UCIP: User Controlled Internet Protocol

Morteza Kheirkhah[1], Truong Khoa Phan[1], XinPeng Wei[2], David Griffin[1], Miguel Rio[1]
[1]University College London, [2]Huawei
[2] weixinpeng@huawei.com, [1]{m.kheirkhah@ucl.ac.uk, t.phan,d.griffin,miguel.rio}@ucl.ac.uk

***Abstract*—Internet protocols have developed significantly over the last 50 years but have reached a point where the further improvements in performance, resilience, security and privacy cannot be achieved by simple incremental changes. This paper proposes a new IP protocol that puts the user's end host at the centre of major algorithmic decisions. It consist of three new mechanisms: a private source routing establishment protocol that allows inter-domain traffic routes to be decided by the user and kept private from the providers whilst allowing for anonymous connections where two node can communicate without knowing the identity/address of the other end point; a mechanism to control reception of packets that mitigates denial-of-service attacks and a new directory system that puts the end user at the core of the decisions enabling anycast and mobility with a pub-sub mechanism with fine grain capabilities for describe resources. These changes allow end nodes to have a much tighter control of how they send and receive their traffic and provide a paradigm shift for the Internet ecosystem.**

I. Introduction

Although the Internet has proved to be an incredibly successful tool for communication and information access there are some fundamental architectural issues that prevent higher levels of resilience, security and privacy.

These problems stem partly from major decisions being taken away from the users. Although the end-to-end principle [1] has been a constant presence in the effort to decentralise the Internet, the last decades have put disproportionate decision-making power in stakeholders other than the users: network providers, cloud providers, DNS providers, etc.

Before addressing the changes needed to put users at the heart of decision making, we first outline the decisions that are currently removed from the users:

- Which routes should packets follow? Ideally we would like users to be able to try different routes and adapt or change routes depending on traffic dynamics and application requirements. For example, multi-path solutions such as MPTCP [2] have a small set of paths the algorithm is able to select between (outside of data centres).
- Who sees the identity of senders and receivers of traffic? Due to the nature of hop-by-hop routing packets in the current IP need to carry a destination address. Every router and provider involved in the routing of the packet is able to see this. This could be seen as a fundamental violation of user privacy since knowledge of the destinations of user traffic can reveal a significant amount about a user's online activity.
- Who is allowed to send data to a specific user/end host? Currently every node can send to any other node on the Internet. This default-on mode of operation leads to an architecture where denial-of-service (DoS) attacks are relatively common and easy to implement.
- Which copy of content/service replica a user accesses? Decisions on which replicas of content or server each user accesses are made by service providers without necessarily having the interest of the user as their main priority [3], [4]. End user devices should be able to make these choices based on their own requirements, which they will know better than the service provider.
- Who is able to see where the user is located? Current name resolution through DNS is primitive and does not cover presence services to manage user mobility. Mobility and presence of users is managed by various walled-garden solutions such as social media applications that do not interact with one other. The successor of DNS should implement this functionality in an open manner.

The Internet has evolved without inherent privacy preserving features and there is limited influence users can make on decisions on how their packets are routed and forwarded. This has implications on service performance and accountability. If performance is poor or if the metadata of who connected to whom can be leaked to third parties, users have no visibility of which domains were to blame and have little control on how their traffic can avoid those domains in the future.

This paper proposes UCIP - a user-centric Internet protocol which gives users a higher degree of control on how their data is transferred across the network and who knows about it. It consists of three elements: A private source routing scheme that allows for controllable private paths to be established; a sender control scheme that allows receivers to whitelist senders allowing for default-off communication; and, thirdly, an overhaul of the name resolution service that allows a richer control by the end user of what services and service replicas it uses. By shifting to a source routing paradigm based on light-weight connection state information users can batter match their requirements to delivered performance.

II. Private Source Routing

With private source routing (PSR), nodes establish bidirectional connections using private source-routed packets. The source route will contain an ordered list of the domains the packet will traverse, not the routers themselves. Inside each domain it is up to its network administration to route the packet as they see fit to the next domain in the list. Domains can be seen as equivalent to today's autonomous systems, although there does not need to be a one-to-one mapping between PSR domains and ASs. Note that only the first packet in each

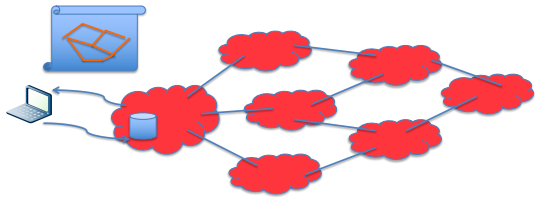

Fig. 1: Map propagation

direction of the connection needs to convey the full source routing information, as explained further below.

A. Map Propagation

The first step in PSR is domain map propagation. The global connectivity map of PSR domains is built using a link-state protocol and is sent to every device and updated accordingly, as illustrated in Figure 1. Our assumption is that this map is pushed periodically to users whenever there are inter-domain topological updates. Although this task may seem challenging we think it is perfectly feasible even today (see discussion section).

The map consists of two parts:

- Static information about the domains: country, administration, contact and the domain's public keys signed by a certificate authority.
- Dynamic information about links between domains. This may change periodically. It also includes type of link (customer-provider, peer-to-peer) to provide sufficient information in order for users to avoid building routes that are not valley-free [5]. A user should never build a route that uses customer-provider peering as transit. If this is violated by a user's path selection the domain will reject the connection.

Note that some edge domains who are not offering publicly-available services may not want to propagate this information to the public and choose instead to selectively disseminate it through other means.

Additional performance information about links and domains (e.g. link load) may be obtained or collected through parallel information systems. UCIP does not require domains to volunteer this information themselves which may be difficult to trust anyway. Providers like Thousand Eyes [6] should be able to provide this information on a domain-neutral basis.

B. Path Construction

The second step in PSR involves route construction. A sender will use an algorithm, e.g. shortest-path or a variant

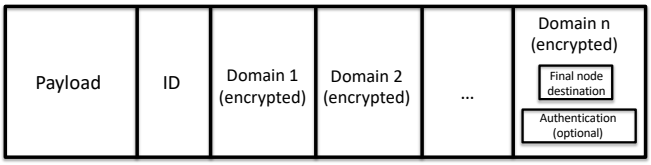

Fig. 2: UCIP packet

to maximise throughput or improve resilience with latency guarantees [7], for example. It may also apply its own specific policies to avoid certain domains or geographical regions, for example. Note that due to this being a source-routing system it is not necessary for all users to use the same routing algorithm. The result of the routing algorithm is the UCIP packet shown in Figure 2.

This is sent as the first packet for both directions. The main part is the list of domains the packet will traverse. The identifier of each domain in this list is encrypted with the public key of the previous domain. This guarantees that no domain in the path knows the full list of domains in the path. Only the origin domain will know who is the sender of the packet and only the destination domain can see the destination identifier/address. When the final domain decrypts the last part it will find the final destination and some optional authentication credentials. We will explore the latter in section III.

Each UCIP packet contains a unique ID which allows subsequent packets to be routed along the same path, without requiring the full list of encrypted domains to be sent in each packet of a flow. The ID is fixed for the lifetime of the flow and is also used in subsequent flow re-routes. If multi-path is used different IDs will be used per sub-flow.

Because clients select their own paths this allows them to build true multi-path (from the inter-domain point of view) connections for the same application. This can increase throughput - by sending a different sub-flow that avoids congested paths - or protect for resilience - by sending the same data through two disjoint inter-domain paths. Technologies like multi-path TCP, which inherently allows a data sender to select and/or dynamically change the path for its sub-flows, has gained popularity recently and PSR increases the benefits of MPTCP by allowing finer-grained control over the paths taken by sub-flows, to increase performance and resilience to failures.

C. Bidirectional Path establishment

The final step in connection establishment in PSR is the reply packet that traverses the reverse path in a way equivalent to RSVP-TE [8]. The reply packet structure is shown in Figure 3

The reply packet (Figure 3 contains the ID of the flow to match against the previous request and the downstream label the domain wishes the packets to be marked with. It is important to note that PSR does not require per-flow state.

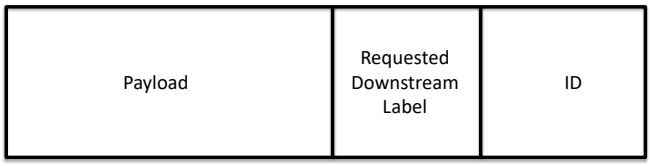

Fig. 3: UCIP reply

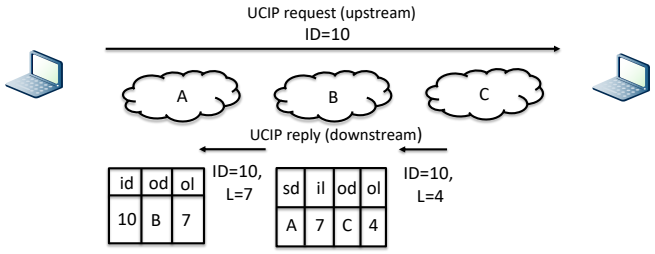

Fig. 4: UCIP label table construction

Domains may ask for labels to be switched in the upstream domain but this can agglomerate many connections.

This interaction can be seen in Figure 4 where the device on the left is establishing a connection with the one on the right. UCIP requests an upstream connection creating temporary state in the control plane. UCIP replies are then sent downstream requesting labels in each downstream domain. These are the labels that domains uses to mark the packets for that flow. In the figure, the path traverses 3 domains. The reply in the downstream direction populates the label tables in each domain. These include the output domain (od), output label (ol), input domain (id) and input label (il). In domain B, when a packet for this flow arrives from domain A with label 7 it should be sent to domain C with label 4.

There is a similar interaction for labels on the PSR return path for packets sent in the downstream direction that is not shown in the figure. Hence, each domain will maintain two label tables per path, one in each direction.

In theory all flows following the same path could use the same table entry to reduce state maintained by the domains to a single entry per path rather than one per flow. However, this could potentially allow malicious domains to circumvent privacy by reverse engineering paths by establishing flows to a range of destinations and checking the labels to discover the previously hidden destinations of user flows. Further work needs to be done to quantify the trade-off between privacy assurances and label table size.

There is no fundamental reason why a receiver of a connection needs to know which end-point initiated the connection. Like [9] we do not include a source address in the packet. It is not needed and the path is bidirectional anyway. Applications that require source addresses should include this information in the packet payload.

D. Path Tear-down

Connections get torn down with a FIN packet which may or may not alter state in the routers. All UCIP messages can piggyback on transport level messages equivalent to today's TCP messages (SYN, SYNACK, FIN, etc), for example, they do not require additional handshake cycles beyond those needed for TCP.

III. Controlling Senders

The economics of the Internet make it easy to send packets to anybody. Although this default-on approach has advantages, it makes denial-of-service attacks relatively simple to deploy, which is a major problem today. The vast majority of people/systems do not need to accept unsolicited connections from remote senders and therefore do not need default-on. UCIP allows for the deployment of default-off where the network only delivers the packet to the destination if it was previously authorised. This works by including in the connection packet cryptographic credentials. These are checked in the home domain of the recipient when it decrypts the last hop in the PSR packet and the packet is only delivered if this authentication matches the list of authorised senders. In case an initiator tries to establish a connection for which they are not authorised the UCIP reply will contain an error which is propagated through to the initiator's domain. This domain can blacklist the initiator or take some other action.

IV. Overhauling Internet Resolution

The final piece of the puzzle is an overhaul of the name resolution system with a new directory service. Since the 1980s this function has been performed by the DNS. However, DNS has evolved to become a relatively complex system where entities like content distribution networks and ISPs make resolution decisions without considering user preferences on how to select between replicas.

UCIP implements a new directory service to replace DNS resolution with the following goals:

- The directory service should be generalised to include all networked resources including people, content and services, e.g. IoT devices, full documents and media resources. The directory should maintain mobility/presence information and inform users when resources move or are replicated.
- Resolution will return the requested resource locator together with its domain identifier so that in a source routing context, users can construct PSR routes to the destination domain(s). It should also allow for a range of name spaces to be used; not just IP addresses.
- The directory should be based around a pub-sub system so that changes due to locator changes or additional replicas being created can be pushed to the requesting systems. This allows for users to dynamically adapt to remote system mobility and to changes in replica availability. While UCIP allows both stateful and stateless services, dynamic selection is more easily handled for

stateless servers where applications can change to a better copy as conditions change or new replicas come online.

- The directory should be able to indicate preference and load information so that the users' selection can be informed to optimise service quality. Service providers should be able to indicate their preferences between replicas based on service availability and network congestion levels.
- The directory service should be built with inherent privacy and security principles. Recent controversy regarding DNS over HTTPS [10] brought to light privacy concerns regarding users' DNS queries. It is paramount that the resolution mechanism is kept private and as anonymous as possible. By leveraging on PSR's privacy, the existence of directory queries cannot be revealed to third parties by observing source and destination addresses. Furthermore the client's identity does not need to be revealed to the directory service itself. It should be noted that information queries to the directory service do not need to be sent to the local server operated by the users' ISP since all raw data is available to everybody. Results should always be the same independently of where they come from if users are able to select only trusted directory servers this removes the possibility of untrusted directories snooping on user queries.

V. Discussion and Open Questions

A. Scalability of domain map propagation

Given that we are, in fact, making inter-domain routing link-state, the size of this data may be problematic. Every client/end-host needs a copy of this map and to receive updates. This can be challenging but we believe it is achievable in the long-term. Although we have potentially tens of thousands of domains, not all need to be propagated to everybody. Furthermore, if one takes studies on BGP update frequency into account [11], [12] one can conclude that it is not very high. Finally, as networks become more reliable the numbers of updates due to failures will tend to reduce.

B. Size of connection packets

The size of the connection initiation packet may be large. A minimum of 255 bytes are needed for each encrypted domain. In the case of long path lengths in terms of domains being traversed and if the de facto limit of approximately 1500 bytes at layer 2 continues then UCIP packets will need to be properly fragmented.

C. Connections inside the same domain

UCIP does not prescribe how packets are routed within domains. Providers will still have full flexibility for intra-domain traffic engineering.

D. Connections traversing a small number of domains

Privacy is compromised when the UCIP path traverses less than three domains. If the packet's destination is within the same domain as the source then no path privacy can be achieved. For paths of two domain hops senders could repeat either the source or destination domain in the source routing packet in order to avoid the full domain path being exposed to either of the two domains. In this case the repeated domain would simply ignore the additional fake hop. Consider an example of a path which crosses just two domains, D1 and D2 where the user has determined that the path should be D1->D1->D2. Domain 1 when decrypting the first hop will see that the second domain is itself and will also decrypt the second hop which identifies the true next hop as being domain 2. Domain 2, on the other hand, will see that the path includes two prior hops but will not be able to decrypt the first two hops meaning it will not know that the first hop was actually domain 1. As mentioned previously we do not expect there to be a one-to-one mapping between ISPs/ASs and domains, for example we anticipate that cloud providers will have their own domains which will add at least another hop to the PSR path and therefore enable another level of privacy.

E. Sticky routes may impact resilience

PSR connections are routed along the set of domains determined by the originating node and established during flow set-up. Intermediate network domains are unaware of the final destination and are therefore unable to dynamically reroute the connection in the case of network outages. Sticky routes can exhibit low resilience to failures. However, note that resilience inside each domain can still be dealt with in the same way as today. For inter-domain resilience, UCIP allows the end node to be much more involved in path selection. Users can establish several routes with minimal common links for critical applications. Given that inter-domain link state is now being maintained in UCIP with state information being propagated to users, they are able to react quickly to broken inter-domain paths.

F. Multicast

Multicast presents challenges from the point of view of privacy. If one wants the network to play a role in replicating packets for network efficiency it is very hard to keep this information entirely private. UCIP maintains the current IGMP model where rendezvous points are always located in the receivers' domain. While the sender's path to the receiver domain(s), including the inter-domain branching points, can be kept private, the receivers' local provider will always know which groups they have joined. In summary, UCIP can maintain privacy for multicast senders but the privacy of receivers in terms of the visibility of group membership to their local domain is a subject of further study.

G. Attacks on network devices and links

UCIP mitigates denial-of-service in the end systems by enabling a default-off system in packet forwarding. It does not mitigate DoS in routing/switching devices. There are two types of problems here: economic attacks and DoS. A user can use a link which is not the most efficient just to incur a cost to a domain (or to benefit another domain). Moreover,

a user or a set of users, can coordinate an attack on a given inter-domain link and to overload it with traffic. Mitigation strategies against these potential attack vectors is a subject of ongoing study.

H. Compatibility with inter-domain traffic engineering

UCIP reduces the capability of providers to do inter-domain traffic engineering. Although we believe this is a price worth paying, it is also worth noting that providers are not compelled to announce all links and they are able to include prioritisation metrics (similar to the MED attribute in BGP). Allowing users to choose routes also challenges the current charging practices of ISPs where users pay independently of the remote origin/source of traffic they receive/send. Users can always choose more expensive routes despite ISP preferences. We believe this to be a minor issue, transit costs have been reducing and users would want, in the majority of cases, to have efficient routes. However, charging a premium to users who adopt UCIP PSR is also an option.

I. Overhead of PSR

Since the next-hop domain information in initial UCIP route establishment packets needs to be decrypted in each domain, this could be a source of non-negligible overheads. The advent of SDN [13], where initial packets are escalated to central controllers in each domain, can help here since the decryption can be done in high performance servers potentially with dedicated hardware rather then in the routers themselves. It should be noted that individual flows to the same destination can be multiplexed over the same path. Depending on traffic patterns and user behaviour some paths can be long lived and hence new PSR paths do not be established for each user flow, so reducing the decryption overhead for new paths.

VI. Related Work

This protocol builds on a plethora of previous work on the diverse fields of source routing, security and domain name resolution.

Despite the clear advantages of source routing and the fact that it has been present in IP since the beginning [14], its use has been discouraged due to security reasons [15]. Nevertheless, several proposals have been put forward. Examples include the Nimrod architecture [16], Pathlets [17], NIRA [18], MIRO [19] and [20]. More recently segment routing [21] has gained significant acceptance from the community which opens the door for source routing being a fundamental idea on how to connect devices. Furthermore, source routing has been successfully implemented in data centres [22].

Our private source routing has similarities with Tor/onion routing [23] in the way that the full path is hidden to other routers. However, rather than implementing overlay routing as in Tor, UCIP is designed as a network infrastructure protocol that allows nodes to have even more efficient routes than today.

Our default-off ideas have been inspired by [24] but have no requirement to change how routing information gets propagated. We can see our system as an architectural extension of a firewall, but inside the network.

Evolution and adoption of directory services in the Internet has been disappointing over recent decades. In the late 80s X.500 [25] promised significantly more functionality than today's DNS [26]. UCIP reintroduces some of those ideas to build a new directory system that upgrades DNS in a similar way to proposals such as [27].

VII. Conclusions

This paper proposes a new IP protocol for the Future Internet. By significantly transferring control to the user, it addresses several main challenges of the Internet in the following manner:

Performance: Applications will be able to explore path diversity and by choosing one or more appropriate paths they can target specific requirements and increase throughput.

Reliability: End systems can exploit redundant paths to destinations. Paths can be used simultaneously or activated when loss is detected.

Network security: While many security threats to computer systems can be addressed by end systems or nearby middleboxes, denial-of-service attacks are the exception. The network needs to actively mitigate against them. UCIP proposes a default-off mechanism that allows the network to filter packets before they reach the destination. Negative replies can be cached to prevent further unauthorised transmissions. Furthermore, PSR allows for users to prevent specific domains being used to route their traffic allowing for a level of trust in the domains along the path not available today.

Privacy: Privacy has become a major societal concern and path privacy is part of this. While end-to-end encryption can secure data the sources and destinations of traffic cannot be easily hidden today without resorting to overlay obfuscation techniques such as Tor. UCIP's private source routing significantly limits the source and destination information visible to intermediate routing domains making it much harder for an interceptor to identify the communicating entities. UCIP deals only with network path privacy and not with the privacy of the data itself. While end-to-end encryption is the major established technique for securing in-flight data there is a interesting possible collaboration between UCIP and alternative efforts like Databox which provide techniques for keeping personal and private data on site rather than being stored and processed in remote cloud nodes [28].

Anycast and mobility: UCIP includes a new directory service that allows for knowledge of resource replicas to be distributed to everybody with enough granularity for user selection and for updates to be distributed as resources/people move to different network attach points.

Acknowledgments

This work was partially funded through the support of Huawei Technologies Co. Ltd.

References

[1] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-End Arguments in System Design. In *International Conference on Distributed Computing Systems*, 1981.

[2] Damon Wischik, Costin Raici, Adam Greenhalgh, and Mark Handley. Design, Implementation and Evaluation of Congestion Control for Multipath TCP. In *USENIX NSDI*, 2011.

[3] T. K. Phan, D. Griffin, E. Maini, and M. Rio. Utility-centric Networking: Balancing Transit Costs with Quality of Experience. *IEEE/ACM Trans. Networking*, 2018.

[4] M. Rocha, T. K. Phan, J. Reis, D. Griffin, and M. Rio. Triptych: Multi-objective Optimisation of Service Deployment Costs, Application Delay and Bandwidth Usage. In *IFIP NETWORKING*, 2019.

[5] Sophie Y Qiu, Patrick D McDaniel, and Fabian Monrose. Toward Valley-free Inter-domain Routing. In *IEEE International Conference on Communications*, 2007.

[6] https://www.thousandeyes.com.

[7] J. Li, T. K. Phan, W. K. Chai, D. Tuncer, G. Pavlou, D. Griffin, and M. Rio. DR-Cache: Distributed Resilient Caching with Latency Guarantees. In *IEEE INFOCOM*, 2018.

[8] E. A. Farrel, A. Ayyangar, and JP. Vasseur. RFC 5151 - Inter-Domain MPLS and GMPLS Traffic Engineering Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions, 2008.

[9] M. B. Braun and J. Crowcroft. SNA: Sourceless Network Architecture. Technical report, University of Cambridge, Computer Laboratory, 2014.

[10] Geoff Huston. DNS Privacy and the IETF. *Internet Protocol Journal*, July 2019.

[11] O. Bonaventure, C. Filsfils, and P. Francois. Achieving Sub-50 Milliseconds Recovery Upon BGP Peering Link Failures. In *ACM CoNEXT*, 2005.

[12] J. Rexford, J. Wang, Z. Xiao, and Y. Zhan. BGP Routing Stability of Popular Destinations. In *ACM SIGCOMM Workshop on Internet Measurement*, 2002.

[13] Hyojoon Kim and Nick Feamster. Improving Network Management with Software Defined Networking. *IEEE Communications Magazine*, 2013.

[14] RFC 791 - Internet Protocol DARPA Internet Program Protocol Specification, 1981.

[15] David Hoelzer. The dangers of source routing. Technical report, Enclave Forensics.

[16] I. Castineyra, N. Chiappa, and M. Steenstrup. RFC 1992 - The Nimrod Routing Architecture, 1996.

[17] P. B. Godfrey, I. A. Ganichev, S. J. Shenker, and I. Stoica. Pathlet Routing. In *ACM SIGCOMM*, 2009.

[18] X. Yang, D. Clark, and A. W. Berger. NIRA: a New Inter-domain Routing Architecture. *IEEE/ACM Trans. Networking*, 2007.

[19] W. Xu and J. Rexford. MIRO: Multi-path Interdomain Routing. In *ACM SIGCOMM*, 2006.

[20] X. Yang and D. Wetherall. Source Selectable Path Diversity via Routing Deflections. In *ACM SIGCOMM*, 2006.

[21] C. Filsfils, S. Previdi, B. Decraene, S. Litkowski, and R. Shakir. RFC 8402 - Segment Routing Architecture, July 2018.

[22] M. Kheirkhah, I. Wakeman, and G. Parisis. MMPTCP: A Multipath Transport Protocol for Data Centers. In *IEEE INFOCOM*, 2016.

[23] https://www.torproject.org.

[24] H. Ballani, Y. Chawathe, S. Ratnasamy, T. Roscoe, and S. Shenker. Off by Default. In *Hot Topics in Networks (Hotnets)*, 2005.

[25] D. Chadwick. *Understanding X.500: The Directory*. Chapman & Hall, October 1994.

[26] Paul V. Mockapetris and Kevin J. Dunlap. Development of the Domain Name System. In *ACM SIGCOMM*, 1988.

[27] Abhigyan Sharma, Xiaozheng Tie, Hardeep Uppal, Arun Venkataramani, David Westbrook, and Aditya Yadav. A Global Name Service for a Highly Mobile Internetwork. In *ACM SIGCOMM*, 2014.

[28] Amir Chaudhry, Jon Crowcroft, Heidi Howard, Anil Madhavapeddy, Richard Mortier, Hamed Haddadi, and Derek McAuley. Personal Data: Thinking Inside the Box. In *Proceedings of the Fifth Decennial Aarhus Conference on Critical Alternatives*. Aarhus University Press, 2015.