HARVARD
UNIVERSITY

Enterprise Architecture
and
Information Security

Advisory

Use of HTTP Cookies

Authors: Greg Charest, Raoul Sevier	**Audience Level:** • Solution Architect and Program Manager • IT Development Manager and Operations Manager • Application Developer and Designer
Version: 1.0 **Last Revised:** 01-Nov-2019 **Status:** draft **Document Type:** Single Topic Guidance	**Distribution Scope:** Harvard-wide
Workgroup Members:	**Reviewers:** Mahbub Rahman Jefferson Burson Mike Thomas Sandy Silk

1. Executive Summary:

The use of http cookies in web sites and applications can cause operational and security issues. A cookie's Domain directive should **NOT** be set to harvard.edu.

2. Problem Statement:

An **HTTP cookie** is a small piece of data sent from a website and stored on the user's computer, usually in clear text, by the user's browser. Cookies can cause operational and security incidents if not configured correctly. Inappropriate use of the 'Domain' cookie directive in particular has caused problems with Harvard applications in the past and represents a potential system vulnerabilitiy. Cookies need to be created, used, and modified carefully.

3. Discussion

3.1. Background

Cookies are based on a number of Internet RFC documents dating back to 1994. They are typically used for:

- Session management
- Personalization
- Tracking

While cookies perform essential functions, it is important to remember that:

- They are sent to the server with every request as long as the URL requested is within the same domain and path defined in the cookie
- They are inherently insecure. Information should be stored in cookies with the understanding that all cookie values will be visible to and can be changed by the end-user and client

Cookies are sent to a web browser in the response header and can include a number of directives in the form of name-value pairs. These include:

- Max-Age=<number> *Optional*
- Domain=<domain-value> *Optional*
- Path=<path-value> *Optional*
- Secure *Optional*
- HttpOnly *Optional*
- SameSite=<samesite-value> *Optional*
  - Strict
  - Lax

The Domain and Path cookie directives define to which URLs the cookies should be sent (the scope) and are of the primary focus of this advisory.

3.2. Cookie Scope

The **Domain directive** specifies the hosts that are allowed to receive the cookie. Browsers vary in implementing the RFC requirements, but in general:

1) The *origin domain* of a cookie is the domain of the originating request.
2) If a cookie's domain attribute is not set, the cookie is only applicable to its origin domain.
3) If a cookie's domain attribute is set,
    a) the cookie is applicable to that domain and all of its subdomains;
    b) the cookie's domain must be the same as, or a parent of, the origin domain
i) the cookie's domain must not be a top-level domain (TLD) or a public suffix.

c) Modern browsers respect the newer specification RFC 6265, and will ignore any leading dot on a domain name.

3.3. Risk Associated with the Domain Directive

3.3.1. Performance and Privacy

Setting the domain attribute means that the cookie is transmitted within the header to that domain and all its subdomains in every HTTP request. A cookie set at the harvard.edu level will be sent in response to requests from any harvard.edu subdomain server. Large numbers of cookies may exceed browser limits and/or result in long page loading times. Moreover, cookie data should be assumed to be relevant to a specific site or application and should not be sent elsewhere without reason.

If you set a cookie's domain to harvard.edu, then you are sharing your cookie with student-run websites and with websites managed by various service providers employed by Harvard.

3.3.2. Name Collisions

Sending a cookie to multiple subdomains may lead to unexpected behavior in the event of a name conflict. In at least one case at Harvard, a cookie set at the harvard.edu level by a server in a subdomain caused unwanted session terminations because the cookie name was meaningful to an unrelated application.

Using default names for session cookies (like JSESSIONID, PHPSESSID, etc.) and setting the domain to harvard.edu is dangerous. Modern browsers will deliver multiple cookies with the same name, and browsers are not required to order them so the one with the most specific domain comes first. Another site may get your session cookie when they were expecting their own.

3.3.3. Escalation of Privileges

When related subdomain-based applications share authentication services, unintended domain-level cookies may lead to privilege escalation.

3.3.4. Session Hijacking

Setting the Domain attribute to a too permissive value, such as harvard.edu, may allow an attacker to launch attacks on the session IDs between different hosts and web applications belonging to the same domain.

4. Recommendations

4.1. Domain Directive Recommendations

- Remove the domain attribute completely to limit cookies to the origin host only
- If you must set the Domain attribute, **DO NOT** set it to harvard.edu
- Add the appropriate path attribute if multiple applications are deployed on the same host

4.2. General Recommendations

- Always use the https protocol
- Never store sensitive data in unencrypted cookies
- Always set the Secure and HTTP Only directives unless you have specific requirements for not doing so
- Consider setting the SameSite directive to prevent cookies being sent with cross-site requests (where site is defined by the registrable domain), which provides some protection against cross-site request forgery attacks (CSRF)
- Don't rely solely on behavior across all browsers and versions for proper clean-up of session cookies during a given browsing session. Authenticated sessions should be expired on the server-side periodically
- Consider more modern APIs for client storage such as Web Storage

5. References

- Mozilla MDN Web Docs - https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
- Open Web Application Security Project https://www.owasp.org/images/a/a0/OWASPLondon20171130_Cookie_Security_Myths_Misconceptions_David_Johansson.pdf