Deploying secure multi-party computation for financial data analysis[**]
(Extended version)[*]

Dan Bogdanov[1,2], Riivo Talviste[1,2,3], and Jan Willemson[1,3]

[1] Cybernetica, Akadeemia tee 21, 12618 Tallinn, Estonia
{dan,riivo,janwil}@cyber.ee
[2] University of Tartu, Institute of Computer Science, Liivi 2, 50409 Tartu, Estonia
[3] Software Technology and Applications Competence Centre, Akadeemia tee 15A,
Tallinn 12618, Estonia

Abstract. In this paper we describe a secure system for jointly collecting and analyzing financial data for a consortium of ICT companies. To guarantee each participant's privacy, we use secret sharing and secure multi-party computation (MPC) techniques. While MPC has been used to solve real-life problems beforehand, this is the first time where the actual MPC computation on real data was done over the internet with computing nodes spread geographically apart. We describe the system architecture, security considerations and implementation details. We also present the user feedback analysis revealing that secure multi-party computation techniques give sufficient assurance for data donors to submit their sensitive information, and act as a critical enabling feature for privacy-preserving data mining.

Keywords: financial data analysis, privacy-preserving data mining, secure multi-party computation

1 Introduction

In order to analyze the economic situation of an industrial sector, financial metrics must be collected from the companies and analyzed jointly. However, since this data is largely confidential, the process can not be carried out just by sending the data from one company to another.

There are numerous ways for maintaining the confidentiality of data in such cases, including anonymization and pseudonymization techniques or including

[*] This is an extended version of the paper presented at Financial Cryptography and Data Security 2012.

[**] This research was supported by the European Regional Development Fund through the Estonian Center of Excellence in Computer Science, EXCS, and the Software Technology and Applications Competence Centre, STACC; the European Social Fund's Doctoral Studies and Internationalisation Programme DoRa; the target funded theme SF0012708s06 and the Estonian Science Foundation, grant No. 8124.

a trusted third party. In this paper, we present how secret sharing and secure multi-party computation (MPC) techniques can be used to guarantee that confidential data is processed without compromising business secrets. The main idea of this research is the observation that the use of MPC distributes the role of a trusted third party among many parties so that none of them has to be trusted unconditionally. The most significant added value for the companies is that no single data value can be seen by a single outside party after it leaves the user's computer.

MPC has been studied for almost thirty years. Until recently it has been mostly academic work, because MPC protocols add a fair amount of computational and network communication overhead. However, in recent years, many MPC projects aim to be also usable in practice [9,10,2,8,14,16,5,11].

In this paper we describe a secure system for jointly collecting and analyzing financial data. While the system is easily usable in other similar scenarios, we concentrate on the case of collecting financial data for the Estonian Association of Information Technology and Telecommunications (officially abbreviated as ITL). ITL is a non-governmental non-profit organization with the primary goal of promoting co-operation between its members—Estonian companies engaging in the field of information and communication technology (ICT).

We built the data collection and analysis system on top of the Sharemind secure computation framework [8]. While MPC technology has been already previously used to solve real-life problems [9], then to the best of our knowledge, this is the first time where the actual secure multi-party function evaluation was done over a wide area network (the internet) using real data. Another significant contribution of our work is the development of a JavaScript library that can be used to turn an ordinary web-based questionnaire form with numerical or categorical answers into a secure input source for MPC with minimal effort.

1.1 Related work

In 2004, J. Feigenbaum et al. implemented a privacy-preserving version of the Taulbee Survey[4] using MPC [11]. Their implementation used secret sharing at the data source and two parties evaluating a Yao circuit over a wide area network. However, their implementation was never used with real data [12].

Secure multi-party computation was first used in a large-scale practical application in Denmark in 2008, when P. Bogetoft et al. implemented a secure double auction system that allowed Danish sugar beet farmers to trade contracts for their production on a nation-wide market [9]. For data submission, their system required each end user to download a special program to their computer. Similarly to the Sharemind framework, the Danish system used three secure computation servers. However, the shares of private data were not directly sent from the farmers' computers to the servers. Instead, each share was encrypted with a public key of one of the computation nodes and all the encrypted shares were then stored in a central database.

[4] Computing Research Association, Taulbee Survey, http://www.cra.org/statistics

In the data analysis phase, a representative of each computation node downloaded their corresponding shares from the central database and decrypted them using their private key. After that, the actual MPC process was performed in a local area network set up between the three computation nodes.

Other MPC frameworks that aim for practical use include the SEPIA library [10] and the SecureSCM project [2]. The SEPIA library is strongly optimized for privacy-preserving aggregation of multi-domain network data and is therefore capable of near real-time data analysis. SecureSCM concentrates on investigating how MPC technology can be used to build confidentiality-preserving supply chain management. Both libraries have also been tested in both local and wide area networks. However, to the best of authors' knowledge, neither of them has been deployed to solve real life problems for continuous use.

Fully homomorphic encryption is yet another way to analyze data in a privacy-preserving manner. Unfortunately, the current implementations [18,13] have key and ciphertext sizes and execution times that are nowhere near suitable to be used in real-life applications.

2 Preliminaries

Secure multi-party computation is a technique for evaluating a function with multiple parties so that each of them learns the output value but not each other's inputs. There are various ways for implementing secure MPC with a different number of nodes and various security guarantees. In this work, we concentrate on systems based on secret sharing (also called share computing systems).

Share computing systems use the concept of secret sharing introduced by Blakley [7] and Shamir [17]. In secret sharing, a secret value s is split into a number of shares s_1, s_2, . . ., s_n that are distributed among the parties. Depending on the type of scheme used, the original value can be reconstructed only if the shares belonging to some predefined sets of parties are known. For example, in a t-threshold setting, any group of t or more parties can combine their shares to reconstruct the original value. However, the result of combining less than t shares provides no information about the value they represent.

Secure multi-party computation protocols can be used to process secret shared data. These protocols take secret shared values as inputs and output a secret shared result that can be used in further computations.

2.1 Sharemind

Sharemind [8] is a distributed virtual machine for performing privacy-preserving computations. The Sharemind framework can perform various operations on secret shared 32-bit integers, vectors of 32-bit integers and booleans.

The framework allows the developer to write algorithms where public and private data are separated. The Sharemind virtual machine guarantees that private data is not leaked while such an algorithm is evaluated.

The Sharemind system uses three servers to hold the shares of secret values. In Sharemind terminology, these servers are *data miners*. The miners are connected with each other over the network using secure channels and use secure MPC protocols to evaluate a function on the secret shared data. The Sharemind computation protocols are provably secure in the *honest-but-curious* model with no more than one corrupted party. The honest-but-curious model means that security is preserved when a malicious miner attempts to use the values it sees to deduce the secret input values of all the parties without deviating from the protocol.

Secret sharing of private data is performed at the source and each share is sent to a different miner over a secure channel. This guarantees that no-one except the data owner will know the original value. Sharemind uses additive secret sharing scheme in the ring $\mathbb{Z}_{2^{32}}$ as this allows it to support the efficient 32-bit integer data type.

2.2 The development process of Sharemind applications

Creating applications with the Sharemind framework involves three main steps. First, we have to find three independent parties who will host the miner servers. Each of those hosts has to set up a server and install Sharemind miner software on it. In a distributed data collection and analysis scenario, it is possible to select the parties from the organizations involved in the process.

Second, we have to develop the necessary data mining applications that take advantage of the privacy-preserving guarantees that the Sharemind framework provides. Sharemind has a low-level assembly language that the virtual machine can execute. As implementing an algorithm in low-level language is tedious and error-prone, the framework also provides the developers with a more high-level programming language called SecreC. SecreC [15] is a high-level language with a C-like syntax that is capable of separating public and private data flows. It means that the public computations are done in an ordinary manner, while private computations involving sensitive information (shares of secret values) are evaluated using secure MPC protocols. SecreC applications are compiled into Sharemind assembly, which is then given to each Sharemind miner and that can be then executed by the Sharemind virtual machine.

The use of a separate programming language to represent the data mining algorithm allows the miner hosts to validate the code that processes confidential data before it is executed. This is especially important since SecreC programs also control which results are published in the reports. For better security, only the final values should be published and the code files should be distributed securely during miner server setup.

In the third step, we need to use the Sharemind controller library to build end-user applications. These applications are used to insert the data into the Sharemind miners, run analysis on that data and also generate the required reports. These applications are made available to the end users—both for data entry and report generation.

3 The application scenario

In Estonia, the Ministry of Economic Affairs and Communications publishes an economic report every year, combined from all of the annual reports of Estonian companies. However, while this report is accurate and gives a detailed overview of the country's economic situation, it is only compiled once a year and by the time it is published, the data is already more than half a year old.

Since ICT is a rapidly evolving economic sector, ITL members would like to get more up-to-date information about the sector to make better business decisions. There is an initiative within ITL that ITL itself, as a consortium, should collect some basic financial data from its members twice a year and publish them as anonymized benchmarking results for its members. As the collected data does not have to be audited, the data collection periods can be shorter, which means that the published benchmarking results will be up-to-date.

During the interviews conducted with ITL representatives, they described a possible solution they had imagined. This was to deploy a typical anonymization service that strips the identities from the data. According to this plan, ITL should collect these financial indicators with the following frequency:

Indicator	**Collected**
total return	annually and semi-annually
number of employees	annually and semi-annually
percentage of export	annually and semi-annually
added value	annually and semi-annually
labour costs	annually
training costs	annually
profit	annually

After each collection period, the data set would be anonymized (i.e. the company identifiers are removed) and each indicator would be sorted independently to reduce the risk of identifying some companies by just looking at a set of financial indicators. For example, combining total return, number of employees and profit, it could be easy to identify some ICT companies. However, when sorting by each indicator independently, a company that is the first when sorted by one indicator might not be the first when sorted by another indicator.

Sorting the collected data by each indicator separately gives us a slightly stronger privacy guarantee than just stripping away the identifying information. However, as seen on Figure 1, all of the collected data is still accessible by ITL board, which consists of the leaders of competing ICT companies. This means that ITL members (including the board members themselves) must trust the ITL board not to misuse or leak the collected information. Consequently, ITL member companies might be reluctant to participate and give away their sensitive economic information, as it can be seen by their competitors. ITL members are required to trust the board with their data and this is quite a significant assumption.

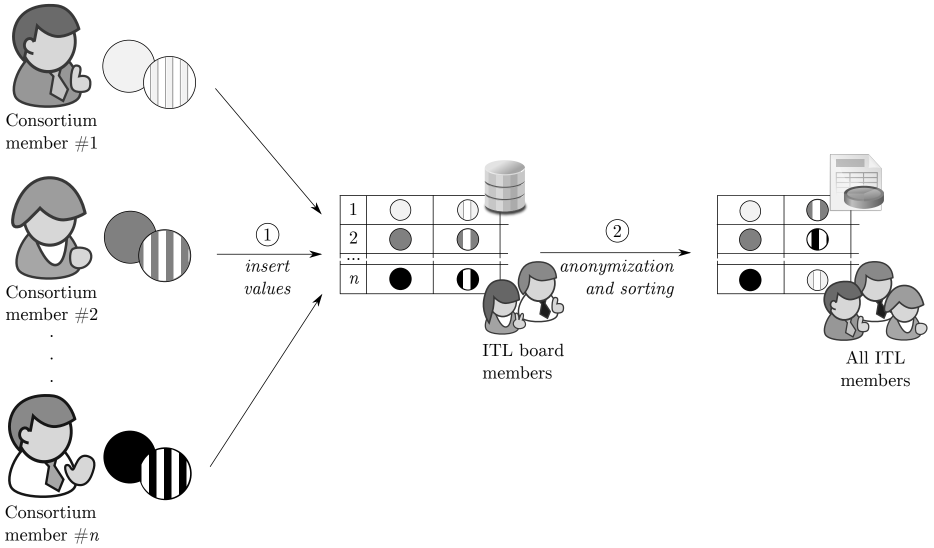


Fig. 1: Data flow and visibility in the initial proposed solution.

4 Reducing the trust requirements

To address the shortcomings of the initial solution, we proposed to use the Sharemind framework to collect and analyze the financial data. That way, all of the collected sensitive information is secret-shared at the source and distributed among the three Sharemind data miners. This gives us the additional benefit that no single party has access to the original values and lowers the risk of anyone misusing the collected data. Also, we have a much lower threat of insider attacks and unintentional disclosures (e.g. compromise of economic data by a leaked backup). Most importantly, the use of MPC reduces the trust that ITL members need to have towards any single party.

The idea of using MPC in this scenario is simple. After the data has been collected from all of the members, three data miners engage in secure MPC protocols and sort all the collected economic indicators independently. These sorted indicators are then published as a spreadsheet and made accessible to the board members of ITL. The board will then either give these spreadsheets directly to all of the members or first compute some aggregate values and/or charts and give this edited report to the members.

Making the spreadsheet with the sorted values initially available only to the board members is a procedural decision that allows the board to tailor the presentation options and provide comments. However, we stress that even the board members will not see any identifying information, as this is removed while sorting the collected values. Hence, ITL members do not have to trust the board members not to misuse their sensitive information and are hopefully more prone to participate in the data collection process. The latter is the main advantage and the critical enabling component of the described solution using the Sharemind framework over the initial solution using only anonymization techniques.

The data flow and visibility to different parties for this solution is shown on Figure 2.

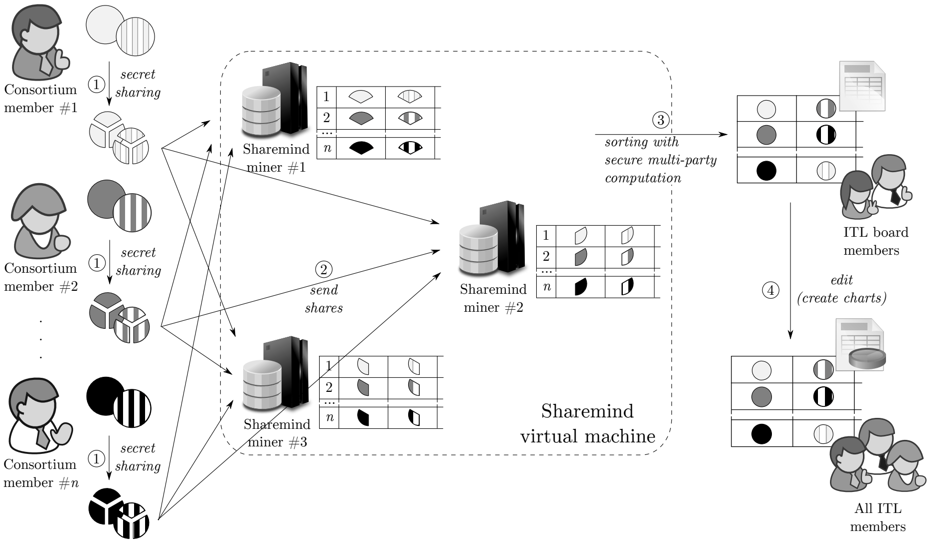


Fig. 2: Data flow and visibility in the improved solution using the Sharemind framework.

The use of a general MPC framework such as Sharemind is beneficial in this scenario, as new reports can be added with relative ease. When the data has been collected and stored, Sharemind acts like a database and application server that can perform simple operations like sorting and ratio computation, but can also be extended in the future to perform trend computations and data mining. After a couple of successful data collection periods, when the ITL board has agreed on the analyses to be done, we can implement these analyses with the secure MPC protocols in the Sharemind framework. In this case, we would not have to disclose the sorted data vectors with the original values anymore. Instead, the MPC protocols will compute the statistics required for the report and publish only the aggregated values. This can be done effectively by using the SecreC language to implement the required algorithms and feeding the results into a report generation system.

It is also important to mention that the ITL board has considered the possibility that some of ITL member companies could submit false financial information which would invalidate the calculated statistical results. However, the ITL board feels that such risk is very low, as ITL members themselves are interested in getting correct benchmarking data. Furthermore, generated reports with sorted indicator vectors are given to ITL board members who can easily spot any outliers and remove them if necessary. In the future, when we add more complex data analysis capabilities to the system, we can implement and use a privacy-preserving outlier detection to filter out potentially malicious data.

The described solution was proposed to the ITL board. The board accepted the proposal and we developed the necessary applications. In the following sections we will describe the architecture, its components and their deployment in more detail.

5 The ITL secure data aggregation system

5.1 Deployment

In the deployed application, the three Sharemind miners are hosted by three companies—Cybernetica, Microlink and Zone Media, all of which are Estonian ICT companies and also members of ITL. Choosing the miner hosts among the consortium members fulfills the following requirements set for the data miners:

1. They are motivated to host the miners, as this project would also be beneficial for themselves.
2. They are independent and will not collude with each other as they are also inserting their own data into the system and want to keep it private.
3. Also, ITL members act in the field of information technology, thus they have the necessary infrastructure and competence to host a server that runs the Sharemind miner software.

Both Microlink and Zone Media set up a virtual machine in their environments and Cybernetica used one of its public servers to host the data miner. In addition to the Sharemind miner instance, each of those servers also has an installation of a web server together with a web-based data collection interface, database backend and a proxy application. This is required as we have a web-based data submission form (see Section 5.2 for details) which distributes the shares of secret shared data using the HTTPS protocol. However, the current version of Sharemind miner does not have a web interface and thus cannot receive the shares directly from the submission form. To overcome the problem, a simple web application at each host receives the shares from the data submission form and saves them to a local buffer database. After the data collection period has ended, and before the secure MPC protocols are executed, a proxy application transfers the shares from the local buffer database to the local miner's internal database.

As the miner hosts provided their servers with no cost, they wished to reduce the effort needed to maintain the servers. Thus, all of the three miner hosts were set up by a single administrator who also regularly executes the computations. Ideally, each host should be maintained by its respective owner and this should be a rule in all future deployments of the technology. We consider it an important challenge to reduce the administrative attention required for managing a Sharemind miner to a minimum as this makes miner host selection easier and makes the technology easier to deploy in practice.

5.2 Securing web-based data collection

ITL requested that the online financial data submission form has to be integrated into their web page that already had both public pages and a member area. This way, the representatives of ITL member companies can access everything related to ITL from one place and the environment is also more familiar. Moreover, it allows us to reuse the authentication mechanisms of the ITL web page without implementing and deploying one ourselves. Thus, the users can access the submission form with the credentials they already have.

For the purpose of making web-based privacy-preserving data collections easier, we have developed a JavaScript library that can be used to turn a basic HTML form into an input source for secure MPC applications with minimal effort. The JavaScript library [19] handles everything from secret sharing of the user-entered data in the web browser to distributing the shares among the three miner hosts.

When initialized, the library first contacts all of the three miner hosts and asks each one of them for 256-bit vector of randomness. These random vectors are then XOR-ed together in the user's web browser and the result is used to initialize a pseudo-random number generator. This way, the JavaScript application has access to a good entropy pool available to the web server and also does not depend on just one entropy source. We use AES in counter mode to set up a cryptographically secure pseudo-random number generator for the purpose of secret sharing [1]. While communicating with the miner hosts, the JavaScript library automatically overcomes the Same Origin Policy enforced in web browsers. It uses either a generic solution of dynamically adding a HTML script tag into the web page source; or the HTML5 cross-document messaging API[5] available in newer web browsers to load the randomness from and send answer shares to the three remote domains.

Security The representatives of an ITL member company can log in to ITL web page member area over an HTTPS connection using either their credentials (username and password) or even more securely, using the Estonian ID-card or Mobile-ID.

The shares of financial data are also distributed among the miners using HTTPS connections. For this, each miner host server requested a SSL certificate for its web server. Cybernetica and Microlink requested their certificates from StartCom Ltd., while Zone Media got its certificate from GeoTrust, Inc. The root certificates of these Certification Authorities are installed in most web browsers, making the deployment of the application easier. If the data collection needs to be limited, a special Certification Authority can be used to ensure that.

To make sure that only representatives of ITL member companies are able to send shares to the miners, we use access tokens. As shown in Figure 3, a random

[5] HTML5 Web Messaging, W3C Working Draft 17 March 2011, http://www.w3.org/TR/webmessaging/

access token[6] is generated by the ITL web server and sent together with the form each time the financial data submission form is requested by one of the logged-in users. The JavaScript library used in the submission form sends this token together with the corresponding shares and other submission data to each miner. Before saving the received shares into the buffer database, the miner contacts the ITL web server and confirms that this token was really generated for the current submission form, the current company and has never been used for any submission before. The latter means that access tokens also act as nonces to rule out any replay attacks. All the communication between a miner and the ITL web server is done over the HTTPS protocol and a unique, previously agreed and pre-configured passphrase is used to identify each miner to the ITL web server. If a miner receives a positive reply from the ITL web server, it saves the received shares to its local buffer database and notifies the submission form. If the latter receives these notifications from all three miners, it marks this submission form as "submitted" in the ITL web server. This also invalidates the used nonce.

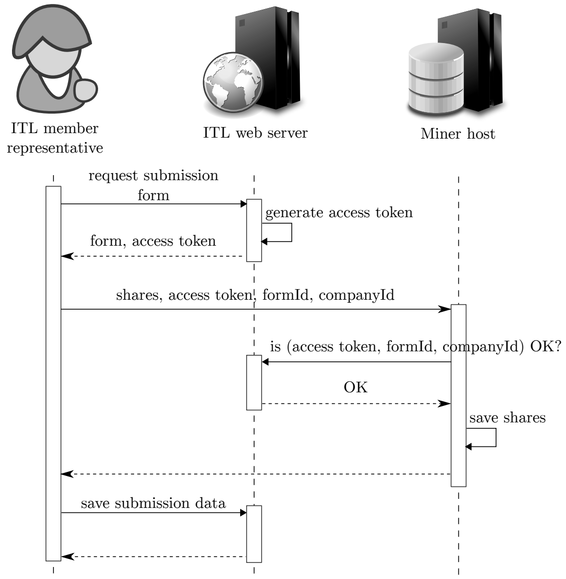


Fig. 3: Submitting shares of financial data. Communication between the user's computer, ITL web server and a miner.

[6] We use two 31-bit randomly generated numbers for access tokens. This is due to the fact that the PHP language that is used by the ITL web server does not have a 32-bit unsigned integer datatype. Thus, on 32-bit systems we can generate non-negative numbers up to $2^{31} - 1$. So for increased security, we use two 31-bit values instead on just one.

5.3 Maintaining confidentiality during data analysis

After the data collection period has ended, the proxy applications at each miner host synchronously copy the shares from the buffer database to the SHAREMIND miner's internal database and secure MPC protocols can be started.

Each SHAREMIND miner has a copy of a SECREC script that loads shares from the miner's database and uses a secure MPC implementation of an oblivious Batcher's odd-even merge sorting network [4] to sort the underlying private data vector. All of the collected financial indicator vectors are sorted separately in that manner and the results are published on the ITL web page member area for the ITL board members as an Excel spreadsheet. After reviewing the results, the board forwards this report to all other ITL members.

Security The SHAREMIND framework uses the RakNet library[7] for its network layer. The RakNet library provides a possibility to encrypt connections between the peers using efficient 256-bit elliptic curve key agreement and the ChaCha stream cipher [6]. While the latter choice is not standard, the best known attacks against ChaCha are still infeasible in practice [3], hence the used combination provides high-performance secure channels. These are critical for MPC protocols since the performance of share computing protocols depends mostly on the communication efficiency between the computing parties.

This technique is used to encrypt all the communication between the SHAREMIND miners as well as between the miners and the controller applications (e.g. proxies and analysis applications). To use encryption, each miner host has to generate a key pair for its SHAREMIND miner and send its public key to the other two miner hosts. The same has to be done for the proxy applications. Again, generating the key pairs and securely distributing public keys between the miner hosts is a responsibility of each miner's administrator. However, since this is a one-time procedure during system setup, we do not consider it an administrative burden.

6 Secure financial statistics in practice

ITL uses the financial data submission system to collect the indicators twice a year:

- At the beginning of a new calendar year, ITL members have a 45 day period to fill in two forms: one concerning financial data for the whole previous calendar year containing all seven financial indicators; and another one for the second half of the previous calendar year, containing four indicators (see Section 3).
- In the third quarter, ITL members have 30 days to fill in a form concerning financial data for the first half of the current year, asking for four indicator values (see Section 3).

[7] RakNet – Multiplayer game network engine, http://www.jenkinssoftware.com

The described solution using the Sharemind framework was deployed in the beginning of 2011 and has been already used to collect financial data concerning three periods: the whole year of 2010, the second half of 2010 and the first half of 2011.

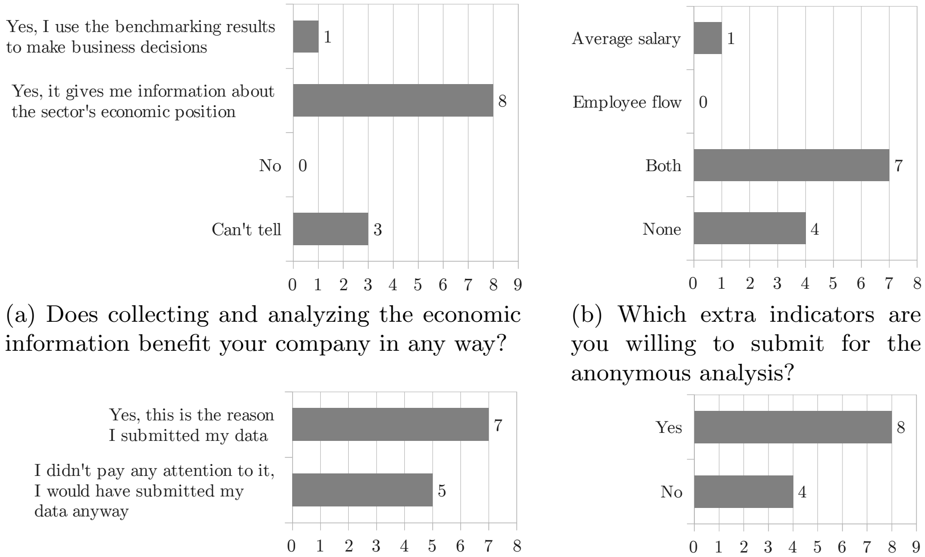
After both data collection periods, we used secure MPC protocols to sort each financial indicator vector independently and published the results as a spreadsheet for the ITL board. In addition to this, the ITL board requested a few extra reports. A list of the analyses performed on the collected financial data, together with the required computational routines, are listed in Table 1.

Analysis operation	**Required MPC primitives**
Sorting each financial indicator vector.	Oblivious sorting algorithm using a sorting network. Requires multiplication, addition and comparison.
Privacy-preserving filtering to keep only the data values that were really submitted by the end user.	Casting boolean to integer, vector multiplication.
Calculating a new composite indicator, *added value per employee.*	Division of secret shared values.
Time series for each financial indicator over all of the three forms.	Sorting the columns in a secret shared matrix by the values in one of the rows.

Table 1: A list of analyses performed on the collected financial data, together with the corresponding required secure MPC primitives and algorithms.

Implementing those extra requests from the ITL board was relatively effortless as we had all of the individual indicator values available in secret shared form and were able to implement and run new algorithms without having to recollect the financial data. This also justifies the choice to use a general-purpose secure MPC framework.

Together with the second data collection period, we also conducted a survey among ITL members, asking about the motivation and possible privacy issues of participating in such a data collection system. While the number of responders is not large enough to draw statistically significant conclusions, they still cover the most important players in the Estonian ICT market. As seen in Figure 4a, most of the participants feel that collecting and analyzing the sector's financial indicators is beneficial for themselves in one way or another. We can also see that most of the participants are concerned about their privacy as they familiarized themselves with the security measures taken to protect the privacy of the collected data (Figure 4d) and about half of the participants submitted their data only because they felt that the system is secure in that matter (Figure 4c). The fact that most of the participants are willing to submit even more indicators (see Figure 4b) shows once more that ITL members are pleased with the security measures employed in this system to protect the participants' privacy.



(a) Does collecting and analyzing the economic information benefit your company in any way?

(b) Which extra indicators are you willing to submit for the anonymous analysis?

(c) Did the explanation of applied security measures make it easier for you to submit your sensitive information?

(d) Did you familiarize yourself with the provided materials that explained which security measures were taken to protect the sensitive information?

Fig. 4: Results from the feedback questionnaire.

7 Conclusions and Future Work

In this paper we describe a solution of how to securely collect and analyze financial data for a consortium of ICT companies. As we are dealing with sensitive data, companies are usually reluctant to disclose their financial indicators, as it is difficult for them to trust the parties who have access to their data for the purpose of analyzing it. To solve this problem, we use the secure MPC technology, so the companies do not have to trust any one party unconditionally and their sensitive data stays private throughout the analysis process.

The described solution was implemented and deployed in the beginning of 2011 and is in continuous use. It has already been used to collect and analyze data for three periods. To the best of our knowledge, this is also the first practical secure MPC application where the computation nodes are in separate geographic locations and the actual MPC protocol is run on real data over the internet.

A survey conducted together with one of the collection periods shows that ICT companies are indeed concerned about the privacy of their sensitive data and using secure MPC technology gives them enough confidence to actually participate in the collective sector analysis process. Moreover, thanks to the increased security and privacy measures, many companies are also willing to submit some extra indicators during the data collection process in the future.

Based on the experience of the ITL financial statistics application we conclude that MPC-based applications can be successfully deployed for real-life problems. Performance of the available implementations is no more a bottleneck, but more effort needs to be put into making application deployment and administration easier. Our current setup works over open internet, but still assumes relatively well controlled environment for the miner hosts. The next logical step is to study the challenges arising from cloud-based installations, and this remains a subject for future developments.

References

1. NIST SP 800-90, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, 2007.
2. SecureSCM. Technical report D9.1: Secure Computation Models and Frameworks. http://www.securescm.org, July 2008.
3. Jean-Philippe Aumasson, Simon Fischer, Shahram Khazaei, Willi Meier, and Christian Rechberger. New Features of Latin Dances: Analysis of Salsa, ChaCha, and Rumba. In Kaisa Nyberg, editor, *Fast Software Encryption*, volume 5086 of *Lecture Notes in Computer Science*, pages 470–488. Springer Berlin / Heidelberg, 2008.
4. K. E. Batcher. Sorting networks and their applications. In *Proceedings of the April 30–May 2, 1968, spring joint computer conference*, AFIPS '68 (Spring), pages 307–314, New York, NY, USA, 1968. ACM.
5. Assaf Ben-David, Noam Nisan, and Benny Pinkas. FairplayMP: a system for secure multi-party computation. In *CCS '08: Proceedings of the 15th ACM conference on Computer and communications security*, pages 257–266, New York, NY, USA, 2008. ACM.
6. D.J. Bernstein. ChaCha, a variant of Salsa20. *http://cr.yp.to/chacha.html*, 2008.
7. G.R. Blakley. Safeguarding cryptographic keys. In *Proceedings of the 1979 AFIPS National Computer Conference*, pages 313–317, Monval, NJ, USA, 1979. AFIPS Press.
8. Dan Bogdanov, Sven Laur, and Jan Willemson. Sharemind: A Framework for Fast Privacy-Preserving Computations. In Sushil Jajodia and Javier Lopez, editors, *Computer Security – ESORICS 2008*, volume 5283 of *Lecture Notes in Computer Science*, pages 192–206. Springer Berlin / Heidelberg, 2008.
9. Peter Bogetoft, Dan Christensen, Ivan Damgård, Martin Geisler, Thomas Jakobsen, Mikkel Krøigaard, Janus Nielsen, Jesper Nielsen, Kurt Nielsen, Jakob Pagter, Michael Schwartzbach, and Tomas Toft. Secure multiparty computation goes live. In Roger Dingledine and Philippe Golle, editors, *Financial Cryptography and Data Security*, volume 5628 of *Lecture Notes in Computer Science*, pages 325–343. Springer Berlin / Heidelberg, 2009. 10.1007/978-3-642-03549-4_20.
10. Martin Burkhart, Mario Strasser, Dilip Many, and Xenofontas Dimitropoulos. SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics. In *USENIX Security Symposium*, pages 223–239, Washington, DC, USA, 2010.
11. J. Feigenbaum, B. Pinkas, R. Ryger, and F. Saint-Jean. Secure computation of surveys. In *EU Workshop on Secure Multiparty Protocols*, 2004.
12. J. Feigenbaum, B. Pinkas, R. Ryger, and F. Saint-Jean. Some requirements for adoption of privacy-preserving data mining. *PORTIA Project White Paper*, 2005.

13. Craig Gentry and Shai Halevi. Implementing Gentry's Fully-Homomorphic Encryption Scheme. In Kenneth Paterson, editor, *Advances in Cryptology – EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 129–148. Springer Berlin / Heidelberg, 2011.
14. Wilko Henecka, Stefan Kögl, Ahmad-Reza Sadeghi, Thomas Schneider, and Immo Wehrenberg. TASTY: tool for automating secure two-party computations. In *CCS '10: Proceedings of the 17th ACM conference on Computer and communications security*, pages 451–462, New York, NY, USA, 2010. ACM.
15. Roman Jagomägis. SecreC: a Privacy-Aware Programming Language with Applications in Data Mining. Master's thesis, Institute of Computer Science, University of Tartu, 2010.
16. Lior Malka and Jonathan Katz. VMCrypt - modular software architecture for scalable secure computation. Cryptology ePrint Archive, Report 2010/584, 2010. http://eprint.iacr.org/.
17. Adi Shamir. How to share a secret. *Commun. ACM*, 22:612–613, November 1979.
18. Nigel Smart and Fre Vercauteren. Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. In Phong Nguyen and David Pointcheval, editors, *Public Key Cryptography – PKC 2010*, volume 6056 of *Lecture Notes in Computer Science*, pages 420–443. Springer Berlin / Heidelberg, 2010.
19. Riivo Talviste. Deploying secure multiparty computation for joint data analysis — a case study. Master's thesis, Institute of Computer Science, University of Tartu, 2011.