# CWE Detail – CWE-1004

## Description

The product uses a cookie to store sensitive information, but the cookie is not marked with the HttpOnly flag.

## Extended Description

The HttpOnly flag directs compatible browsers to prevent client-side script from accessing cookies. Including the HttpOnly flag in the Set-Cookie HTTP response header helps mitigate the risk associated with Cross-Site Scripting (XSS) where an attacker's script code might attempt to read the contents of a cookie and exfiltrate information obtained. When set, browsers that support the flag will not reveal the contents of the cookie to a third party via client-side script executed via XSS.

## Threat-Mapped Scoring

Score: 3.0

Priority: P2 - Serious (High)

## Observed Examples (CVEs)

**•** CVE-2022-24045: Web application for a room automation system has client-side Javascript that sets a sensitive cookie without the HTTPOnly security attribute, allowing the cookie to be accessed.

**•** CVE-2014-3852: CMS written in Python does not include the HTTPOnly flag in a Set-Cookie header, allowing remote attackers to obtain potentially sensitive information via script access to this cookie.

**•** CVE-2015-4138: Appliance for managing encrypted communications does not use HttpOnly flag.

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Read Application Data — Notes: If the HttpOnly flag is not set, then sensitive information stored in the cookie may be exposed to unintended parties.

**•** Impact: Gain Privileges or Assume Identity — Notes: If the cookie in question is an authentication cookie, then not setting the HttpOnly flag may allow an adversary to steal authentication data (e.g., a session ID) and assume the identity of the user.

## Potential Mitigations

**•** Implementation: Leverage the HttpOnly flag when setting a sensitive cookie in a response. (Effectiveness: High)

## Applicable Platforms

**•** None (Class: Not Language-Specific, Prevalence: Undetermined)

## Demonstrative Examples

**•** The snippet of code below establishes a new cookie to hold the sessionID.