Distributed Management of Online Data

JC Cannon, Microsoft Online Services Division, jccannon@microsoft.com

Abstract

Online tracking has become an enormous concern for consumers, advocacy groups and regulators. Each is seeking ways to understand and manage it from their perspective. On the other hand, advertisers, publishers and service providers, that makeup the online business ecosystem, are looking for ways to understand consumer intent and provide personalized services in ways that generate revenue, but do not infringe on privacy. Using cookies or browser plug-ins suffer from the multi-multi issue. The multi-multi problem makes persisting and protecting user preferences difficult. At any point in time a user could be using multiple computers, multiple browsers and multiple identity services. There could also be multiple users on a single computer. And how should kiosks be managed? Forcing consumers to log onto the Internet is too heavy weight and privacy invasive. A personal, portable, private store is needed.

Introduction

When Sally goes to website.com it may ask for demographic data to access the site. This can help with personalization based on location, gender or age range. The site may drop a cookie in the browser in order to track Sally's browsing habits and understand her intent. While at work, Sally may enter some demographic data for a site she visits because she likes the personalization. She may opt-out of behavioral advertising at the NAI site. Now that her preferences are set she feels she is ready to go. Now she goes home and goes to the same sites, but her preferences aren't there. Even after she sets them, they could be changed by her daughter. Back at work where she clears cookies once a week to minimize tracking, she also removes her opt-out cookies that kept advertisers from tracking her. If she brings up a different browser she notices that her preferences for some reason aren't there. Managing preferences is so difficult for Sally that she decides to just set the intermediate privacy and security level for each browser and hope for the best.

Current State

Persistence and synchronization of preferences is not a new problem. Consumers and companies have had to deal with this for many years now. Unfortunately there is still not a solution that works for well. Let's look at some of these solutions and their weaknesses.

Cookie Synchronization

One of the simplest ideas for preference persistence is to merely copy cookies to each environment. There are several applications that will let users synchronize their cookie files with different computers. This permits Sally to set her preferences once and have the same settings on all the computers she controls. She could even copy them across browser environments so no matter which browser she was using, Internet Explorer, Chrome or Firefox, the settings

would be the same. Depending on the program she was using she might have a default set of cookies she wants to maintain and have all other cookies purged on a regular basis.

The weakness in this solution is in finding an application that only persists the cookies you want to persist and not have the values overwritten. There is also the problem of managing multiple user computers and kiosk scenarios that isn't resolved. Sally's cookie values could always be overwritten by another user or her synchronization utility may not exist on the computer she is using.

BHOs and LSOs

Browser Help Objects (BHOs) are browser plug-ins or toolbars. They are able to persist settings or possibly synchronize values. They would either be used to manage cookies or store data themselves for later access by websites. Locally Stored Objects (LSOs) are browser-based storage such as Flash, Silverlight or HTML 5 storage. They permit the local storage of settings in a persistent fashion, but don't provide much flexibility or synchronization capabilities. Their storage is not easy to manage for most consumers. In fact they can circumvent a user's desires by recreating cookies that a user doesn't want in her browser.

BHOs and LSOs have the basic flaw of availability. They may not exist for the version of the browser being used. If they do exist they have to be installed on each machine being used. This can be difficult in the multi-user or kiosk scenario where the BHO or LSO does not exist. These mechanisms can also have a negative by-product where they are able to capture a user's browsing habits and share them with the manufacturer or user of the BHO or LSO.

Cloud Identities

One of the best ways to persist and protect one's preferences is to use an online identity to store them in the cloud. Facebook, Google, Microsoft and Yahoo provide the ability for users to create an identity and associate storage and preferences with the identity. The identities can also be linked to other sites so that the preferences can be shared amongst multiple sites.

The first weakness in this strategy is it takes away a user's anonymity. The website integrating the identity service knows specifically who the visitor is to the website. The websites also typically have direct access to all of the user's data as well as browsing history. This strategy also has the limitation in that a chosen identity service may not work at all the sites that one visits.

Scoping the Problem

So let's look at what is trying to be accomplished:

1. Make settings accessible across multiple computers and browsers
2. Protect my settings from modification by other computer users
3. Let the user control which settings a website can access and when
4. Don't permit the uncontrolled association of the settings with an identity

Portable Private Storage

To address the parameters above let's start with a personal storage area for storing Sally's settings. Sally can control access to the storage via an ID and password. Inside the storage she could store settings and personal attributes and even categorize them. She could have demographic, clothing, entertainment, travel and many other categories of settings stored over time after accessing multiple websites. The storage can be synchronized between computers, copied to the cloud or placed on removable storage.

Access to the data within the device could be controlled by APIs so a website couldn't just read the data without permission. A website wishing to retrieve data from the storage device would use an API to request the value or category of values desired and the purpose for which the request was being made.

A Simple Usage Scenario

Installation

Sally decides that she wants to use this portable private storage service she's heard about. She installs it on her home computer and during setup it defaults all cookies to session cookies to prevent cross-session identification. It also obfuscates the browser and system configuration data to mitigate the risk of fingerprinting. Now she browses the web as usual. Alternatively, Sally could install the service on a removable storage device and run the service from that device as needed on other computers.

Usage

When arriving at a website that takes advantage of the new service, Sally gets prompted for the ability to read and write to her storage. Most sites want to read her demographic data in exchange for access to their content. The request appears as an extension to the browser that does not interrupt Sally's access to the site, though the site may block content until Sally provides the data. If it is the first time the request was made, Sally can enter the requested data and indicate that the site may access it or all sites may read her demographic data in return for providing access to content. She may also permit the site to store content and product categories that are retrievable by other sites with permission.

Portability

An encrypted copy of Sally's data could be kept in the cloud for synchronization with her other computers. Sally could also export her data in encrypted fashion to a removable storage device and import it to other devices. For kiosks, Sally could run the service from removable storage. The service would interact with websites via web services and have access to her encrypted data.

Conclusion

Consumers can no longer trust their online data or preferences to cookies, BHOs and proprietary identity mechanisms. Consumers should have a flexible, portable and private mechanism for sharing their online preferences and data as needed. This mechanism should be

agnostic of the type of operating system, browser or web identity that the consumer wishes to use.