Personal Containers: Yurts for Digital Nomads[1]

Malte Schwarzkopf †, Anil Madhavapeddy †, Theodore Hong †, Richard Mortier ‡

†University of Cambridge
Computer Laboratory
15 JJ Thomson Avenue
Cambridge CB3 0FD
firstname.lastname@cl.cam.ac.uk

‡Horizon Digital Economy Research
Sir Colin Campbell Building
Triumph Road
Nottingham NG7 2TU
richard.mortier@nottingham.ac.uk

1 Introduction

In the youthful days of the Internet, there was a clear division between public data (web homepages, FTP sites, etc.) and private (e-mail, personal documents, etc.). Many people archived their personal e-mail and home directories and thus were able to keep a simple history of all their digital activities. The pace of change in recent years has been tremendous, not only in the *variety* of personal data, but in *where* that data is held. It has moved out of the confines of desktop computers to datacentres hosted by third-parties such as Google, Yahoo and Facebook, who provide "free" hosting of data in return for mining information from millions of users to power advertising platforms.

These sites are undeniably useful, and hundreds of millions of users voluntarily surrender private data in order to easily share information with their circle of friends. Hence, the variety of personal data available online is booming—from media (photographs, videos), to editorial (blogging, status updates), and streaming (location, activity).

Unfortunately by sharing data using sites like this, we lose much control over our data by handing it over to third-parties, decreasing our online privacy. We have also become digital nomads: we have to fetch data from many third-party hosted sites to recover a complete view of our online presence. Why is it so difficult to go back to managing our own information, using our own resources? Can we do so while keeping the "good bits" of existing shared systems, such as ease-of-use, serendipity and aggregation?

Although the immediate desire to regain control of our privacy is a key driver, there are several other longer-term concerns about third-parties controlling our data. The incentives of hosting providers are not aligned with the individual: we care about preserving our history over our lifetime, whereas the provider will choose to discard information when it ceases to be useful for advertising.

1.1 Hunting for a Digital Home

When e-mail and home pages were the height of Internet presence, it was fashionable to obtain "shell accounts" to manage one's own e-mail. This represented a home location on the Internet from where messages could be managed and archived. Nowadays, shell accounts are less useful for keeping our data, for a few reasons:

1. *Data is spread* around the Internet and over many devices such as laptops, mobile

[1] Draft 1, under submission

phones and iPods. It is no longer centralised in e-mail or home directories.

2. *The ever-increasing variety of data* means that tools and standards for aggregating content have not kept up with the growth of new forms of personal information such as photos, status updates, and tweets.

3. *There is no single best place* to store personal information which meets all of our needs for private, high-capacity, accessible, long-lived, reliable, flexible, and cheap storage. Each of the numerous possibilities available (e.g., cloud hosting, home appliances, mobile phones) offers a different set of trade-offs among these qualities.

A number of existing services have attempted to tackle these problems, but no comprehensive solution has yet emerged. FriendFeed[2] aggregates data across social networks, but is a hosted service with the privacy and archival problems that entails. OpenSocial [4] is a standard for federating content, but does not deal with how to store or manage it. Vis-a-Vis [1] is a privacy-preserving proxy system, but only for mobile devices and requires cloud hosting.

1.2 Building a Nice Yurt

We are building "personal containers" as a complete way to tackle all these problems—*where* to store personal data without trusting third parties, *how* to access it, and how to understand *what* that data represents. Every individual has their own personal container as a logically centralised place for their information, although it may be physically distributed over available resources and devices.

Crucially, instead of simply archiving raw data, personal containers also contain the *logic* to handle and manage that data, such as serving it through standard protocols such as HTTP, IMAP or XMPP. This means that users can have a single logical location at which to point their devices, and it can be easily upgraded in the future to handle new formats and keep legacy data alive whether by periodically transcoding it, or by preserving the access libraries and even operating system platform alongside the data itself.

The personal container has "data drivers" for a variety of data sources: cloud services such as Twitter, Facebook, and Google; desktop operating systems such as MacOS X SyncServices, and individual applications such as iPhoto and iTunes; and mobile devices such as the iPhone and Android for SMS and call records, as available from the device.

1.3 Meeting the Neighbours

Synchronising and storing data is of limited use unless we can share it in a controlled manner. Social networks have taken off because of a strong desire to share information with friends or other like-minded people, yet users remain uneasy about the lack of clarity over how much is shared and with whom. Personal containers give the owner the ability to selectively export information to others through filters, returning control to the user.

Personal containers can communicate back to social websites through their APIs, e.g., to publish an activity stream. However, they also advertise their services over standard DNS,

[2] `http://friendfeed.com/`

with an identity certificate.[3] This permits one personal container simply to perform a DNS lookup, confirm the identity of a host it connects to, and directly transmit data without a trusted third-party involved. Communication occurs over the XMPP protocol, which permits the user to join the session using a standard messaging client and monitor the automated messages.

The user can also choose to hide their identity and share information on an anonymous or pseudonymous basis. This is particularly useful when dealing with commercial organisations, where the user may want to reveal a limited set of characteristics in exchange for something they want, without revealing a full identity which might be linked to information from other sources. For example, I might be happy to tell a news site my age and city of residence in exchange for access to the site, but not want the site to see a name which they could link to other databases.[4]

2 Personal Containers

The core element of the personal containers vision is to return to you control of your digital footprint. Ideally, every Internet user would have a personal container, containing and aggregating all of their digital "belongings," moving along with them as they roam the Internet. The user could then decide both *who* to grant access to data in their personal container, and *what* to share for everyone to see.

As noted earlier, there is no single perfect platform to run personal containers on, as the cloud, mobile and home all vary in their capabilities. We have identified the key dimensions to consider as storage, bandwidth, accessibility, computational power, cost, and reliability. Table 1 shows how various possible backend platforms for handling personal data are positioned in this space.

Platform	*Google AppEngine*	*VM (e.g., on EC2)*	*Home Computer*	*Mobile Phone*
Storage	moderate	moderate	high	low
Bandwidth	high	high	limited	low
Accessibility	always on	always on	variable	variable
Computation	limited	flexible, plentiful	flexible, limited	limited
Cost	free	expensive	cheap	cheap
Reliability	high	high	medium (failure)	low (loss)

Table 1: Comparison of different platforms to store and handle personal data.

We thus aim to use all the above platforms, emphasising and combining their relative strengths. We use Google AppEngine to run a *data collector node*, pulling data out of other services, and a POSIX-compatible server application that can be run on e.g., a home computer as the *permanent storage backend*. Additionally, the server can run on a VM in the cloud, acting as a *flexible compute proxy* and *high-bandwidth caching node*. Client applications for the personal container can run on a multitude of devices, including mobile ones. The client applications are chiefly used as access portals and *data drivers*, pushing locally created data into the personal container. This high-level architecture is depicted in Figure 1.

[3] We have previously explored confirming friends' identities through physical contact [5].

[4] Facebook's *instant personalization* is a recent example of unexpected sharing of even *public* data causing concern to users.

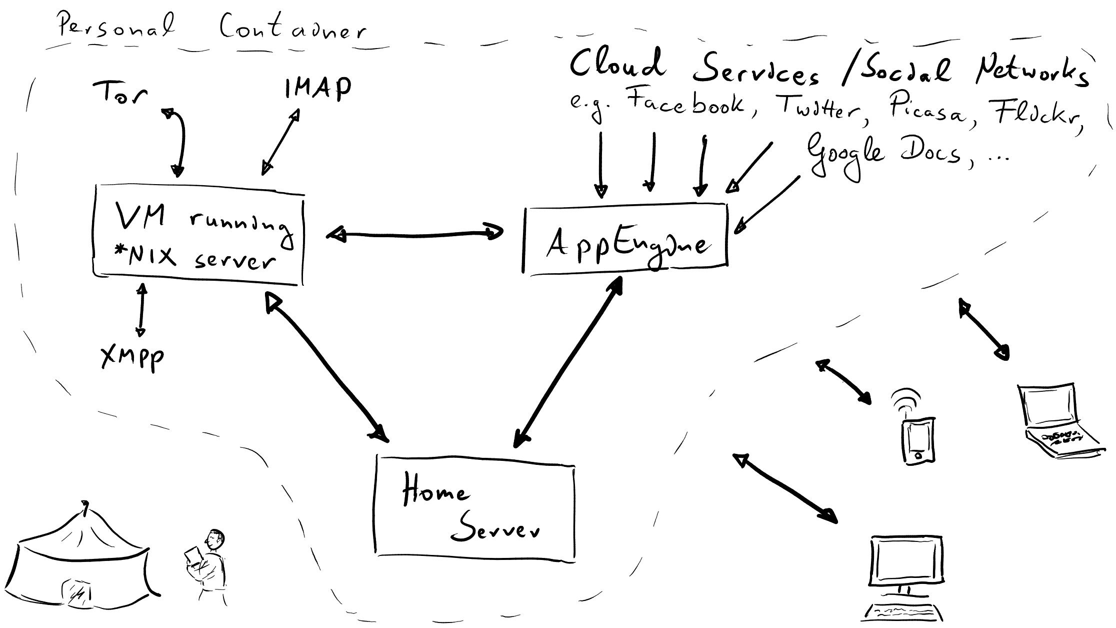

Personal Container
Cloud Services / Social Networks
e.g. Facebook, Twitter, Picasa, Flickr, Google Docs, ...
Tor
IMAP
VM running *NIX server
AppEngine
XMPP
Home Server

Figure 1: Preliminary schematic architecture of a personal container.

We are currently implementing a prototype personal container, with components running on Google AppEngine, locally on clients and on Android devices, and a separate server application—written in OCaml—that can be run on a POSIX-compatible OS. A web UI allows events and aggregate data in the personal container to be viewed (see Figure 2).

The prototype uses a flexible plugin architecture, making it straightforward to add new data sources or capabilities to the personal container. The source code is freely available, and we welcome input and contributions.[5]

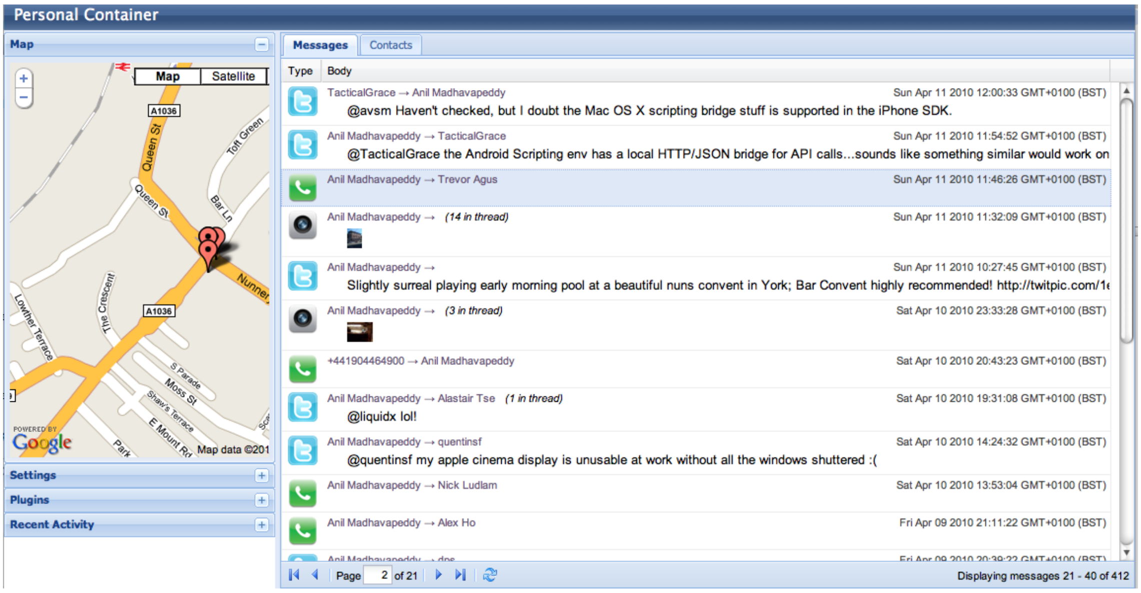
3 Perspectives

3.1 Secure Aggregation

Aggregation of personal data stored with different services, despite its perceived benefits and uses, is a major threat to online privacy. Users often use pseudonymns when interacting in online platforms, and may rely on the fact that, to the best of their belief and knowledge, there is no connection between, e.g., their work email address and their Twitter account. At the same time, there is increasing demand for aggregation of this data, since users' digital identities are increasingly fragmented across the Internet.

Personal containers enable secure aggregation: since the interfaces to other cloud services allow them to pull data out of existing services and aggregate it inside the trusted environment of the personal container. From here, it can be visualised, or even re-exported to other platforms. Crucially however, no potentially untrusted entity gains access to the raw input data for aggregation, and the results can be exposed in a controlled fashion.

[5] `http://www.perscon.net/`



Type Body

TacticalGrace → Anil Madhavapeddy Sun Apr 11 2010 12:00:33 GMT+0100 (BST)
@avsm Haven't checked, but I doubt the Mac OS X scripting bridge stuff is supported in the iPhone SDK.

Anil Madhavapeddy → TacticalGrace Sun Apr 11 2010 11:54:52 GMT+0100 (BST)
@TacticalGrace the Android Scripting env has a local HTTP/JSON bridge for API calls...sounds like something similar would work on

Anil Madhavapeddy → Trevor Agus Sun Apr 11 2010 11:46:26 GMT+0100 (BST)

Anil Madhavapeddy → *(14 in thread)* Sun Apr 11 2010 11:32:09 GMT+0100 (BST)

Anil Madhavapeddy → Sun Apr 11 2010 10:27:45 GMT+0100 (BST)
Slightly surreal playing early morning pool at a beautiful nuns convent in York; Bar Convent highly recommended! http://twitpic.com/1e

Anil Madhavapeddy → *(3 in thread)* Sat Apr 10 2010 23:33:28 GMT+0100 (BST)

+441904464900 → Anil Madhavapeddy Sat Apr 10 2010 20:43:23 GMT+0100 (BST)

Anil Madhavapeddy → Alastair Tse *(1 in thread)* Sat Apr 10 2010 19:31:08 GMT+0100 (BST)
@liquidx lol!

Anil Madhavapeddy → quentinsf Sat Apr 10 2010 14:24:32 GMT+0100 (BST)
@quentinsf my apple cinema display is unusable at work without all the windows shuttered :(

Anil Madhavapeddy → Nick Ludlam Sat Apr 10 2010 13:53:04 GMT+0100 (BST)

Anil Madhavapeddy → Alex Ho Fri Apr 09 2010 21:11:22 GMT+0100 (BST)

Anil Madhavapeddy → dps Fri Apr 09 2010 20:39:22 GMT+0100 (BST)

Page 2 of 21 Displaying messages 21 - 40 of 412

Figure 2: Personal Container prototype displaying tweets, pictures and phone calls.

3.2 Confidential Computation

Recent developments in the field of homomorphic encryption [3] offer the promise of eventually being able to perform confidential computation, in which the data being processed does not have to be revealed to the provider doing the computation. A homomorphic encryption scheme has the property that an arbitrary function can be computed on encrypted data to yield an encrypted result, without ever having to decrypt the original data. For example, this could allow you to store all of your photos in encrypted form on a cloud computing server and do a computationally-intensive face recognition scan over those photos without revealing them to the server.

3.3 Anonymous Communication

A personal container can be linked into existing anonymous communication networks, such as Tor [2]. For example, a VM running as part of the personal container could be running a Tor client, providing the owner with an anonymous channel to communicate with other users, as well as to share data. The Tor architecture of "hidden services" and rendezvous points could be leveraged to establish anonymous communications between previously and otherwise unrelated personal containers. For instance, one could imagine performing a large aggregate query across a large number of personal containers, which are happy to contribute data provided they can remain anonymous in the process.

3.4 Provenance Tracking

An important component of online privacy and its maintenance is being able to track where information is flowing, and finding out who knows what. Personal containers form an ideal platform for this as they are hooked up with all potential data sources and sinks, and can act as an arbitrator between them. As an example, the personal container could provide a large append-only memory that holds every copy of every data object that has ever passed through it, along with information about the object's source and destination.

Using an appropriate interface, the provenance of data stored in the personal container can be visualised and the user can be given precise information about who can access various bits of their personal data. This helps the user to get a better handle on the different profiles of themselves that are visible to different audiences. For example, if you email a photo to someone, it is only visible to the recipient (unicast), but if you upload it to Facebook, it is visible to all of your Facebook friends (multicast), and if you tweet a shortcut to it, the entire Internet can see it (broadcast).

3.5 Moving On

Longevity over the long haul is an important goal of our system. Inevitably, new paradigms of computing will appear, displacing current ones, just as traditional desktop computing has been displaced by web applications and cloud computing. Personal containers have the flexibility to evolve as new computing paradigms appear, as they are designed as a federated network of heterogeneous computing nodes. New types of computing nodes can be added to a container and begin synchronizing and replicating data held by the existing nodes, while also offering new services. Gradually the new nodes will handle more and more data, while outdated nodes fade in importance and can eventually be archived or removed. In this way, your personal information is continuously migrated to the latest platforms rather than being lost in the shuffle (e.g., those 5.25" floppy disks containing your old WordStar documents).

4 Conclusions

We envision the *Personal Container* as a system that can control, manage and store personal data in the present and future Internet. Having presented this vision, we are currently focusing on continued development of our prototype in order to perform a large scale evaluation. We call upon the community to assist us in this effort to re-attain user control over personal data whether by providing comments, feedback or code.

References

[1] R. Cáceres, L. Cox, H. Lim, A. Shakimov, and A. Varshavsky. Virtual individual servers as privacy-preserving proxies for mobile devices. In *Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds - MobiHeld '09*, page 37, New York, New York, USA, 2009. ACM Press.

[2] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium.* USENIX Association, 2004.

[3] C. Gentry. Computing arbitrary functions of encrypted data. *Commun. ACM*, 53(3):97–105, 2010.

[4] L. Grewe. *OpenSocial Network Programming.* Wrox Press Ltd., Birmingham, UK, UK, 2009.

[5] R. Sharp, A. Madhavapeddy, R. Want, and T. Pering. Enhancing web browsing security on public terminals using mobile composition. In *MobiSys '08: Proceeding of the 6th international conference on Mobile systems, applications, and services*, pages 94–105, New York, NY, USA, 2008. ACM.