Ben Antonellis, Gabe Smith, Alex Nguyen

Smart Water Metering (Title TBD)

Preface

The way we monitor and manage vital resources has changed dramatically as a result of the incorporation of cutting edge technologies into critical infrastructure systems, such as water distribution networks. The idea of "smart water grids", similar to its smart energy grid counterpart, has gained traction recently as a viable way to improve the sustainability, dependability, and efficiency of water supply systems. Smart water grids use state-of-the-art technologies, such as data analytics and Internet of Things (IoT) devices, to maximize water distribution, find leaks, reduce cyber attacks, and enhance system performance.

A thorough examination of one such technology, fully homomorphic encryption, or FHE as it relates to smart water grids is provided in this document. With FHE, it is possible to protect sensitive data transported across smart water infrastructure and still use encrypted data for the operation of advanced analytics and anomaly detection algorithms. The main goals, driving forces, and contributions of this research project are described in the introduction.

Introduction

Communities depend on water delivery systems as essential infrastructure because they guarantee the availability of clean water for many uses, including drinking, sanitation, and industrial use. Like its electrical counterparts, Advanced Metering Infrastructure (AMI) is essential to properly monitoring and controlling water consumption in contemporary smart water systems. The Internet of Things (IoT) is used in these smart water grids, which include data management systems, communication networks, and smart meters deployed at each customer's location.

Smart water networks are vulnerable to data integrity threats, which jeopardize the precision and dependability of usage data, just like electrical grids. Both utilities and customers are seriously at risk from these kinds of attacks. Similar to deductive assaults in electrical systems, data falsification in smart water grids frequently takes the form of water theft, when reported consumption figures are lower than the real usage. Additive attacks, on the other hand, entail manipulating reported consumption figures, which may result in increased expenses and improper use of resources. Additionally, more subtle threats come in the form of camouflage attacks, which combine additive and deductive attacks to maintain the mean consumption while secretly changing individual consumption records.

Ensuring the integrity, dependability, and effectiveness of smart water grids requires the detection and mitigation of various data integrity assaults. Conventional anomaly detection systems present a promising defense against these kinds of attacks since they can detect departures from typical usage patterns. Deploying such systems, however, raises privacy concerns over user data, as providing detecting systems with individual consumption information may breach privacy laws and damage customer confidence.

Driven by the necessity to balance privacy concerns with security requirements, researchers have investigated a range of privacy-preserving methods for anomaly detection in smart water grids. While approaches like Secure Multiparty Computation (SMC) and Differential Privacy (DP) provide theoretical guarantees of privacy, they frequently have scalability or accuracy issues. Fully Homomorphic Encryption (FHE), in particular, offers a strong substitute by allowing calculations to be performed on encrypted data without the requirement for decryption. The problem lies in striking a balance between processing overhead and detection accuracy when including FHE into anomaly detection frameworks.

Using a Look-Up Table (LUT) based FHE approach combined with Private Information Retrieval (PIR), we provide a privacy-preserving anomaly detection system inspired by the privacy-preserving anomaly detection system in smart energy grids designed for smart water grids in this research. By employing LUTs and PIR, our method addresses the key challenges of integrating FHE into anomaly detection frameworks, providing an accurate solution that maintains user confidentiality. Early findings suggest that this innovative approach can enhance the integrity, dependability, and effectiveness of smart water grids. The following sections will detail the technology behind our method, explore implementation challenges, and present evidence of its impact on improving water management systems.

Glossary

Fully Homomorphic Encryption (FHE): A cryptographic technique that allows computations to be performed on encrypted data without the need for decryption, enabling privacy-preserving data processing and analysis.

Homomorphic Encryption (HE): A cryptographic scheme that supports certain algebraic operations on encrypted data, enabling computations to be performed on ciphertexts while still ensuring the correctness of the results.

Internet of Things (IoT): A network of interconnected devices embedded with sensors, software, and other technologies that enable them to collect and exchange data over the internet.

Private Information Retrieval (PIR): A cryptographic protocol that allows a user to retrieve information from a database without revealing which specific data items they are accessing.

Look-Up Table (LUT): A data structure used in computer science and cryptography to store precomputed values for efficient retrieval during computation.

Advanced Metering Infrastructure (AMI): A system of smart meters, communication networks, and data management systems that enable the remote monitoring and management of utility services, such as electricity or water consumption.

Differential Privacy (DP): A concept in data privacy that aims to provide strong guarantees for the protection of individual privacy in statistical databases or data analysis systems. The goal of DP is to enable the analysis of sensitive data while minimizing the risk of revealing information about specific individuals contained in the data.

Brakerski-Fan-Vercauteren (BFV): A fully homomorphic encryption scheme that enables computations to be performed on encrypted data. It is one of several FHE schemes that provide the ability to execute a range of arithmetic operations on ciphertexts such that, when decrypted, the result matches the outcome of the same operations performed on the plaintexts.

Single Instruction, Multiple Data (SIMD): Parallel computing paradigm where multiple data points are processed simultaneously under one instruction. This approach is useful when the same operation are used many times across a large dataset

User Requirements Definition

- The system should enable anomaly detection in smart water grids while preserving the privacy of customer consumption data. Anomaly detection algorithms should operate directly on encrypted data without the need for decryption, ensuring that sensitive information remains protected.
- Implementing Fully Homomorphic Encryption to encrypt customer consumption data at the point of collection would allow us to detect anomalies without the need to decrypt any additional private information.
- Utilizing a homomorphic LUT-based approach to support privacy-preserving anomaly detection between utility providers, customers, and other parties involved in security services. We intend to store data pairs of input and output values for each function required by the anomaly detection framework in LUTs, facilitating the Arithmetic mean -Harmonic mean and ratio calculations over FHE.

- Implementing the Brakerski-Fan-Vercauteren (BFV) scheme allows the difference calculations and implementing the LUT-based approach (BFV doesn’t support division) allows us to calculate the ratio to compare effectiveness between both detection formulas.
- The BFV scheme's ability to perform additive and multiplicative operations on encrypted data makes it suitable for computing complex anomaly detection metrics without compromising data privacy. Shown below will introduce each communication component with the system architecture:
  - Smart Meters
  - Data Collectors
  - Computation Servers
  - LUT Provider
  - Utility Provider
- We aim to provide real-time or near-real-time anomaly detection capabilities to promptly identify suspicious water usage patterns and potential attacks and minimize processing delays to ensure timely response to anomalies detected.
- The system will be subjected to a comprehensive array of simulated attacks to validate its detection capabilities.
- The system shall implement functionality to compare the effectiveness of two anomaly detection formulas: the Harmonic Mean (HM) to Arithmetic Mean (AM) ratio, and the difference between AM and HM. This comparative testing is essential to identify which formula yields the lowest error rate in anomaly detection within the smart water grid data.

System Requirements Specification

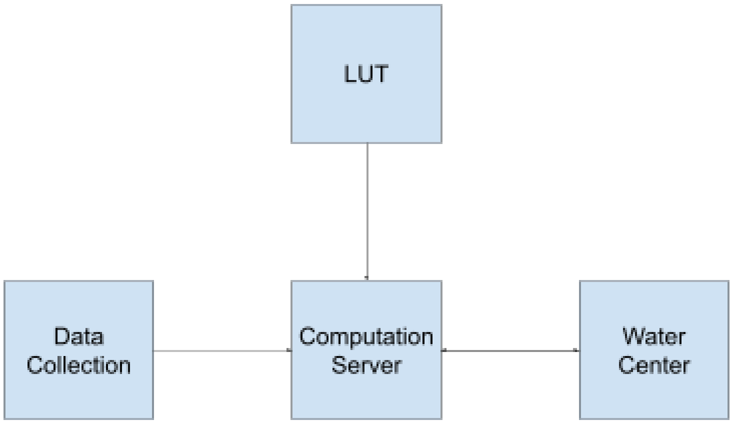
Microsoft SEAL: Cryptograph library that performs operations on encrypted data. Supports FHE schemes including the Brakerski-Fan-Vercauteren scheme

CMake: Tool to assist in building processes of software

OpenMP: API that supports multi-platform shared memory multiprocessing which helps in multi-core processors work efficiently to enhance the performance of software

System Architecture

The experimental setup will be similar to that of the electrical grid experiment, having four main components. These components are the server where computations are performed, the LUT provider, the utility provider, and the region of water collection that has N smart meters. With this design each component will communicate with the computational server alone, and any necessary calculations will be gathered and done there. The figure below shows a basic representation of the setup.



Smart Meter

- **Data Collection:** The smart meter will be responsible for collecting water consumption data from customers houses
- **Data Encryption:** Each smart meter will encrypt the collected data using a public key before transmission to ensure data remains confidential during transmission to the data collector

Data Collector

- **Data Hub:** The data collector will take in all the encrypted data from multiple smart meters across a region
- **Data Transmission:** The data will be forwarded to the computation server

Computation Server

- **Data Processing:** The computation server processes the encrypted data utilizing homomorphic encryption techniques
- **Anomaly Detection:** Based off the processed data, detecting anomalies based off certain metrics

LUT Provider

- **LUT Maintenance:** Generates the Look-Up Tables that are essential to perform complex calculations on encrypted data
- **Complex Operations:** Provides support for operations not directly supported by homomorphic encryptions

Utility Provider/Water Center

- **Analysis and Response:** Analyze the processed results sent by the computation server to detect any discrepancies or anomalies in water consumption usage that indicates any potential attacks

Operational Flow

- Key generations include public, secret, relinearization, and galios keys
    - Public Key: Initial encryption key
    - Secret Key: Decryption key
    - Relinearization Key: Manageable ciphertext sizes post multiplication operations
    - Galios Key: Rotations in SIMD processing and processes multiple data points simultaneously instead of needing to process one at a time
- Public keys will be shared with smart meters that encrypt the data at the source. Relinearization and Galios keys are sent to the computation server to support processing of encrypted data
- The public key encrypted water consumption information will be sent to the data collectors ensuring the information is protected
- The data collectors collect the encrypted data and forward all the information to the computation server
- The computation server performs all necessary computations including calculating the harmonic mean, arithmetic mean, and look up table
- The look up table is prepared by the LUT provider for operations that cannot be computed by the BFV scheme. The computation server utilizes the LUT to perform the necessary computations
- The computation server will then calculate the ratio and difference for anomaly detection over certain timeslots
- The processed data should remain encrypted throughout the entire computation process
- Final encrypted results is decrypted and indicates potential anomalies within the utility center

System Models

The HM and AM are fundamental metrics used to analyze water consumption within a given timeframe.

Arithmetic Formula:

$$AM_t = \frac{1}{N}\sum_{i=1}^{N} w_i(t)$$

Description: Calculates average water consumption over a defined period by summing individual meter readings and dividing by total numbers of meters

Harmonic Formula:

$$HM_t = \frac{N}{\sum_{i=1}^{N} \frac{1}{w_i(t)}}$$

Description: Average rates by taking the reciprocal of each rate

To detect anomalies in water consumption data, we will be comparing both the difference and the ratio of the HM and AM.

Difference Over Timeslot Formula:

$$D_d = \sum_{t \in T} (AM_t - HM_t)$$

Description: Subtracting the arithmetic mean from harmonic mean for a specific timeslot

Ratio Over Timeslot Formula:

$$Q_d = \frac{\sum_{t \in T} HM_t}{\sum_{t \in T} AM_t}$$

Description: Dividing the harmonic mean from the arithmetic mean for a specific timeslot

Daily Ratio Formula:

$$Q_d = \frac{\sum_{t=1}^{24} HM_t}{\sum_{t=1}^{24} AM_t}$$

Description: Dividing the harmonic mean from the arithmetic mean over all timeslots of the day

Daily Difference Formula:

$$D_d = \sum_{t=1}^{24} (AM_t - HM_t)$$

Description: Subtracting the arithmetic mean from the harmonic mean over all timeslots of the day

System Parameters

1. N: The number of water meters in the system.
2. T: Timeslots within a 24 hour period meaning every hour is a timeslot.
3. t: Specific time within set T.
4. wi(t): The water consumption reading from the i-th smart meter at timeslot t.
5. Dd: The AM-HM difference for the d-th date/Timeslot.
6. Qd: The HM over AM ratio for the d-th date/Timeslot.

Comparative Analysis

- Our system employs a comparative analysis to determine which metric Dd or Qd effectively identifies anomalies better.

Historical Data Processing

- This analysis will involve processing historical water consumption data to establish baseline patterns and then applying the metrics to detect deviations.

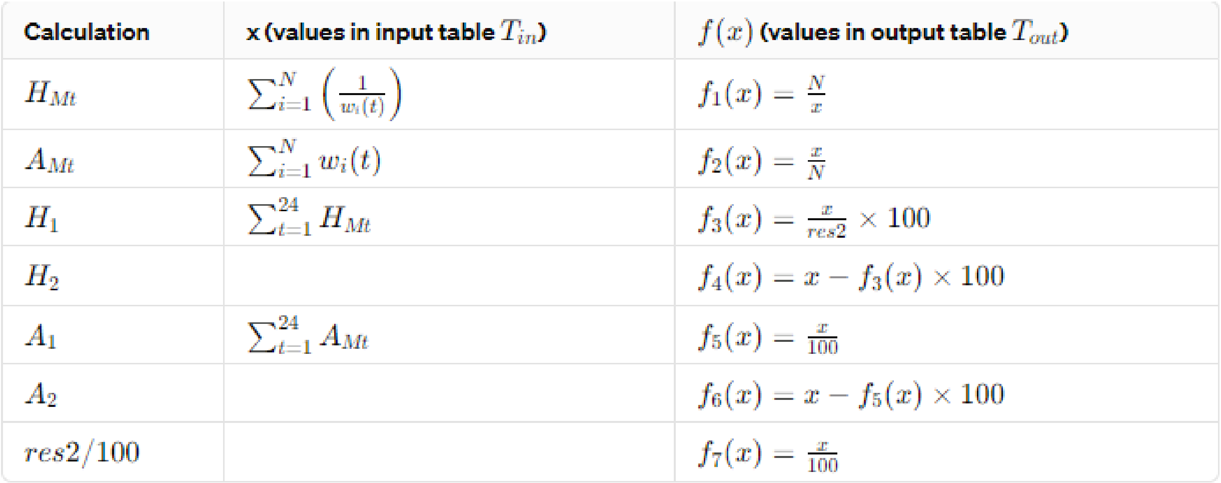
Dynamic Threshold Establishment

- Thresholds for these deviations will be dynamically established based on historical data and the specific characteristics of each distribution network and water consumption patterns.

BFV Characteristics

- Given the characteristics of the BFV scheme which only supports addition, subtraction, and multiplication, we must introduce a look up table to perform the calculations for the ratio (division)

Prepared Look-Up Table for Ratio



Calculation	x (values in input table T_{in})	$f(x)$ (values in output table T_{out})
H_{Mt}	$\sum_{i=1}^{N} \left(\frac{1}{w_i(t)}\right)$	$f_1(x) = \frac{N}{x}$
A_{Mt}	$\sum_{i=1}^{N} w_i(t)$	$f_2(x) = \frac{x}{N}$
H_1	$\sum_{t=1}^{24} H_{Mt}$	$f_3(x) = \frac{x}{res2} \times 100$
H_2		$f_4(x) = x - f_3(x) \times 100$
A_1	$\sum_{t=1}^{24} A_{Mt}$	$f_5(x) = \frac{x}{100}$
A_2		$f_6(x) = x - f_5(x) \times 100$
$res2/100$		$f_7(x) = \frac{x}{100}$

Note: Using 100 as a placeholder value to scale up or down

Precision and Scaling

Adopting the LUT, we must calculate each time slot for HM and AM. Then we need to calculate the sums of HM and AM over a 24 hour period. HM/AM is how we compute Qd. We will be multiplying instead of dividing since the division function requires a two-input function.

Assume the maximum scaled ration of Qd is 9 but the significant digits of the plaintext space is 5, we would have to discard the last 4 digits of the Qd result. Reducing the Qd result by 10000 to discard the last 4 digits since 100*100 = 10000.

HM/AM Ratio

$$Q_d = \frac{\sum_{t=1}^{24} HM_t}{\sum_{t=1}^{24} AM_t} = \sum_{t=1}^{24} HM_t \times \frac{1}{\sum_{t=1}^{24} AM_t}$$

Definition: The ratio of the sum of the harmonic means over the sum of the arithmetic means calculated over a 24 hour period. This ratio is computed using the multiplication of the inverse values due to limitations

Polynomial Representation

Multiplying the daily ratio of HM and AM makes the values too large which results in an overflow of the plaintext, so we drop the least significant digits off the result. Expressing HMt and AMt as a polynomial is shown below with H1, H2, A1, and A2 where these values are coefficients.

$$\sum_{t=1}^{24} HM_t = \mathcal{H}_1 \times 100 + \mathcal{H}_2$$

$$\frac{1}{\sum_{t=1}^{24} AM_t} = \mathcal{A}_1 \times 100 + \mathcal{A}_2$$

Note: 100 is not a fixed value, it depends on the plaintext space in the FHE setting and depends on how many significant digits we would like to keep. Larger plaintext space leads to more execution time. Dropping more significant bits allows us to reduce execution time

Daily Qd Ratio

$$\begin{aligned} Q^d &= \sum_{t=1}^{24} HM_t \times \frac{1}{\sum_{t=1}^{24} AM_t} \\ &= (\mathcal{H}_1 \times 100 + \mathcal{H}_2) \times (\mathcal{A}_1 \times 100 + \mathcal{A}_2) \\ &= \mathcal{H}_1 \times \mathcal{A}_1 \times 10,000 \\ &\quad + (\mathcal{H}_1 \times \mathcal{A}_2 + \mathcal{H}_2 \times \mathcal{A}_1) \times 100 + \mathcal{H}_2 \times \mathcal{A}_2 \end{aligned}$$

Definition: Due to size limitations in the plaintext space, using polynomial representation avoids overflow

Defining res1 and res2

$$res1 = \mathcal{H}_1 \times \mathcal{A}_1$$

$$res2 = \mathcal{H}_1 \times \mathcal{A}_2 + \mathcal{H}_2 \times \mathcal{A}_1$$

Discarding Least Significant Digits

$$Q^d = \text{res1} + \frac{\text{res2}}{100}$$

Note: This is scalability for the ratio formula, not the difference formula.

Definition: A necessary step in processing the Qd values where the last 4 digits are discarded to manage the size within the plaintext space

Precision Parameter and Vector Creation

Water consumption data will be retrieved by the smart meters, encrypted, and sent to the data collector. Using the LUT method, a packed ciphertext filled with elements that are identical to a search query, the smart meter will create two vectors v(wi(t)) and v((1/wi(t)) where wi(t) is water consumption:

$$v(w_i(t))[k] = \text{Round}(2^{p^{AMin}} \times w_i(t)),$$

$$v\left(\frac{1}{w_i(t)}\right)[k] = \text{Round}(2^{p^{HMin}} \times \frac{1}{w_i(t)}).$$

Note: 2p is a precision parameter, introducing this precision parameter allows us to understand trade-offs between execution time and anomaly detection accuracy.

Vector Length

Where 0 <= k < l, where l represents the length of the two vectors. Since we are employing integer encoding rather than bitwise encoding to speed up execution time, the data must be converted into integers before the encryption, allowing the data to be scaled with precision parameters 2p and rounded into integers.

Integer Encoding

- A method of encoding data into integers before encryption. This process enhances the efficiency of operations within cryptographic schemes that do not natively support high-precision floating-point arithmetic

- Converting data to integers reduces computational overhead which is essential when execution speed is a critical part

Difference Scaling

Given the nature of taking the difference, we do not entirely need a LUT to perform the calculations for the difference since addition, subtraction, and multiplication is supported.

To scale the harmonic and arithmetic formulas for the difference you would simply need to multiply the encrypted values by 100 as an example, subtract the encrypted arithmetic from encryption harmonic to get the difference and divide by 100. Even though we are dividing in the formula which is not entirely supported by the BFV scheme, we are dividing by a known constant which can be precomputed. The LUT is mainly targeted towards complex division which involves variables

System Architecture Definitions

- **Smart Meters:** Responsible for monitoring water consumption data
- **Data Collectors:** Collects consumption data over a region of smart meters
- **Computation Server:** Performs all necessary computations on encrypted data
- **LUT Provider:** Generates and provides operations that are not entirely supported by the FHE system
- **Utility Provider/Water Center:** Receives processed, encrypted data from the computation server and decrypts the final results to analyze and detect anomalies

Encrypted Key Generation and Management

- Initiate Encryption Parameters
- The SEALContext object is created with the defined parameters
- The KeyGenerator object is created and responsible for generating public and private keys. Public keys are used for encryption and private keys are used for decryption
- Galios keys are generated to facilitate operations such as rotations in SIMD processing and utilize encrypted vectors to take advantage of multiple computations to reduce execution time
- Relinearization keys are generated for size reduction of ciphertexts after operations utilizing the
- Batching capabilities check to ensure batching is enabled to allow multiple values to be encoded into plaintext and then cipher text, allowing for parallel operations on multiple data values.

- Key serialization provides serialized encryption parameters that ensures the keys can be stored and distributed to the correct components of the system to perform encryptions, decryptions, and other homomorphic operations

Reasons for choosing Brakerski-Fan-Vercauteren scheme

- Supports addition, subtraction, and multiplication
- Batching introduces the ability to have multiple values to be encoded into a single ciphertext which enhances computational efficiency
- Introducing relinearization keys help reduce degrees of polynomials down to a manageable size without the need of decrypting. Expanded ciphertext will be brought back down to either original size or degree after the initial multiplication. The relinearization keys are essential to performing sequences of multiplication without the ciphertexts becoming too large.
- The BFV scheme can easily be adjusted to balance between anomaly detection and performance making it scaleable to small and large data sets
- Noise Management can be an issue when performing homomorphic operations on ciphertext, BFV has efficient noise growth management which is especially critical for accurate computations with homomorphic operations

Anomalies

- **Deductive Attack:** Attacker reports reduced amount of water usage which leads to customers paying less
- **Additive Attack:** Attackers report increased amount of water usage which leads to customers paying more
- **Camouflage Attack:** Attackers divide meters into two equal sets where an additive attack is launched on one set resulting in higher consumption use and a deductive attack on the other set to reduce water consumption. This attack favors one set of customers and is harder to detect since the mean consumption rate would remain the same

System Evolution

Conclusion

Appendices

Index

Bibliography

[1] Smart Grid Detection Simulation
[2] IEEE Privacy Preserving Analytics Publication
[3] Professor Shameek PPA Paper
[4] Professor Shameek Forensics Paper
[5] Detection of False Data Injection in Smart Water Metering Infrastructure