Polaris, A Distributed Online Social Network

Christo Wilson, Troy Steinbauer, Gang Wang, Alessandra Sala, Haitao Zheng and Ben Y. Zhao
CS Department, University of California at Santa Barbara, Santa Barbara, CA 93106-9560
Tel: 805-229-1169 E-mail: troysteinbauer@cs.ucsb.edu

Abstract: Polaris is distributed Online Social Network (OSN) architecture that allows users to make their own trade-off between privacy and cost, while balancing the economic needs of OSN providers. The user's smart phone acts as the identity provider and access control manager for external, commoditized OSN providers who provide specific social network functionality (e.g. status update, photo sharing) with fine-grained privacy control.

Online Social Networks (OSN) are rapidly growing and gaining large amounts of sensitive user data. The centralized OSN model, widely adopted by prevailing social networks such as Facebook, makes an explicit trade-off that maintain viable monetary incentives for the provider at cost of sacrificing user control on their own data. In this “walled garden” OSN architecture, all personal information, social data, and relationships are exposed to a single provider, who can generate revenue through content-driven adds placement, or in some cases, selling user data sets. In contrast, distributed OSN architectures from the research community take the opposite extreme: total privacy. Users manage their own data and use end-to-end encryption to preserve privacy against the providers, but removes the economic incentives for providers to host data. We believe that they are all missing one critical property: choice.

We present a distributed OSN architecture called Polaris. Instead of taking a single point in the provider incentive versus user privacy tradeoff, Polaris allows users to make their own trade-off between cost and privacy, while balancing the economic needs of OSN providers. Polaris consists of an open-source client application on mobile device and an API specification that is implemented by commoditized OSN providers. The client application acts as the identity provider and access control manager for distributed providers. It stores privacy sensitive data, including personal profile, friend list, and access control list, local while distributing social network functionality to providers. Social network functionality is partitioned into application components (status updates, photo sharing, etc) are served by competing providers. This prevents one provider from having all of a user's private data. The standardized API ensures compatibility between providers, and through the use of friend access control, allows users to share content with friends who chose different providers. Users can make their choice among the competing providers: some users may choose add-driven providers for free and always-on service while others may be willing to pay nominal fee to the provider for improved privacy guarantee.

We implement the Polaris client as an application for the Android OS. Polaris uses Android Cloud to Device Messaging (C2DM) to push real-time notifications to the device. Polaris utilizes the Bump API to facilitate friending securely and instantly when users are physically meeting, while traditional, search based friending is also supported. Other basic functionality includes signup for commoditized providers, manage group based privacy policies, and normal OSN usage such as status updates, photo sharing, geo-location check-in, etc. Prototype Polaris providers were created for status, photo, and geo-location, and are implemented with Ruby on Rails using the Polaris API.