A prototype of an online privacy-preserving questionnaire system

Riivo Talviste

June 12, 2010

Theory Days in Elva

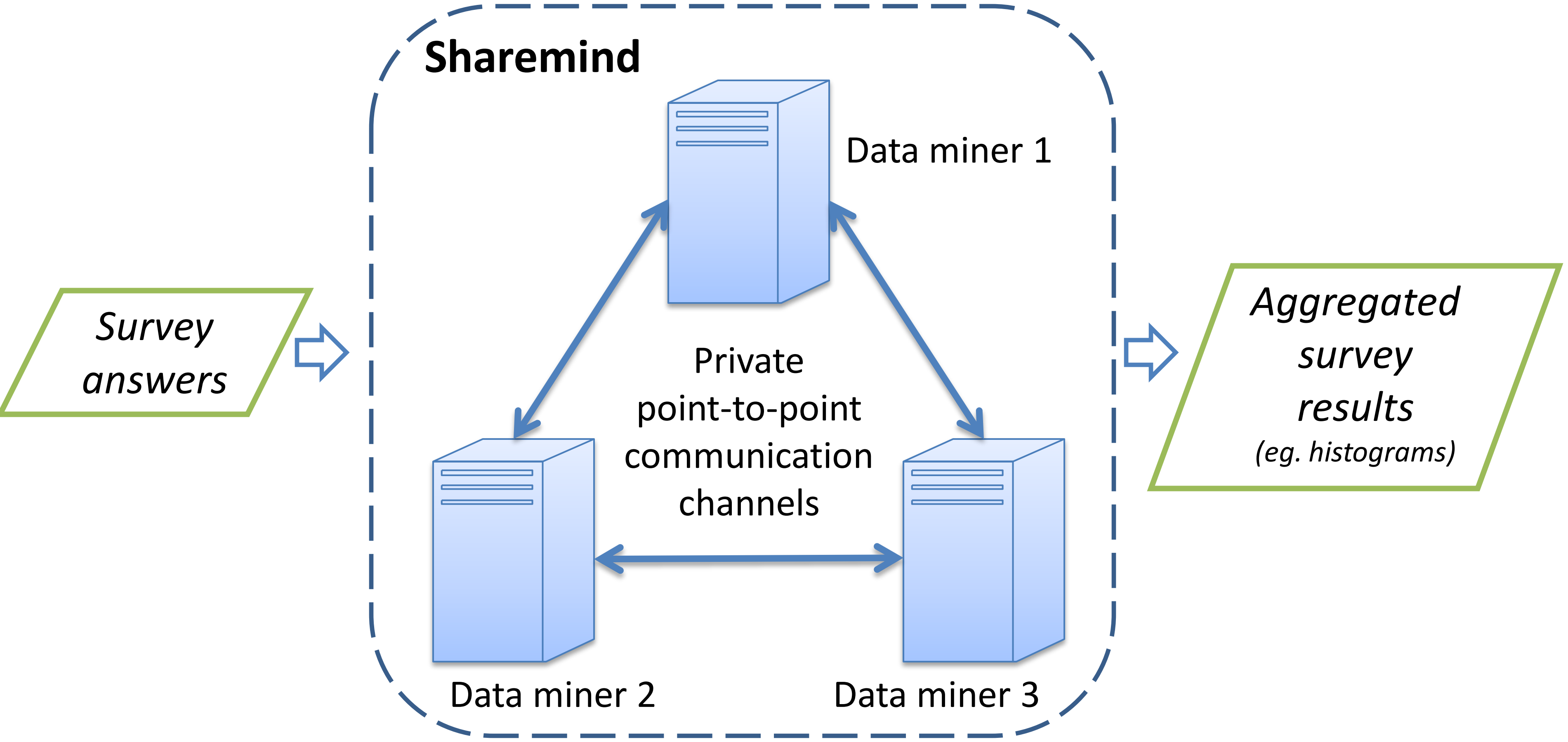
The project

- An ongoing project at STACC
- Develop an online questionnaire system that preserves participants’ privacy
  - i.e. Original answers do not leave user’s computer
- Use the Sharemind framework for privacy-preserving computations

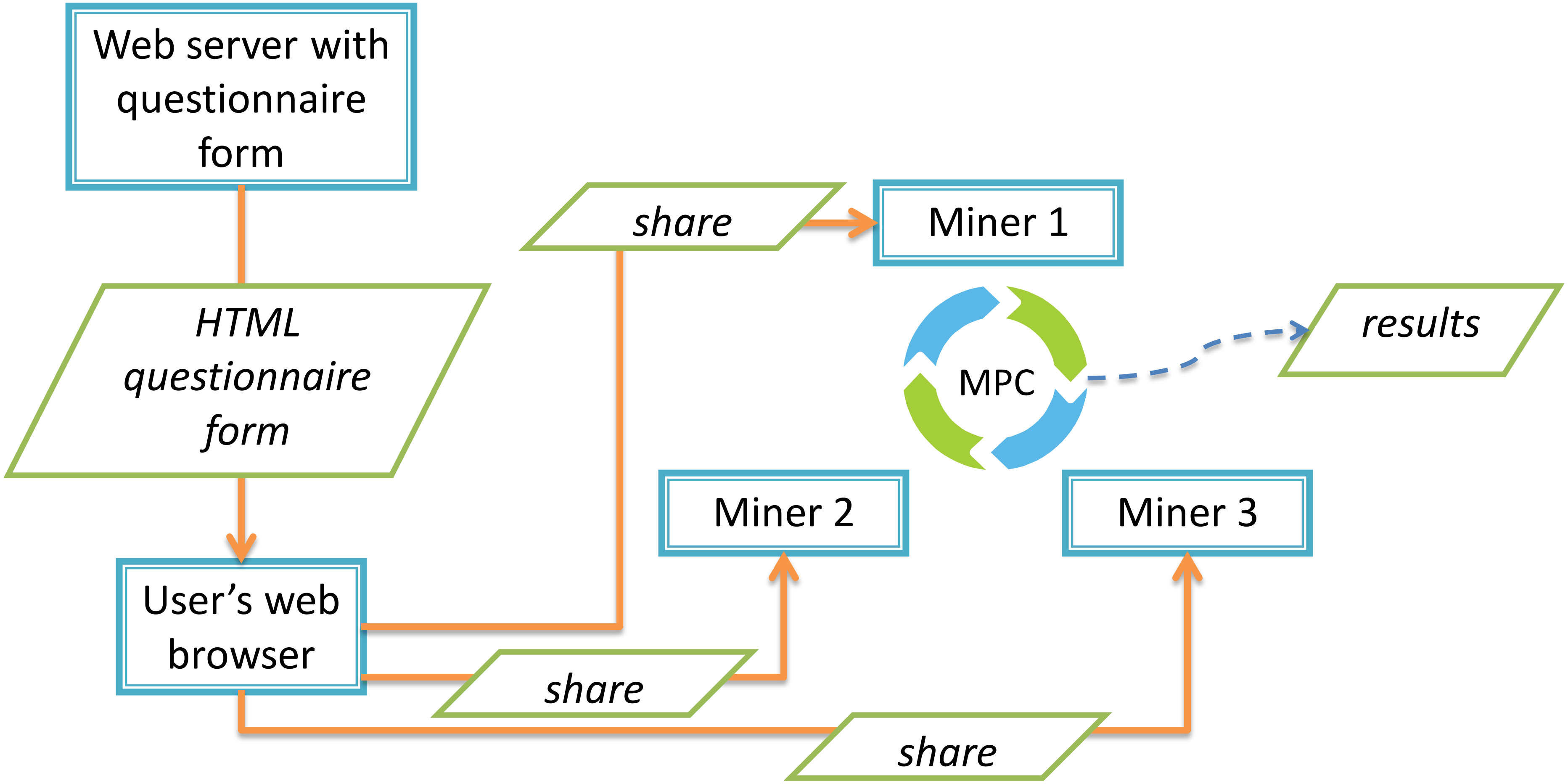
What is Sharemind?

- Distributed virtual machine for privacy-preserving computations
- Uses secure multiparty computation and additive secret sharing scheme
- Three **independent** miners
- API for controller application

What is Sharemind?



Architecture: the idea

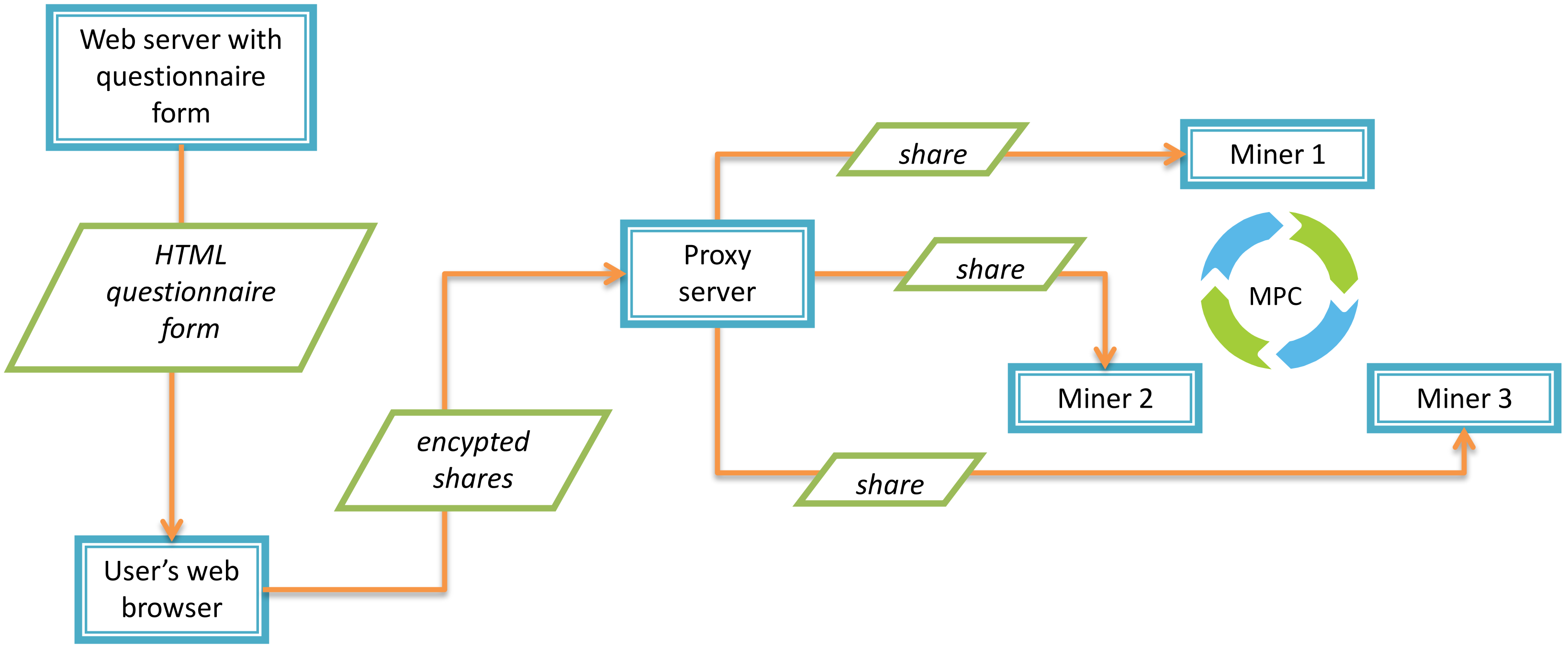


Previous work (1)

- Using Adobe Flex
  - + Direct HTTPS connections to miners
  - + Miners trusted by their SSL certificates
  - – Not supported on all platforms

Previous work (2)

- Using JavaScript
  - Same Origin Policy
  - Have to use proxy server and encrypted shares
    - Proxy server has to be in the same domain



This project

- Flex security with JavaScript
- Use “script tag hack”
  - <script>, <img> and other resources are not restricted by Same Origin Policy

Script tag hack

- How?
  - Dynamically add a new <script> element to (X)HTML DOM tree
  - The new script is automatically loaded and executed
  - It has to be valid JavaScript
- Use JavaScript Object Notation (JSON)
  - To send/receive arbitrary data
- <script> has no state like XMLHttpRequest
  - How do we know when the script is loaded?

Response format (1)

- Server sends response:

```
var someVariable = <JSON-encoded data>;
```

- Client periodically checks:

```
if (someVariable != undefined) {
  // Requested file is loaded, do something
  ...
}
```

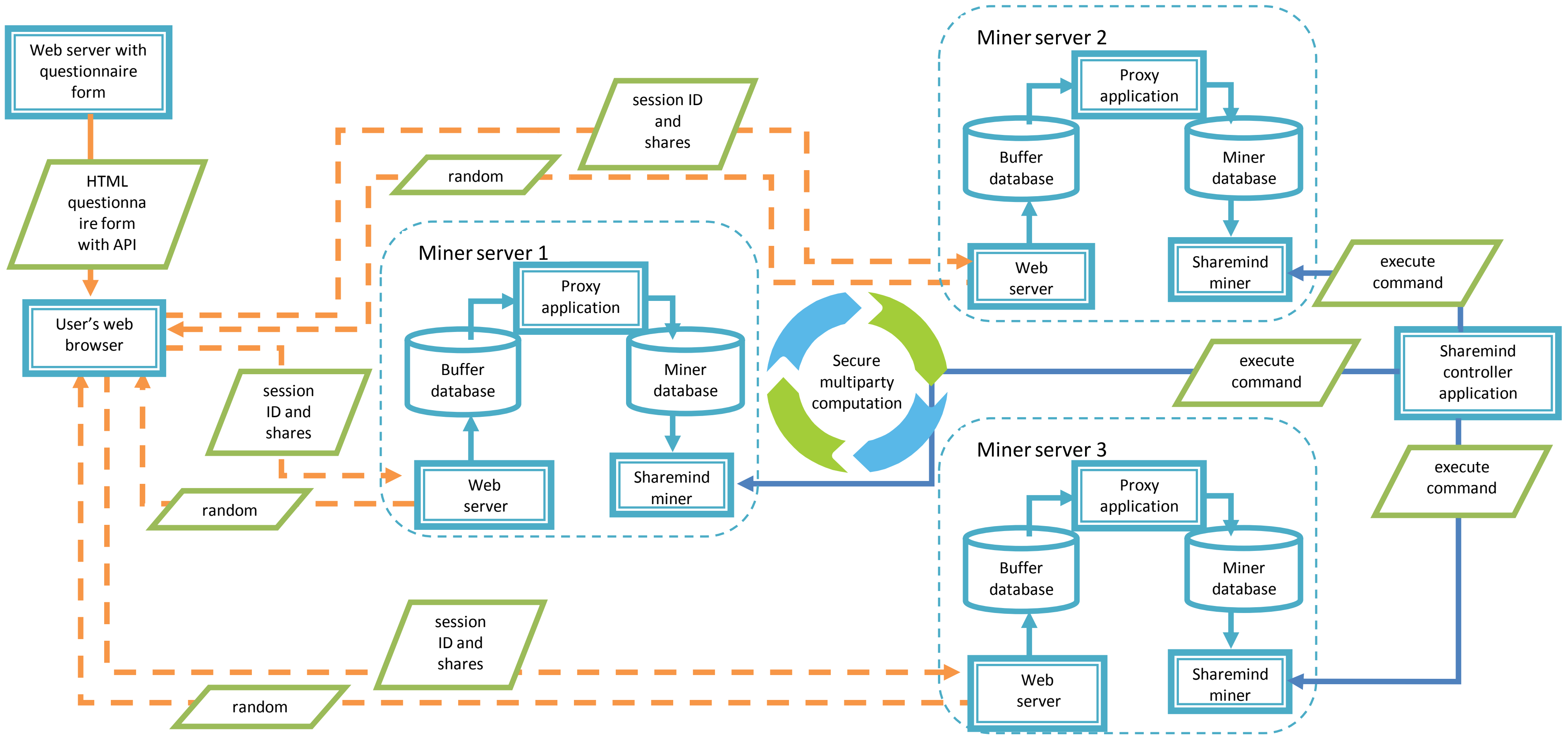
Response format (2)

- Server sends response:

```
callback(<JSON-encoded data>);
```

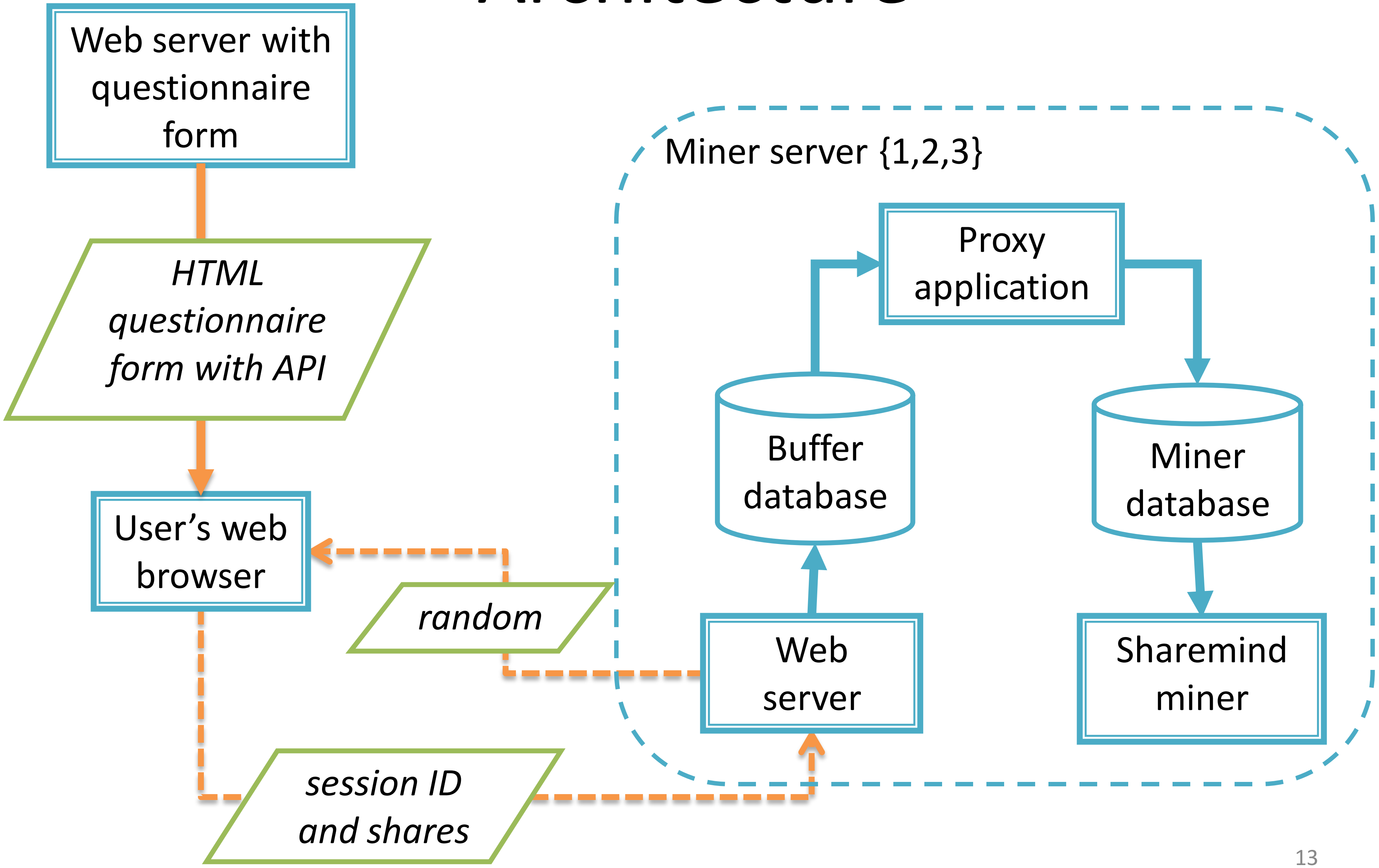
- Client automatically executes predefined callback() function with received data
- This is called JSON with padding (JSONP)

Architecture



▬ (blue)	UDP connection
▬ (orange)	HTTPS connection
╍ (orange dashed)	HTTPS connection with script tag hack and JSONP

Architecture

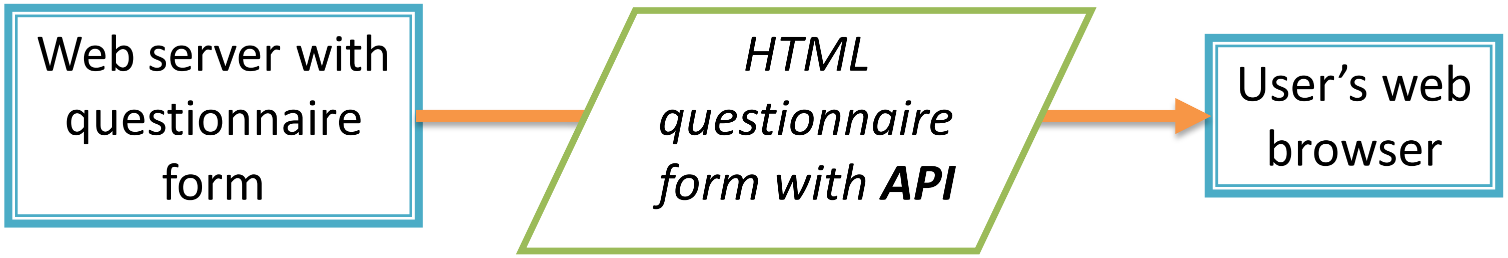


Dealing with secure data

- Use HTTPS to send/receive data
- Miner's web server certificates have to be trusted beforehand
  - Otherwise loading the script fails silently with timeout
  - This is a browser limitation

API

- Written in JavaScript
- Uses Dojo Toolkit implementation of the script tag hack
- Handles everything from collecting random to secret sharing and distributing the shares
- Needs a secure random number generator


Web server with questionnaire form
HTML questionnaire form with ***API***
User's web browser

Randomness in web applications

- Math.random()
- Java VM has access to OS entropy pool via java.security.SecureRandom
- Microsoft Silverlight also has access to OS entropy pool via RNGCryptoServiceProvider in .NET framework
- JavaScript and Adobe Flex lack this option

Randomness in JavaScript (1)

- Server with the web application also sends some random data with it
- Could use that random directly or as a seed for some PRNG
- Risk: server knows all the random values

Randomness in JavaScript (2)

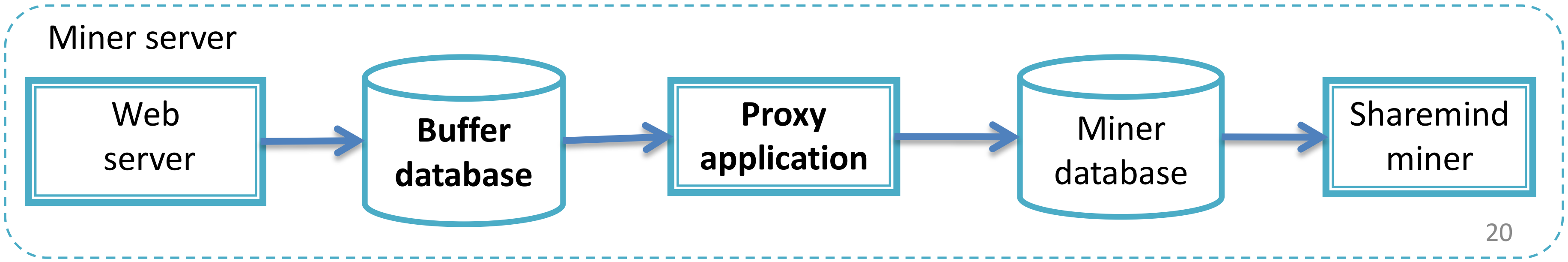
- Ask random data from the miners.
  - They must be independent anyway
  - We do it using the same script tag hack over HTTPS
- XOR the received random values together
- Use this combined value to initialize AES in counter (CTR) mode
- Split the resulting ciphertexts in 32-bit chunks

Randomness in JavaScript (3)

- There is also a possibility to collect randomness from user
- Some JavaScript crypto libraries implement this
  - jsCrypto collects entropy from user mouse movements, among other sources

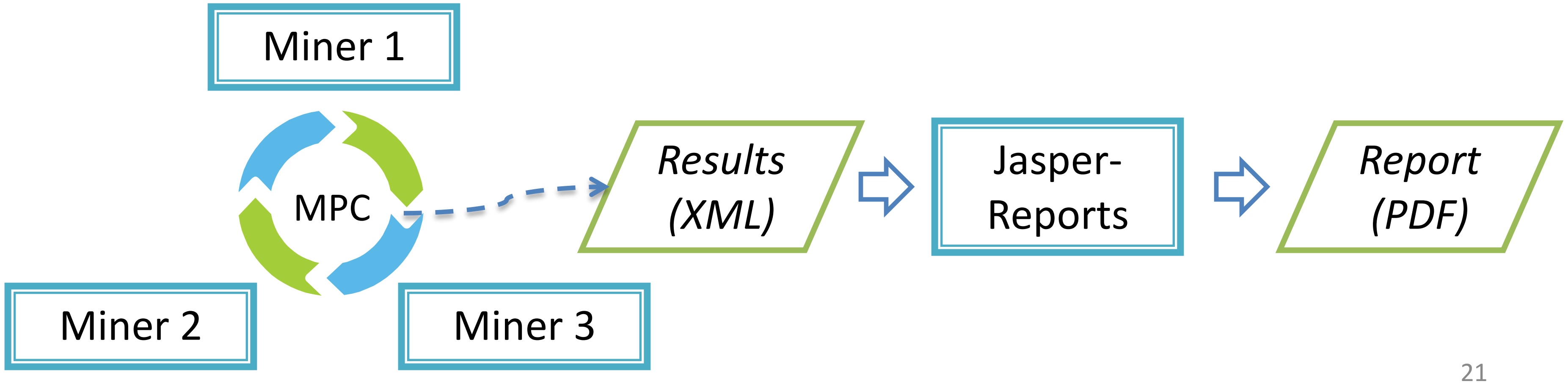
Buffer database and Proxy application

- Miner buffer database used for simplicity and robustness
- Questionnaire application also sends a random 32-bit session identifier with the answers
- WebControllerProxy uses that ID to synchronize the records of the three miners



Privacy-preserving data aggregation

- Sharemind miners periodically compute histogram for each question
- Output is in XML format
- JasperReports is used to generate a PDF with bar plots based on that data


Miner 1
MPC
Miner 2
Miner 3
Results (XML)
Jasper-Reports
Report (PDF)