

Research Statement

Bogdan Carbunar

I work at the intersection between information privacy, security and distributed systems. I strive to provide efficient and secure solutions for end-to-end content distribution as well as privacy aware solutions for data and computation outsourcing.

One of today's ubiquitous trends is the online hosting of data and services. Users can access a wide range of content from Video on Demand services and sites such as Netflix or Hulu on their personal devices. Users can also store personal content on sites offering specialized services such as Yahoo Mail, Google Docs or YouTube and outsource complex computations to "cloud" providers such as Amazon, Google and Microsoft. At the other end of the spectrum, users can access content stored on other user devices in a wired, peer-to-peer fashion or over-the-air, in an ad-hoc manner. Such environments raise a multitude of scalability and efficiency issues as well as security and privacy concerns. My work so far has focused on providing solutions that allow users to efficiently access remote content and services without fear of unwanted side-effects, e.g., leaking personal information which can lead to further behavioral profiling, spam and targeted ad placement. In the following, I summarize my current work and then I provide plans for the future.

Secure Content Distribution: Part of my work, consisting in the efficient distribution of content, is an important component in most of today's networked environments. Of particular interest were Video on Demand (VoD) services, which provide a wide range of content options and enable subscribers to select, retrieve and locally consume desired content. They rely on proprietary Content Distribution Networks (CDNs) to transfer the content from a central library to the subscriber population. Current CDNs of VoD providers such as Comcast, Charter or Time Warner consist of a central content library and several regional streaming servers all connected through a high-speed fiber ring. They require all the regional servers to locally mirror the entire content library. While this solution provides excellent service quality and content availability, it introduces significant scalability costs: The disks, RAMs and flash drives of all regional servers need to be updated when the central library is upgraded. In [1], I worked toward addressing this problem by proposing the use of smart content placement and caching solutions and allowing regional servers to only store items that were predicted to be locally popular. Then, the hardware of regional servers needs to be upgraded only based on local demand.

To achieve this goal, several metrics for quantifying the quality of a solution needed to be identified – the network load, the lifetime of the cache storage technology and the user satisfaction. In order to address then the constraints introduced by the newly proposed content placement model, novel predictive and collaborative caching algorithms needed to be devised. The proposed solutions rely on the ability to predict a *penalty* value for each item: the cost of not storing the item for a future interval. The penalty values of items are used to drive not only the replacement algorithm (which items to evict from a cache) but also the decision of which items to cache and which to stream (and not cache). The penalty prediction was built using observations from several VoD deployment logs: Recorded values of metrics of interest (e.g., number of requests received for an item in a minute) were used to predict their future values. Using several Comcast, Charter and Time Warner logs, each consisting of more than 4.5 million user requests during more than 2 week intervals, the solutions were compared against existing approaches such as LRU, LFU or Greedy-Dual. The conclusions are that unlike existing algorithms, several of the proposed solutions are able to simultaneously reduce and balance the total network traffic, significantly reduce the daily amount of cache overwrite and allow user requests to be satisfied at their required consumption rate.

The content offered by VoD services is requested, transferred over the CDN and consumed by users on devices ranging from Set-Top-Boxes (STBs) to smartphones or tablet PCs (e.g., iPad). Since such devices often have a few tens of GBs available storage space, they can also be used to cache relevant content. The prediction techniques mentioned can be also used to detect user preferences which can drive smart pre-caching strategies and save bandwidth at peak load times. More importantly however, the widespread adoption of Wi-Fi or Bluetooth interfaces on most "companion" devices leads to a natural next step: requested content can be searched for and transferred from other devices, over alternate network interfaces. By providing bandwidth savings, this approach has the potential to significantly impact CDNs. Part of my work in this area has been on devising efficient methods for advertising and discovering content stored on neighboring devices. In [2] I devised a Content and Presence Multicast Protocol (CPMP) that allows higher level applications to share their user's content consumption actions (e.g., metadata of music currently listened, videos watched, thumbnails of pictures taken) with anyone in the user's communication range. CPMP shares local context over Wi-Fi

through periodic (multicast) updates, enabling a new type of experience: Users can see what their neighbors do, choose an interesting source of content and “tune-in” by transferring the content consumed as and when advertised by its source. Many companion devices are battery operated and Wi-Fi, even when inactive, is one of the most power consuming components. Distributed and localized algorithms needed then to be developed, that use the information contained in CPMP packets to synchronize the periodic transmissions of all the nodes. This approach enables devices to switch off their Wi-Fi cards between transmissions without missing neighbor updates.

The reliance on CPMP information makes however the synchronization process vulnerable to Sybil attacks: A single malicious participant, able to spoof valid device identifiers, can prevent the convergence of the synchronization process of multi-hop networks. This problem was addressed by rating the quality of neighbors based on the consistency of their behavior: nodes that always send their updates at a pre-agreed upon interval act as anchors of synchronization in the network. This solution, implemented on Motorola smartphones and simulated on large scale networks, quickly stabilizes the synchronization process and significantly reduces the number of updates lost even in the presence of aggressive Sybil attackers while also extending battery lifetimes by up to 30%.

Secure and Private Remote Data Access: When accessing data stored on potentially untrusted servers, users unwittingly reveal personal information. For instance, in the case of Video on Demand services, information on user browsing and consumption habits is currently used by the service providers to infer user interests and improve content and ad placement techniques. Users however have no control over who can access and the type of processing that can be performed on this sensitive data. On multiple occasions such data *has* been sold to other companies or even made public. Another part of my work has been on this problem.

Computational private information retrieval (cPIR) techniques do exist and their goal is to allow users to access remote data without leaking items of interest or even access patterns to the server storing them. However, in [3] I showed that deployment of non-trivial single server PIR protocols on real hardware of the recent past would have been orders of magnitude less time-efficient than trivially transferring the entire database for each user access. This reasoning was validated in an experimental setup on modern off-the-shelf hardware. However, this result is likely to hold on non-specialized traditional hardware even in the foreseeable future.

To address this challenge, in [4] I designed a Bloom filter based construction that reduces the amortized complexity of a cPIR protocol to $O(\lg n \lg \lg n)$ server computation overhead, in the presence of $O(\sqrt{n})$ client working memory. The proposed solution retains the pyramid-shaped database layout and reshuffling schedule employed by an Oblivious RAM (ORAM) structure but stores the database in a series hash table maintained by the server. Using a series of encrypted Bloom filters to query the location of a data item, the user can retrieve the item from the appropriate hash table. The use of encrypted Bloom filters allows the user to query an item directly at each level without revealing the success, instead of relying on a series of $O(\lg n)$ fake blocks for each stored block to hide the success of each level query. In addition to reducing the time complexity, this reduces the amount remote storage required, from $O(n \lg n)$ to $O(n)$. Overall, this solution increases the throughput by nearly an order of magnitude, and reduces the storage required by over an order of magnitude.

This solution allows users to also modify the data stored on the server. This ability is fundamental when clients outsource their data (e.g., e-mail, documents pictures) to remote (in the “cloud”) servers. When the users of the system are company employees an important obstacle remains – the need for *assured* lifecycle storage of records. Over 10,000 regulations are believed to govern the management of information in the US alone, including the Gramm-Leach-Bliley Act, the Sarbanes-Oxley Act, or the Health Insurance Portability and Accountability Act (HIPAA), to mention only a few. The main goal there is to support Write Once Read Many (WORM) semantics: once written, data cannot be undetectably altered or deleted before the end of its regulation-mandated life span. To balance the two requirements, in [5] I introduced WORM-ORAM, a first mechanism that combines Oblivious RAM access privacy and data confidentiality with WORM regulatory data retention guarantees. Clients can outsource their database to a server with full confidentiality and data access privacy, and, for data retention, the server ensures client access WORM semantics. WORM-ORAM is built on a set of novel efficient zero knowledge (ZK) proofs: The client is allowed unfettered ORAM access with full privacy to the server-hosted encrypted data set while simultaneously proving to the server *in zero-knowledge* – at all stages of the ORAM access protocol – that no existing records are overwritten and WORM semantics are preserved.

Computation Outsourcing: The well established volunteer computing projects and the recently introduced “cloud” computing services show that users are not only willing to outsource their available CPU cycles to large computing projects but they also have computing jobs for which they lack the required resources. In such computing “markets”, clients, or outsourcers, generate jobs that they would like other servers, or workers, to perform. Given that anyone can be an outsourcer or a worker and Sybil identities can easily be generated, an essential problem is verifying the correctness and completeness of outsourced computations. In [6] I proposed a solution that aggregates mutual feedback of interacting peers into a *reputation* metric for each participant. This is then available to prospective outsourcers for the purpose of

evaluation and subsequent selection of workers. The solution uses threshold witnessing, a mechanism in which a minimal set of “witnesses” are used to provide service interaction feedback and sign associated ratings for the interacting parties. Witnessing relies on a challenge-response protocol in which servers provide verifiable computation execution proofs. This endows traditional feedback rating with trust while handling both “ballot-stuffing” and “bad-mouthing” attacks.

The lack of efficient incentives is in itself an incentive for worker laziness and incomplete job computations. Top participant lists may be sufficient for enthusiastic volunteers, but may prove ineffective for arbitrary computations. Motivating participation through the use of financial incentives raises additional concerns as both outsourcers and workers can attempt to cheat. In particular, workers now also need to trust outsourcers to provide valid payments at job completion. In [7] I proposed a solution that solves both sides of the trust problem: Outsourcers can trust that workers can retrieve the payments only if they complete the jobs. Workers trust that the embedded payments are valid: if they complete the jobs they will also be able to deposit the payments. The solution works by embedding payments into jobs. The validity of payments can be verified in zero knowledge but payments cannot be obtained unless the job has been completed. The solution proposed requires minimal involvement from a bank trusted only to act as a fair financial institution. By making the bank oblivious to the actual transaction details, the solution enables the bank to perform hundreds of payment transactions per second. Moreover, the computation and communication overheads imposed on outsourcers and workers are negligible.

Anonymity is also of concern, as computing projects may be sanctioned in certain societies, while in general workers can be profiled based on the jobs they choose to execute. A straightforward use of payments can immediately link participants. Then, given the small nature of the payments likely to be used in such markets, a new problem arises: Design an anonymous payment system where the cost of providing anonymity is significantly smaller than the value of the payments. In [8] I worked on this problem, by proposing several anonymous micropayment solutions. The outsourcer’s unique identifier is divided into multiple identity shares using threshold cryptography. Each coin in a micropayment chain embeds a unique share. Overspending is prevented probabilistically, by requiring the worker to select one random coin (and associated identity share) for each job it computes. If the outsourcer spends more coins from a micropayment chain than the chain’s value, with high probability enough shares are publicly available to reconstruct the outsourcer’s identity. The solutions support thousands of transactions per second and provide full anonymity for both outsourcers and workers.

Future Work

I am very excited about the prospect of applying my expertise – in privacy and security as well as in distributed and networked environments – into related and relevant domains. My vision is to extend my existing work to create the basis for a private “cloud”: Users should expect and trust that their data is stored and handled with intrinsic privacy and security assurances. In the following I summarize several of my plans for the future.

The Private Data Cloud: Users are becoming ever more aware of the fact that their online actions are captured and mined for personal information. However, users are also often willing to trade their privacy in exchange for an improved user experience. This is a fundamental problem as collected private user information, including age, e-mail address, data (e-mails, documents, health care information) or preferences (ads and movies watched, items bought, places visited) is frequently sold and often used in ways that do not always coincide with the users’ best interests. One of my research interests is to reconcile the seemingly incompatible goals of preserving user privacy and providing ease of use for online services. The challenge consists in solving this problem for a wide variety of services while simultaneously preserving the service provider’s goals – ease of deployment and the ability to collect aggregate statistics over user interests. For instance, can a private ad watching experience be offered: Ad providers have assurances on the number of and attention paid by viewers but are not provided with individual viewer interests. Moreover, can a “free” map service be extended to provide private directions: individual user end-points of interest are not leaked, but providers can collect aggregate data (e.g., popular spots). I am planning to use the expertise I developed when working on private data and computation outsourcing as a starting point. I also intend to focus on the “usability” aspect of the problem. The user experience should be minimally changed, and required user feedback should not become a nuisance.

Private Cloud Computing: Major cloud services provide users with the ability to rent compute resources on which they can execute their outsourced computations. By scaling with user demand and by being elastic – users only pay for the resources used – this concept caters to users that have computations that by far exceed their hardware capabilities, e.g., web services outsourcers. Elasticity allows outsourcers to escape the traps of over/under provisioning their resources before launching the service and experiencing the real user loads. These advantages come today at a steep price – privacy. Malicious providers may use the outsourced code and their huge amount of resources, to finish the computations way ahead and perhaps use the results to their advantage. They may also replicate outsourced web services and provide them as their own, while denying service to the users of the original service providers. Defenses against such attacks go way beyond code watermarking as reversed engineered code can be re-implemented and offered as new. Code obfuscation techniques have produced so far mostly negative results. I am planning instead to investigate the use of trusted hardware

to approach this problem: attested hardware provided by the “cloud”, which outsourcers employ to run their code. This approach is likely to slow down the service and require significant investments on the part of the provider. Instead, the idea I would like to investigate is whether the trusted hardware can be used to run only efficient snippets of sensitive code, without which the overall codebase cannot be reverse engineered.

Outsourcing Access Control: In most distributed systems, access control decisions are made at a central site. While convenient for its simplicity this approach unduly increases the response times and introduces a single point of failure. The alternative solution, of outsourcing the access control structures to the “cloud”, introduces immediate privacy concerns. A good solution should ensure that the access control structures are not leaked to the cloud provider, while still allowing the remote sites to use them to verify access rights and help in enforcing access decisions. A research question I am interested in studying is whether the data outsourcing techniques I have previously explored can be applied or extended to solve this problem. A related question is whether a Role Based Access Control (RBAC) structure can be used to provide access to a private, outsourced data structure (e.g., ORAM). Access rights would be associated with items and users may be granted roles, giving them access to items. While this approach allows the owner to easily change the access rights granted to users, it introduces additional difficulties. For instance, the cloud provider should be oblivious to the RBAC structure even when the RBAC structure changes – the provider should not learn what roles users have, even when they are exercised.

Social Network Privacy: The popularity of online social networks (OSNs) has increased tremendously since their recent inception, with some sites exceeding half a billion users. While extensive research exists on properties of general social networks, the study of the privacy provided by such systems is still in its infancy. This has become a major issue and only recently OSN providers have started treating privacy more seriously. Keeping privacy as an afterthought when building any system can create significant problems. In particular, malicious users can easily crawl OSNs and collect large numbers of personal profiles. Such information can immediately be monetized, e.g., by selling collected e-mail addresses together with user context to spammers. High levels of privacy can easily be achieved by severely restricting access to the accounts of other users. However, such an approach would likely deter people from using OSNs, eventually turning to other, less restrictive sites. I am interested in studying the tradeoffs between privacy and usability that are achievable in OSNs. One intermediate step of this project is to compile an exhaustive list of privacy attacks and study practical methods to launch them on a large scale. An interesting result in itself is showing that such attacks can indeed improve the chance of collecting private profile information without effort. The goal of this work includes also devising a privacy framework for OSNs that carefully balances the human interaction required, ease of use, the privacy levels provided, and the work imposed on an attacker.

References

- [1] Bogdan Carbunar, Michael Pearce, Michael Needham, and Venu Vasudevan. Network aware caching for video on demand services. Document under submission.
- [2] Bogdan Carbunar, Michael Pearce, Shivajit Mohapatra, Loren J. Rittle, Venu Vasudevan, and Octavian Carbunar. Secure synchronization of periodic updates in ad hoc networks. *IEEE Trans. Parallel Distrib. Syst.*, 21(8):1060–1073, 2010.
- [3] Radu Sion and Bogdan Carbunar. On the practicality of private information retrieval. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2007.
- [4] Peter Williams, Radu Sion, and Bogdan Carbunar. Building castles out of mud: practical access pattern privacy and correctness on untrusted storage. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, pages 139–148, 2008.
- [5] Bogdan Carbunar and Radu Sion. Regulatory compliant oblivious ram. In *Proceedings of the Applied Cryptography and Network Security Conference (ACNS)*, pages 456–474, 2010.
- [6] Bogdan Carbunar and Radu Sion. Uncheatable reputation for distributed computation markets. In *Proceedings of Financial Cryptography and Data Security (FC)*, pages 96–110, 2006.
- [7] Bogdan Carbunar and Mahesh Tripunitara. Fair payments for outsourced computations. In *Proceedings of the 7th IEEE Sensor, Mesh and Ad Hoc Networks and Communications (SECON)*, 2010.
- [8] Yao Chen, Radu Sion, and Bogdan Carbunar. Xpay: practical anonymous payments for tor routing and other networked services. In *Proceedings of the 8th ACM workshop on Privacy in the electronic society (WPES)*, pages 41–50, 2009.