.org/author_page.cfm?id=81508706813&coll=DL&dl=ACM&trk=0), [Mauro Tempesta](https://dl.acm.org/author_page.cfm?id=99658671563&coll=DL&dl=ACM&trk=0) – April 2017.

Last viewed on Internet at: [**https://dl.acm.org/citation.cfm?id=3038923**](https://dl.acm.org/citation.cfm?id=3038923)