Whitepaper

SecurEyes
Infusing Security

Proxy Cache as a Security Threat

The Cached Logout Page

Anant Kochhar, SecurEyes, December 2008

TABLE OF CONTENTS

Abstract

Improper implementation of cache control directives for an authentication based web application can have serious consequences for the application users. This paper aims to demonstrate this in a typical web application design implementation.

Introduction

Web Cache is a memory allocation on browsers and web proxies where 'representations', i.e., web pages, images, and other files, rendered to users over HTTP are stored. Caching is done to enhance the user experience by reducing the time taken to render web content. In the case of network web proxies, caching also allows organizations to significantly reduce their bandwidth usage and costs.

Though there are several security concerns regarding caching, typically the cache is associated with the Information Disclosure vulnerability. In this attack, an attacker with access to a victim's computer, can view internal restricted pages previously accessed by the victim. This risk has a Low security threat impact for affected organizations since the attacker can only 'view' pages, and not ‘access’ them. As a result, it is common for web application owners and developers to not take cache directives very seriously.

A recently observed behavior, involving a typical web caching proxy and a typical web application design implementation, has revealed that ignoring HTTP cache directives in a web application can have deadly serious consequences for an organization: ***What Happens When The Logout Page Is Cached in the Proxy Cache?***

A Likely Scenario

A Typical Web Application Design:

A web application enforces strong web proxy cache directives for restricted pages behind authentication using the 'Cache-Control: Private' directive in their respective HTTP headers.

The Typical Logout Page

The 'Logout' page is the page which appears when a user clicks on the 'Logout' button. It usually has a message informing the user that he has successfully logged out of the application. Since this page is neither internal nor restricted, the application developer usually enforce weak cache directive in its HTTP header, allowing web caches to store this page.

Are Users Behind a Web Proxy Really Logging Out Of The Application?

Consider the instance when a user (user A) behind a web proxy logs in and logs out of the application and immediately after, another user (user B) logs into the application and logs out as well. Is user B really logged out of the application?

Logout Page- Server Side edition:

On the server side, the 'Logout' page is where the authenticated session is terminated for a user, i.e., after this page has been successfully processed by the server, a user's session is invalidated and he cannot access internal pages without logging in again.

The Exploit

Subsequent Users...

As you may have already conjectured, if this page is being served from the cache and not from the server, its purpose has been defeated! All users, except the first user who logged in from behind the proxy, are being fooled into believing that they have been successfully logged out of the application.

Malicious User C...

A slightly tech savvy user can easily discover and exploit this vulnerability to use valid sessions of other users of the application. He may use some social engineering

to trick users into letting him use their computers, after they have logged out from the application.

Recommended Resolution

According to the HTTP standards, cache-control directives apply across all caching devices and, unless explicitly prohibited by a cache-control directive, a caching device is allowed to store a page response as a cache entry. The response header sent from the server must set cache control directives to ensure that important pages are not cached on any caching device. The directives to be set are

(i) **Cache-control: no-cache** (ii) **Cache-control: no-store**

About The Author

Anant Kochhar is a senior IT Security Consultant at SecurEyes. He has led many application security projects. He can be reached at anant.kochhar@secureyes.net.

About SecurEyes

SecurEyes is a Bangalore, India, based firm specializing in IT security. SecurEyes offers a wide range of security services and products to its clients. For more information, please visit our website: http://www.secureyes.net/.